PSP0201 Week 5 Writeup

Group Name: study group

Members

ID	Name	Role
1211101157	Lo Pei Qin	Leader
1211102017	Siow Yee Ceng	Member
1211101534	Tan Chi Lim	Member
1211102835	Chew Ming Yao	Member

Day 16 Help! Where is Santa?

Tools used: Kali Linux, python, firefox

Question 1

We use Nmap to scan for the website and find all the open ports for this website.

```
┌──(1211102017㉿kali)-[~]
└─$ nmap -v 10.10.65.8
```

We found that port 22 was open for ssh and port 80 was open for HTML

```
┌──(1211102017㉿kali)-[~]
└─$ nmap -v 10.10.65.8
Starting Nmap 7.92 ( https://nmap.org ) at 2022-07-05 04:07 EDT
Initiating Ping Scan at 04:07
Scanning 10.10.65.8 [2 ports]
Completed Ping Scan at 04:07, 0.20s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 04:07
Completed Parallel DNS resolution of 1 host. at 04:07, 0.17s elapsed
Initiating Connect Scan at 04:07
Scanning 10.10.65.8 [1000 ports]
Discovered open port 22/tcp on 10.10.65.8
Discovered open port 80/tcp on 10.10.65.8
Stats: 0:00:11 elapsed; 0 hosts completed (1 up), 1 undergoing Connect Scan
Connect Scan Timing: About 79.50% done; ETC: 04:07 (0:00:03 remaining)
Stats: 0:00:11 elapsed; 0 hosts completed (1 up), 1 undergoing Connect Scan
Connect Scan Timing: About 80.27% done; ETC: 04:07 (0:00:03 remaining)
Stats: 0:00:11 elapsed; 0 hosts completed (1 up), 1 undergoing Connect Scan
Connect Scan Timing: About 81.30% done; ETC: 04:07 (0:00:03 remaining)
Completed Connect Scan at 04:07, 19.59s elapsed (1000 total ports)
Nmap scan report for 10.10.65.8
Host is up (0.20s latency).
Not shown: 998 closed tcp ports (conn-refused)
PORT   STATE SERVICE
22/tcp open  ssh
80/tcp open  http
```

Question 2

Open the terminal, copy and paste the source code from day 15 and replace it with the URL into the required URL.

```
┌──(1211102017㉿kali)-[~]
└─$ bpython3
bpython version 0.22.1 on top of Python 3.10.4 /usr/bin/python3
>>> import requests
>>> from bs4 import BeautifulSoup
>>>
>>>
>>> html = requests.get('http://10.10.173.123/')
>>>
>>>
>>> soup = BeautifulSoup(html.text, "lxml")
>>>
>>> links = soup.find_all('a')
>>> for link in links:
...     print(link)
```

We found that there was a hidden URL called API inside the webpage

```
1211102017@kali: ~
File  Actions  Edit  View  Help
<a href="https://tryhackme.com">Santa</a>
<a href="https://tryhackme.com">humans</a>
<a href="https://tryhackme.com">click</a>
<a href="https://tryhackme.com">Python</a>
<a href="https://tryhackme.com">notice</a>
<a href="https://tryhackme.com">Skidy</a>
<a href="https://tryhackme.com">TryHackMe</a>
<a href="https://tryhackme.com">man</a>
<a href="https://tryhackme.com">613</a>
<a href="https://tryhackme.com">jumper</a>
<a href="#">Lorem ipsum dolor sit amet</a>
<a href="#">Vestibulum errato isse</a>
<a href="#">Lorem ipsum dolor sit amet</a>
<a href="#">Aisia caisia</a>
<a href="#">Murphy's law</a>
<a href="#">Flimsy Lavenrock</a>
<a href="#">Maven Mousie Lavender</a>
<a href="#">Labore et dolore magna aliqua</a>
<a href="#">Kanban airis sum eschelor</a>
<a href="http://machine_ip/api/api_key">Modular modern free</a>
<a href="#">The king of clubs</a>
<a href="#">The Discovery Dissipation</a>
<a href="#">Course Correction</a>
<a href="#">Better Angels</a>
<a href="#">Objects in space</a>
<a href="#">Playing cards with coyote</a>
<a href="#">Goodbye Yellow Brick Road</a>
```

Question 3

We use this python code to determine which API was the correct API for this website

```
┌──(1211102017㉿kali)-[~]
└─$ bpython3
bpython version 0.22.1 on top of Python 3.10.4 /usr/bin/python3
>>> import requests
>>>
>>> api_key = 1
>>>
>>> for i in range(api_key, 101):
...     html = requests.get(f'http://10.10.65.8:80/api/{api_key}')
...     print(html.text)
...     api_key += 1
```

We found that with API 57 it's shown 3 places and we believe that was the places where Santa at

```
{"item_id":42,"q":"Error. Key not valid!"}
{"item_id":43,"q":"Error. Key not valid!"}
{"item_id":44,"q":"Error. Key not valid!"}
{"item_id":45,"q":"Error. Key not valid!"}
{"item_id":46,"q":"Error. Key not valid!"}
{"item_id":47,"q":"Error. Key not valid!"}
{"item_id":48,"q":"Error. Key not valid!"}
{"item_id":49,"q":"Error. Key not valid!"}
{"item_id":50,"q":"Error. Key not valid!"}
{"item_id":51,"q":"Error. Key not valid!"}
{"item_id":52,"q":"Error. Key not valid!"}
{"item_id":53,"q":"Error. Key not valid!"}
{"item_id":54,"q":"Error. Key not valid!"}
{"item_id":55,"q":"Error. Key not valid!"}
{"item_id":56,"q":"Error. Key not valid!"}
{"item_id":57,"q":"Winter Wonderland, Hyde Park, London."}
{"item_id":58,"q":"Error. Key not valid!"}
{"item_id":59,"q":"Error. Key not valid!"}
{"item_id":60,"q":"Error. Key not valid!"}
{"item_id":61,"q":"Error. Key not valid!"}
{"item_id":62,"q":"Error. Key not valid!"}
{"item_id":63,"q":"Error. Key not valid!"}
{"item_id":64,"q":"Error. Key not valid!"}
{"item_id":65,"q":"Error. Key not valid!"}
{"item_id":66,"q":"Error. Key not valid!"}
{"item_id":67,"q":"Error. Key not valid!"}
{"item_id":68,"q":"Error. Key not valid!"}
```

Question 4

We use the same method as the question 4 to determine the correct API number

**Side note for this task

My IP address may show differently because for question 3 and 4 if I try too many times it will block my access so that I have to terminate the machine every time I failed. Hope y'all don't mind.

Thought process/Methodology:

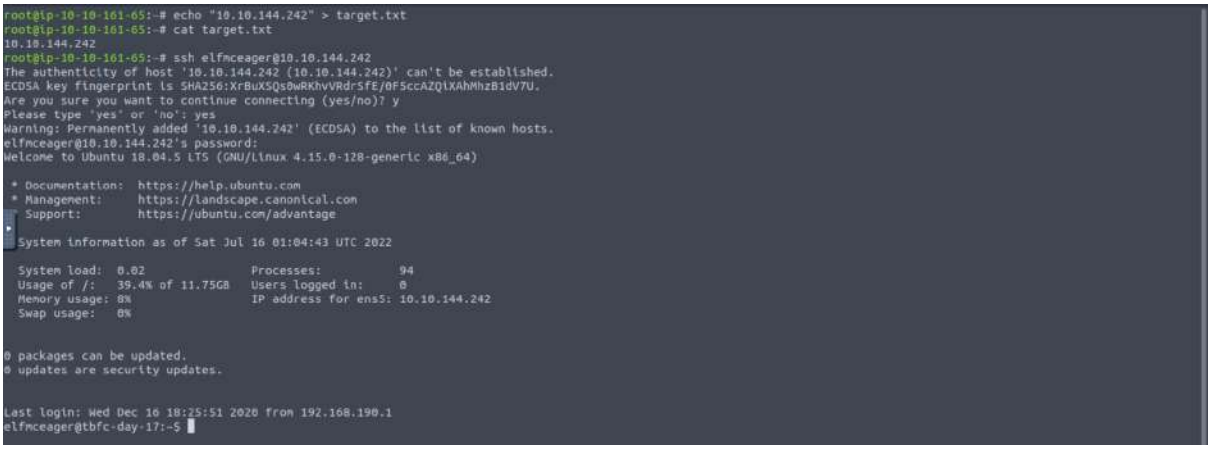
For question 1 we used Nmap to scan all the open port for this website and we found that port 80 were actually open for HTML. And then, we used python given on day 15 to find all the links inside this website. We found that there was a link for the API directory. Lastly, we used python to try all the API keys in ranges 1 to 100 to determine which one was the correct API key. We found that key 57 was the correct key and it has shown with the location of the Santa.

Day 17 ReverseELFneering

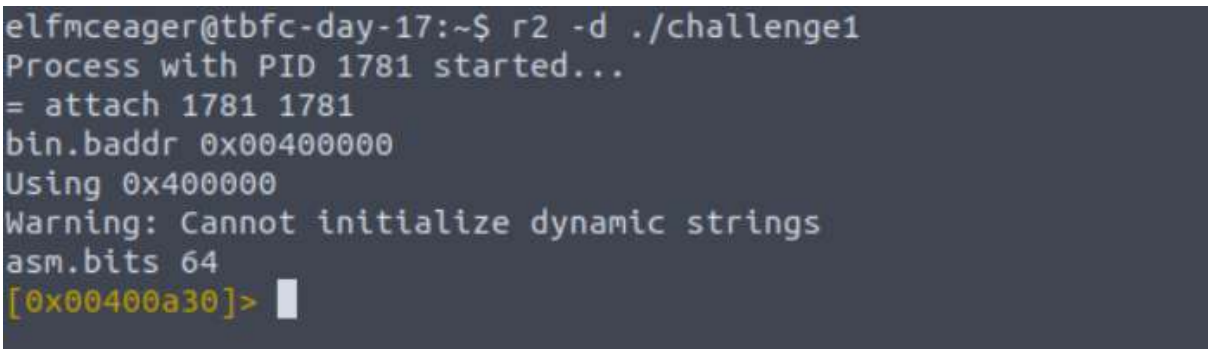
Tools used: Kali Linux

Question 1

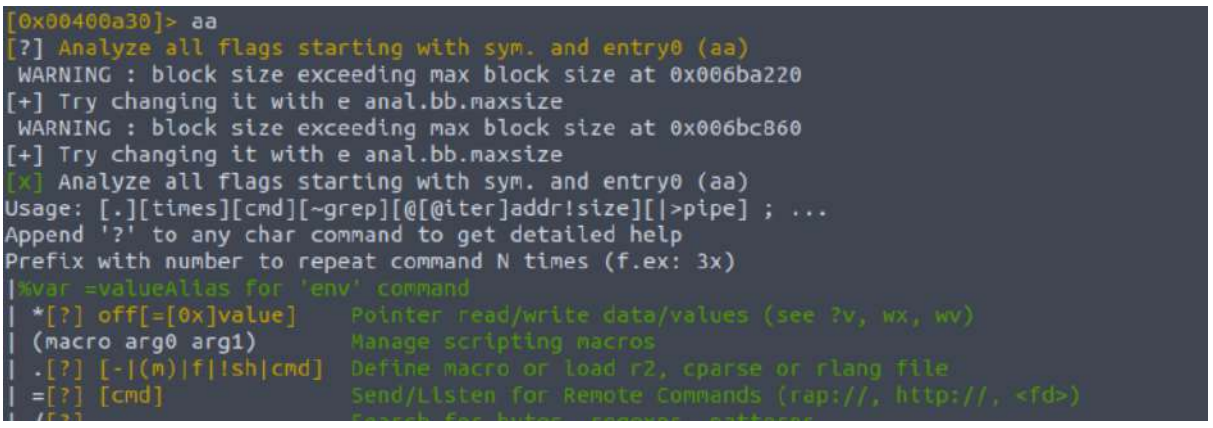
We log in to the username elfmceager using the username and password and ip address given.



We run the command r2 -d ./challenge1 to open the binary in debug mode



We use aa to ask r2 to analyze the program



We type in pdf@main to print the disassembly function and we get to know the answer

Question 2

We get the answer by referring to the pdf@main

Question 3

Since the eax is copy from the previous variable, so it have the same answer as the previous one.

Thought process/Methodology:

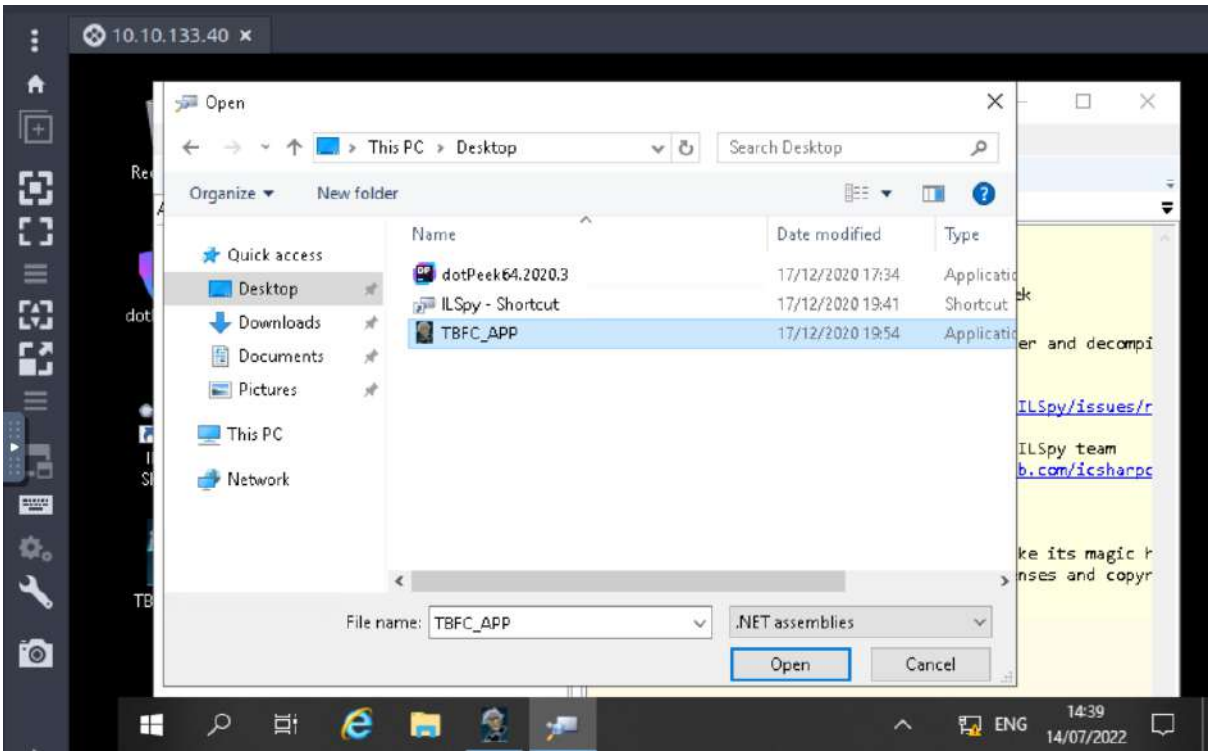
By using the IP address, username, and password given, we log in to the account successfully using echo and ssh. To open the binary debug mode, we run r2. By referring to the disassembly function, we get the answer for question1, and 2. Since the eax is copy from the previous variable, it has the same answer as the variable.

Day 18 The Bits of Christmas

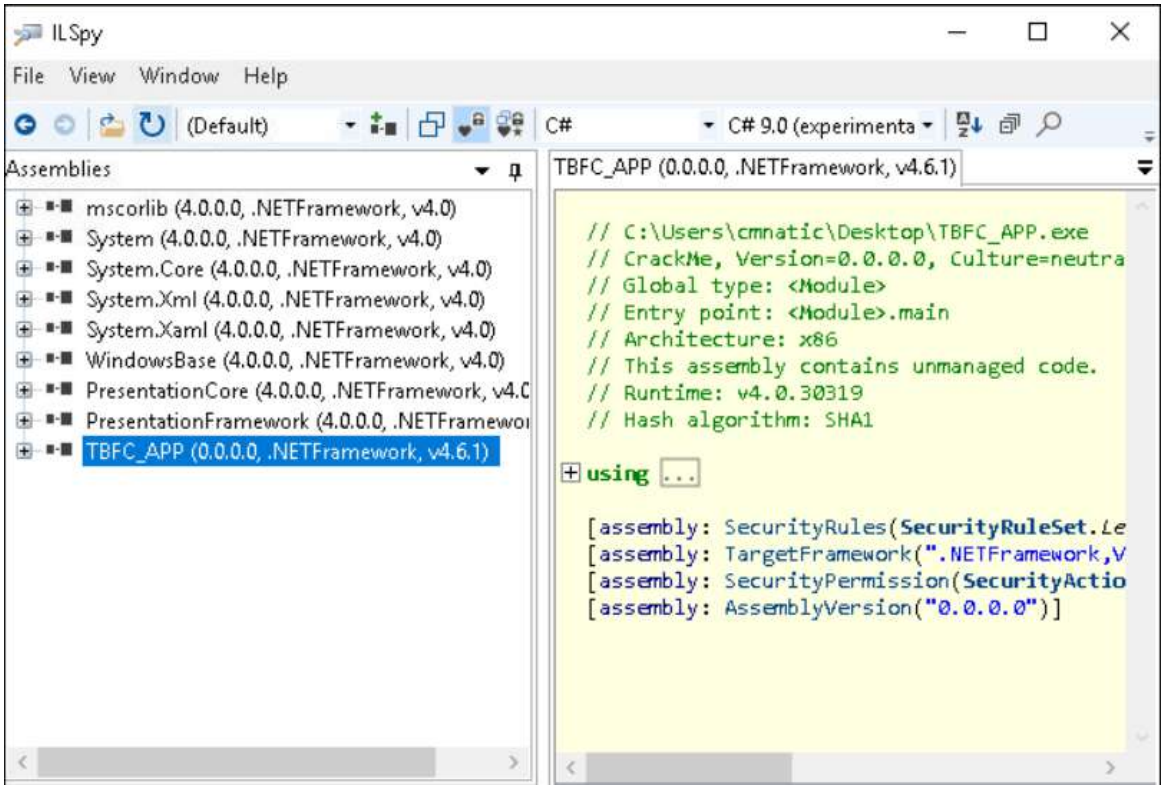
Tools used: Kali Linux, IL SPY, CyberChef

Question 1

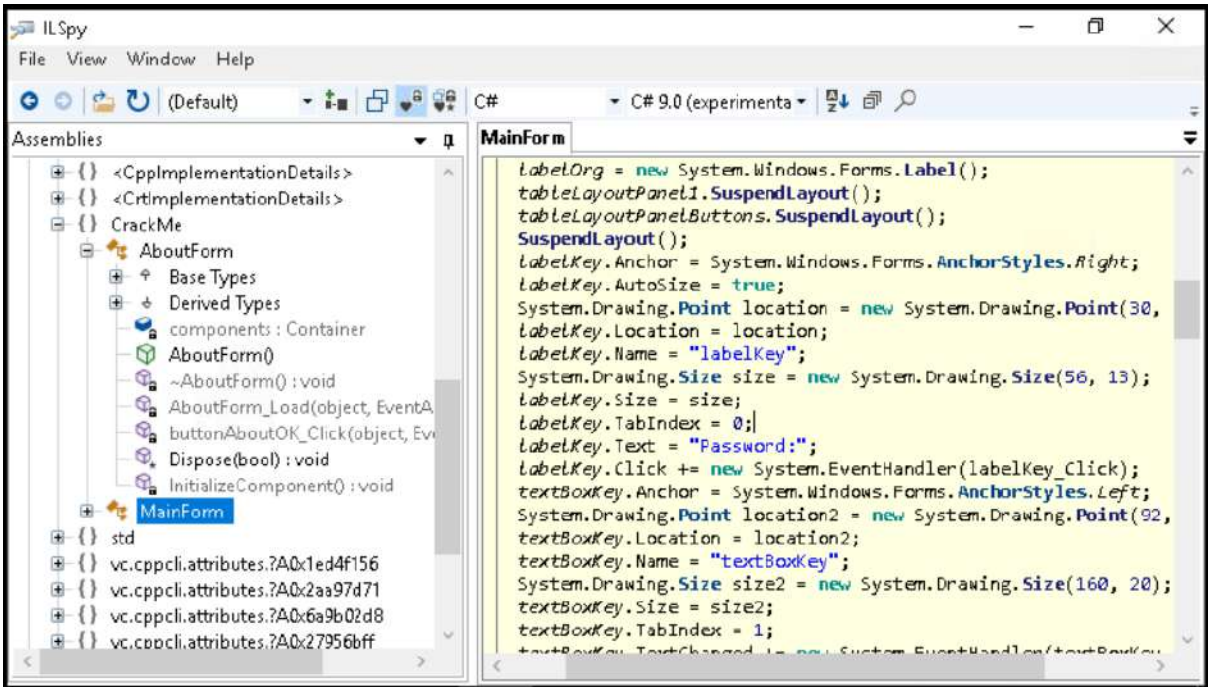
After login to the remmina, open the IL SPY, then load the TBFC_APP.



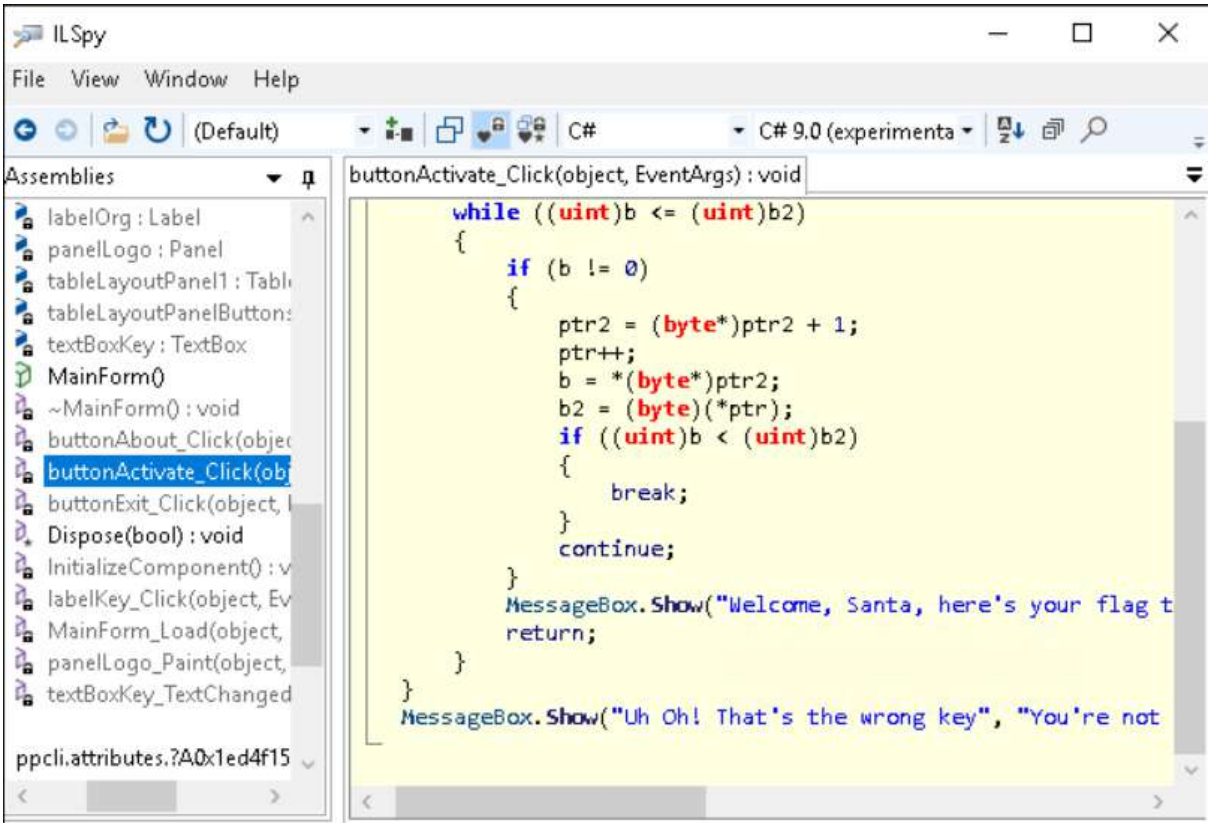
Press the plus button to look in more detail.



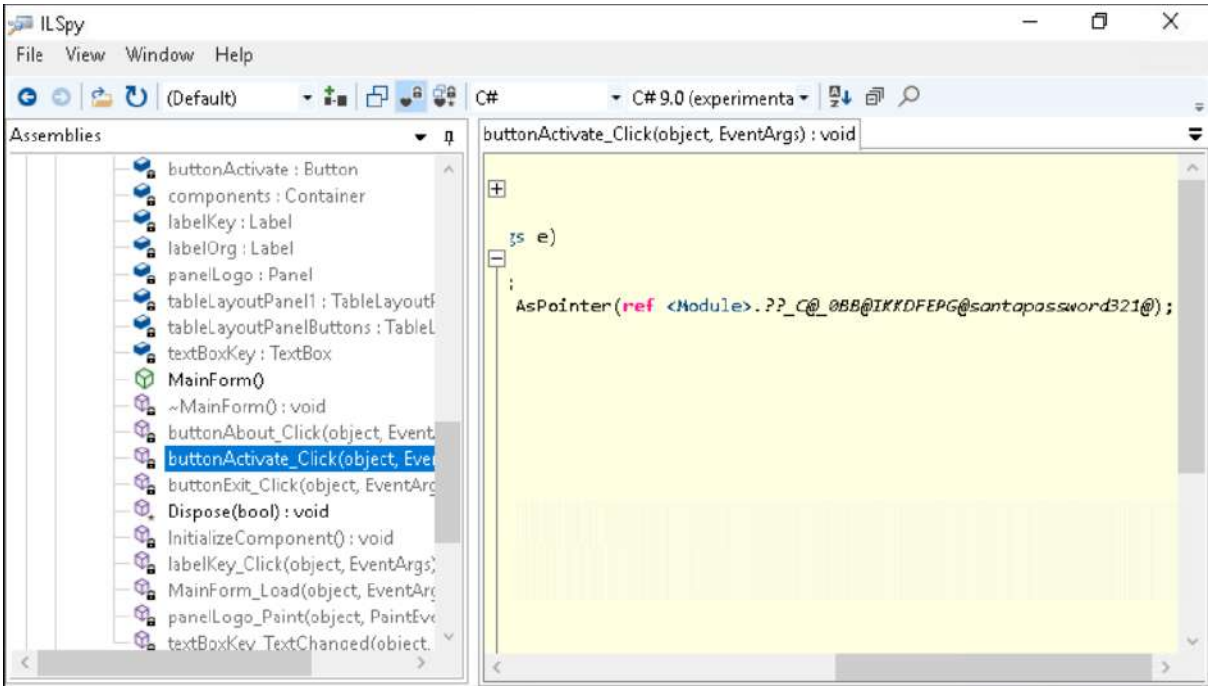
You will see CrackMe and the MainForm in it. Expand it by clicking on the plus button.



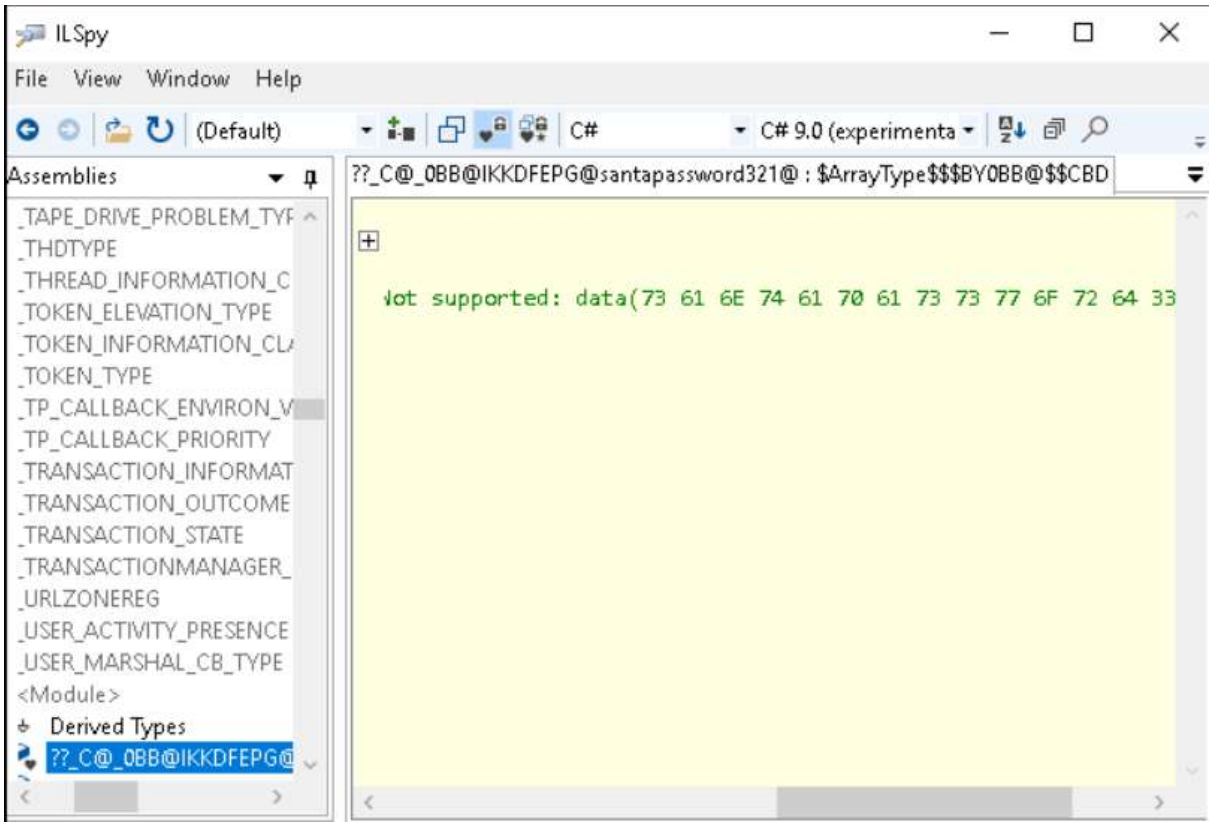
You will see the buttonActivate_Click(object, EventArgs) : void. Double click it and it looks like the button that submit the input when we enter the password when login in.(It print out the “Welcome, santa, …” and the “Uh Oh! …”)



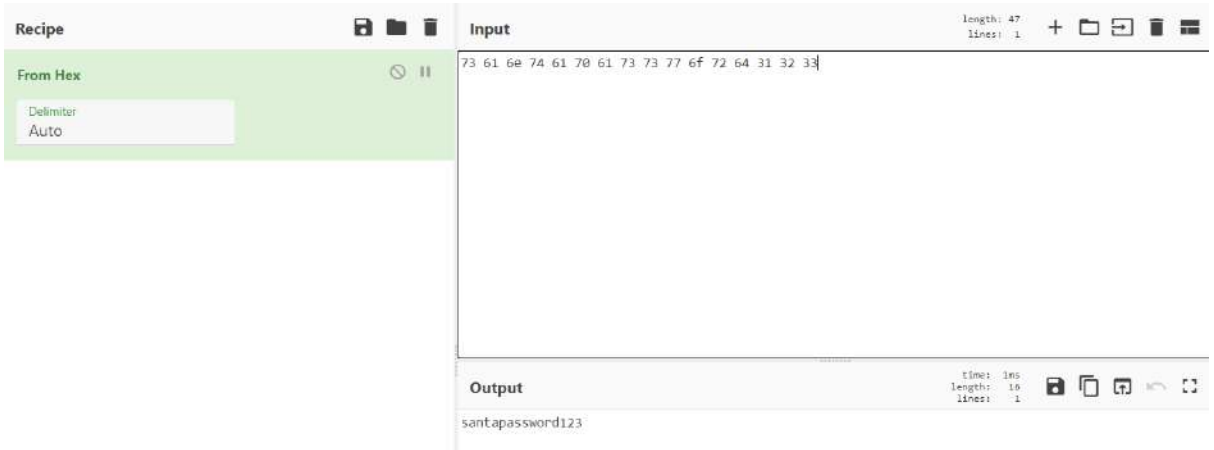
There is a string that looks like a password. Double click it.



You would see some data behind and it is hexadecimal.

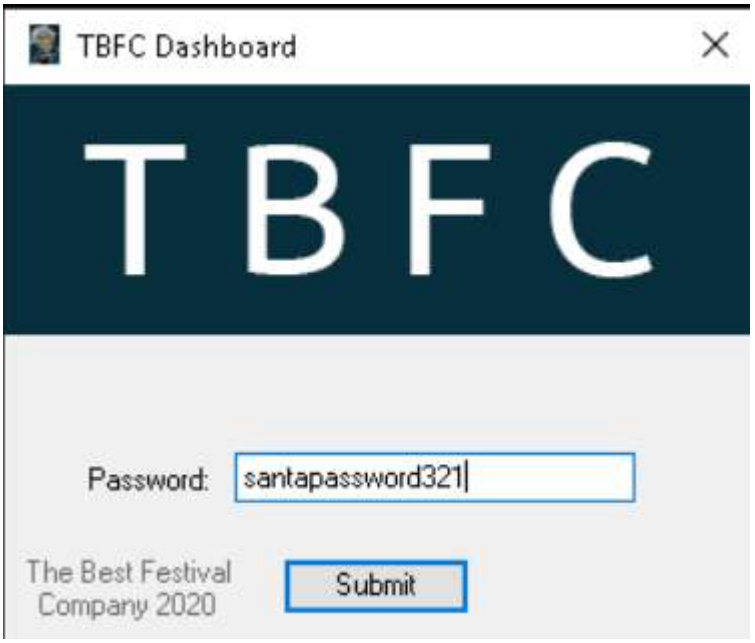


Transform the hexadecimal to word using CyberChef and you will get the password.

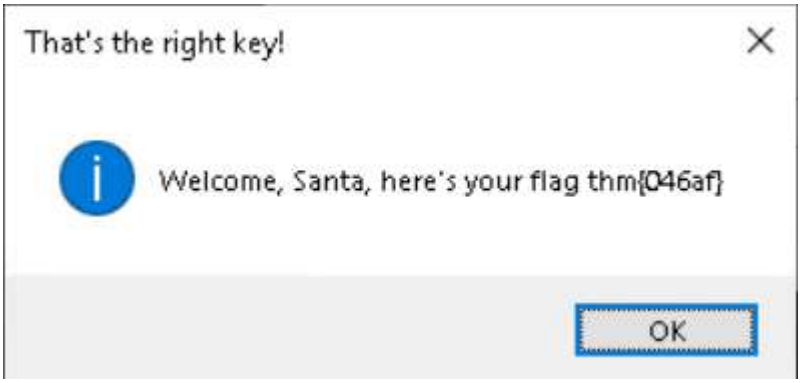


Question 2

Login to the TBFC using the password.

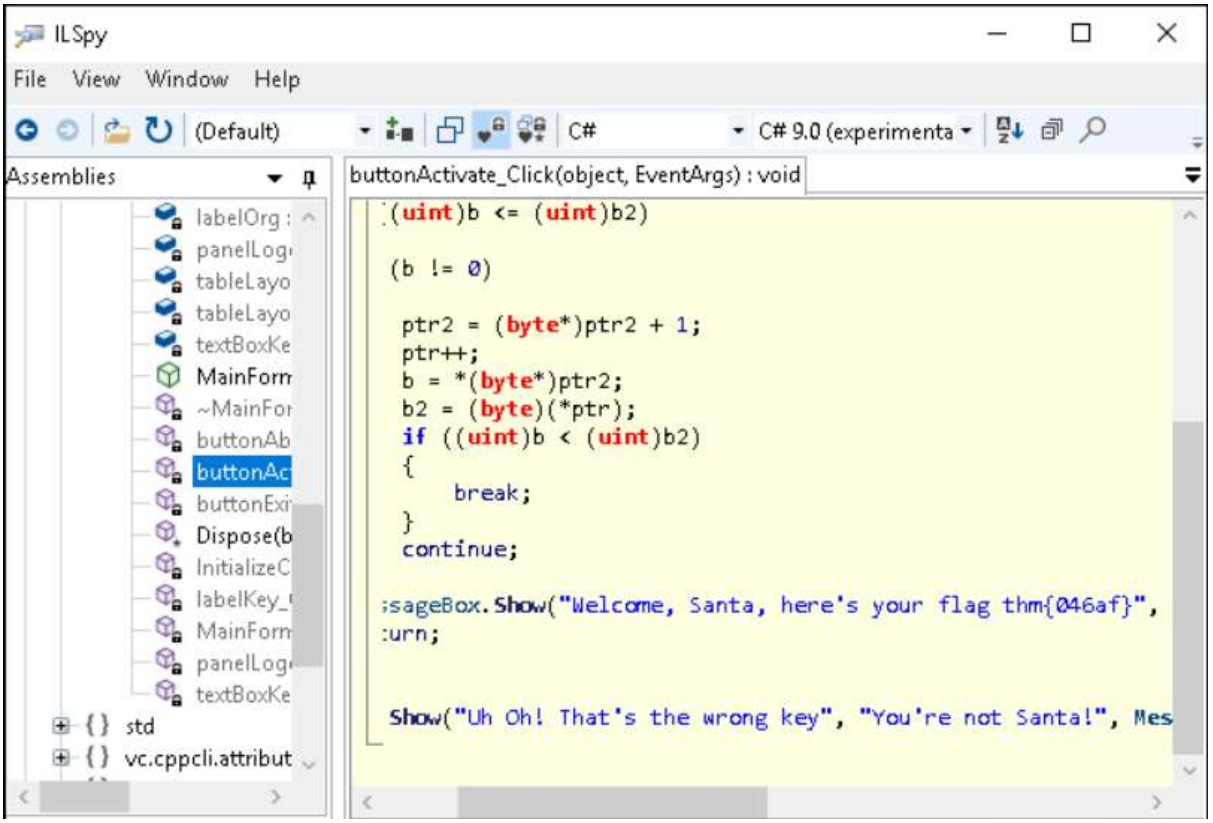


You get the flag.



Or

You can get the flag at the bottom of the buttonActivate_Click(object, EventArgs) page in “Welcome, Santa, …”



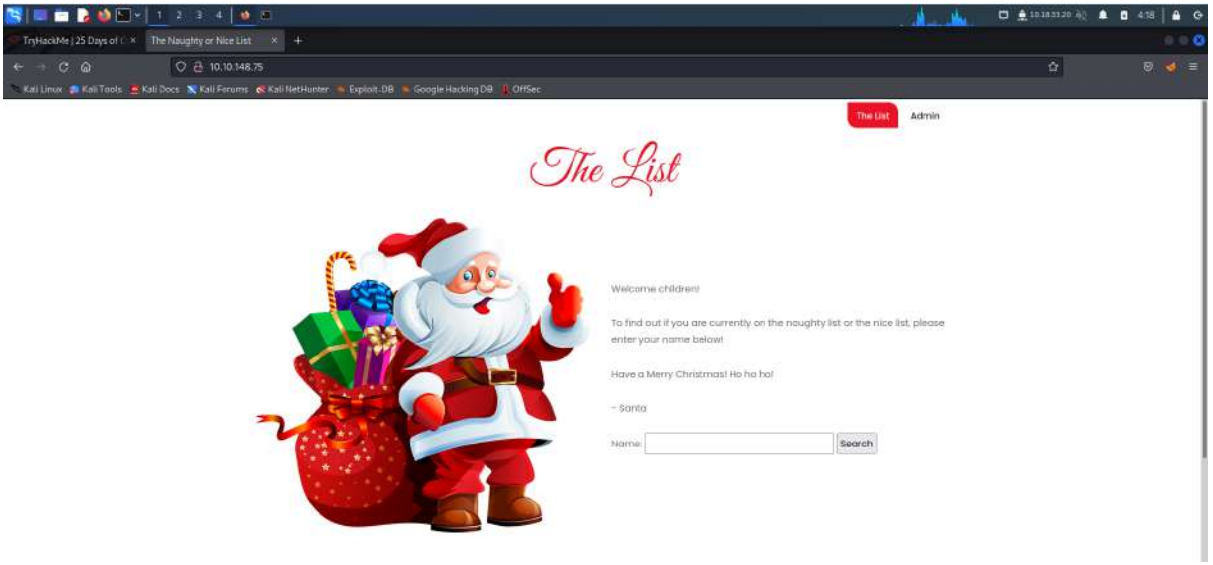
Thought process/Methodology:

For Question 1. Login to the remmina, open the IL SPY, then load the TBFC_APP. Then, press the plus button to look in more detail. You will see the CrackMe and the MainForm in it. Expand it by clicking on the plus button. Then, you will see the buttonActivate_Click(object, EventArgs) : void. Double click it and it looks like the button that submit the input when we enter the password when login in.(It print out the “Welcome, santa, …” and the “Uh Oh! …”). There is a string that looks like a password. Double click it. After that, you would see some data behind and it is hexadecimal. You would see some data behind and it is hexadecimal. You need to transform the hexadecimal to word using CyberChef and you will get the password. For Question 2, login to the TBFC using the password. Then, you will get the flag. Or you can get the flag at the bottom of the buttonActivate_Click(object, EventArgs) page in “Welcome, Santa, …”

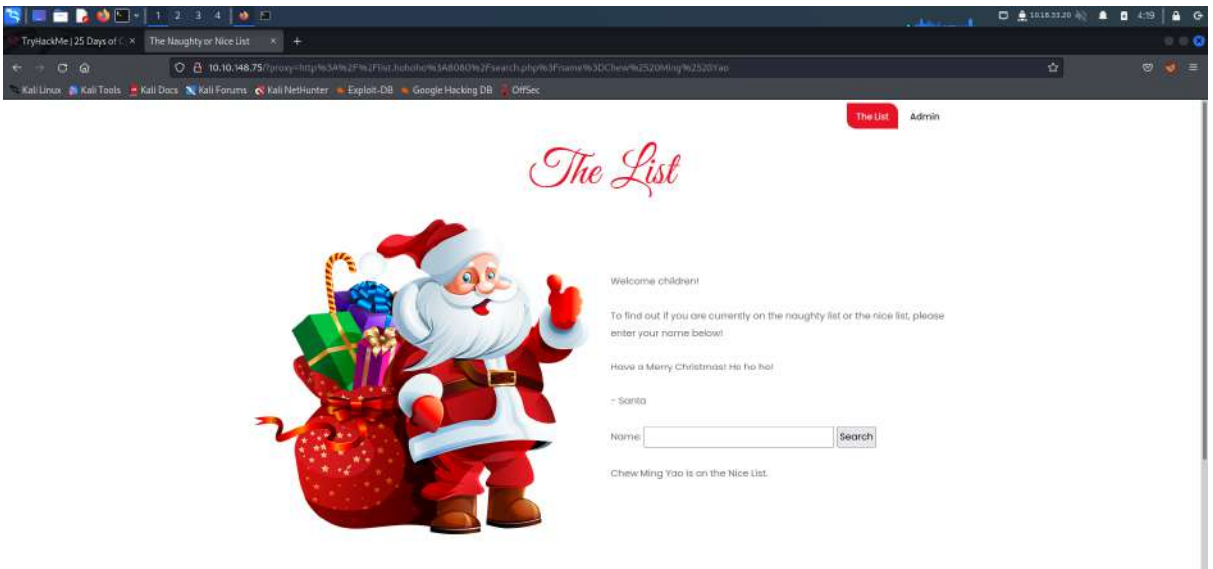
Day 19 The Naughty or Nice List

Tools used: Kali Linux, firefox, CyberChef

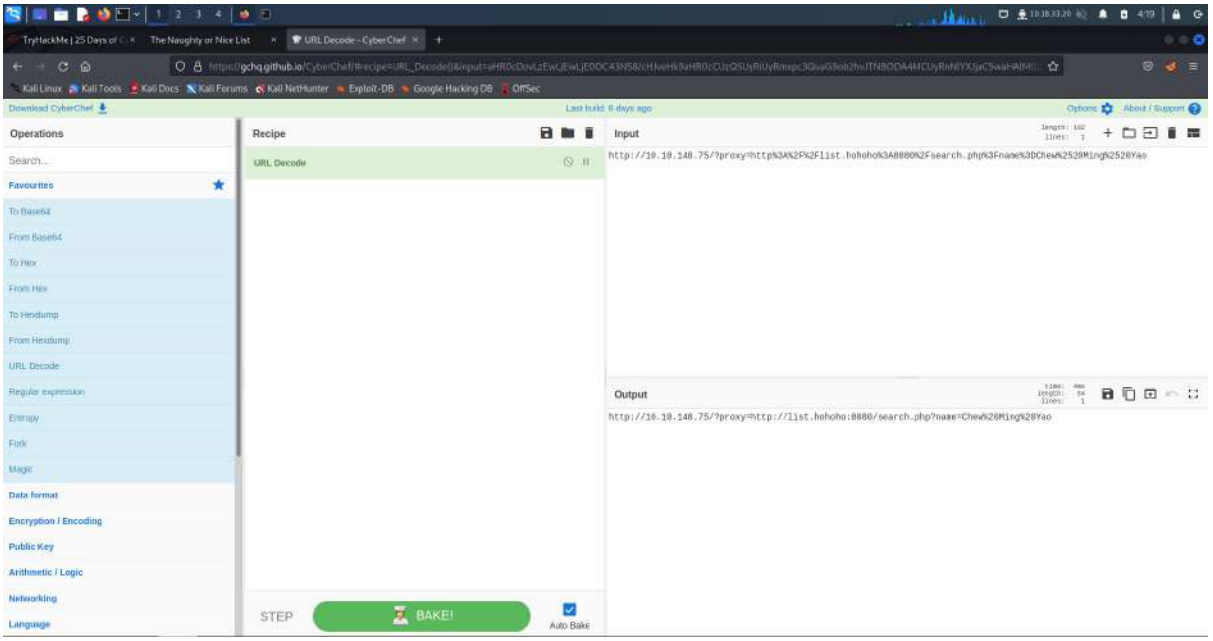
Copy and paste the IP address given.



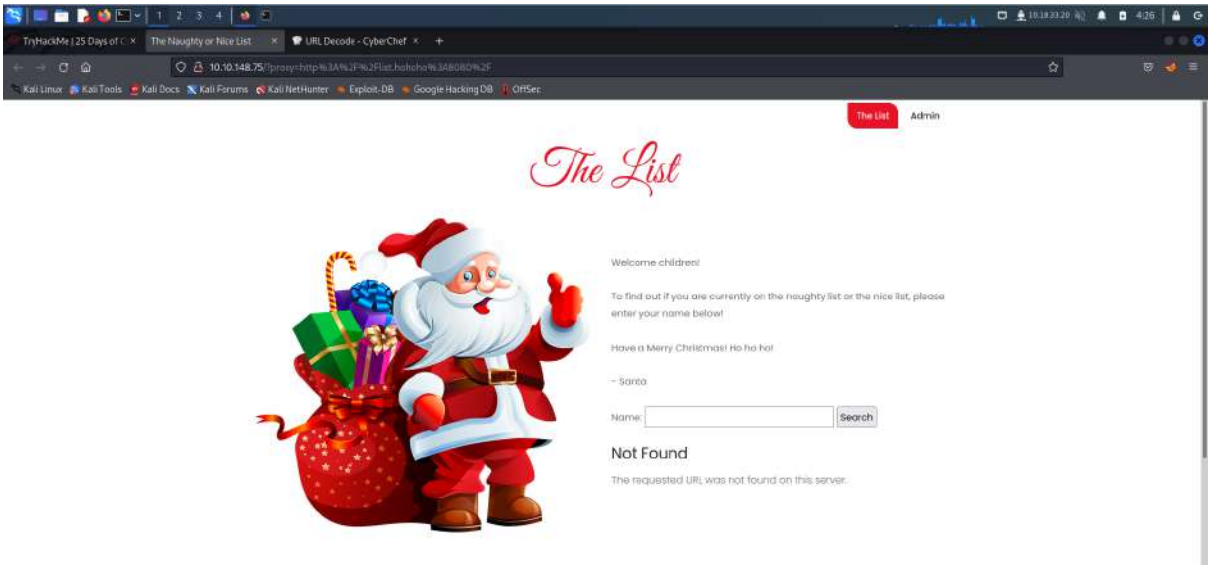
Type the name on the Name bar and it will show the name on the Nice List.



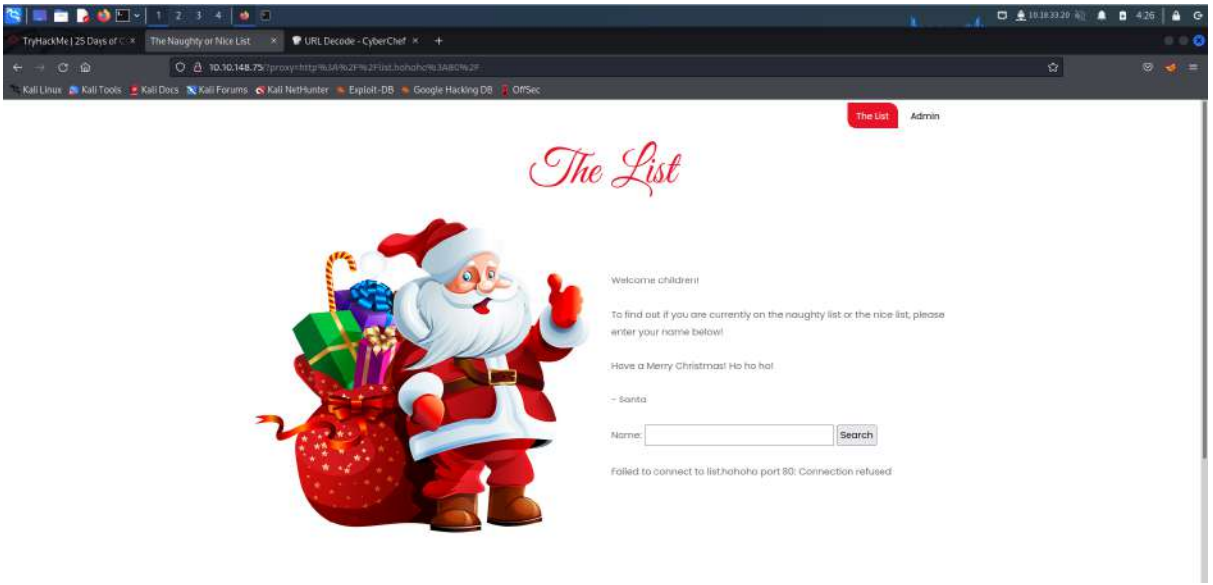
Copy the link and paste into cyberchef. Use URL Decode to find the port.



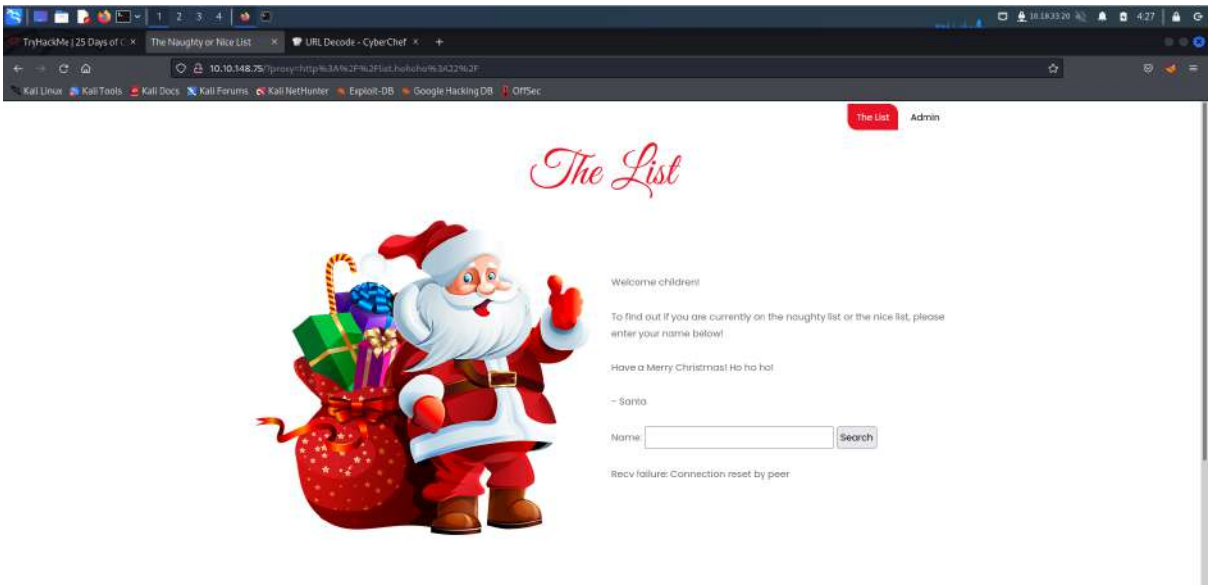
Delete extra words to go port 8080 and it appears the message URL is not found.



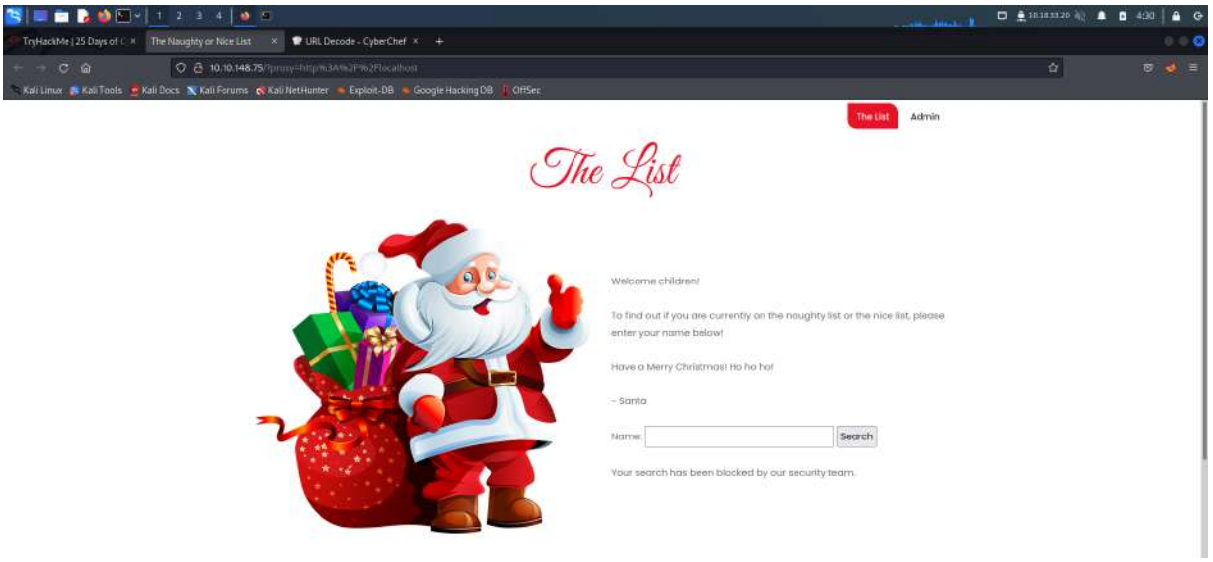
Try to change port 8080 into 80. The messages will be replaced by show the connection refused.



Change to port 22 because port 22 is the default SSH port. The messages will replace connection refused to connection reset by peer.

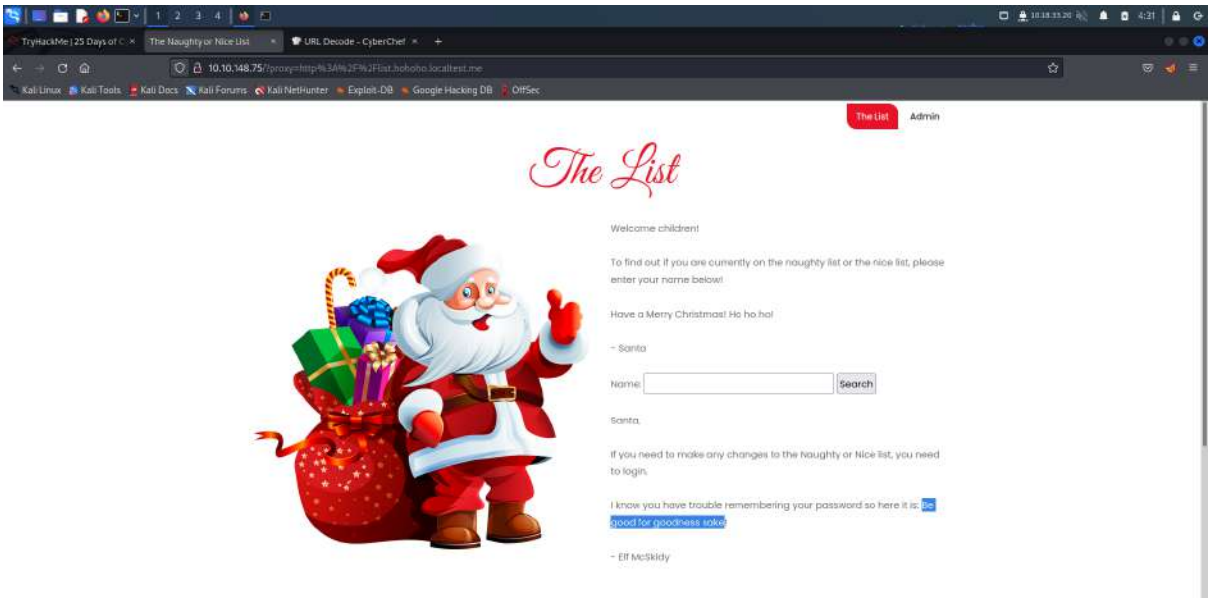


We change the list hoho to localhost and the message will replace connection reset by peer to Your search has been blocked by our security team. It means that it can easily be bypassed.



Question 1

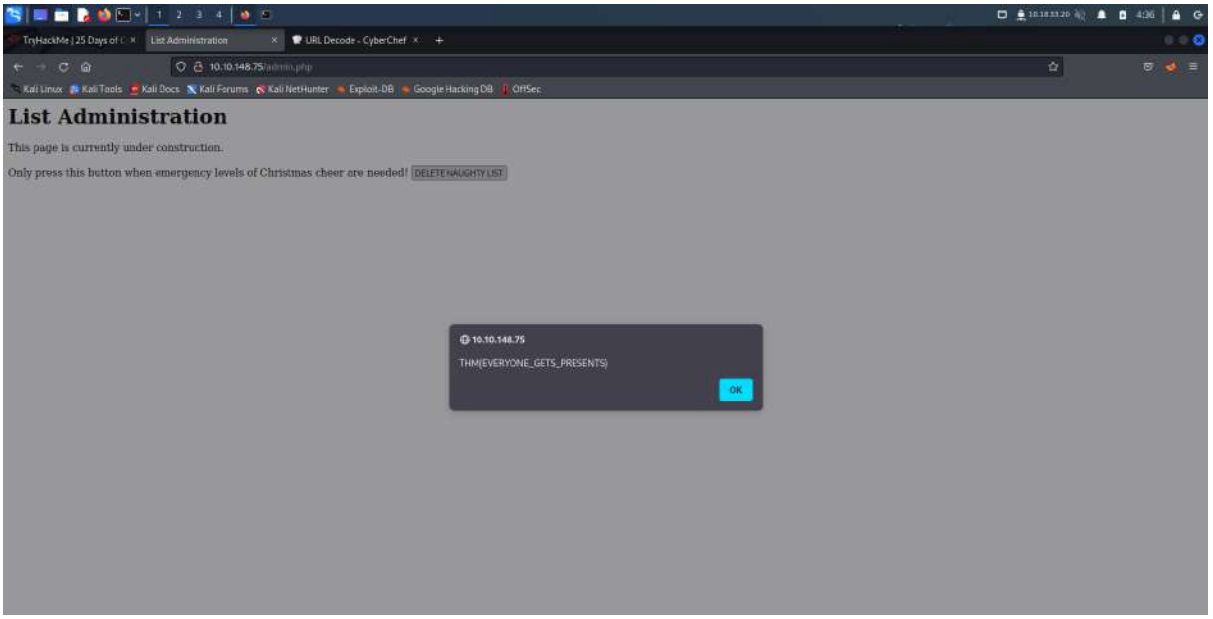
We changed localhost to list.hohoho.localtest.me and the messages showed the password of the santa.



Question 2

Login into the admin with username Santa and password given

We logged in and deleted the naughty list to get the answer.



Thought process/Methodology:

Firstly, we need to copy and paste the ip address that is given. After that, we need to type the name of the name bar to get the URL that changed. We get the URL and know the port is 8080. We need to change the port into 22 to reset the connection peer because port 22 is the default SSH port. Furthermore, we change list hoho to localhost and it is blocked by the security team. It means that it can easily be passed. After that, we change localhost to list.hohoho.localtest.me and get the password for question 1. For question 2, login into admin with username Santa and use the password given. We need to delete the naughty list and get the answer for question 2.

Day 20 PowershELIF to the rescue

Tools used: Kali Linux

Question 1

We used the command given to login using the ip given

```
root@ip-10-10-3-8:~# ssh -l mceager 10.10.113.119
r0ckStar!
Microsoft Windows [Version 10.0.17763.737]
(c) 2018 Microsoft Corporation. All rights reserved.

mceager@ELFSTATION1 C:\Users\mceager>
```

We use the Get-ChildItem command to get to know the Hidden file

```
PS C:\Users\mceager\Documents> Get-ChildItem -file -Hidden


    Directory: C:\Users\mceager\Documents


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a-hs-         12/7/2020  10:29 AM            402 desktop.ini
-arh--        11/18/2020   5:05 PM             35 e1fone.txt
```

We used the Get -Content command to know what Elf 1 wants

```
PS C:\Users\mceager\Documents> Get-Content e1fone.txt
All I want is my '2 front teeth'!!!
PS C:\Users\mceager\Documents>
```

Question 2

We change the directory to desktop to get to know the Hidden directory which is elf2wo

```
PS C:\Users\mceager\Documents> cd..
PS C:\Users\mceager> Set-Location .\Desktop\
PS C:\Users\mceager\Desktop> Get-ChildItem -file -Hidden


    Directory: C:\Users\mceager\Desktop


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a-hs-        12/7/2020  10:29 AM            282 desktop.ini


PS C:\Users\mceager\Desktop> ls -Hidden


    Directory: C:\Users\mceager\Desktop


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d--h--        12/7/2020  11:26 AM                elf2wo
-a-hs-        12/7/2020  10:29 AM            282 desktop.ini


PS C:\Users\mceager\Desktop>
```

By opening the file in the directory using the Get-Content command we know that the movie Elf 2 like is Scrooged

```
PS C:\Users\mceager\Desktop> cd elf2wo
PS C:\Users\mceager\Desktop\elf2wo> Get-ChildItem -file


    Directory: C:\Users\mceager\Desktop\elf2wo


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a----        11/17/2020  10:26 AM             64 e70smsW10Y4k.txt


PS C:\Users\mceager\Desktop\elf2wo> Get-Content e70smsW10Y4k.txt
I want the movie Scrooged <3!
PS C:\Users\mceager\Desktop\elf2wo>
```

Question 3

After changed the directory to C:\Windows\System32, We use the Get-ChildItem command to know the hidden folder name which is 3lfthr3e

```
PS C:\Windows\System32> Get-ChildItem -Hidden -Directory


    Directory: C:\Windows\System32


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d--h--        11/23/2020   3:26 PM                3lfthr3e
d--h--        11/23/2020   2:26 PM                GroupPolicy
```

Question 4

After we logged in to the directory, we use the Get-Content command to open the file and measure the word count using command- Measure-Object

```
PS C:\Windows\System32\3lfthr3e> Get-Content 1.txt |Measure-Object


Count    : 9999
Average  :
Sum      :
Maximum  :
Minimum  :
Property :
```

Question 5

By typing the command as (Get-Content -file.txt)[number of index], we get to know the two words in each index

```
PS C:\Windows\System32\3lfthr3e> (Get-Content 1.txt)[551]
Red
PS C:\Windows\System32\3lfthr3e> (Get-Content 1.txt)[6991]
Ryder
```

Question 6

```
PS C:\Windows\System32\3lfthr3e> Get-Content 2.txt | Select-String -Pattern "red
ryder"
```

By typing the command as Get-Content -file.txt | Select String -Pattern “pattern of the index”, we get to know what Elf 3 want which is red ryder bb gun

Thought process/Methodology:

We have learned to use the Get-ChildItem to know the name of hidden file. By using the Get-Content command, we read the content in the file. We repeated the same step to get to know what Elf 2 and 3 wanted. Besides, we used the Measure–Object command to know the total word count in the text file. By using the format of command- (Get-Content-file. txt)[number of indexes], we successfully get to know the words in the index. Finally, we used the Select-String command to get to know the same format of answers.