CSE 5382: Secure Programming -Spring 2021
Capture the Flag Bonus Exercise

Student Name: Aparna Krishna Bhat
Student ID: 1001255079

	Cryptography	Malware	Network Capture	Reverse Engineering	Steganography
Level 1 (2 Points)	backoff_malware	sandworm_apt	M0d1c0nF14G	lm	covert_channel
Level 2 (3 Points)	WANNACRY	2e1afcef9baa15b8db764274b0e45d3f	YWRtaW46YWRtaW4 =	ClippyHasReturned	cookies_n_milk
Level 3 (4 Points)	FLAGSLASHDOTDASH	duqu_aint_dooku	NTLMSSP_NEGOTIATE	infected flag	R0075RUS 6th File.
Level 4 (5 Points)	did cicero say anything	123123	1255	73C972DF8DB0E1EDC289F491A09AF330	alan_turing_edm
Level 5 (6 Points)	NA	flag{kragany_use s_up$x}	passwordBasisk	{.}	NA

Explanation of Approaches

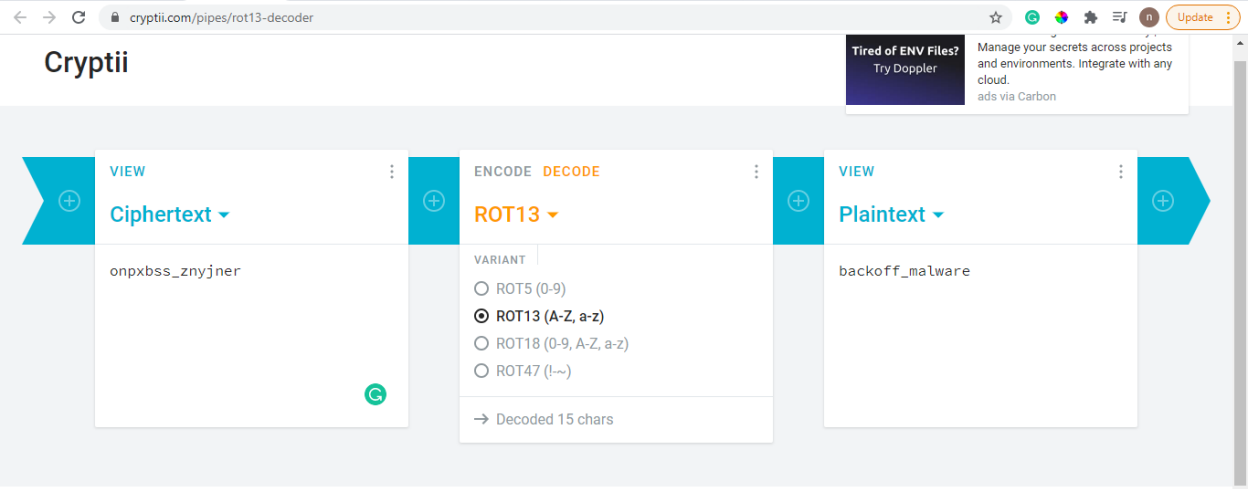
Cryptography

Level 1 – Orange Julius

I googled cipher text de-Cryptor and found a website that could decrypt the flag synt{onpxbss_znyjner}

https://cryptii.com/pipes/rot13-decoder

We get plain text ‘backoff malware' after decryption, as seen in the screenshot below.



Level 2 – Block Chain

The phrase "the pawn can beat the king" gives us the clue that we should use a chess board in this job, and the malware is concealed in the letters given. As a result, I built an 8-square chessboard. We can get MALWARE in the last column by arranging all the letters in this chess board row by row.

I	L	P	L	N	C	W	N
N	U	H	A	A	H	E	M
T	M	E	G	C	I	L	A
H	N	R	I	R	S	L	L
I	A	T	S	Y	N	K	W
S	R	H	W	W	O	N	A
C	C	E	A	H	W	O	R
O	I	F	N	I	A	W	E

However, there was no flag on this chess board. As a result, I tried using an online base 64 decoder to see if I could find any flags.

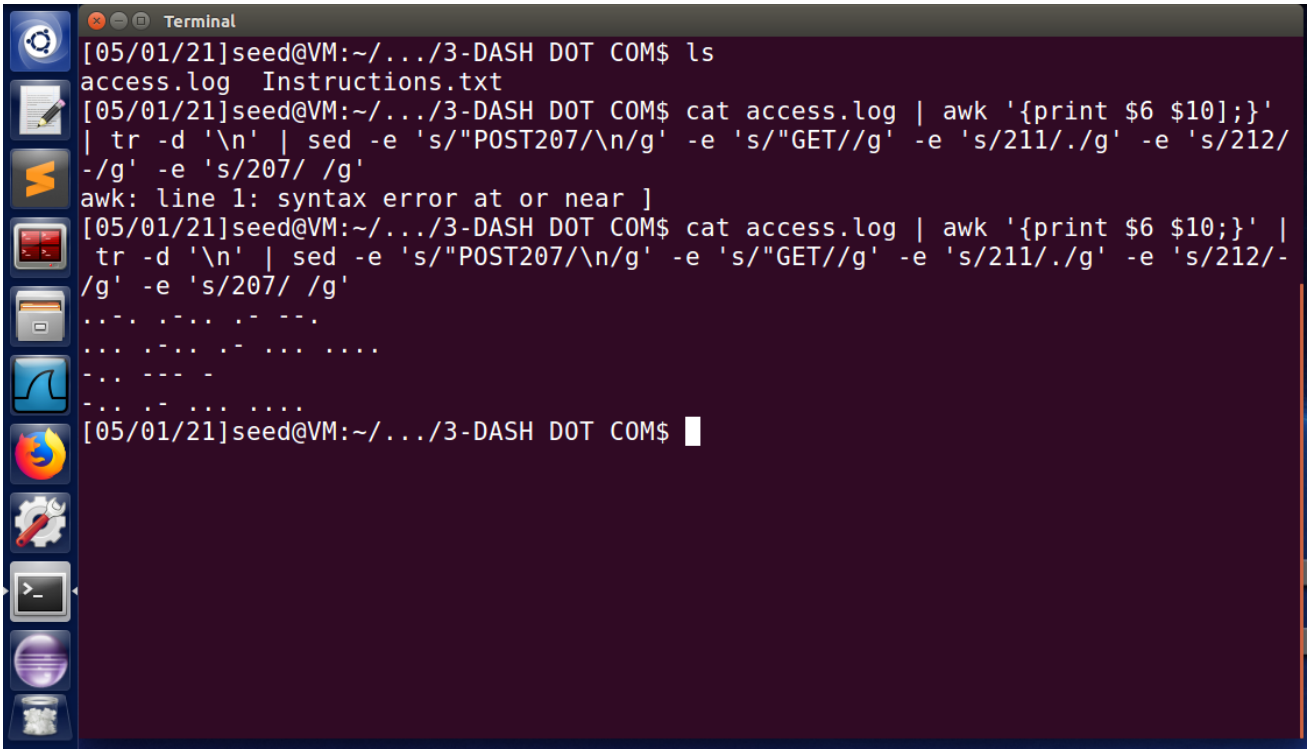
We can see in the screenshot that we get a list of decoded strings on the left. If we look at the first string we have, we can see the flag highlighted below in that string.

INTHISCOLUMNARCIPHERTHEFLAGIS**WANNACRY**WHICHISNOWAWELLKNOWNMALWARE

WANNACRY, a ransomware, is the secret flag in this string.

Level 3:- DASH DOT COM

The only difference in the logs was the request form and the requested resource. Often the image format is specified in the request, and other times it is not. The flag was written in morse code in the apache log file. The logs were translated to morse code, which was then converted to a human-readable string text format using a simple conversion method.



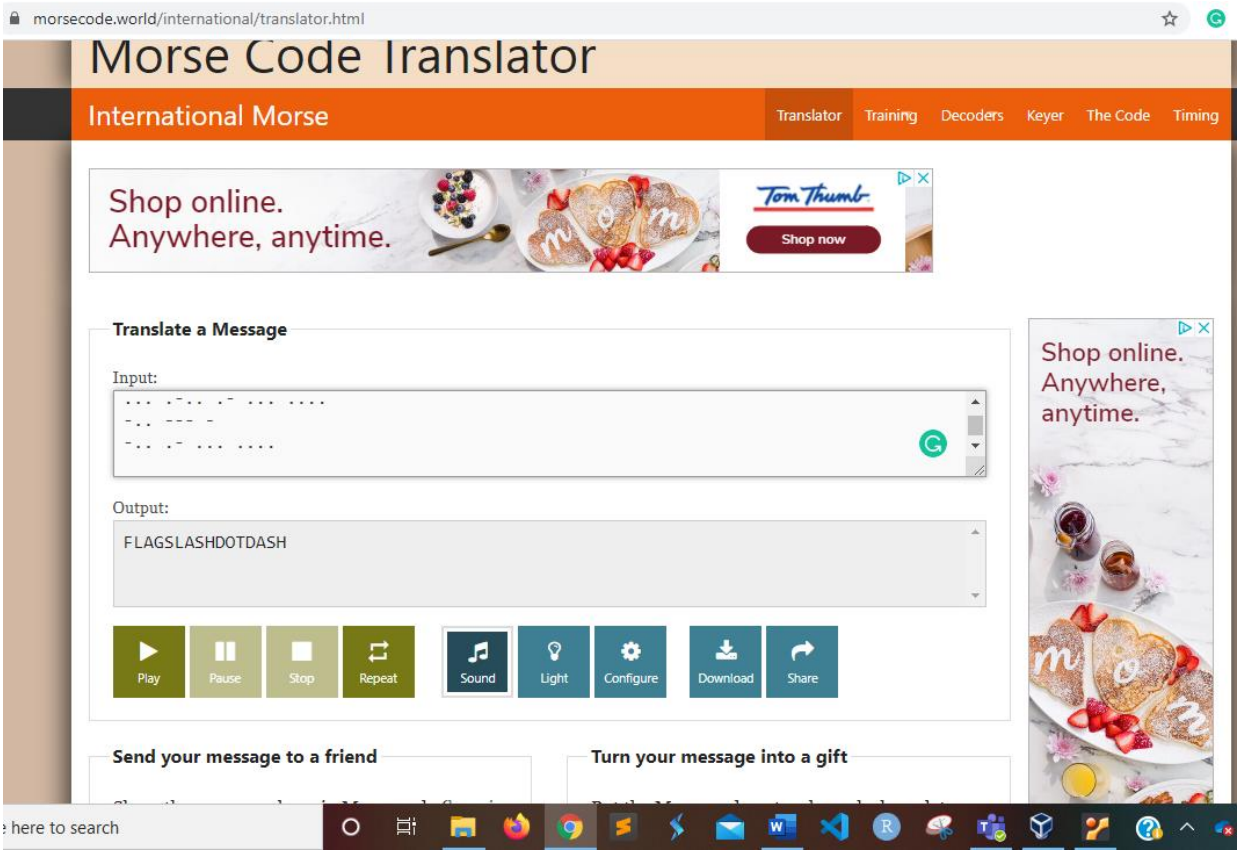
The morse code is:

..-. .-.. .- --.

... .-.. .- ... .

-.. --- -

-.. .- ... .



The flag is :- **flagslashdotdash**

Level 4:- VIII A Small Example

Using RSA decyprion process, values: d=157, n=2773

Message, M= (4 bits)^157 mod 2773

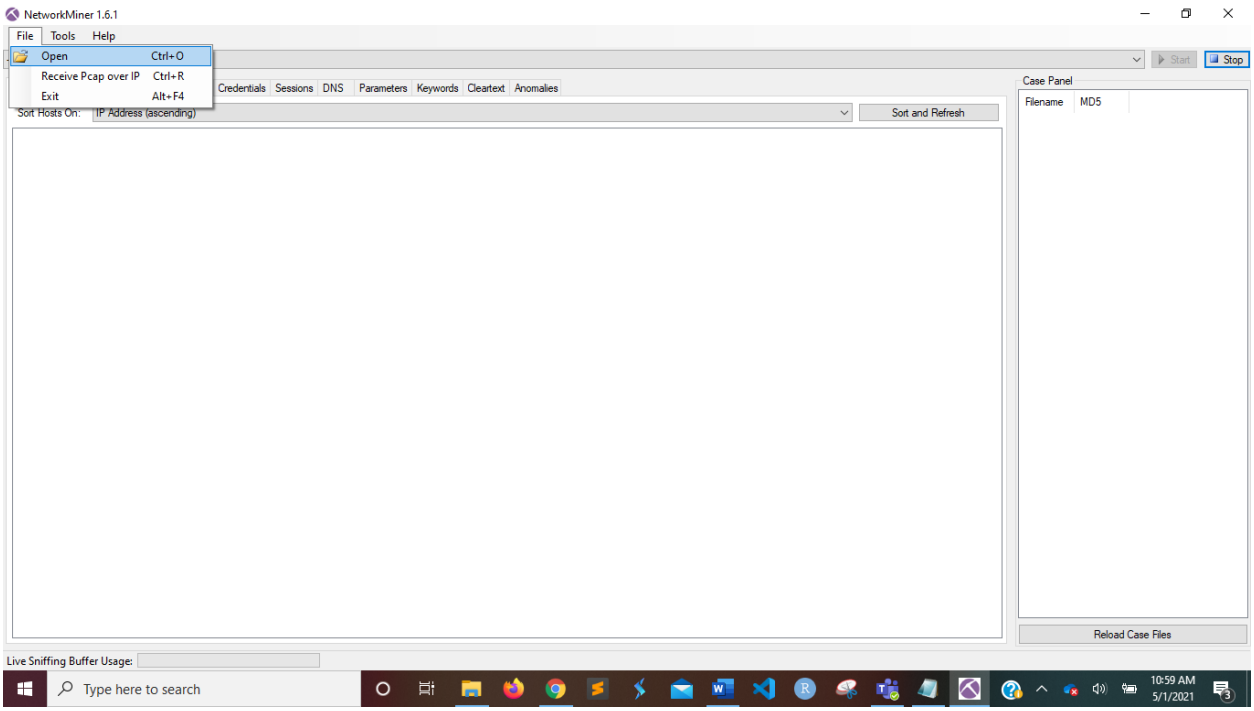
The flag is :- **“did cicero say anything”**

Level 5:- Big Blue Randu

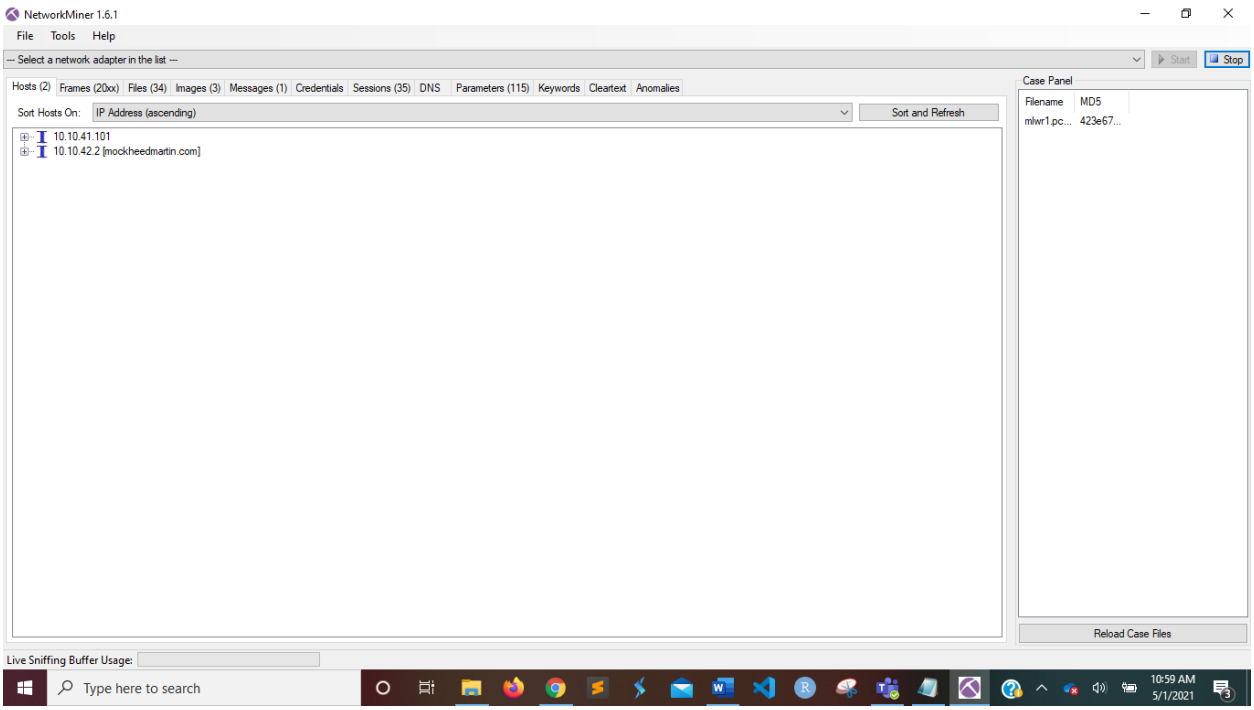
Malware

Level 1:- MLWR-1-PCAP

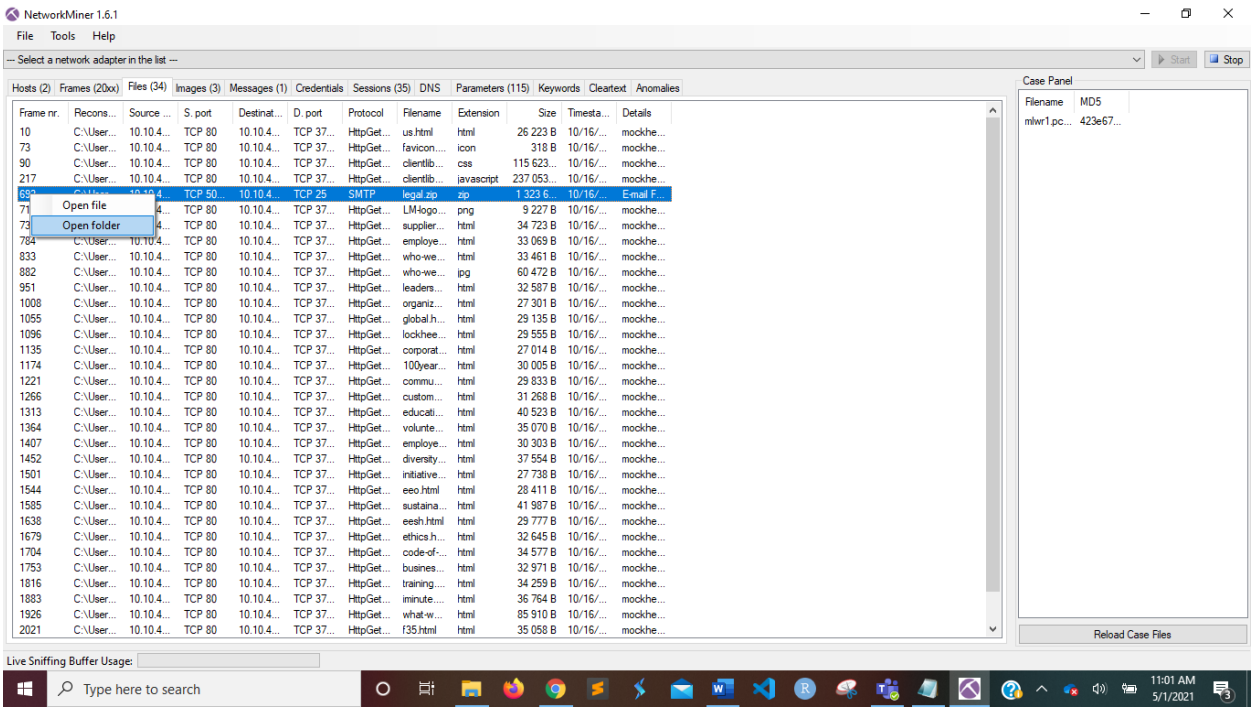
I came across a program called Network Miner. It was what I used to open the capture file.



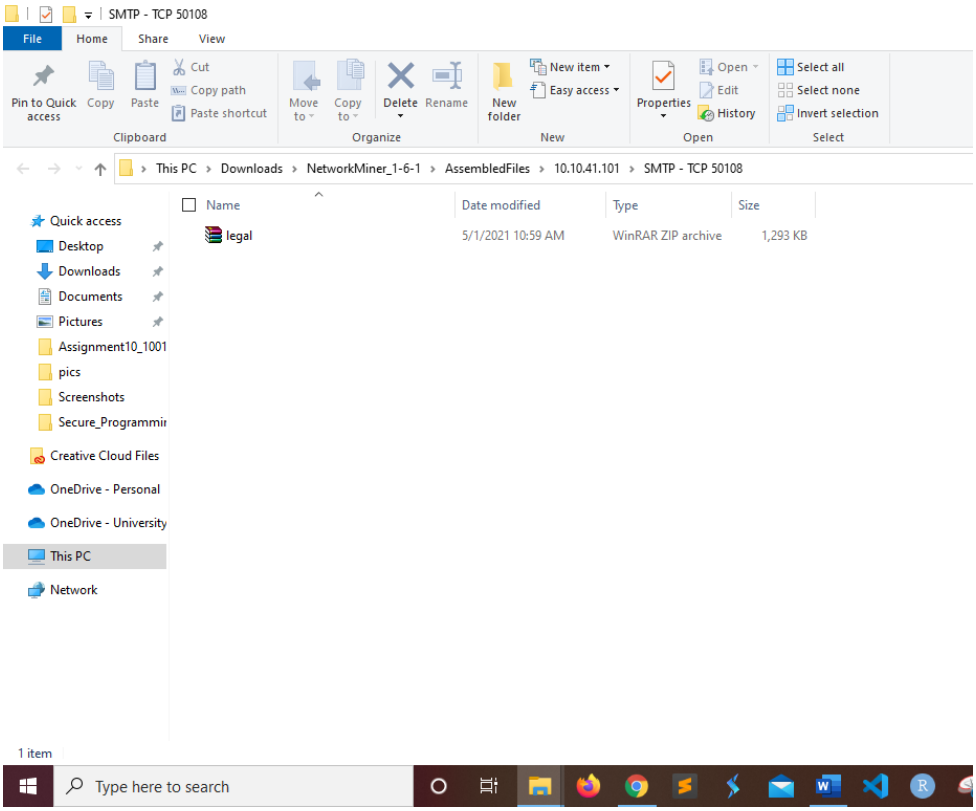
When I first opened it, it displayed two hosts, as seen in the screenshots.



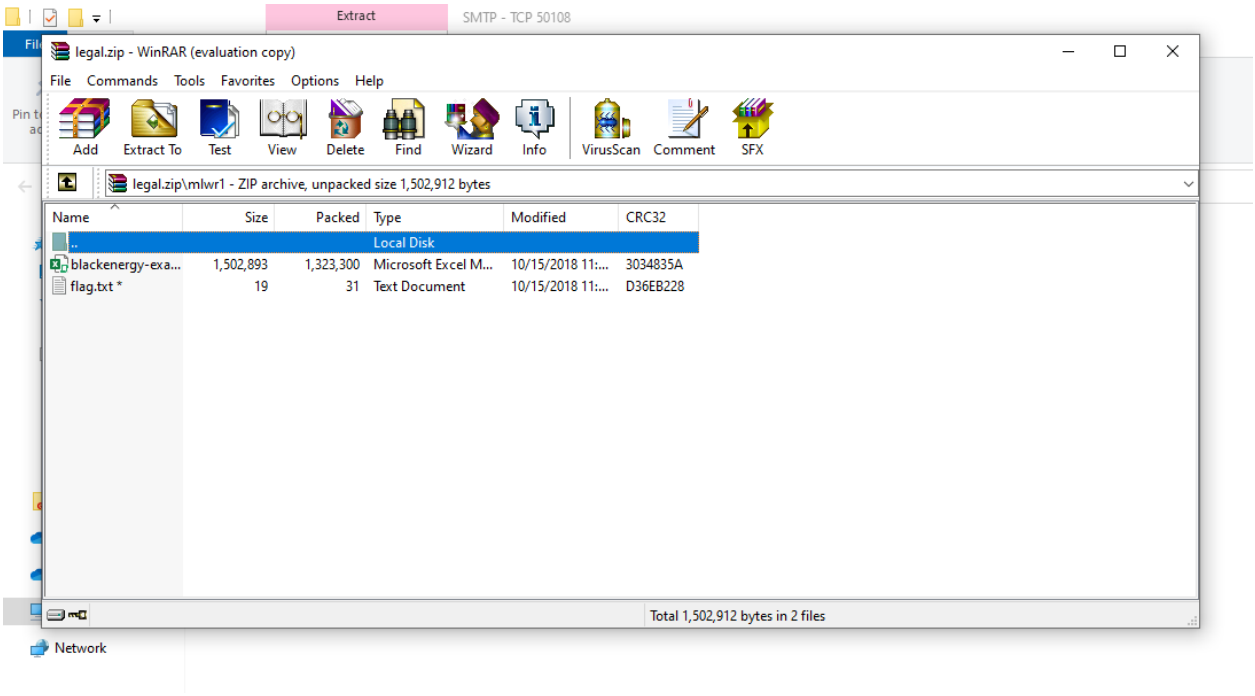
I went to the Files tab and looked for the legal.zip zip file. As shown below, I opened the folder.

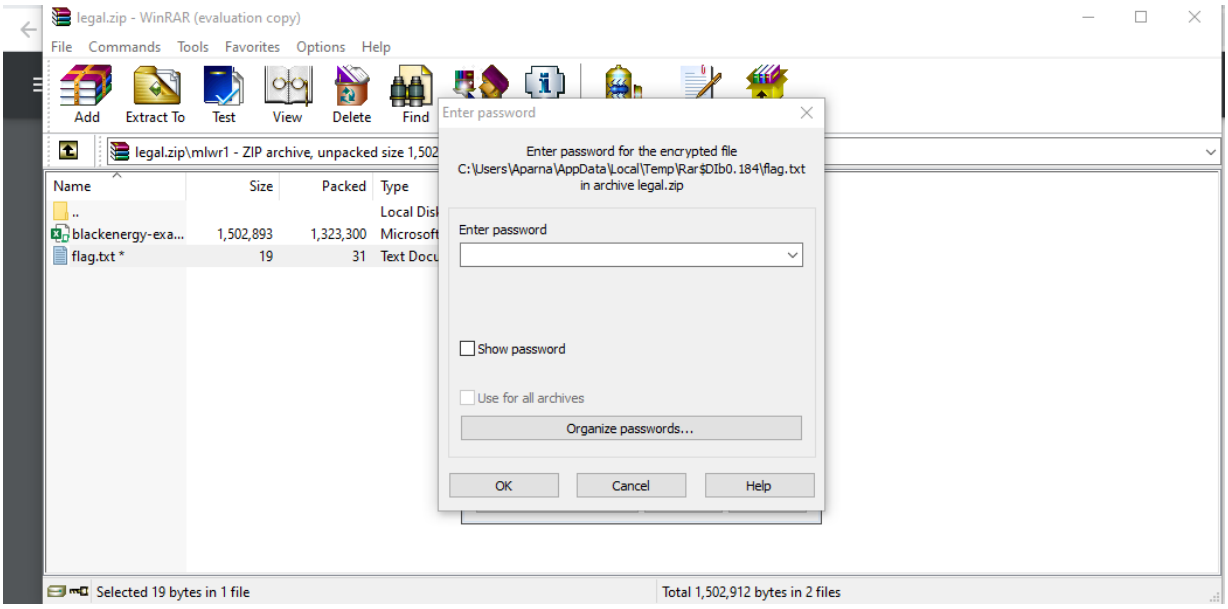


I extracted the mlwr1 folder from the legal.zip file.

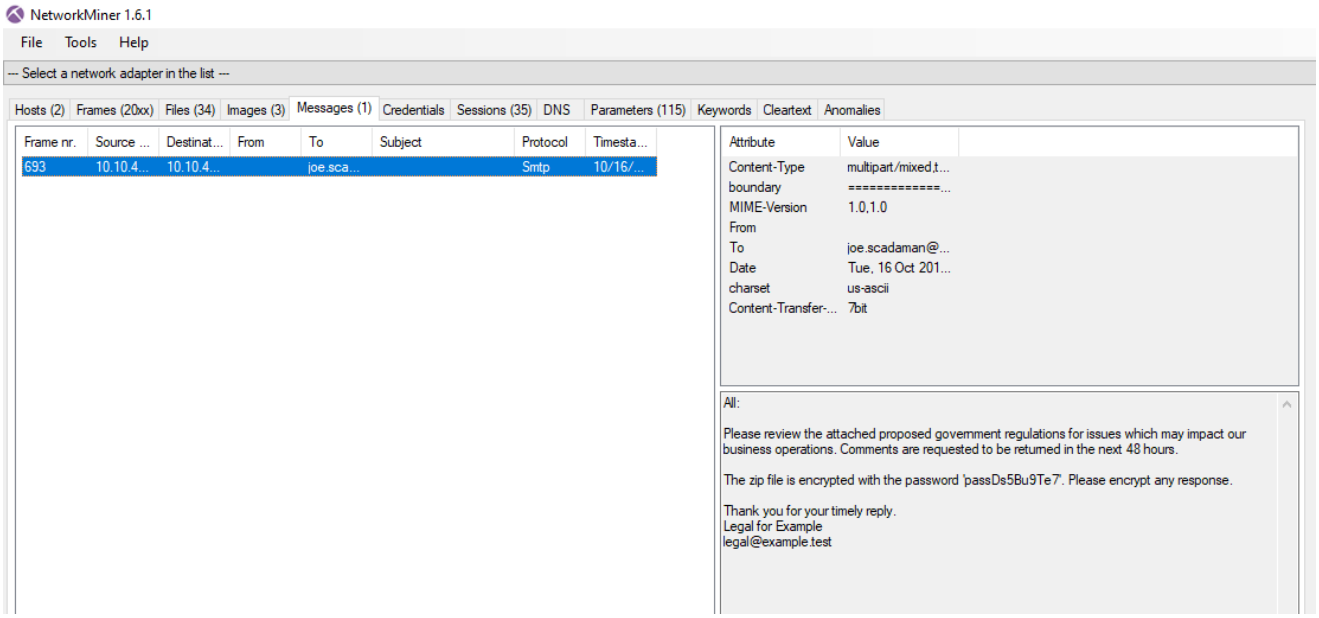


One Excel file and one text file called flag can be found in the mlwr1 folder. When I clicked on it, it asked for my password.

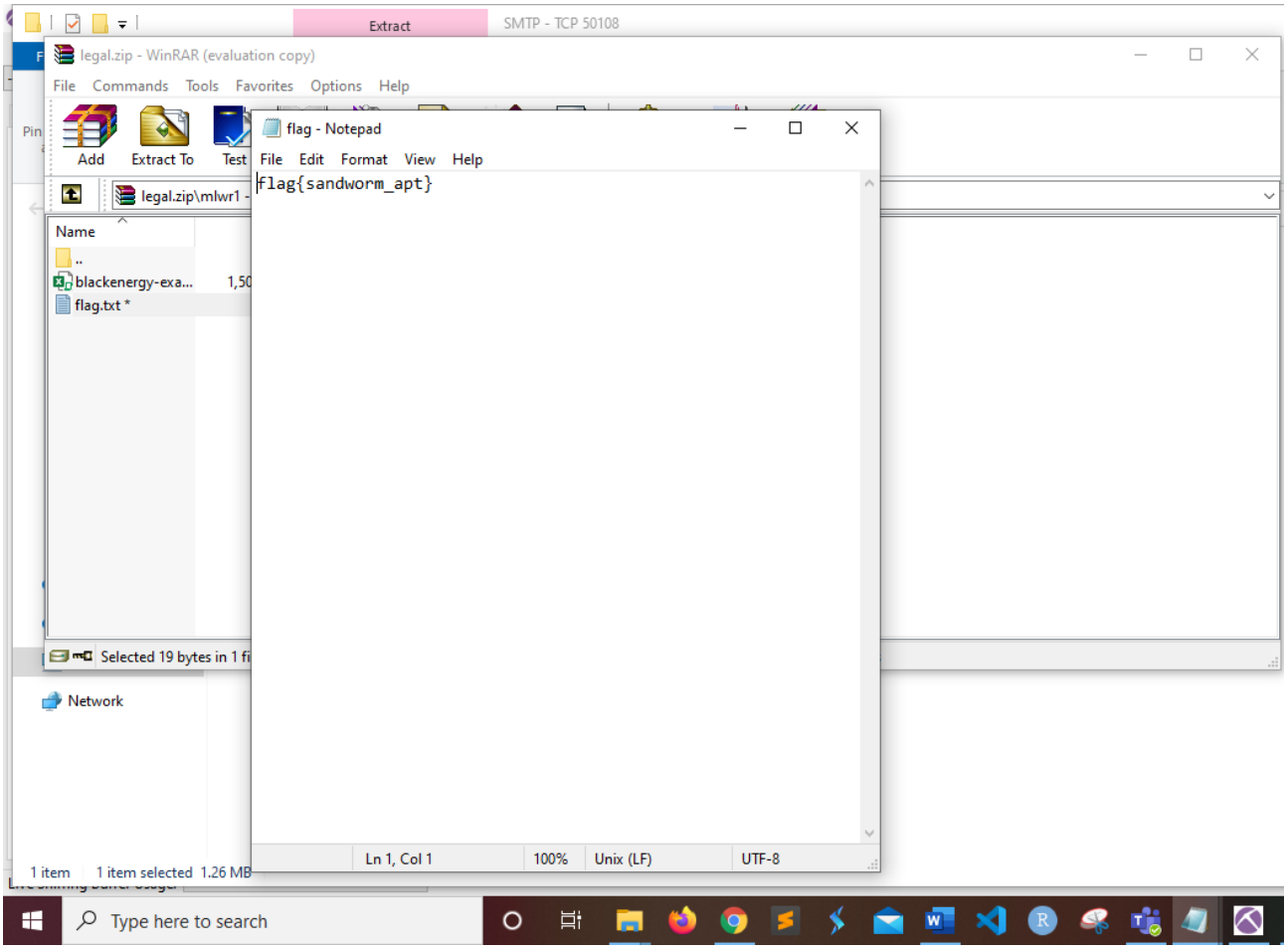




As a result, I returned to NetworkMiner and double-checked all of the other tabs. As shown in the screenshot below, I discovered the password in the Messages tab.



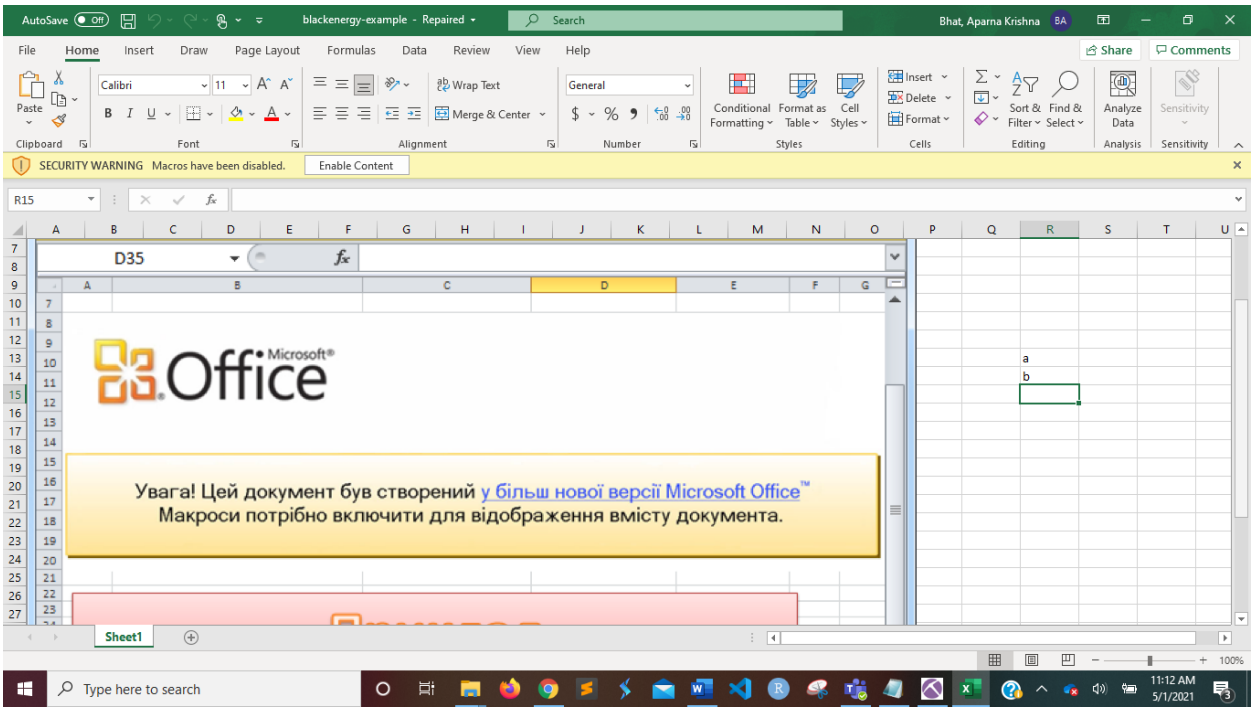
For the flag text file, use the **password passDs5Bu9Te7**.



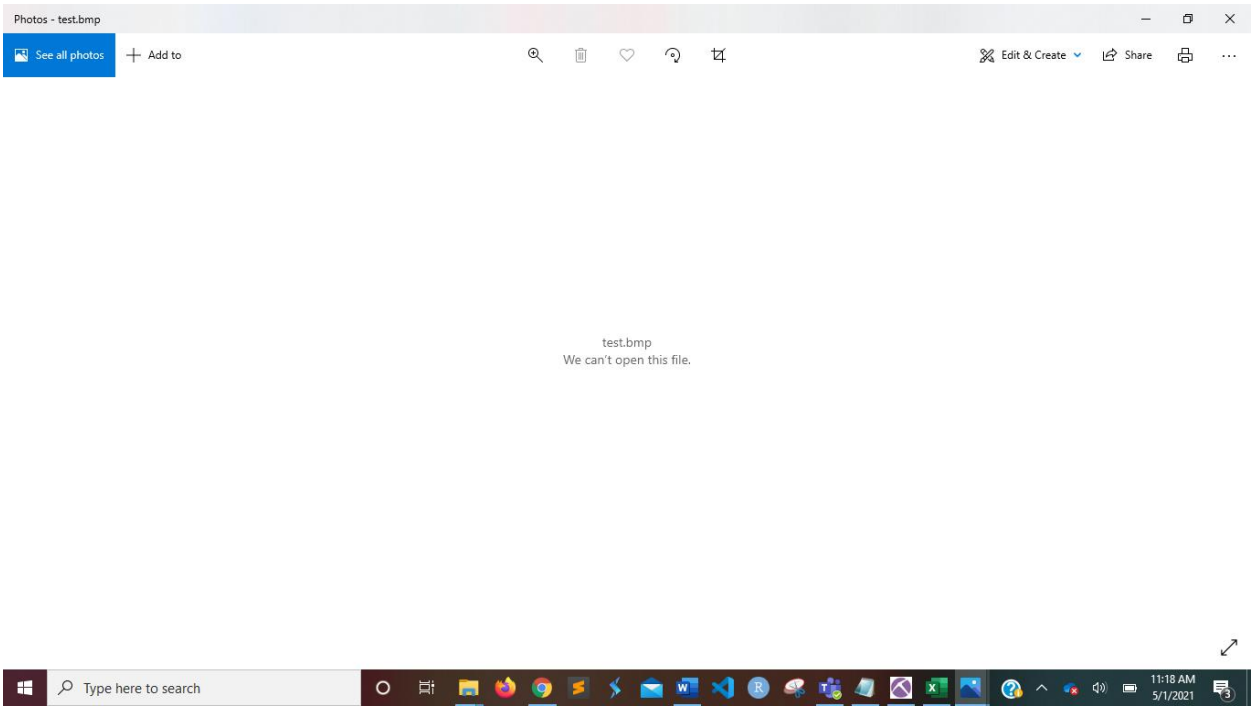
The flag is :- **sandworm_apt**

Level 2

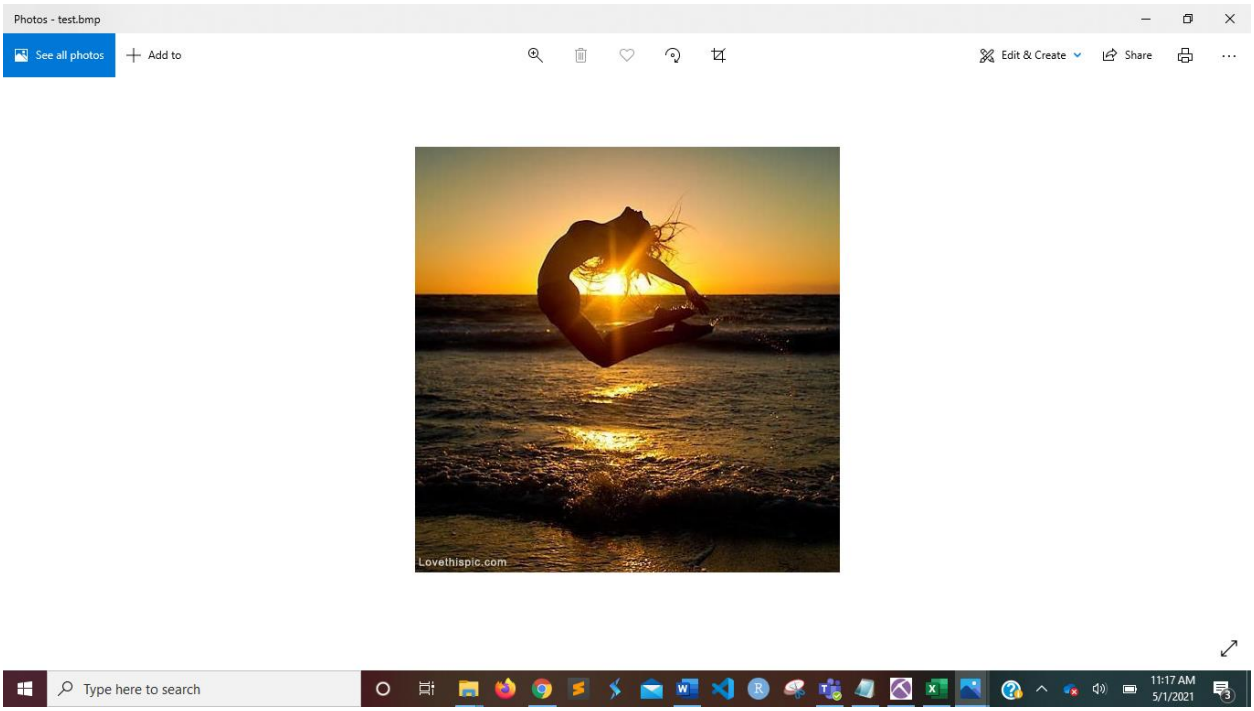
To complete this challenge, we must use the excel file from Level 1 to locate the flag. We check the data in the excel sheet.



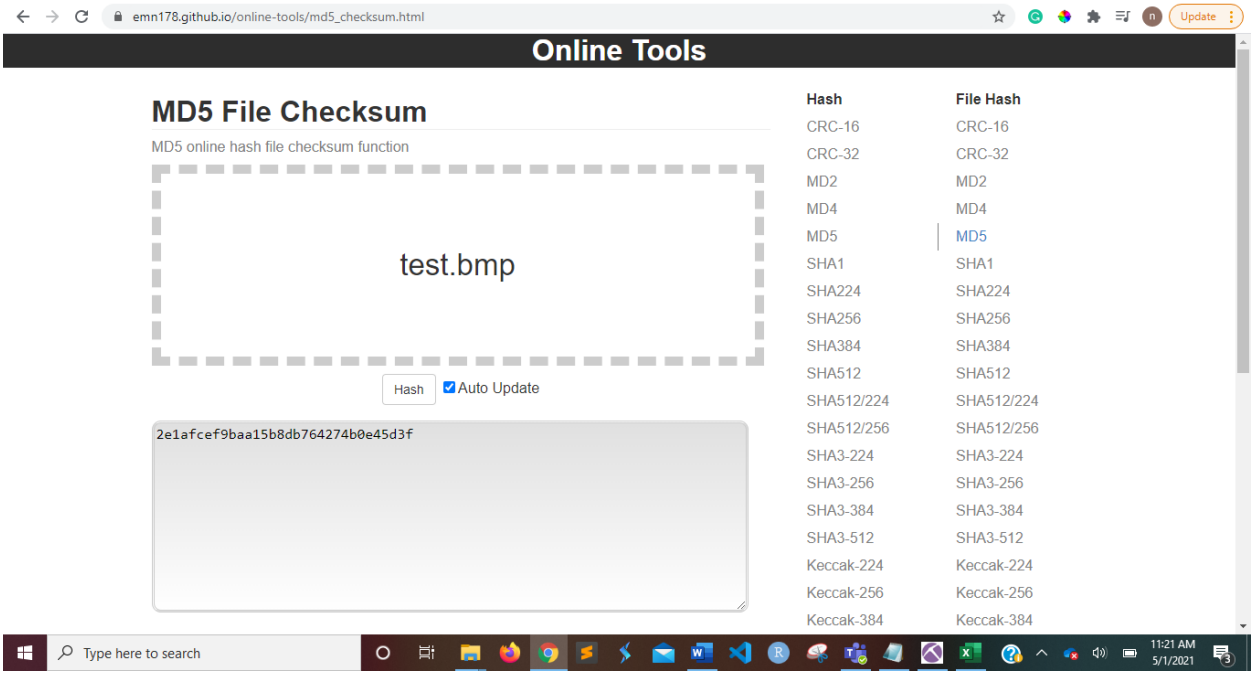
Another image file was attempting to be opened when I opened the excel sheet, as shown below.



As a result, I turned on the macros in the excel sheet, and a picture called test.bmp appeared on the screen, as shown below.



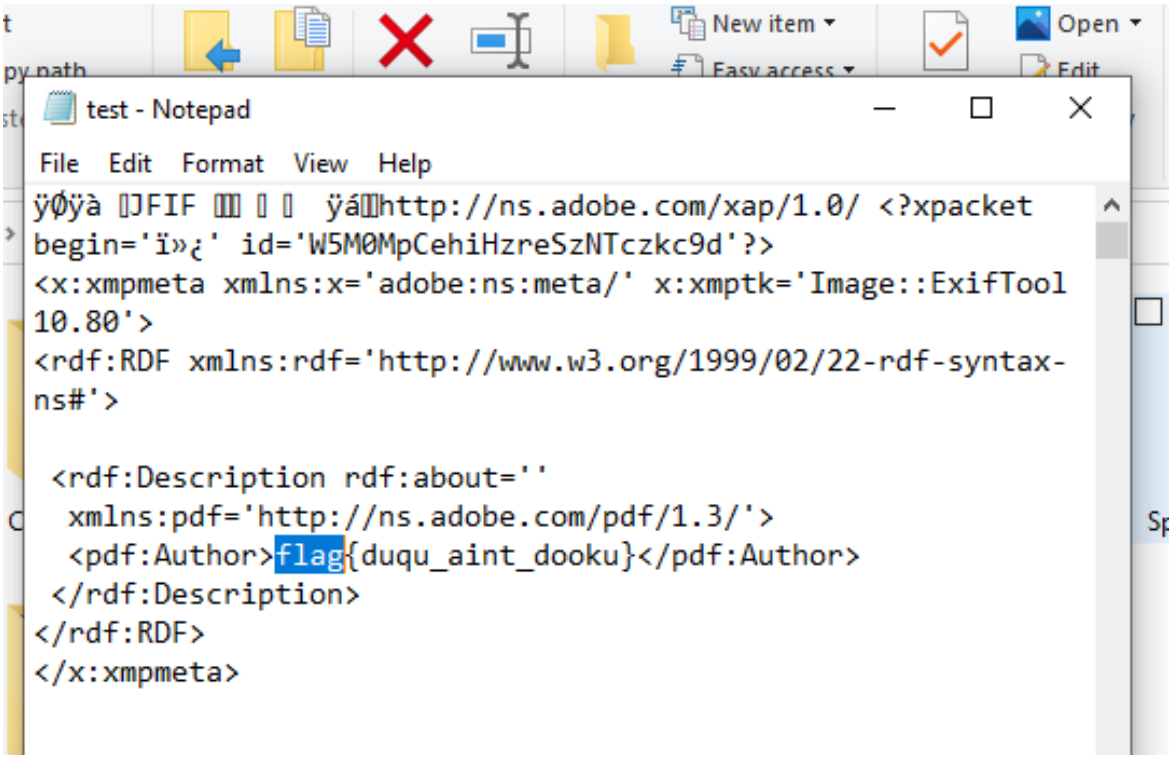
The file test.bmp was saved. The MD5SUM of the file that the spreadsheet drops are defined as the flag. I searched for MD5SUM and found a method to upload and evaluate the test.bmp file. I obtained the hash code, which is the flag.



2e1afcef9baa15b8db764274b0e45d3f is the hidden flag in this task.

Level 3

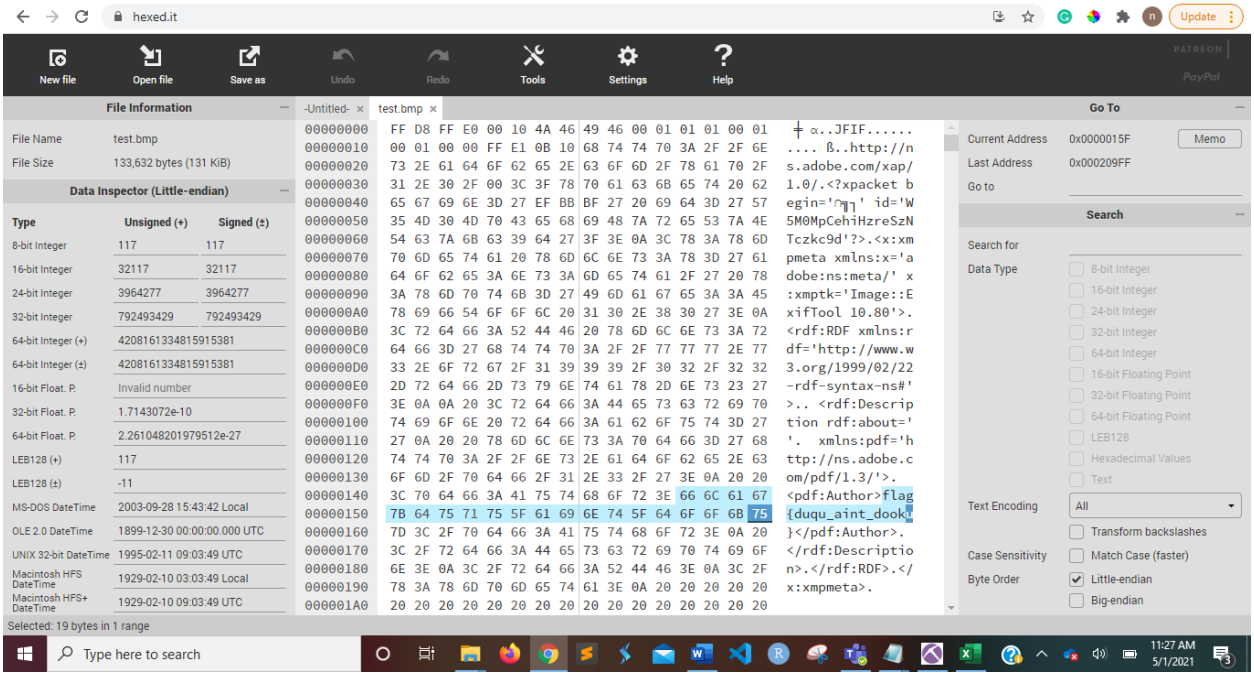
As shown in the screenshot, I attempted to open the bmp file in Notepad text editor.



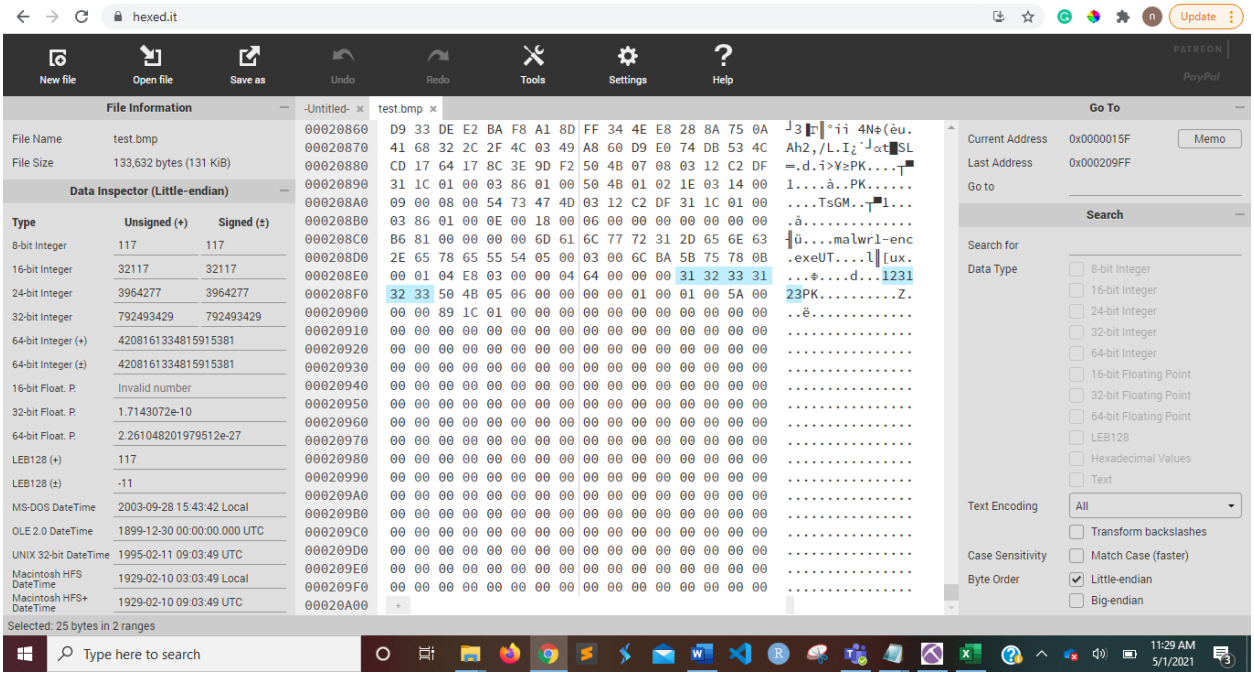
This bmp file has a flag concealed within it. **{flagduqu aint dooku}** is the secret flag.

Level 4

The secret password from the test.bmp file must be discovered in this task. I also found the same flag in the test.bmp file when I opened it in an online hex-editor. Take a look at the following picture.



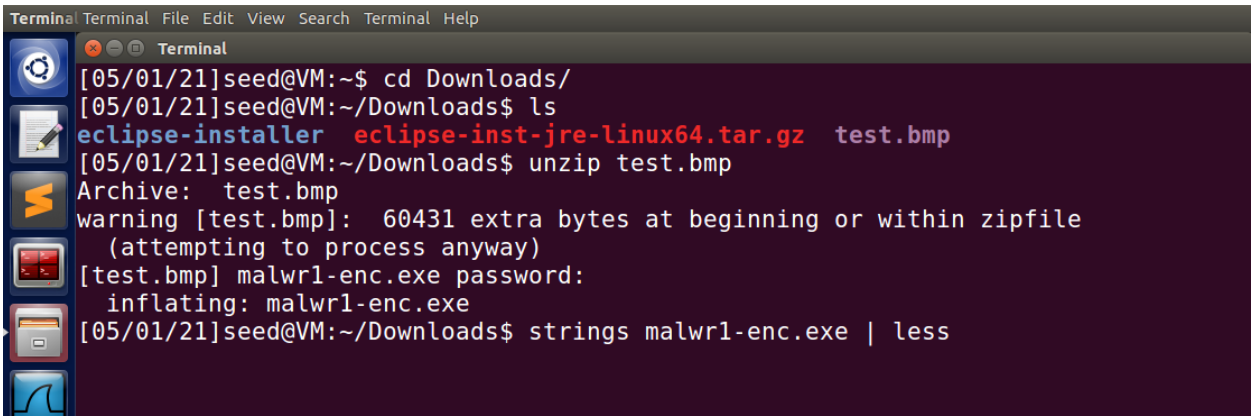
To find the password, I scrolled through the entire file until I found it. Malwr1-enc.exe is the file name. I deduced that the highlighted text is the password based on this hints. Take a look at the following picture.



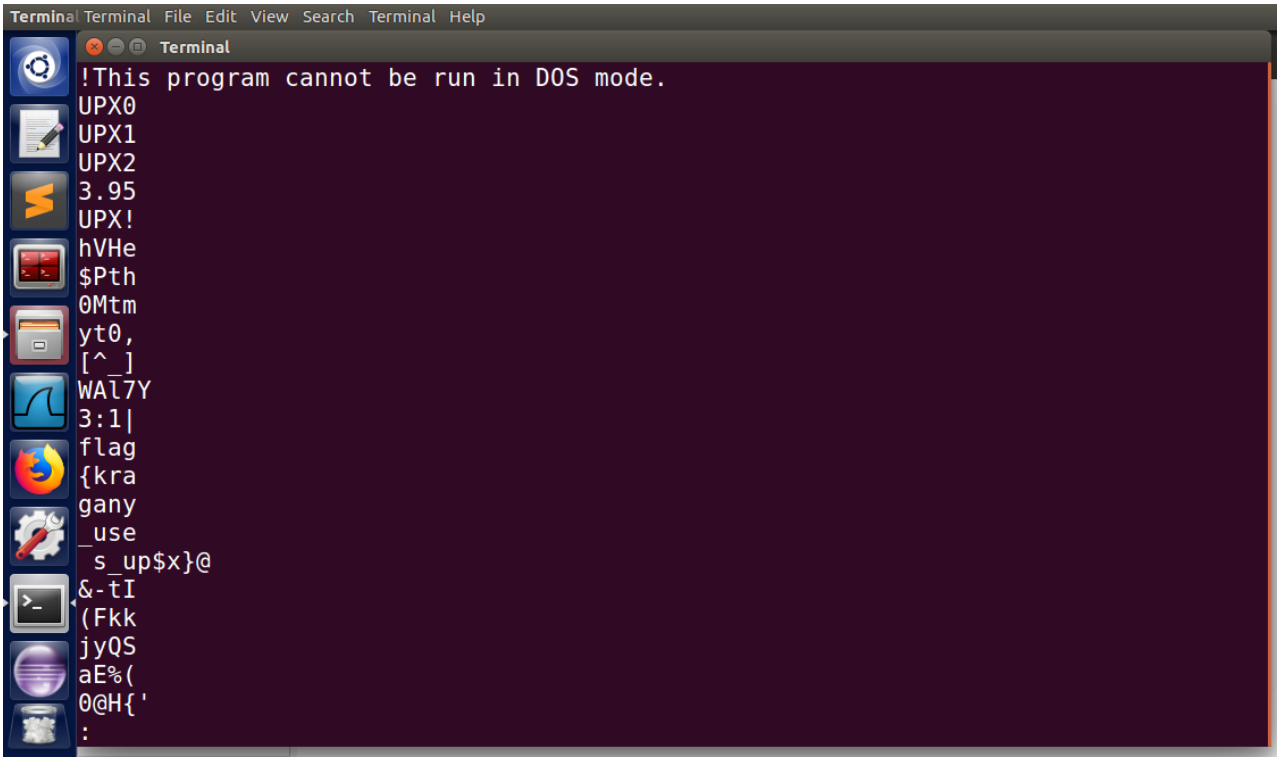
The password **123123** is the task's secret flag.

Level 5

The final malware flag is said to be in the file extracted in this task. We attempt to unzip the test.bmp file using the password we discovered in the previous stage.



Malwr1-enc.exe prompts us that the file is inflating. As a result, the extracted file is used, and the strings command is used to open it.

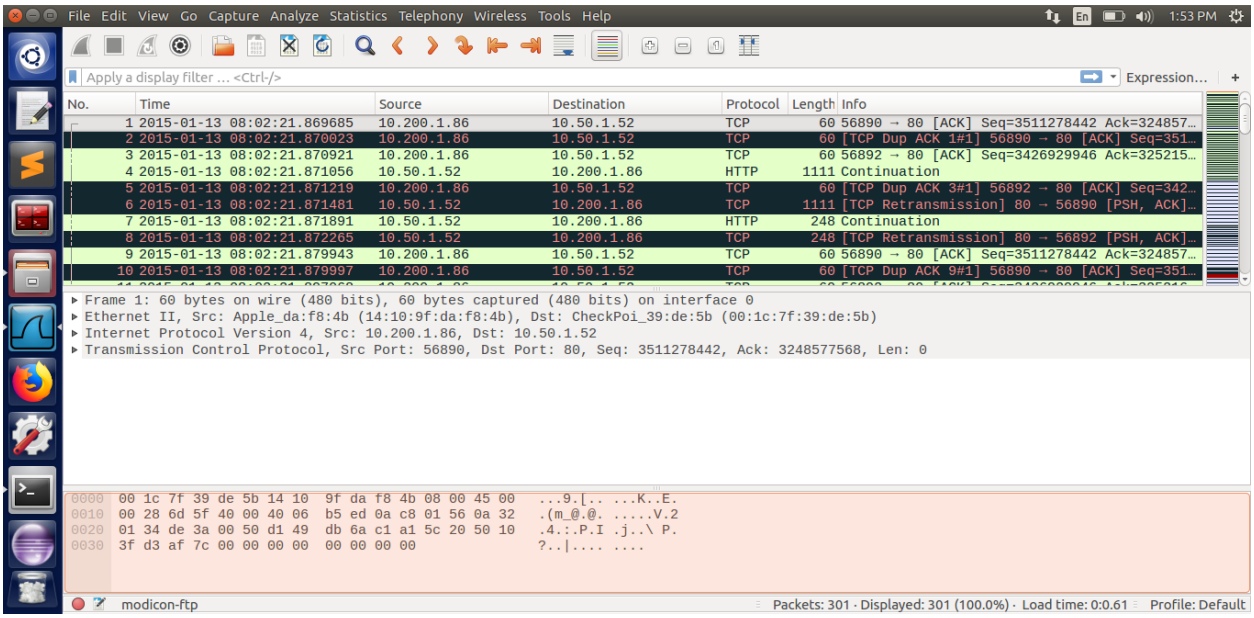


In that exe file, we instantly notice a flag. According to the screenshot, the secret flag is **{kragany_use s_up$x}**

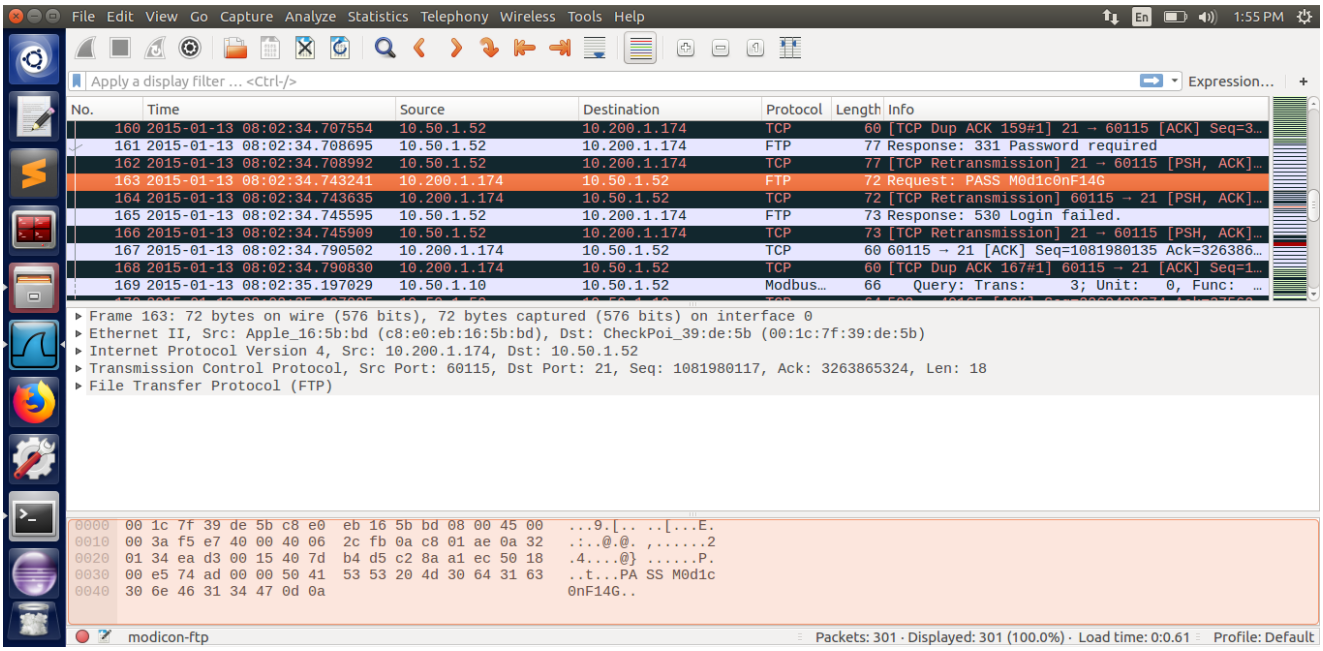
Network Capture

Level 1 – Modicon-FTP

The FTP password used in the failed login attempt must be found in this task. A pcap file was given to us. Wireshark will help us figure out why a login attempt failed. In my virtual machine, I installed Wireshark and imported the provided pcap file.

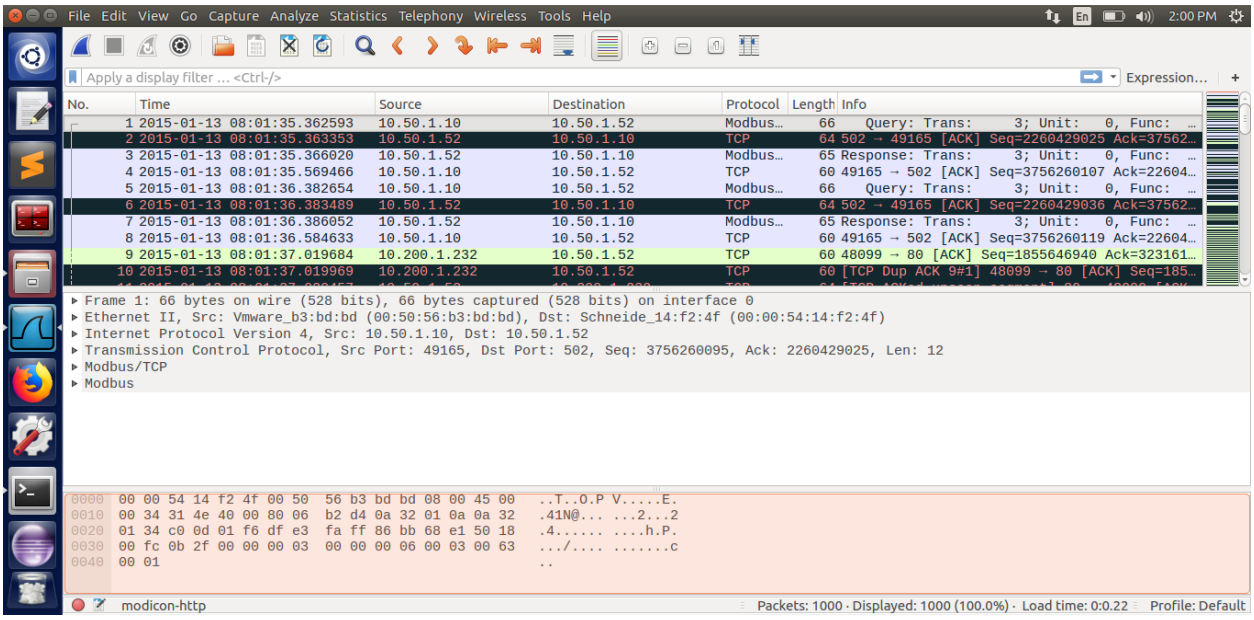


We order the records by protocol used since we need to find the FTP password. We will see the password clearly written in the Info column if we scroll down. M0d1c0nF14G is the password, as seen in the screenshot.

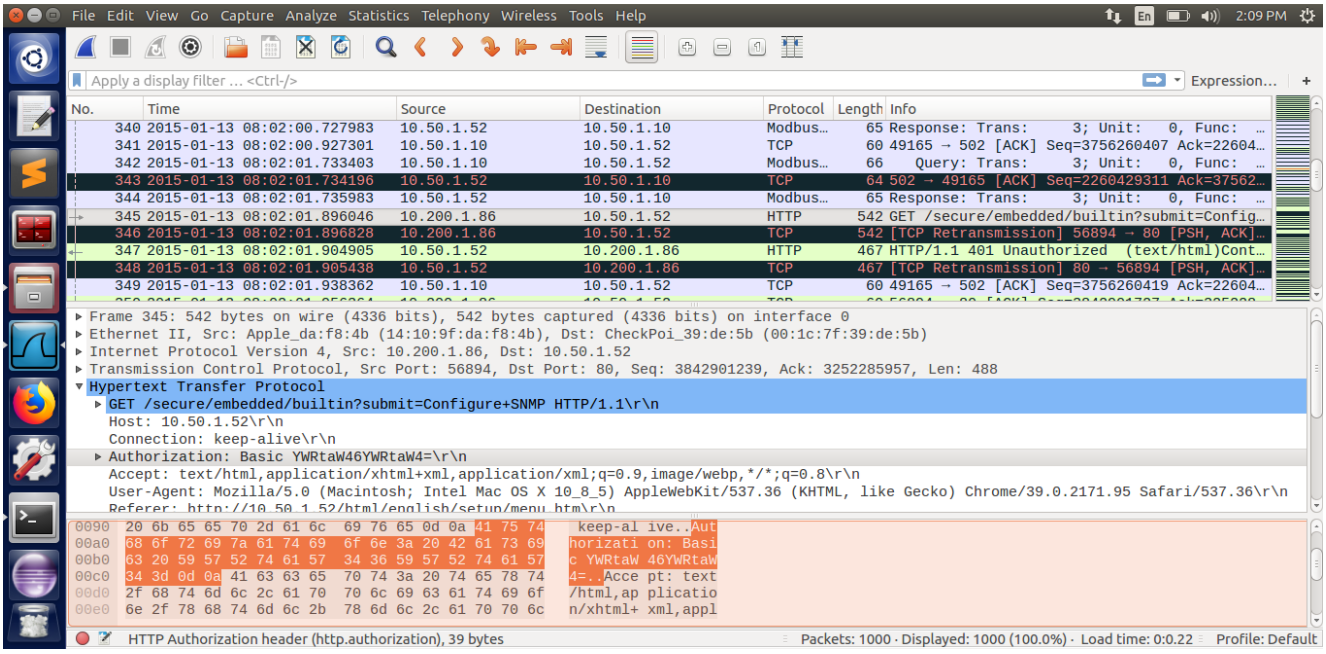


Level 2 – Modicon-HTTP

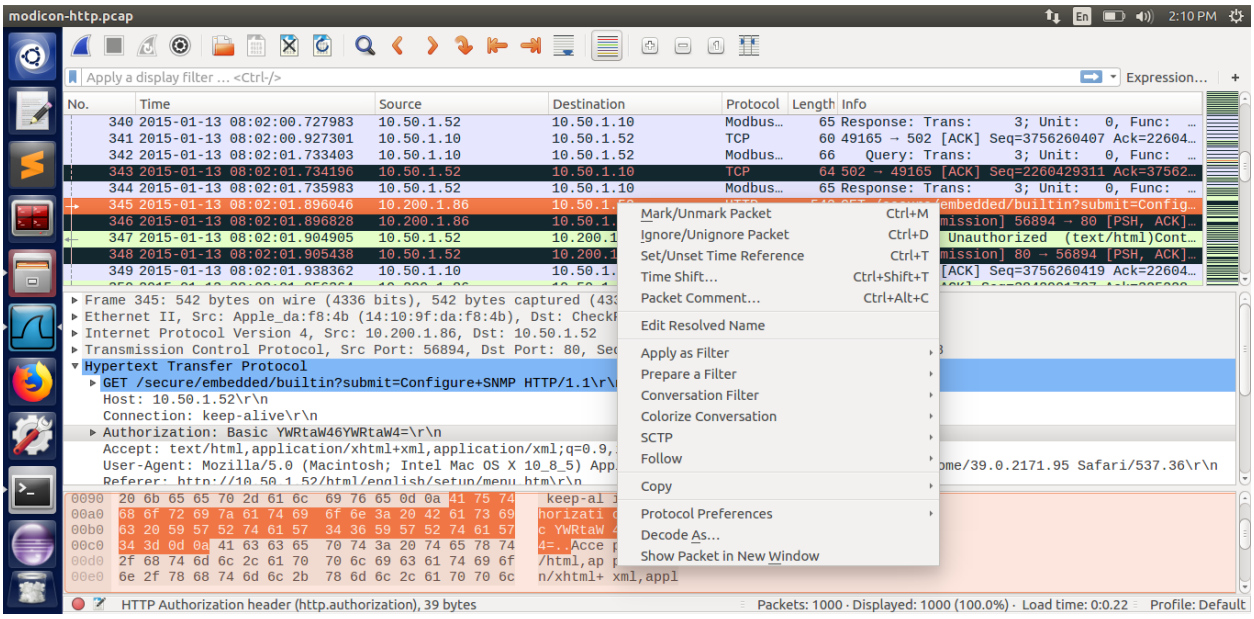
We must try to find out the HTTP Basic Authorization password that was used in a failed user login attempt in this mission. A pcap file was given to us. By importing the pcap file given, we can use Wireshark to find the password.

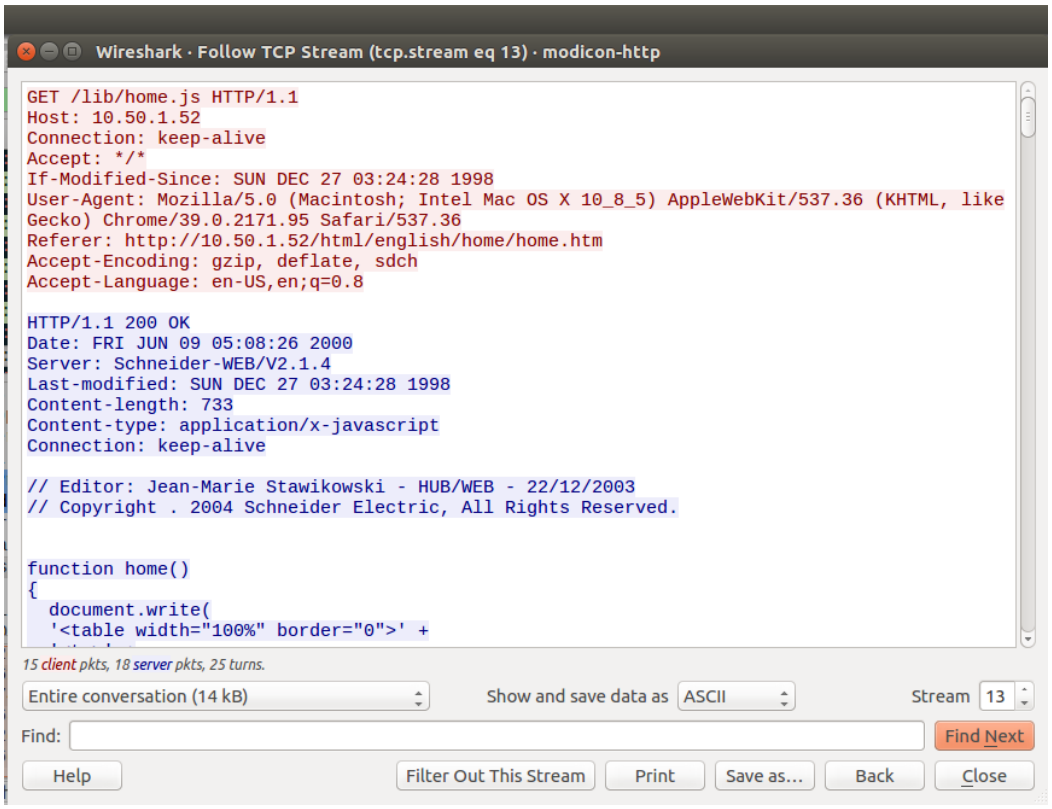


We order the records using the HTTP protocol since we're looking for the HTTP password. We will see 'Unauthorized' written in the Info column if we scroll over. We see a GET request with the password Authorization Basic YWRtaW46YWRtaW4=\r\n



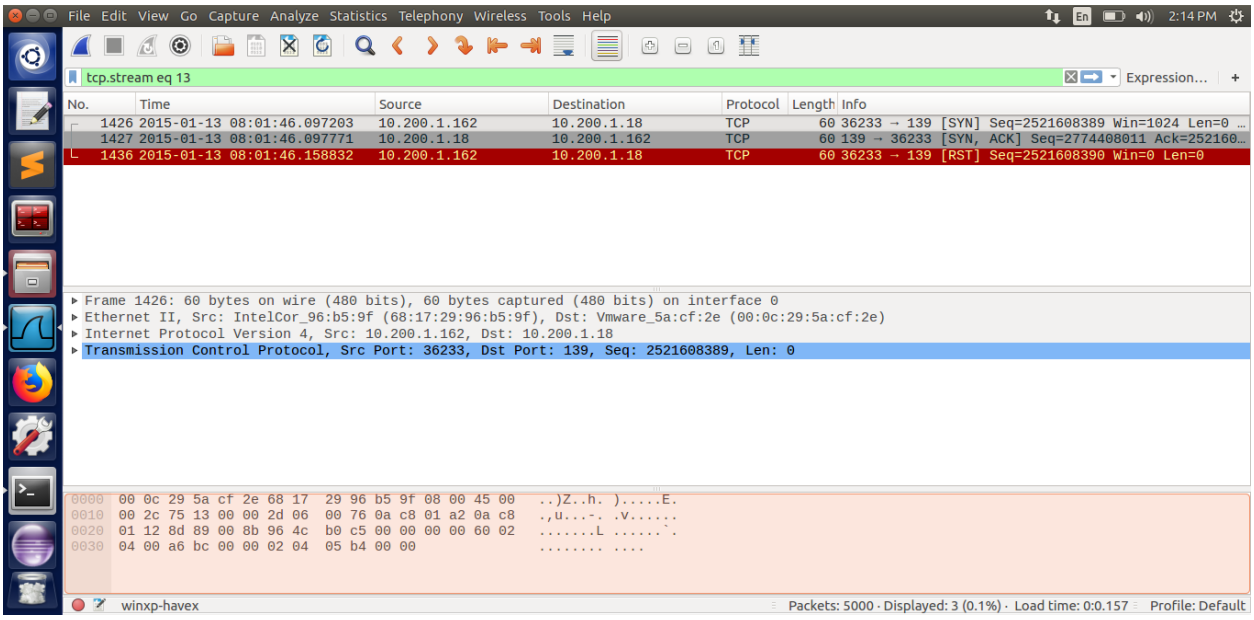
I watched a number of YouTube videos to learn about the Wireshark choices. When we right-click on that record and select Follow->TCP Stream, a new window appears with a failed login attempt highlighted in red when we check for the password YWRt. It contains information such as Authorization: Basic YWRtaW46YWRtaW4=



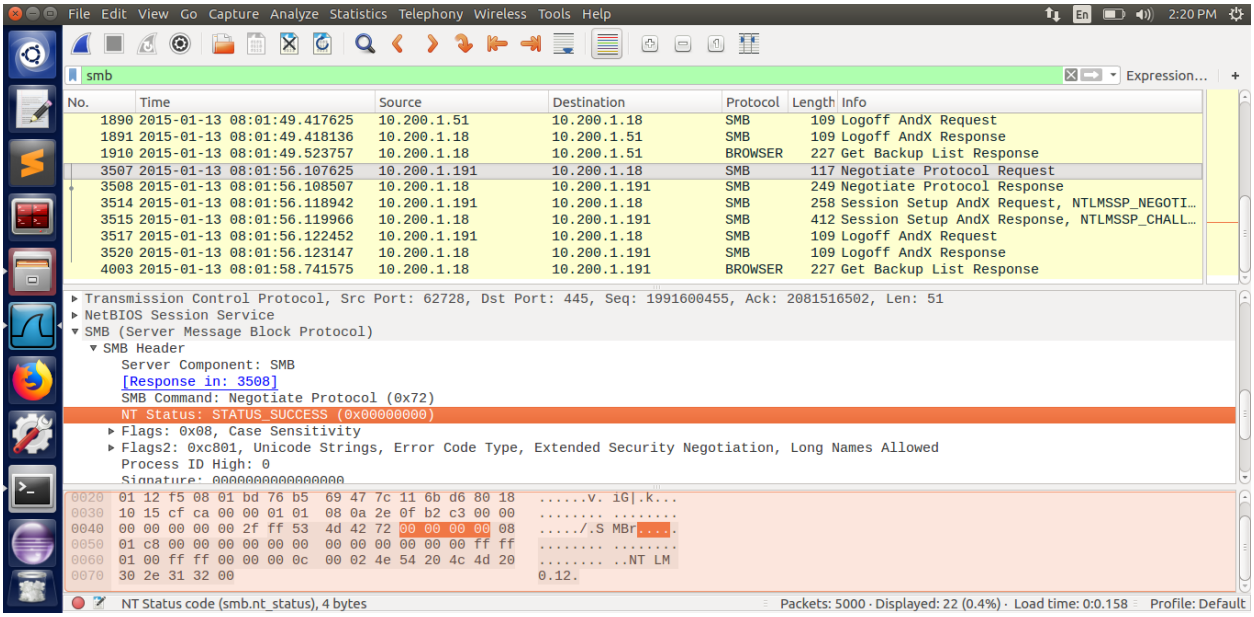


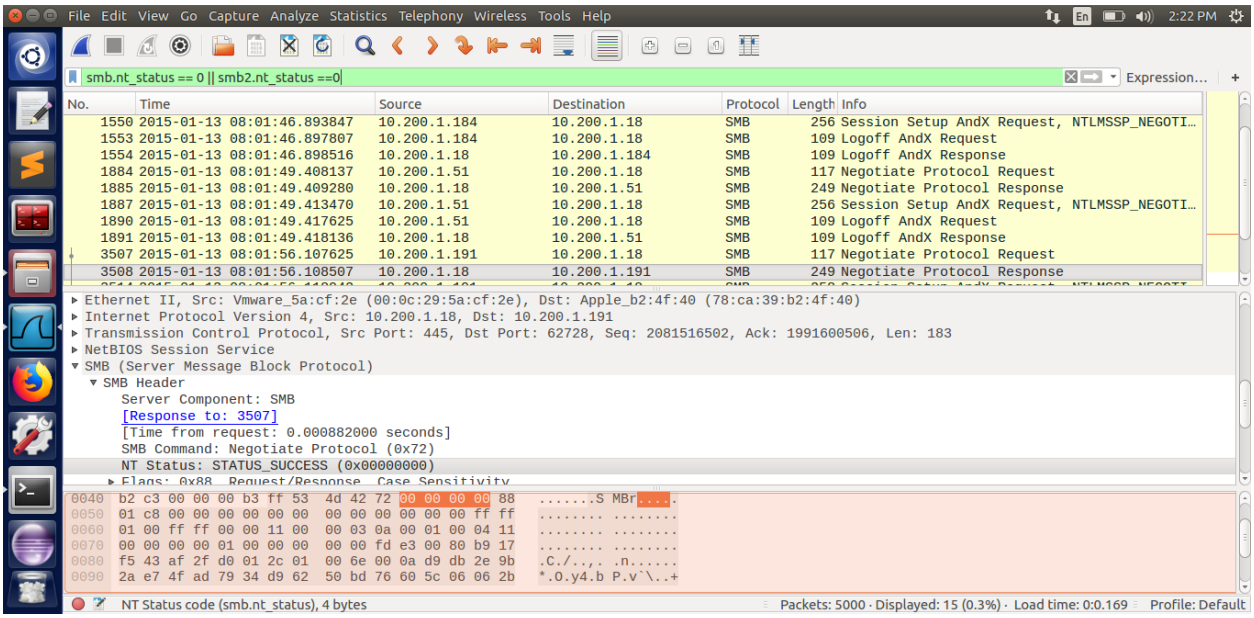
Level 3 – WinXP-HAVEX

We must locate the HAVEX malware IOC in this mission. A pcap file has been given to us. Wireshark can be used to find out by importing the pcap file given.



Backdoor hazard WinXP-HAVEX This threat is detected and removed by Windows Defender. A malicious hacker may gain unauthorized access to and control of your computer as a result of this attack. Unauthorized access to a remote computer is possible thanks to the SMB protocol. As a result, we use the SMB protocol to order the data.



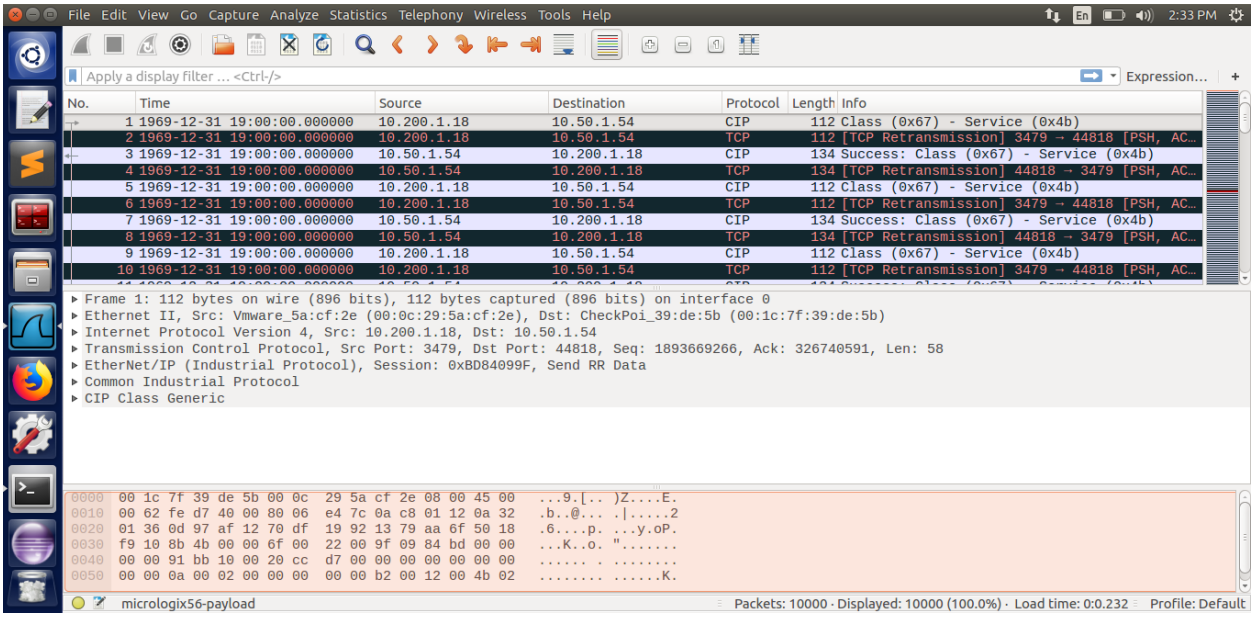


```
tocol  Lengt Info
OWSER   227 Get Backup List Response
3       117 Negotiate Protocol Request
3       249 Negotiate Protocol Response
3       256 Session Setup AndX Request, NTLMSSP_NEGOTIATE
3       412 Session Setup AndX Response, NTLMSSP_CHALLENGE, Error:
3       109 Logoff AndX Request
3       109 Logoff AndX Response
OWSER   227 Get Backup List Response
3       117 Negotiate Protocol Request
```

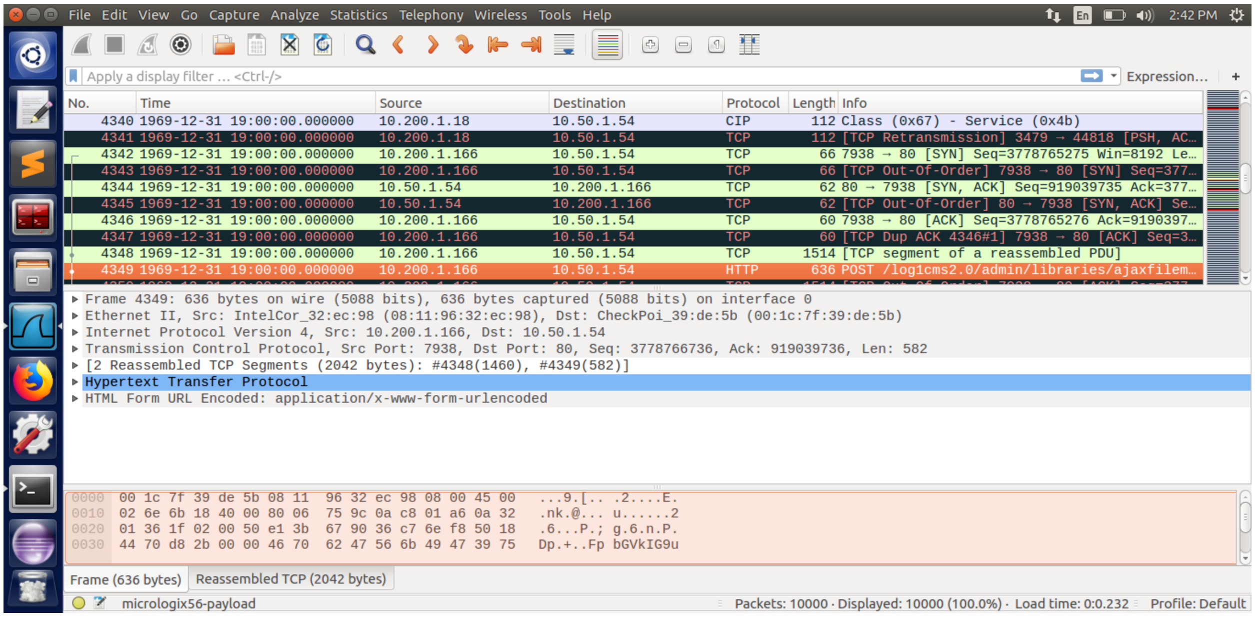
STATUS SUCCESS is the NT Status (Negotiate Status) that we see. This indicates that unauthorized access was granted successfully. The HAVEX malware IOC can be seen in the screenshot.

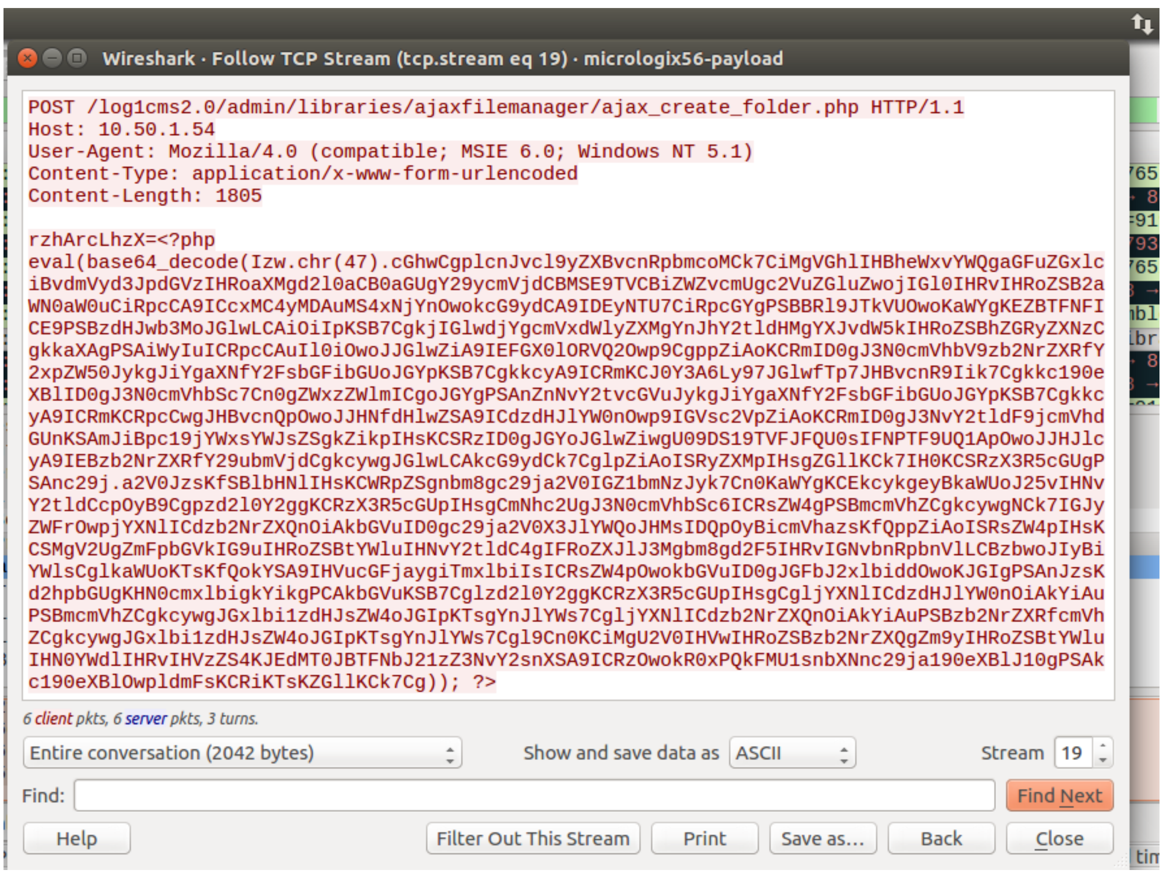
Level 4 – MICROLOGIX56

The aim of this task is to determine the backdoor port number that an intruder attempted to mount. A pcap file has been sent to us. Wireshark can be used to find out by importing the pcap file.

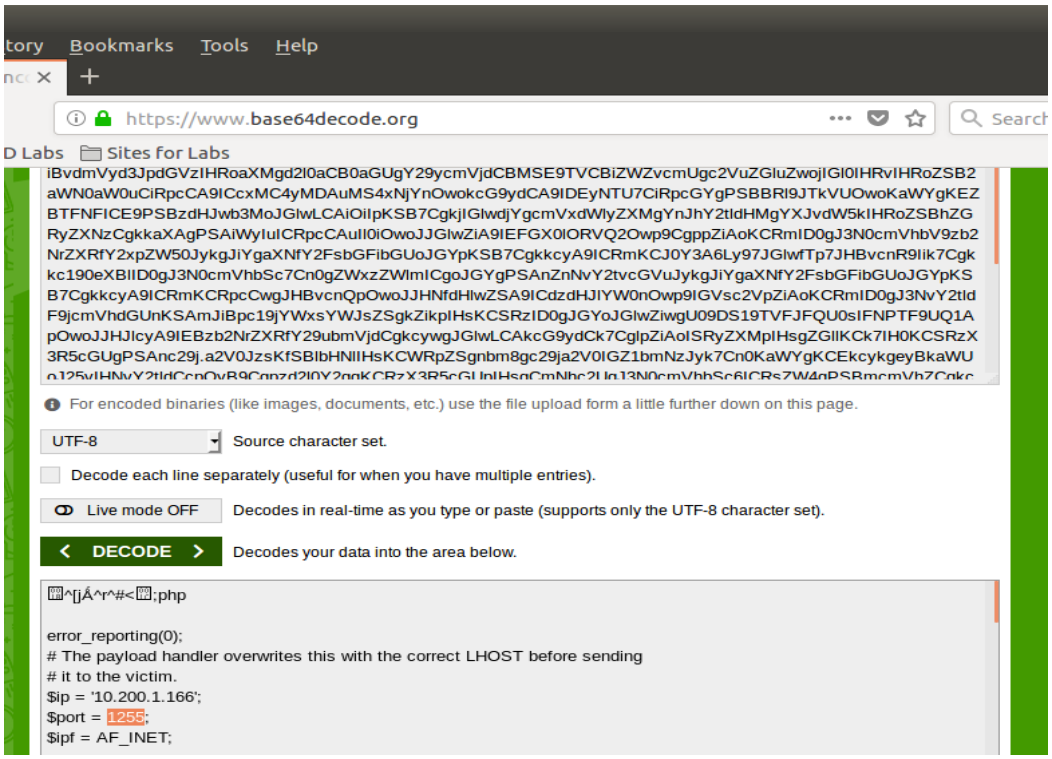


I noticed HTTP protocol when looking through the documents. When I looked at the TCP stream, I noticed a PHP script that shouldn't be there.





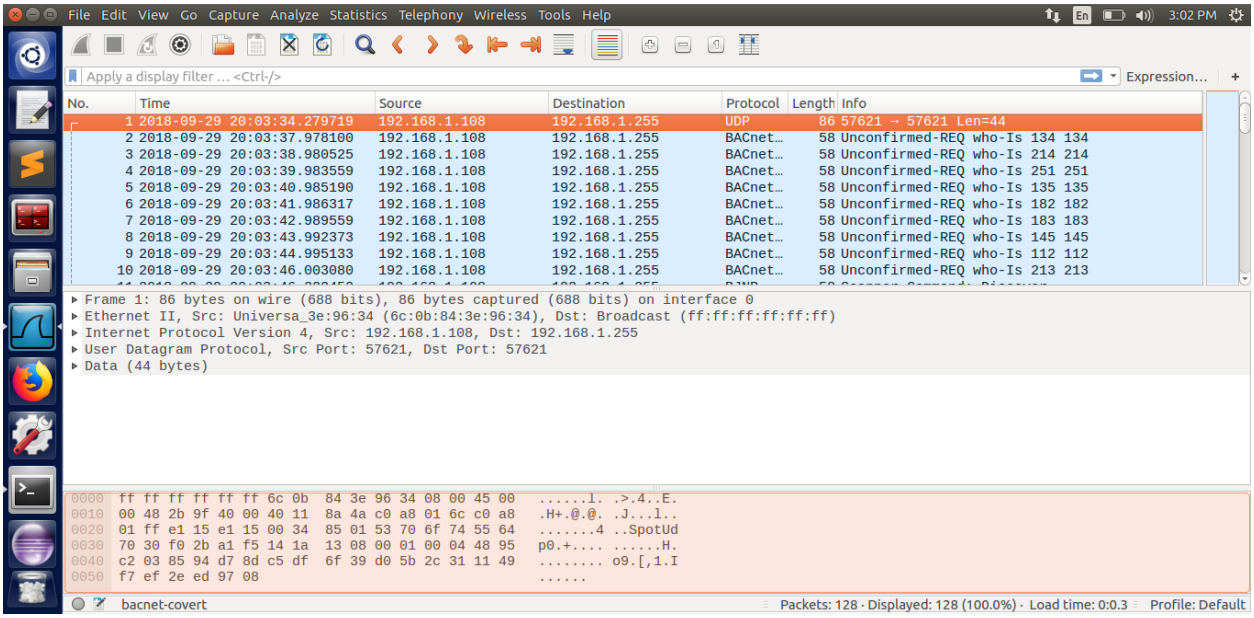
I copied the PHP script and decoded it with a base64 decoder.



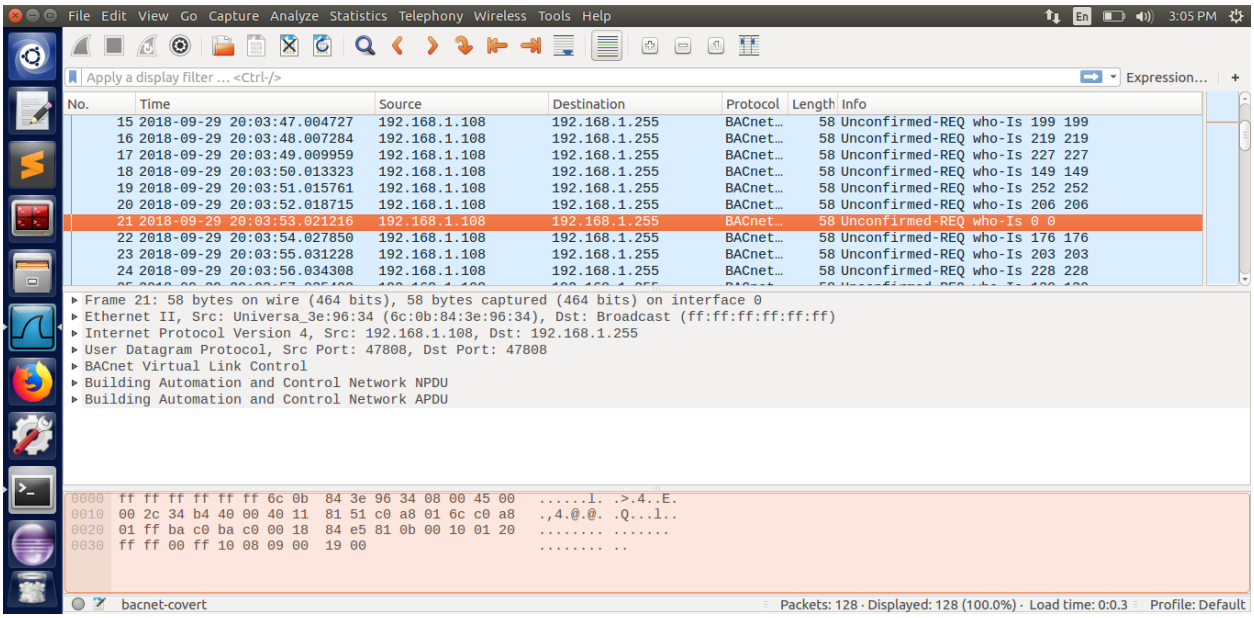
When I tried to decipher the script, I discovered the port 1255, as seen in the screenshot. The backdoor that an intruder tried to mount had a port number of 1255.

Level 5 – BACNET-COVERT

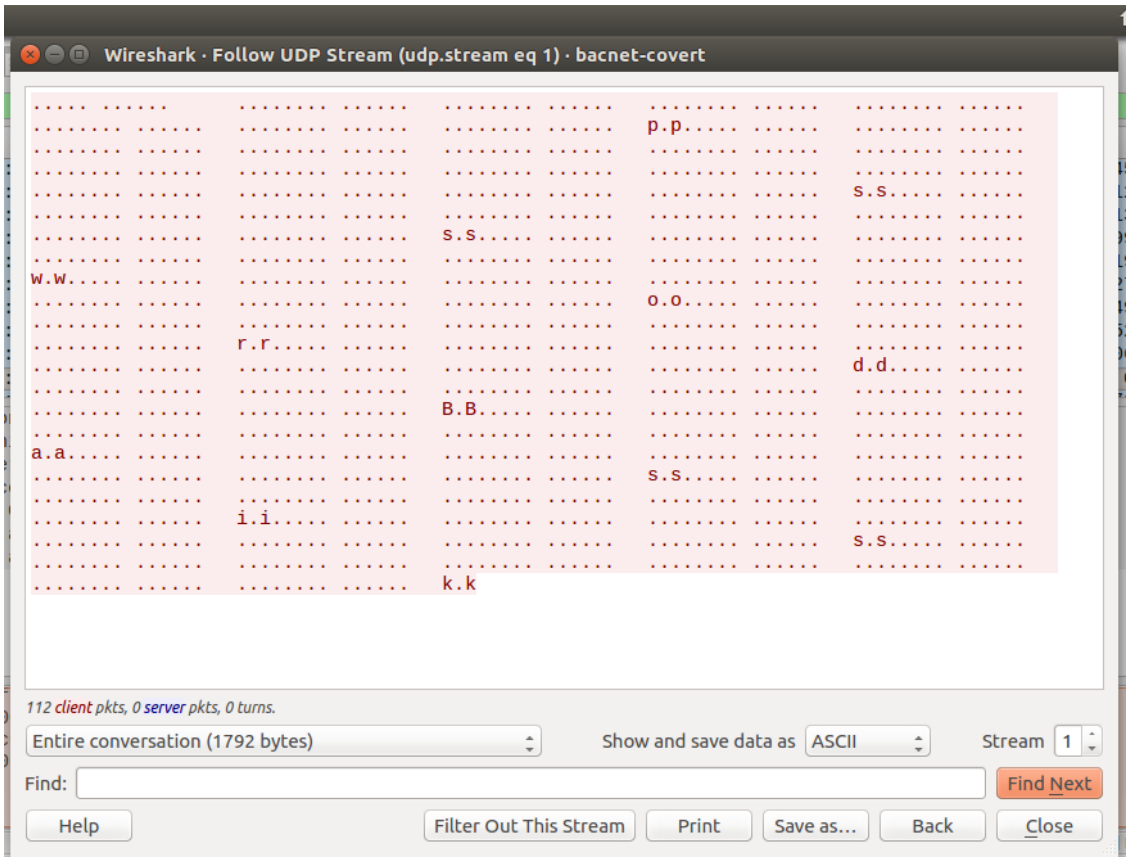
The aim of this challenge is to track down a covert message sent over the BACnet network. A pcap file was given to us. By importing the pcap file given, we can use Wireshark to find out. After that, we use the BACnet protocol to sort the data.



The BACnet protocol is a collection of rules that regulate the exchange of data over a computer network. It includes anything from the type of cable to how to shape a specific request or command in a standard manner.



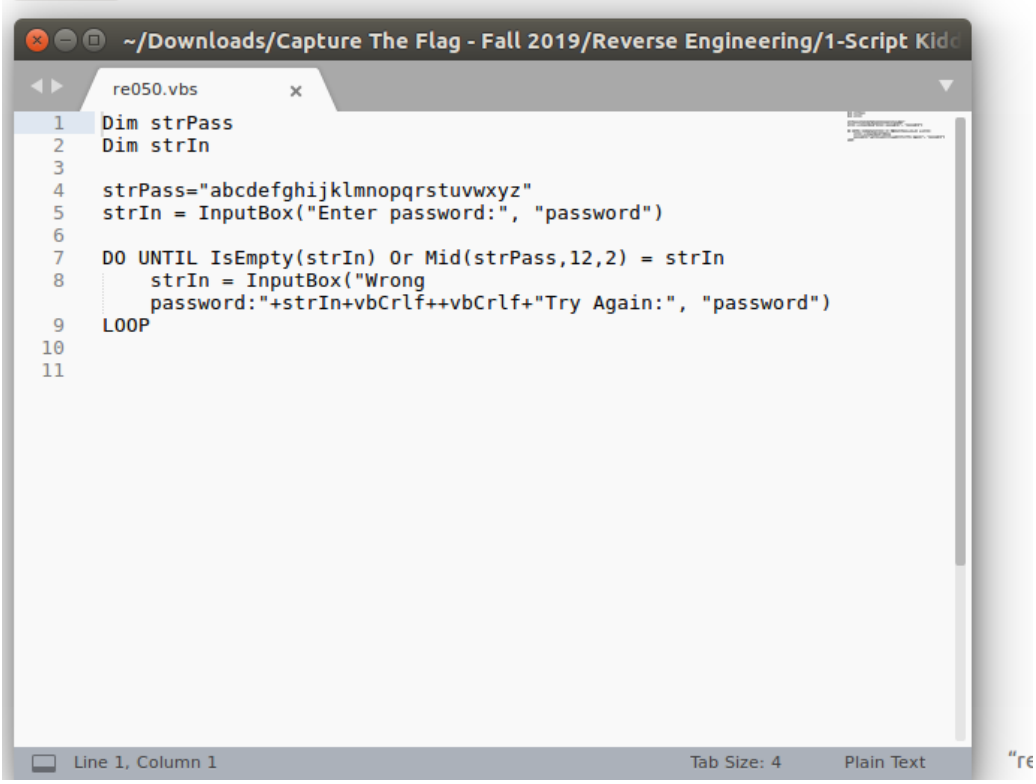
The message “Unconfirmed-REQ who-Is 0 0” is found in the Info column. This clearly indicates that it is an intruder. We get the covert message **passwordBasisk** if we open the UDP stream. The covert message is the coded flag.



Reverse Engineering

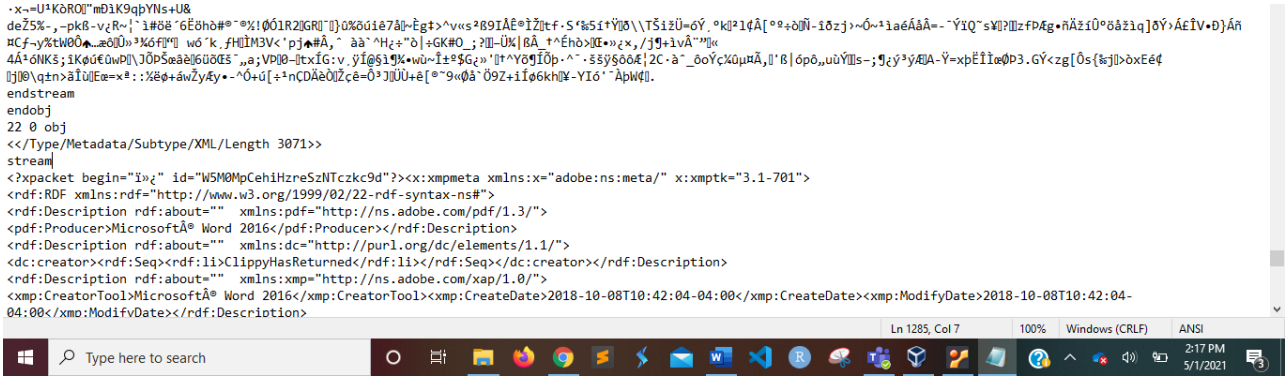
Level 1 – Script Kiddie

We must decipher the code and deduce the password in this task. We can deduce from the code that the password begins at the 12th position of strPass's contents and ends after two positions. lm is ranked 12th and 13th. The password for the flag is lm.



Level 2:- The PDF Your Parents Warned You About I

In this task, we must determine who produced the PDF file that has been provided. I used the Photos program to open the PDF. I was unable to locate something. Then I saved it as a text file in Notepad. I discovered the tag maker when reading through the file.

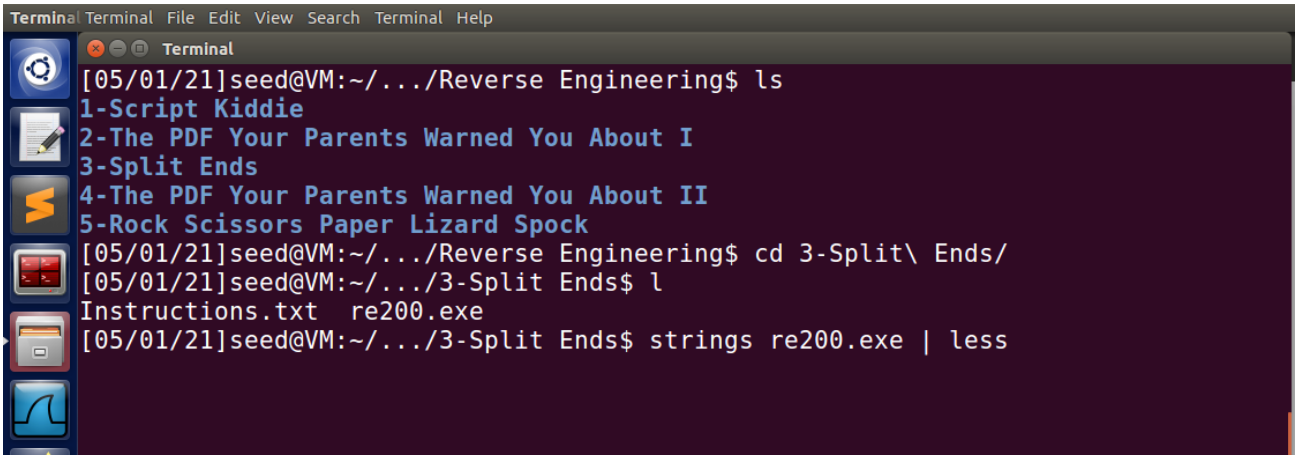


ClippyHasReturned was the creator name on the creator tag.

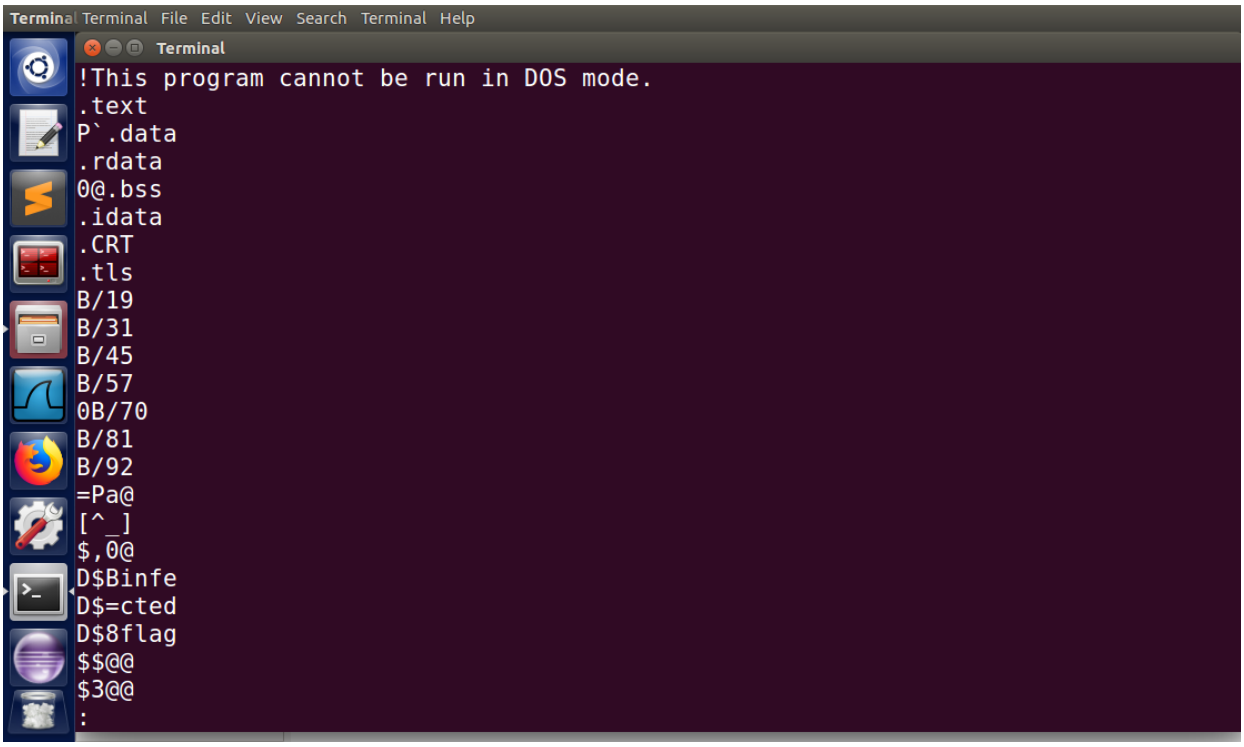
This PDF file was created by a user called "ClippyHasReturned."

Level 3 – Split Ends

We must locate the compiled flag in this task. The executable file is sent to us. I attempted to open Notepad in text mode. It didn't work out. I couldn't come up with anything. To find the flag, I used the strings command.

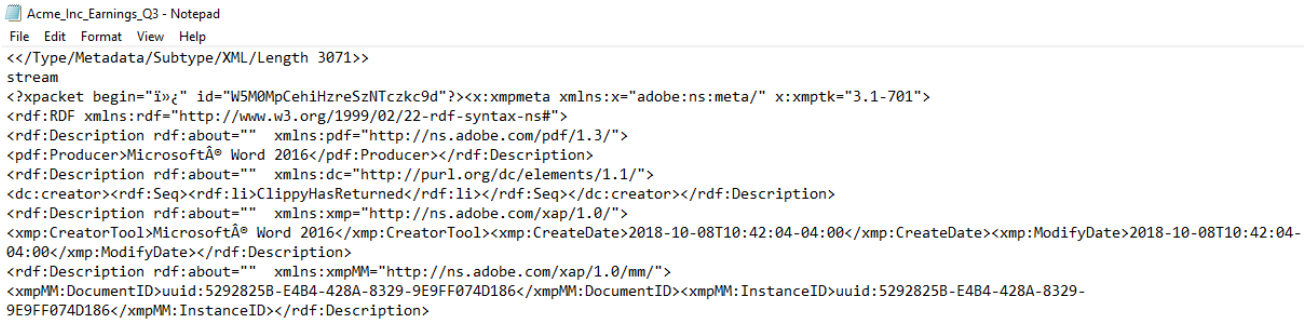


We can see the text "infected flag" towards the end of the strings break into three sections if we keep going through the file. This is similar to the clue we received from SPLIT ENDS. As shown in the screenshot, the compile flag is infected.

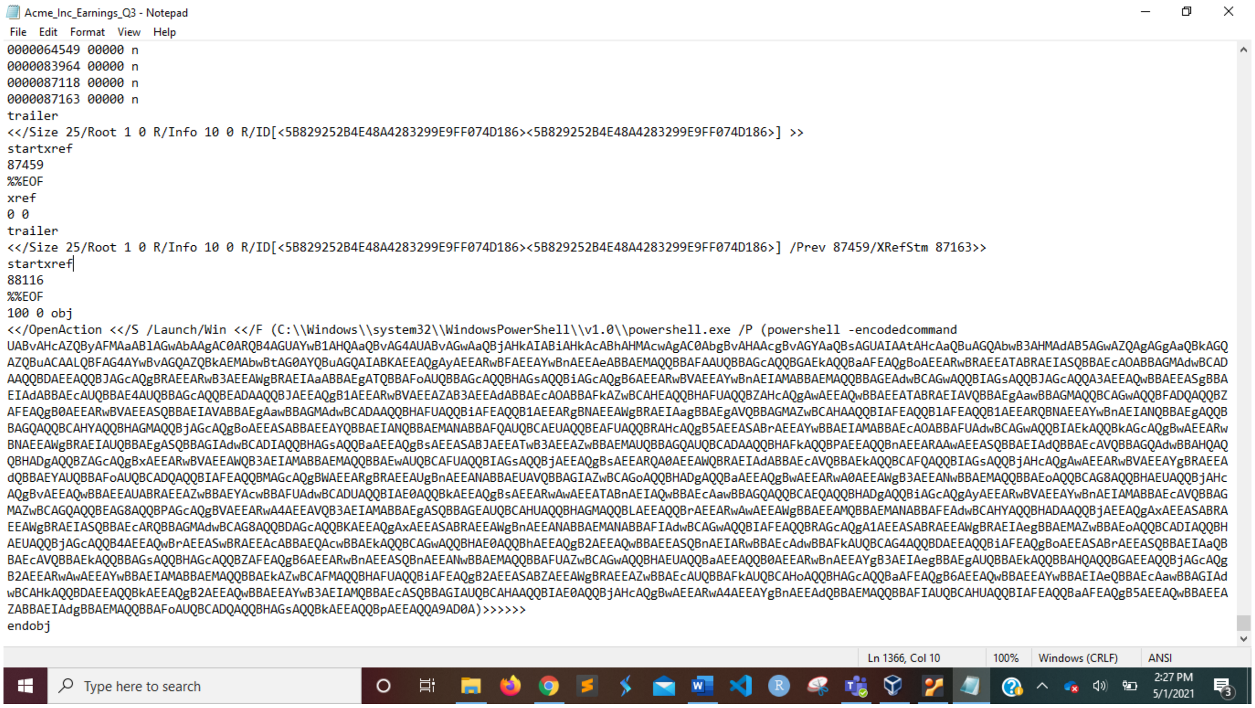


Level 4 – The PDF Your Parents Warned You About II

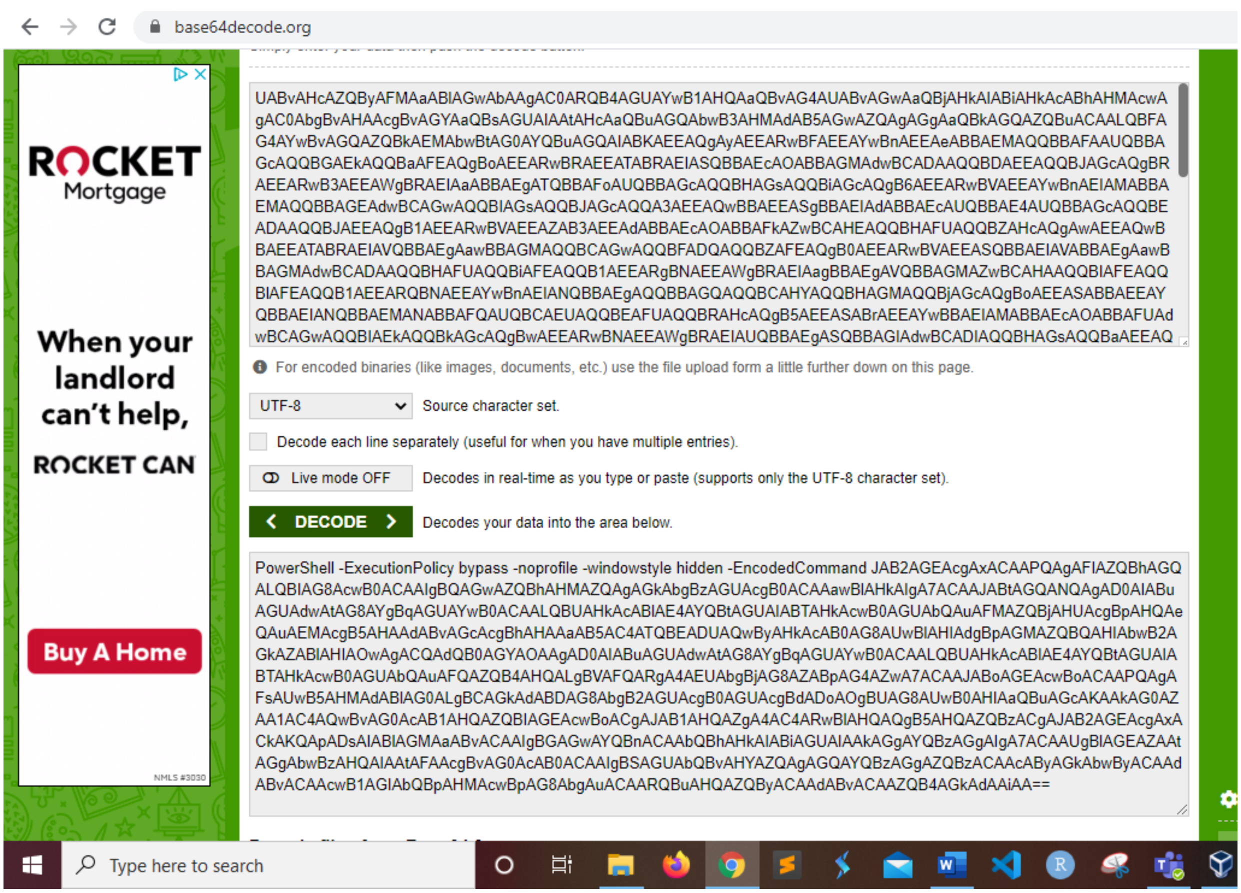
We must find the hidden flag from a PDF file in this task. When we open the PDF in a text editor such as Notepad, we can see an id field with the value W5M0MpCehiHzreSzNTczkc9d, as seen in the screenshot.



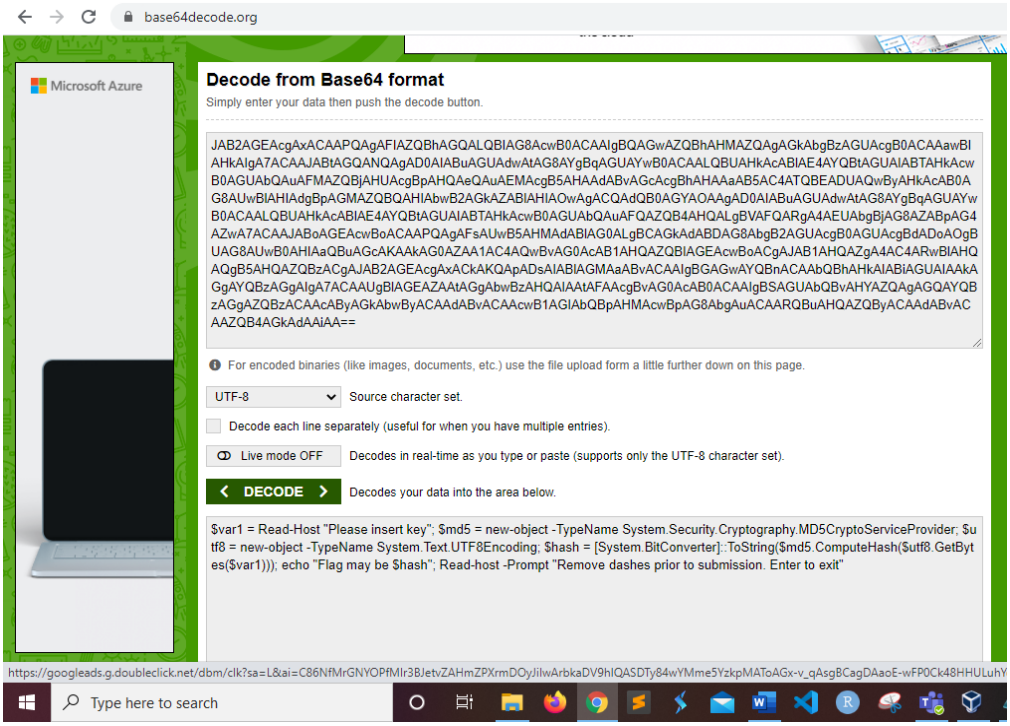
We see a powershell path and a huge string near the end of this file, as seen in the screenshot below.



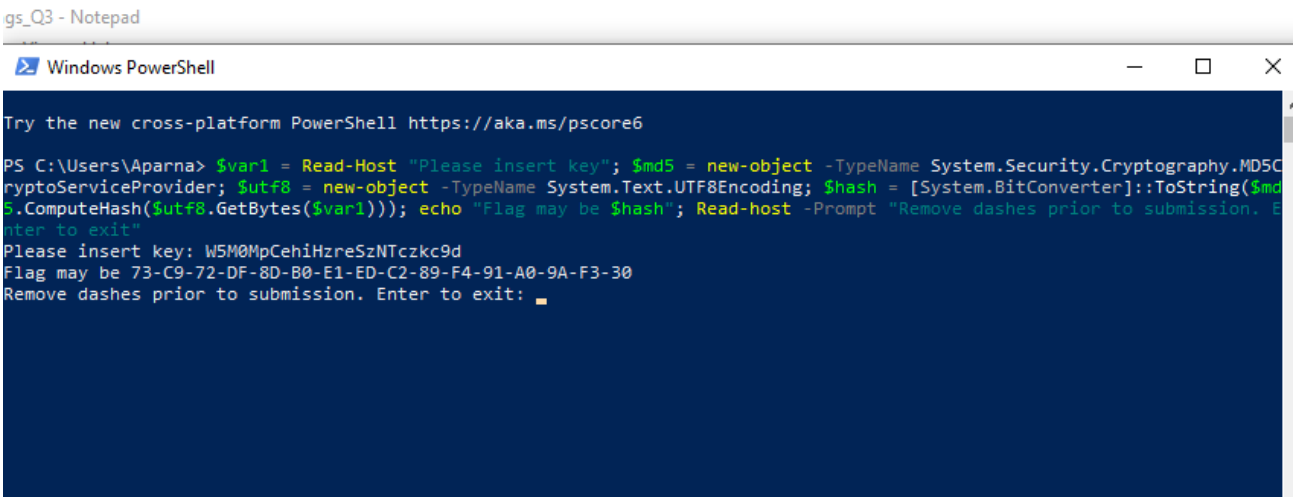
We use a base64 decoder to try to decode this string.



We use the base64 decoder to decode the obtained string again, and we get a new string, as seen in the screenshot.



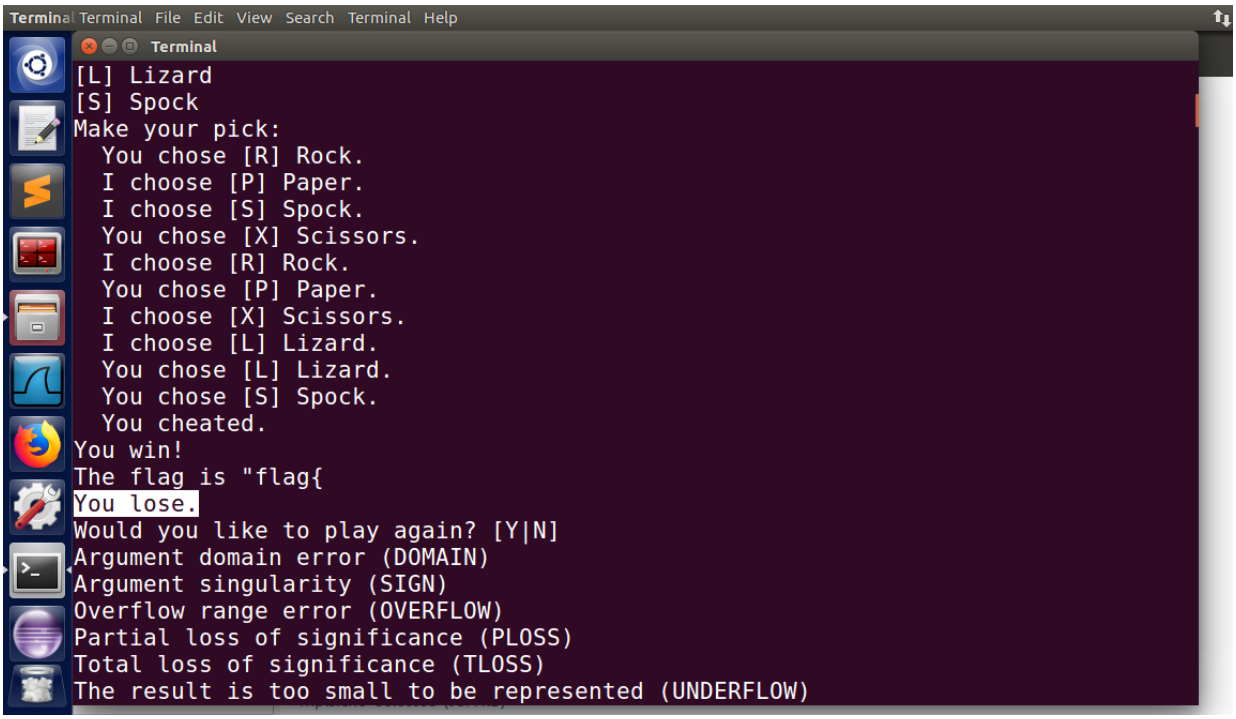
This code is entered into PowerShell. It requests that you insert a key. We use the value W5M0MpCehiHzreSzNTczkc9d from the id field to create this key.



We get the hidden flag 73-C9-72-DF-8D-B0-E1-ED-C2-89-F4-91-A0-9A-F3-30 if we enter this value, as shown in the screenshot. It becomes 73C972DF8DB0E1EDC289F491A09AF330 after we delete the dashes as instructed.

Level 5 – Rock Scissors Paper Lizard Spock

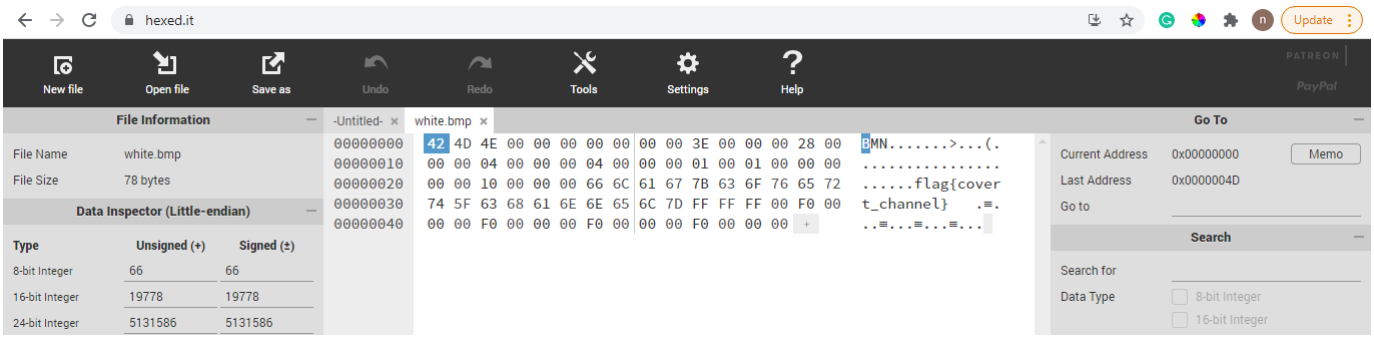
Using the strings command, the .exe file was opened in the command line. The flag found is: “flag{.}”. Screenshot below:



Steganography

Level 1 – Moo Cow

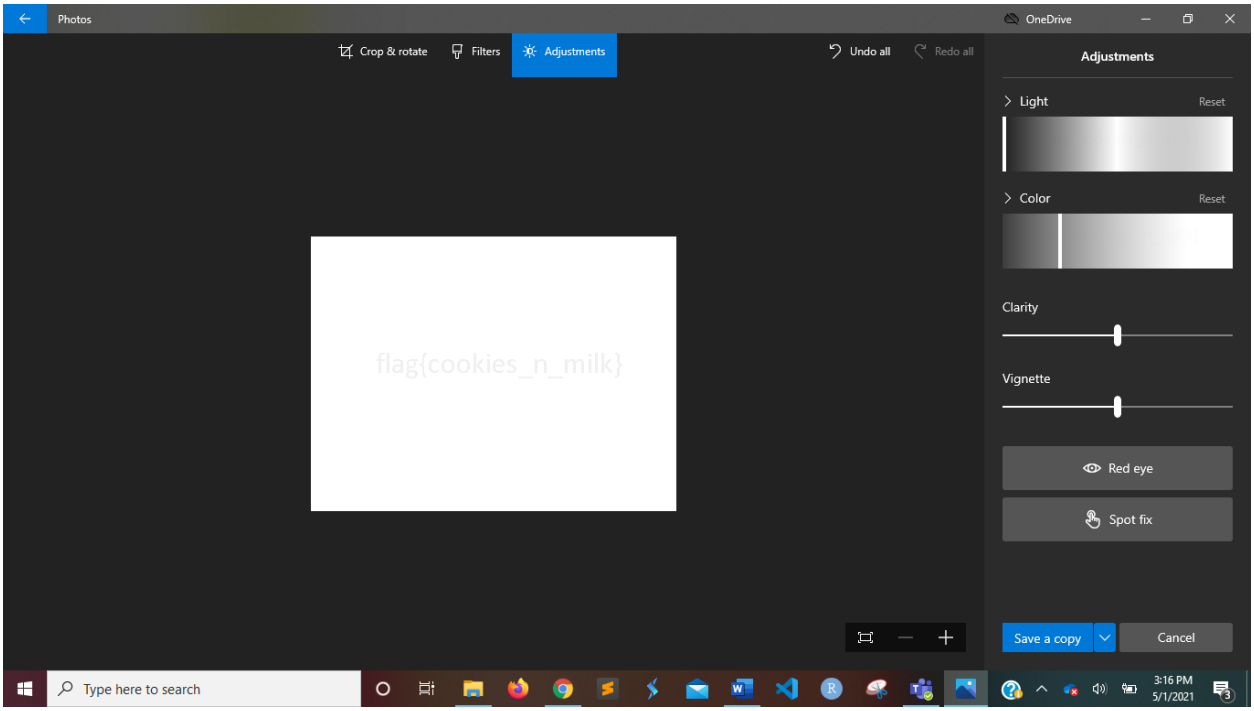
We must locate the flag in the bmp file provided in this task. Using an online hex-editor, I opened the bmp file.



When the bmp file is opened in a hex-editor, it displays **flag{covert channel},** as seen in the screenshot. This is the flag we were looking for.

Level 2 – Milk Run

We must find the white flag in the picture of white cows in a snow field in this task.

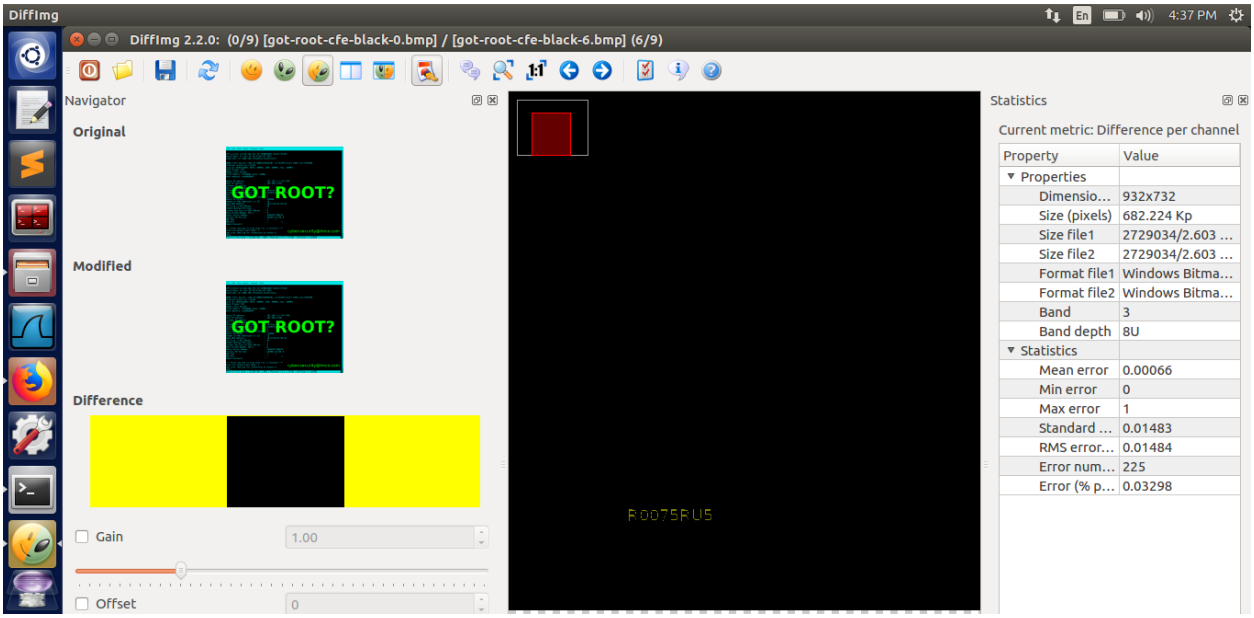


A bmp file has been issued to us. So I used the Photos application to open the bmp file. I discovered that the flag was white based on the clue. As a result, I tweaked the Light. We can see **flag{cookies_n_milk}** in the screenshot. We were looking for a white flag, and this is it.

Level 3 – Got Root

We must determine which of the provided photos contains the secret password in this mission. The pictures all looked the same when I opened them in the Photos program. As a result, I assumed there would be a discrepancy in the images.

I searched the internet for software that detects image differences. DiffImg program was discovered. This was what I used to compare each picture to the next. I discovered a discrepancy between Image 0 and Image 6 by using a brute force technique.

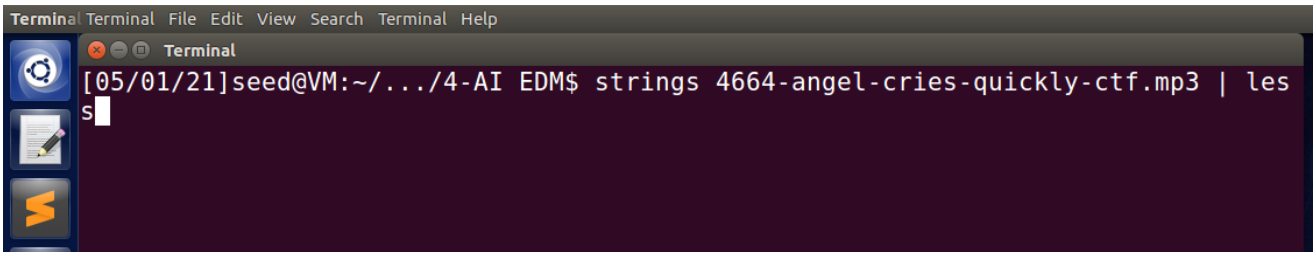


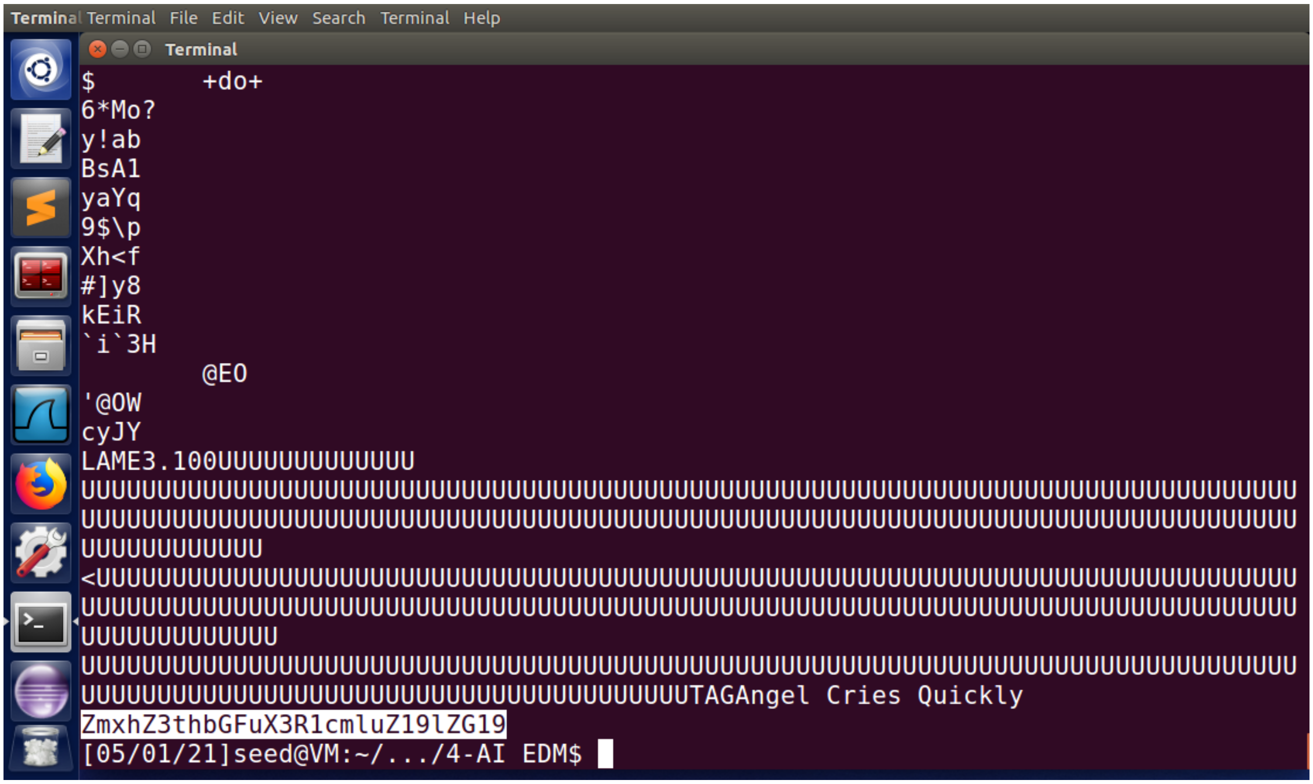
We can see some code written in the picture, as seen in the screenshot. I was able to see the password R0075RUS when I zoomed in. This is the flag we were expected to come across.

Level 4 – AI EDM

We must locate the hidden flag in the provided mp3 file in this mission. I tried listening to it in mp3 format, but all I got was music. As a result, I used Notepad, Photos, and PDF to open the document. The flag was not to be found.

Then I used the command prompt to open with the strings command. I searched the entire file for a flag but couldn't find one. However, at the end of the file, I discovered a string. I looked up the string online and discovered that it is in base64 format.





I attempted to decode the base 64 format online, as seen in the screenshots, and eventually came up with a flag called **flag{alan turing edm}.** We were expected to find a flag like this.