

PSP0201

Week 5

Writeup

Group Name: GLM

Members

ID	Name	Role
1211102971	Leong Chun Kit	Leader
1211103023	Yap Weng Hong	Member
1211101232	Lim Kai Qian	Member
1211101407	Tan Fu Shun	Member

Day 16: Scripting - Help! Where is Santa?

Tools used: Firefox and Terminal

Solution/walkthrough:

Question 1

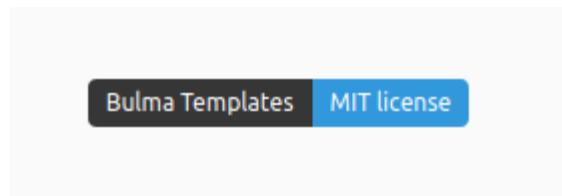
We use nmap and type the machine IP then it shows the port number for us.

```
root@ip-10-10-73-242:~# nmap 10.10.143.111
Starting Nmap 7.60 ( https://nmap.org ) at 2022-07-15 07:09 BST
Nmap scan report for ip-10-10-143-111.eu-west-1.compute.internal (10.10.143.111)
Host is up (0.034s latency).
Not shown: 998 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http
MAC Address: 02:AE:34:23:6C:39 (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 1.99 seconds
```

Question 2

This is the template we used.



Question 3

First, we go Fire fox to search this IP. Then it will lead us to this website. After that, we should right click it and view the page source. Then we will found this link and this is the answer for question 3.

Santa's Tracker

TryHackMe | Learn Cy... TryHackMe Support Offline CyberChef GitHub · swisskyrepo/... Reverse Shell Cheat S...

Home Examples View Source: Template not my own.

Santa's Tracking System

Are you an Elf that Santa has forgotten? Use this system to track Santa! Note: due to how many humans try to find where Santa is, the link is hidden on this webpage. You're going to have to manually click every single link. Or perhaps there is a way to find all the links as fast as a Python?

All deliveries to Skidy for TryHackMe jumpers are to be stopped. That man has asked for 613 on the premise that they are the softest jumper in the world. Please, we need to share them out.

Category	Category	Category
Lorem ipsum dolor sit amet.	Labore et dolore magna aliqua	Objects in space
Vestibulum errato issa	Kanban airis sum eschelor	Playing cards with coyote
Aliquam ipsum dolor sit amet.	Modular modern free	Goodbye Yellow Brick Road
Aliis caisia	The king of clubs	The Garden of Forking Paths
Murphy's law	The Discovery Dissipation	Future Shock
Flimsy Lavenrock	Course Correction	
Maven Mousie Lavender	Better Angels	

Transferring data from 10.10.143.111...

how many humans try to find where Santa is, the link is hidden
haps there is a way to find all the links as fast as a Python?

remise that they are the softest jumper in t them out.

ua

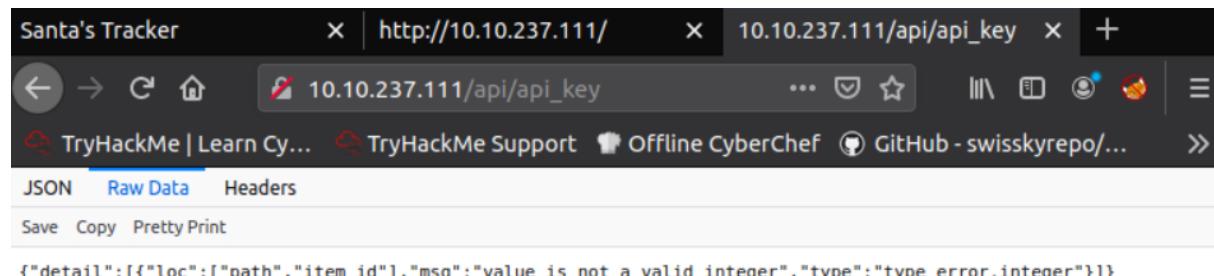
Category

- Objects in space
- Playing cards with coyote
- Goodbye Yellow Brick Road
- The Garden of Forking Paths
- Future Shock

Save Page As...
Save Page to Pocket
Send Page to Device
View Background Image
Select All
View Page Source
View Page Info
Inspect Accessibility Properties
Inspect Element (Q)
Take a Screenshot

Question 4

We just open a new tab and follow the format link like Question 3, then it will show the RAW DATA returned and this is the answer.



Question 5 & 6

By using python, first we go import requests. Then, set the range for api_key from 1 to 100 and space twice. After that, we follow the photo and type the code and go terminal start running it. As you see that 57 is the correct key that we want. Question 6 also same with q5 and when we saw the api_key 57, we found that the places where Santa right now.

```
1 #!/usr/bin/env python3
2
3 import requests
4
5 for api_key in range(1,100,2):
6     print(f"api_key {api_key}")
7     html = requests.get(f'http://10.10.48.42:8000/api/{api_key}')
8     print(html.text)
9
10
```

```
kali@wsl:~/ctf/thm/aoc_day16$ python3 apibruter.py
{"item_id":1,"q":"Error. Key not valid!"}
kali@wsl:~/ctf/thm/aoc_day16$ python3 apibruter.py
api_key 1
{"item_id":1,"q":"Error. Key not valid!"}
api_key 3
{"item_id":3,"q":"Error. Key not valid!"}
api_key 5
{"item_id":5,"q":"Error. Key not valid!"}
api_key 7
api_key 57
{"item_id":57,"q":"Winter Wonderland, Hyde Park, London."}
```

Thought Process/Methodology:

First, we use nmap and type the machine IP then it shows the port number for us. Next, we go Fire fox to search this IP. Then it will lead us to this website. After that, we should right click it and view the page source. Then we will find this link and this is the answer for question 3. For question 4, we just open a new tab and follow the format link like Question 3, then it will show the RAW DATA returned and this is the answer. For question 5, by using python, we go import requests. Then, set the range for api_key from 1 to 100 and space twice. After that, we follow the photo and type the code and go terminal start running it. As you see that 57 is the correct key that we want. For question 6, it also same with q5 and when we saw the api_key 57, we found that the places where Santa right now.

Day 17: Reverse Engineering – ReverseELFneering

Tools used: Attack box

Solution/walkthrough:

Question 1

In 25 Days of Cyber Security in day 17 we can find out the notes had given the solution for Question 1

The core of assembly language involves using registers to do the following:

- Transfer data between memory and register, and vice versa
- Perform arithmetic operations on registers and data
- Transfer control to other parts of the program Since the architecture is x86-64, the registers are 64 bit and Intel has a list of 16 registers:

Initial Data Type	Suffix	Size (bytes)
Byte	b	1
Word	w	2
Double Word	l	4
Quad	q	8
Single Precision	s	4
Double Precision	l	8

Question 2

Run the command `r2 -d ./file1` to opens the binary files in debugging mode. One of the first things to do is ask r2 to analyse the program, and this can be done by typing in: aa

The above program shows that there are 3 variables(a, b, c) where c is the sum of a and b.

Time to see what's happening under the hood! Run the command `r2 -d ./file1`

This will open the binary in debugging mode. Once the binary is open, one of the first things to do is ask r2 to analyze the program, and this can be done by typing in: `aa`

which is the most common analysis command. It analyses all symbols and entry points in the executable. The analysis, in this case, involves extracting function names, flow control information, and much more! r2 instructions are usually based on a single character, so it is easy to get more information about the commands.

I.e. For general help, we can run: `? or if we wish to understand more about a specific feature, we could provide a?`

3. Computer says...Done??

Once the analysis is complete, you would want to know where to start analysing from - most programs have an entry point defined as main. To find a list of the functions run `af`

```
[0x00400030]> af | grep main
```

```
0x00400b4d 1 68      sym.main
0x00400e10 114 1657  sym.__libc_start_main
0x00403870 346 6038 -> 5941 sym._nl_find_domain
0x00415fe0 1 43     sym._IO_switch_to_main
0x0044cf00 1 8      sym._dl_get_dl_main
0x00470520 1 49     sym._IO_switch_to_main
```

```
elfmeager@tbfc-day-17:~
```

```
File Edit View Search Terminal Help
elfmeager@tbfc-day-17:~$ r2 -d ./file1
Process with PID 1332 started...
= attach 1332 1332
bin.baddr 0x00400000
Using 0x400000
Warning: Cannot initialize dynamic strings
asm.bits 64
[0x00400030]> aa
[!] Analyze all flags starting with sym. and entry0 (aa)
```

1h 49m 16s

Question 3

After that start looking at the program from the 4th instruction and set it as a breakpoint by using db to analyse the program while it runs. So let's set a breakpoint using the command db in this case, it would be db 0x00400b55.

The line starting with sym.main indicates we're looking at the main function. The next 3 lines are used to represent the variables stored in the function. The second column indicates that they are integers(int), the 3rd column specifies the name that r2 uses to reference them and the 4th column shows the actual memory location.

The first 3 instructions are used to allocate space on that stack (ensures that there's enough room for variables to be allocated and more). We'll start looking at the program from the 4th instruction (movl \$4). We want to analyse the program while it runs and the best way to do this is by using breakpoints.

A breakpoint specifies where the program should stop executing. This is useful as it allows us to look at the state of the program at that particular point. So let's set a breakpoint using the command db in this case, it would be db 0x00400b55. To ensure the breakpoint is set, we run the pdf @main command again and see a little b next to the instruction we want to stop at.

Question 4

Next we use the dc command to run the program until it hits the break point. Running dc will execute the program until we hit the breakpoint. Once we hit the breakpoint and print out the main function, the rip which is the current instruction shows where execution has stopped.

Now that we've set a breakpoint, let's run the program using dc.

[0x00400a30]> dc
hit breakpoint at: 400b55
[0x00400b55]> pdf
--- main:
--- RAX:
--- RBP:
--- RDI:
--- RSI:
--- RDX:
--- RCX:
--- RBP:

Running dc will execute the program until we hit the breakpoint. Once we hit the breakpoint and print out the main function, the rip which is the current instruction shows where execution has stopped. From the notes above, we know that the mov instruction is used to transfer values. This statement is transferring the

Question 5

We will start through with the challenge file with command r2 d ./challenge1. We follow the same initial steps. After that command pdf @main we can find out that mov dword [local_ch], 1. This means that the local_ch variable is being set to 1.

```
elfmceager@tbfc-day-17:~/Documents$ r2 -d ./challenge1
Process with PID 1517 started ...
= attach 1517 1517
bin.baddr 0x00400000
Using 0x400000
Warning: Cannot initialize dynamic strings
asm.bits 64
[0x00400a30]> aa
[ WARNING : block size exceeding max block size at 0x006ba220
[+] Try changing it with e anal.bb.maxsize
WARNING : block size exceeding max block size at 0x006bc860
[+] Try changing it with e anal.bb.maxsize
[x] Analyze all flags starting with sym. and entry0 (aa)
[0x00400a30]> afl | grep main
0x00400b4d    1 35      sym.main
0x00400de0   10 1007 → 219  sym.__libc_start_main
0x00403840   39 661   → 629  sym._nl_find_domain
0x00403ae0   308 5366 → 5301 sym._nl_load_domain
0x00415ef0    1 43      sym._IO_switch_to_main_get_area
0x0044ce10    1 8       sym._dl_get_dl_main_map
0x00470430    1 49      sym._IO_switch_to_main_wget_area
0x0048f9f0    7 73   → 69  sym._nl_fnddomain_subfreeres
0x0048fa40   16 247   → 237 sym._nl_unload_domain
```

```
[0x00400a30]> pdf @ main
;-- main:
/ (fcn) sym.main 35
sym.main ();
    ; var int local_ch @ rbp-0xc
    ; var int local_8h @ rbp-0x8
    ; var int local_4h @ rbp-0x4
        ; DATA XREF from 0x00400a4d (entry0)
0x00400b4d    55      push rbp
0x00400b4e   4889e5    mov rbp, rsp
0x00400b51   c745f4010000. mov dword [local_ch], 1
0x00400b58   c745f8060000. mov dword [local_8h], 6
0x00400b5f   8b45f4    mov eax, dword [local_ch]
0x00400b62   0faf45f8  imul eax, dword [local_8h]
0x00400b66   8945fc    mov dword [local_4h], eax
0x00400b69   b800000000  mov eax, 0
0x00400b6e   5d      pop rbp
0x00400b6f   c3      ret
```

Question 6

Using the instruction given by we can know that *imulq source, destination*: destination = destination * source. Then, *imul eax, dword [local_8h]* (destination * source or 6*1 = 6)

Some other important instructions are:

- *leaq source, destination*: this instruction sets destination to the address denoted by the expression in source
- *addq source, destination*: destination = destination + source
- *subq source, destination*: destination = destination - source
- *imulq source, destination*: destination = destination * source
- *salq source, destination*: destination = destination << source where << is the left bit shifting operator
- *sarq source, destination*: destination = destination >> source where >> is the right bit shifting operator
- *xorq source, destination*: destination = destination XOR source
- *andq source, destination*: destination = destination & source
- *orq source, destination*: destination = destination | source

Question 7

We confirmed that eax was set to 6. For this question we continue reading. After eax was set to 6, mov dword [local_4h], eax sets [local_4h] to the value of eax, which is 6.

```
mov dword [local_8h], 6
mov eax, dword [local_ch]
imul eax, dword [local_8h]
mov dword [local_4h], eax
mov eax, 0
```

Thought Process/Methodology:

In 25 Days of Cyber Security in day 17 we can find out the notes had given the solution for Question 1. Next, we run the command r2 -d ./file1 to opens the binary files in debugging mode. One of the first things to do is ask r2 to analyse the program, and this can be done by typing in: aa. After that start looking at the program from the 4th instruction and set it as a breakpoint by using db to analyse the program while it runs. So let's set a breakpoint using the command db in this case, it would be db 0x00400b55 . Next,we use the dc command to run the program until it hits the break point. Running dc will execute the program until we hit the breakpoint. Once we hit the breakpoint and print out the main function, the rip which is the current instruction shows where execution has stopped. We will start through with the challenge file with command r2 d ./challenge1. We follow the same initial steps. After that command pdf @main we can find out that mov dword [local_ch], 1. This means that the local_ch variable is being set to 1. Using the instruction given by we can know that *imulq source, destination*: destination = destination * source. Then, imul eax, dword [local_8h] (destination * source or 6*1 = 6). We confirmed that eax was set to 6. For this question we continue reading. After eax was set to 6, mov dword [local_4h], eax sets [local_4h] to the value of eax, which is 6.

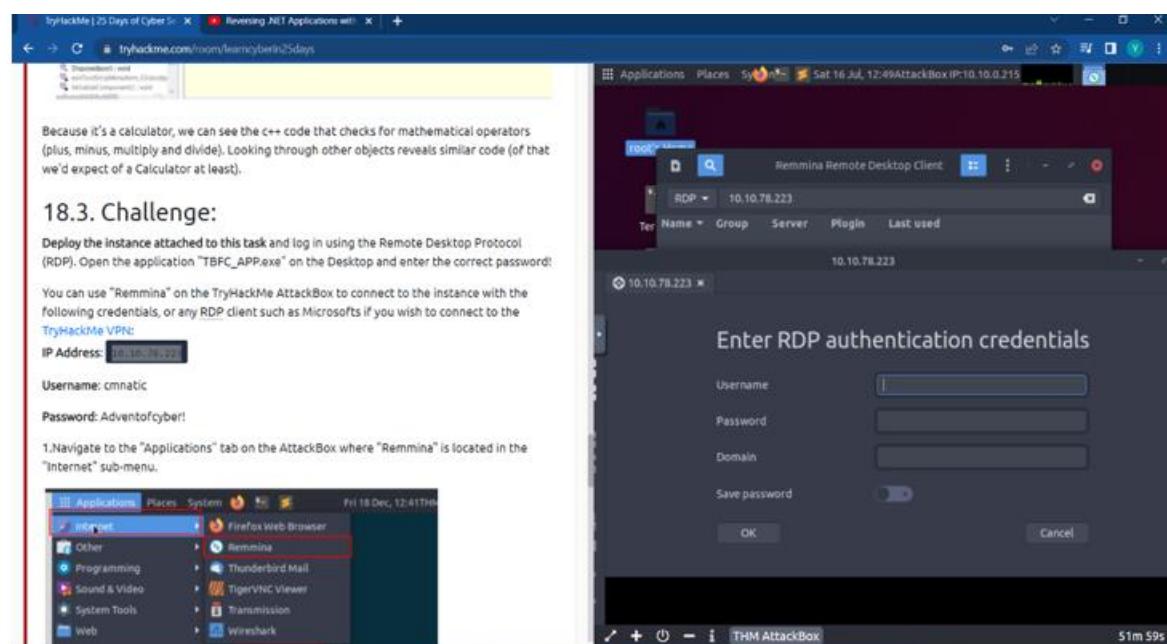
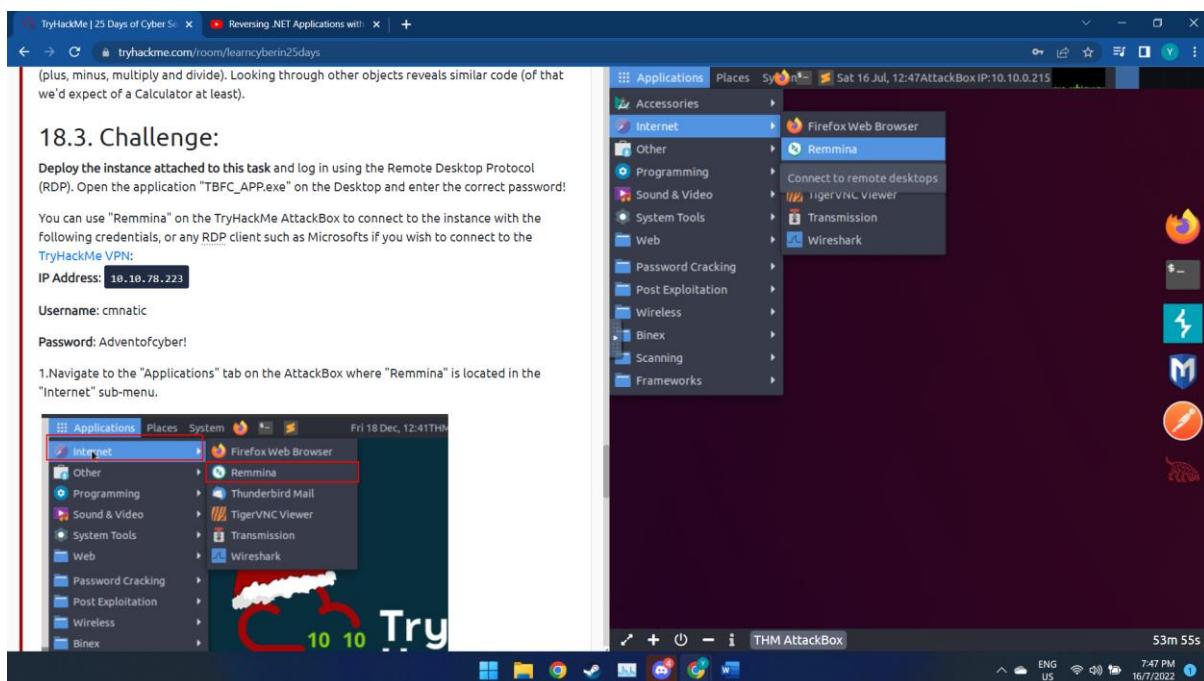
Day 18: Reverse Engineering - The Bits of Christmas

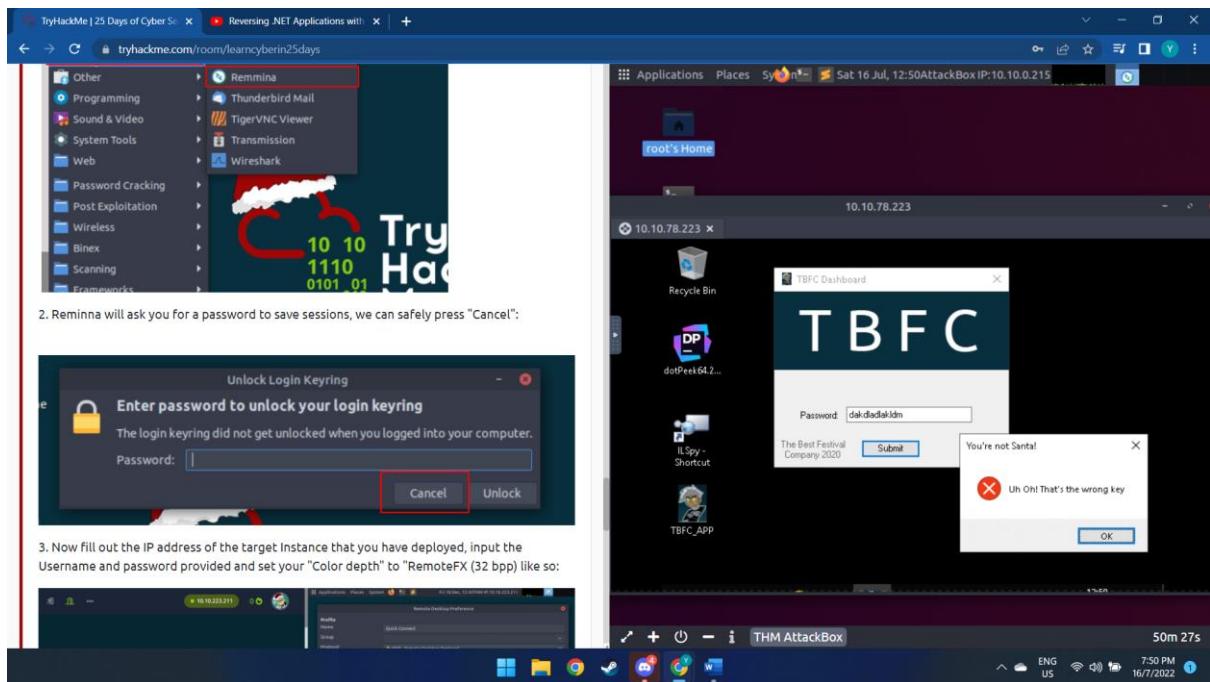
Tools used: IL SPY, Cyberchef, Remmina, TBFC and attack box

Solution/walkthrough:

Question 1

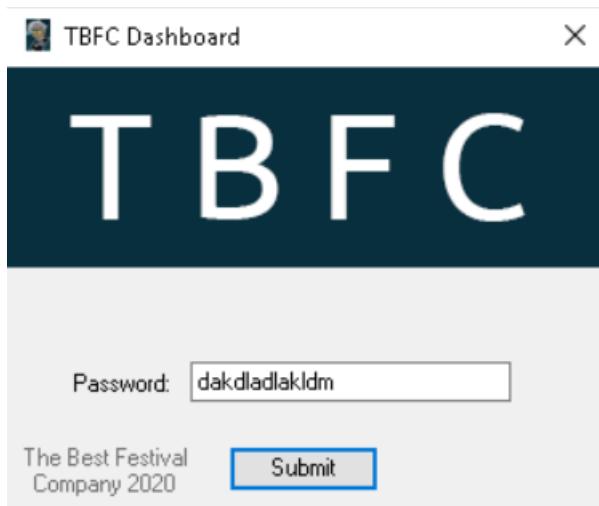
At start, we open Remmina like this. After that, we search the IP address which given in THM. Then, it will come out the page which inside 2nd photo and about username and password, we copy from THM also and it takes us to a new window. Then, we open the TBFC_APP and type any password because we don't know the password yet. At last, it shows a message which is "Uh Oh! That's the wrong key" which is the answer of question 1.





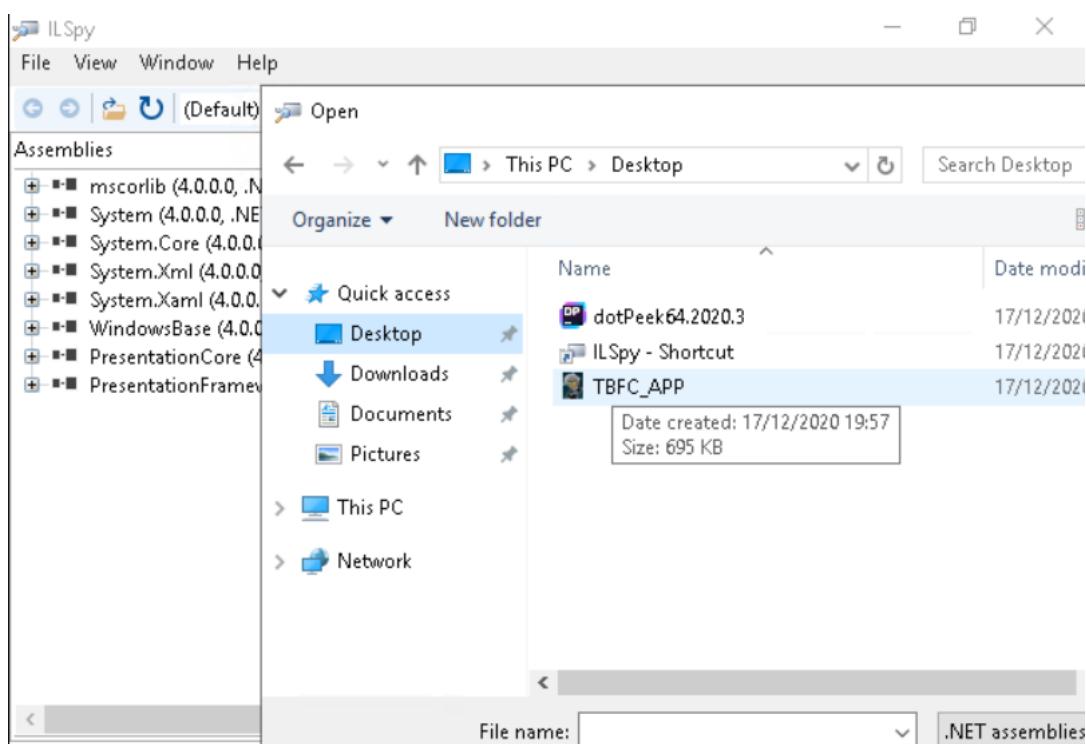
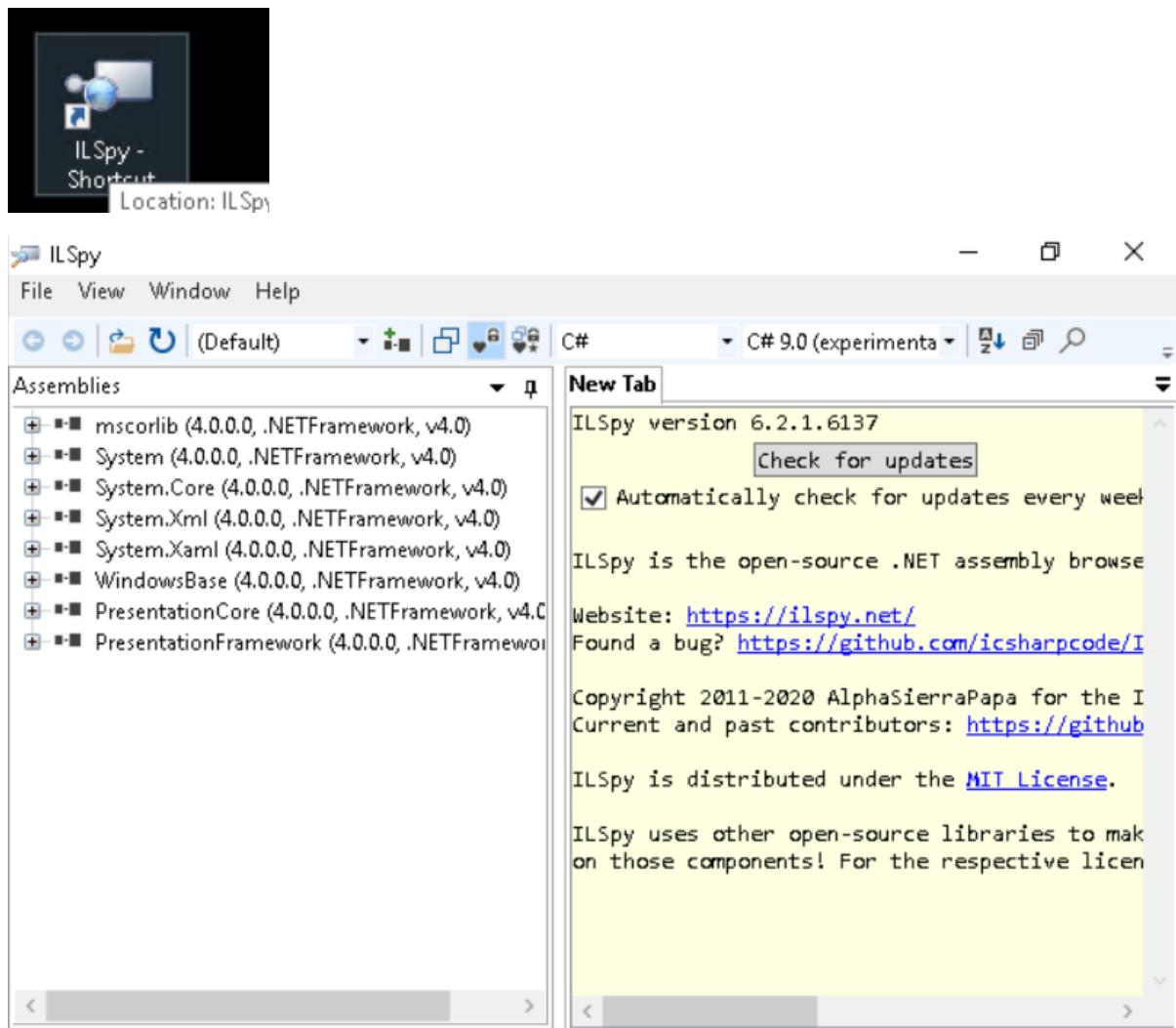
Question 2

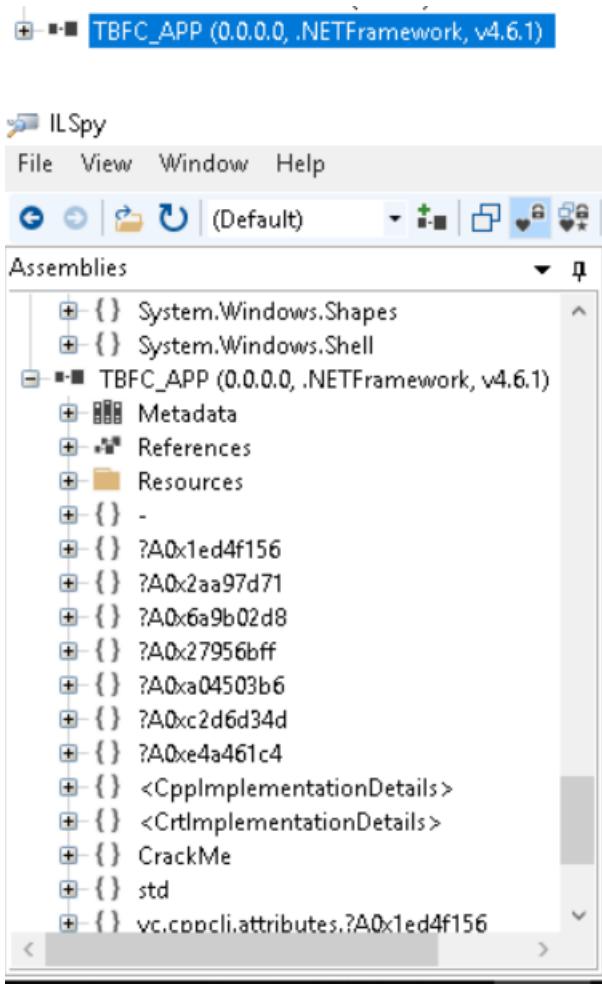
The name of TBFC is at the left bottom corner of the tab as you finish the question 1.



Question 3

We close the TBFC_APP and open another app which is IL Spy. Then, we open TBFC_APP inside IL Spy then it will come out the new things in Assemblies. We open up the applications and there are many stuffs. Metadata, References and Resources are just not important but Crack Me is the module that catches our attention and this module will lead us to solve following questions.





Question 4

About CrackMe, it has 2 form which is AboutForm and MainForm. AboutForm just have a little bit information only and mainly MainForm is the forms that contains many information that we looking for.

ILSpy

File View Window Help

(Default) C# C# 9.0 (experimental)

Assemblies

```
// CrackMe.AboutForm
using ...

public class AboutForm : Form
{
    private Container components;

    public AboutForm()
    {
        try
        {
            InitializeComponent();
        }
        catch
        {
            //try-fault
            base.Dispose(disposing: true);
            throw;
        }
    }

    private void ~AboutForm()
    ...

    private void InitializeComponent()
    ...

    private void AboutForm_Load(object sender, EventArgs e)
    ...

    private void buttonAboutOK_Click(object sender, EventArgs e)
    {
        [HandleProcessCorruptedStateExceptions]
        protected override void Dispose([MarshalAs(UnmanagedType.U1)] bool A_0)
    ...
}
```

CrackMe

- AboutForm
 - Base Types
 - Derived Types
 - components : Container
 - AboutForm()
 - ~AboutForm() : void
 - AboutForm_Load(object, EventArgs)
 - buttonAboutOK_Click(object, EventArgs)
 - Dispose(bool) : void
 - InitializeComponent() : void
- MainForm
- std
- vc.cppcli.attributes.??01ed4f156
- vc.cppcli.attributes.??02aa97d71
- vc.cppcli.attributes.??06a9b02d8
- vc.cppcli.attributes.??027956bff
- vc.cppcli.attributes.??0a04509b6
- vc.cppcli.attributes.??0c26d34d

ILSpy

File View Window Help

(Default) C# C# 9.0 (experimental)

Assemblies

```
// CrackMe.MainForm
using ...

public class MainForm : Form
{
    private Label labelKey;
    private TextBox textBoxKey;
    private Panel panelLogo;
    private TableLayoutPanel tableLayoutPanel1;
    private Button buttonActivate;
    private TableLayoutPanel tableLayoutPanelButtons;
    private Label labelOrg;
    private Container components;

    public MainForm()
    ...

    private void ~MainForm()
    ...

    private void InitializeComponent()
    ...

    private void MainForm_Load(object sender, EventArgs e)
    ...

    private void buttonExit_Click(object sender, EventArgs e)
    ...

    private void buttonAbout_Click(object sender, EventArgs e)
    ...

    private unsafe void buttonActivate_Click(object sender, EventArgs e)
    ...

    private void panelLogo_Paint(object sender, PaintEventArgs e)
    ...

    private void textBoxKey_TextChanged(object sender, EventArgs e)
    ...
}
```

CrackMe

- AboutForm
- MainForm
- std
- vc.cppcli.attributes.??01ed4f156
- vc.cppcli.attributes.??02aa97d71
- vc.cppcli.attributes.??06a9b02d8
- vc.cppcli.attributes.??027956bff
- vc.cppcli.attributes.??0a04509b6
- vc.cppcli.attributes.??0c26d34d
- vc.cppcli.attributes.??0e4a461c4
- System.Data (4.0.0.0, .NETFramework, v4.0)
- System.Drawing (4.0.0.0, .NETFramework, v4.0)
- System.Windows.Forms (4.0.0.0, .NETFramewo

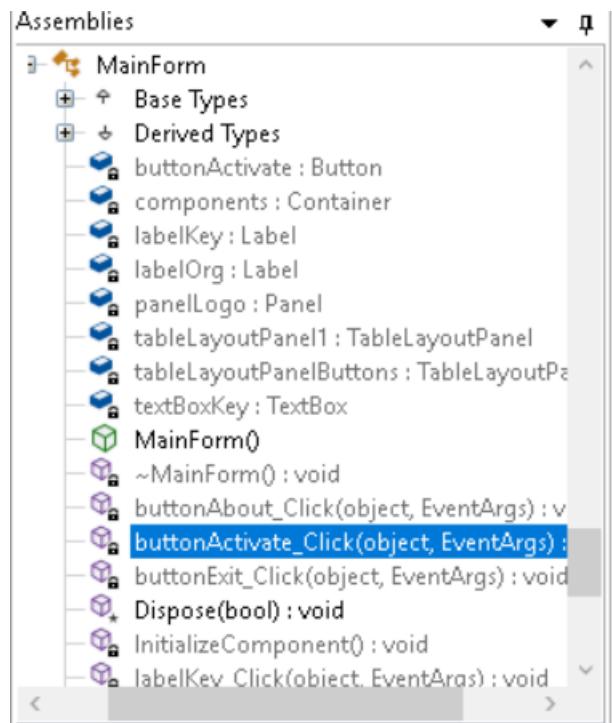
```

private void InitializeComponent()
{
    System.ComponentModel.ComponentResourceManager resources = new System.ComponentModel.ComponentResourceManager(typeof(CrackMe.MainForm));
    labelKey = new System.Windows.Forms.Label();
    textBoxKey = new System.Windows.Forms.TextBox();
    panelLogo = new System.Windows.Forms.Panel();
    tableLayoutPanelPanel1 = new System.Windows.Forms.TableLayoutPanel();
    buttonActivate = new System.Windows.Forms.Button();
    tableLayoutPanelPanelButtons = new System.Windows.Forms.TableLayoutPanel();
    labelOrg = new System.Windows.Forms.Label();
    tableLayoutPanelPanel1.SuspendLayout();
    tableLayoutPanelPanelButtons.SuspendLayout();
    SuspendLayout();
    labelKey.Anchor = System.Windows.Forms.AnchorStyles.Right;
    labelKey.AutoSize = true;
    System.Drawing.Point location = new System.Drawing.Point(30, 14);
    labelKey.Location = location;
    labelKey.Name = "labelKey";
    System.Drawing.Size size = new System.Drawing.Size(56, 13);
    labelKey.Size = size;
    labelKey.TabIndex = 0;
    labelKey.Text = "Password:";
    labelKey.Click += new System.EventHandler(labelKey_Click);
    textBoxKey.Anchor = System.Windows.Forms.AnchorStyles.Left;
    System.Drawing.Point location2 = new System.Drawing.Point(92, 10);
    textBoxKey.Location = location2;
    textBoxKey.Name = "textBoxKey";
    System.Drawing.Size size2 = new System.Drawing.Size(160, 20);
    textBoxKey.Size = size2;
    textBoxKey.TabIndex = 1;
    textBoxKey.TextChanged += new System.EventHandler(textBoxKey_TextChanged);
    panelLogo.BackgroundImage = (System.Drawing.Image)resources.GetObject("panelLogo.BackgroundImage");
    panelLogo.BackgroundImageLayout = System.Windows.Forms.ImageLayout.Center;
    panelLogo.Dock = System.Windows.Forms.DockStyle.Top;
    System.Drawing.Point location3 = new System.Drawing.Point(0, 0);
    panelLogo.Location = location3;
    panelLogo.Name = "panelLogo";
    System.Drawing.Size size3 = new System.Drawing.Size(299, 100);
    panelLogo.Size = size3;
    panelLogo.TabIndex = 2;
    panelLogo.Paint += new System.Windows.Forms.PaintEventHandler(panelLogo_Paint);
    tableLayoutPanelPanel1.ColumnCount = 2;
}

```

Question 5

We take a look at buttonActivate_Click and then we found that the code we looking for.



```

// CrackMe.MainForm
using ...

private unsafe void buttonActivate_Click(object sender, EventArgs e)
{
    IntPtr value = Marshal.StringToGlobalAnsi(textBoxKey.Text);
    sbyte* ptr = (sbyte*)System.Runtime.CompilerServices.Unsafe.AsPointer(<Module>._C@_0BB@IKKDFEPG@santapassword321@);
    void* ptr2 = (void*)value;
    byte b = *(byte*)ptr2;
    byte b2 = 115;
    if ((uint)b >= 115u)
    {
        while ((uint)b <= (uint)b2)
        {
            if (b != 0)
            {
                ptr2 = (byte*)ptr2 + 1;
                ptr++;
                b = *(byte*)ptr2;
                b2 = *(byte*)(*ptr);
                if ((uint)b < (uint)b2)
                {
                    break;
                }
                continue;
            }
            MessageBox.Show("Welcome, Santa, here's your flag thm{046af}", "That's the right key!", MessageBoxButtons.OK, MessageBoxIcon.Asterisk);
            return;
        }
    }
    MessageBox.Show("Uh Oh! That's the wrong key", "You're not Santa!", MessageBoxButtons.OK, MessageBoxIcon.Hand);
}

```

Question 6

We doubleclick the code which under 1st photo and then it lead us to another code. We copy the number inside the bracket and use cyberchef to solve it. Then, we put inside the input and use the hexa recipe to convert it. Then we found the password.



The screenshot shows a debugger interface. A tooltip is displayed over assembly code, showing the memory address `le>.??_C@_0BB@IKKDFEPG@santapassword321@;`. Below the tooltip, the assembly code `.$ArrayType$$$BY0BB@$$CBD global::<Module>._??` is visible. At the bottom of the screen, there is a code editor window containing C# code:

```

// <Module>
using ...

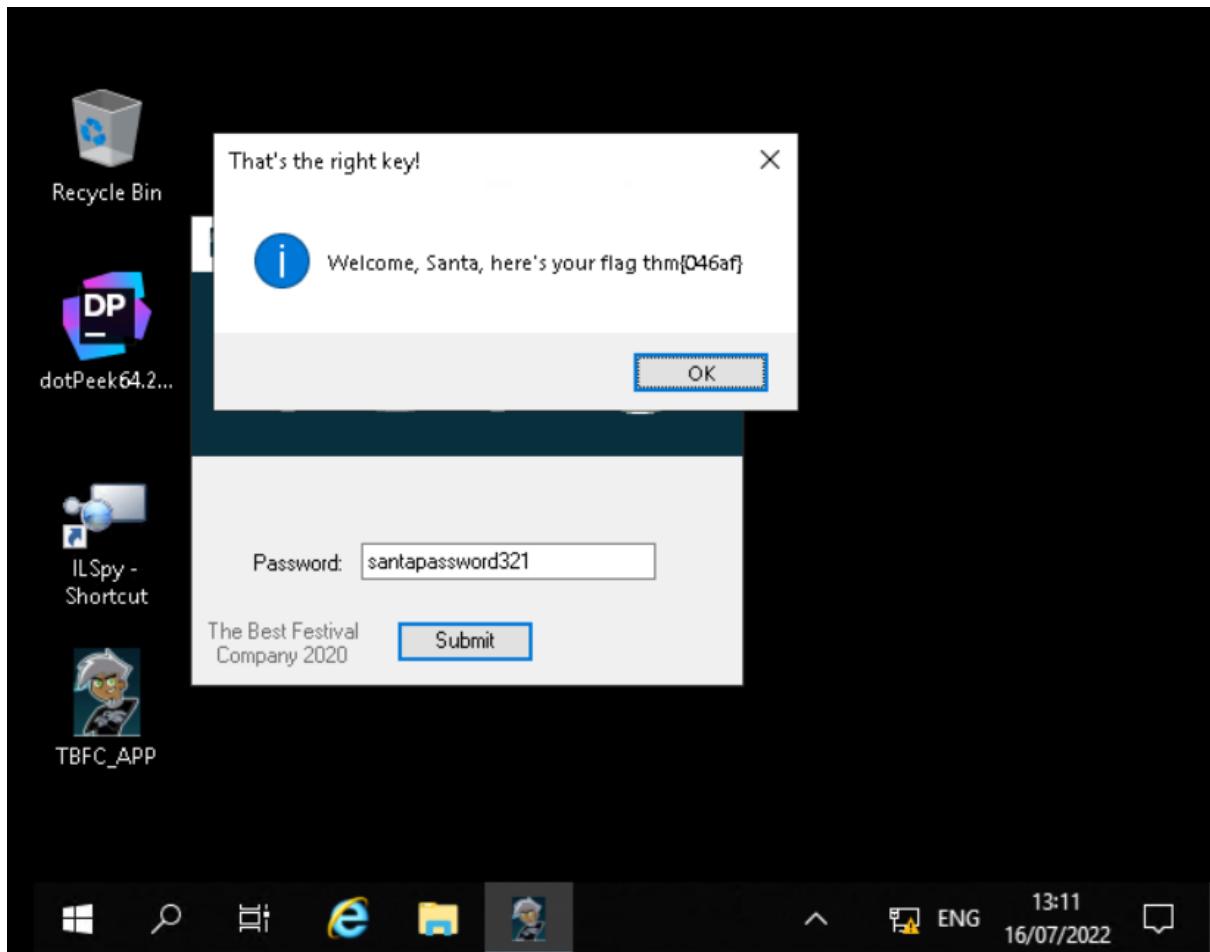
internal static $ArrayType$$$BY0BB@$$CBD _C@_0BB@IKKDFEPG@santapassword321@/* Not supported: data(73 61 6E 74 61 70 61 73 73 77 6F 72 64 33 32 31 00) */;

```

The screenshot shows the CyberChef interface running in Mozilla Firefox. The 'From Hex' recipe is selected in the Recipe list. The Input field contains the hex string: 73 61 6E 74 61 70 61 73 73 77 6F 72 64 33 32 31 00. The Output field shows the ASCII string: santapassword321.. The status bar at the bottom of the browser window says 'AttackBox IP:10.0.2.15'.

Question 7

As we type the correct password which given in Question 6 it will give the flag to us.



Thought Process/Methodology:

At start, we open Remmina like this. After that, we search the IP address which given in THM. Then, it will come out the page which inside 2nd photo and about username and password, we copy from THM also and it takes us to a new window. Then, we open the TBFC_APP and type any password because we don't know the password yet. At last, it shows a message which is "Uh Oh! That's the wrong key" which is the answer of question 1. The name of TBFC is at the left bottom corner of the tab as you finish the question 1. We close the TBFC_APP and open another app which is IL Spy. Then, we open TBFC_APP inside IL Spy then it will comes out the new things in Assemblies. We open up the applications and there are many stuffs. Metadata, References and Resources are just not important but Crack Me is the module that catches our attentions and this module will lead us to solve following questions. About CrackMe, it has 2 form which is AboutForm and MainForm. AboutForm just have a little bit information only and mainly MainForm is the forms that contains many information that we looking for. We take a look at buttonActivate_Click and then we found that the code we looking for. We doubleclick the code which under 1st photo and then it lead us to another code. We copy the number inside the bracket and use cyberchef to solve it. Then, we put inside the input and use the hexa recipe to convert it. Then we found the password. As we type the correct password which given in Question 6 it will give the flag to us.

Day 19: Web Exploitation - The Naughty or Nice List

Tools used: Firefox and Terminal

Solution/walkthrough:

Question 1

Access to URL "http://IP address" and type all the name one by one to see it is naughty or nice.

To find out if you are currently on the naughty list or the nice list, please enter your name below!

Have a Merry Christmas! Ho ho ho!

- Santa

Name: Search

JJ is on the Naughty List.

It looks like you haven't started Firefox in a while. Do you want to clean it up for a fresh, like-new experience? And by the way, welcome back!

Have a Merry Christmas! Ho ho ho!

- Santa

Name: Search

Timothy is on the Naughty List.

It looks like you haven't started Firefox in a while. Do you want to clean it up for a fresh, like-new experience? And by the way, welcome back!

THM Browser-Based

vnc.tryhackme.tech/index.html?host=proxy-3.tryhackme.tech&password=9cf708ff5e7f1e&proxyIP=10.10.170.65&resize=remote

Sun 17 Jul, 07:19

The Naughty or Nice List - Mozilla Firefox

10.10.59.136/?proxy=http%3A%2F%2Flist.hohoho%3A8080%2Fsearch.php%3Fname%3DTib

TryHackMe | Learn Cy... TryHackMe Support Offline CyberChef GitHub - swisskyrepo/... Reverse Shell Cheat S...

The List Admin

Have a Merry Christmas! Ho ho ho!

- Santa

Name: Search

Tib3rius is on the Nice List.

It looks like you haven't started Firefox in a while. Do you want to clean it up for a fresh, like-new experience? And by the way, welcome back!

Refresh Firefox...

THM Browser-Based

vnc.tryhackme.tech/index.html?host=proxy-3.tryhackme.tech&password=9cf708ff5e7f1e&proxyIP=10.10.170.65&resize=remote

Sun 17 Jul, 07:20

The Naughty or Nice List - Mozilla Firefox

10.10.59.136/?proxy=http%3A%2F%2Flist.hohoho%3A8080%2Fsearch.php%3Fname%3DTib

TryHackMe | Learn Cy... TryHackMe Support Offline CyberChef GitHub - swisskyrepo/... Reverse Shell Cheat S...

The List Admin

Have a Merry Christmas! Ho ho ho!

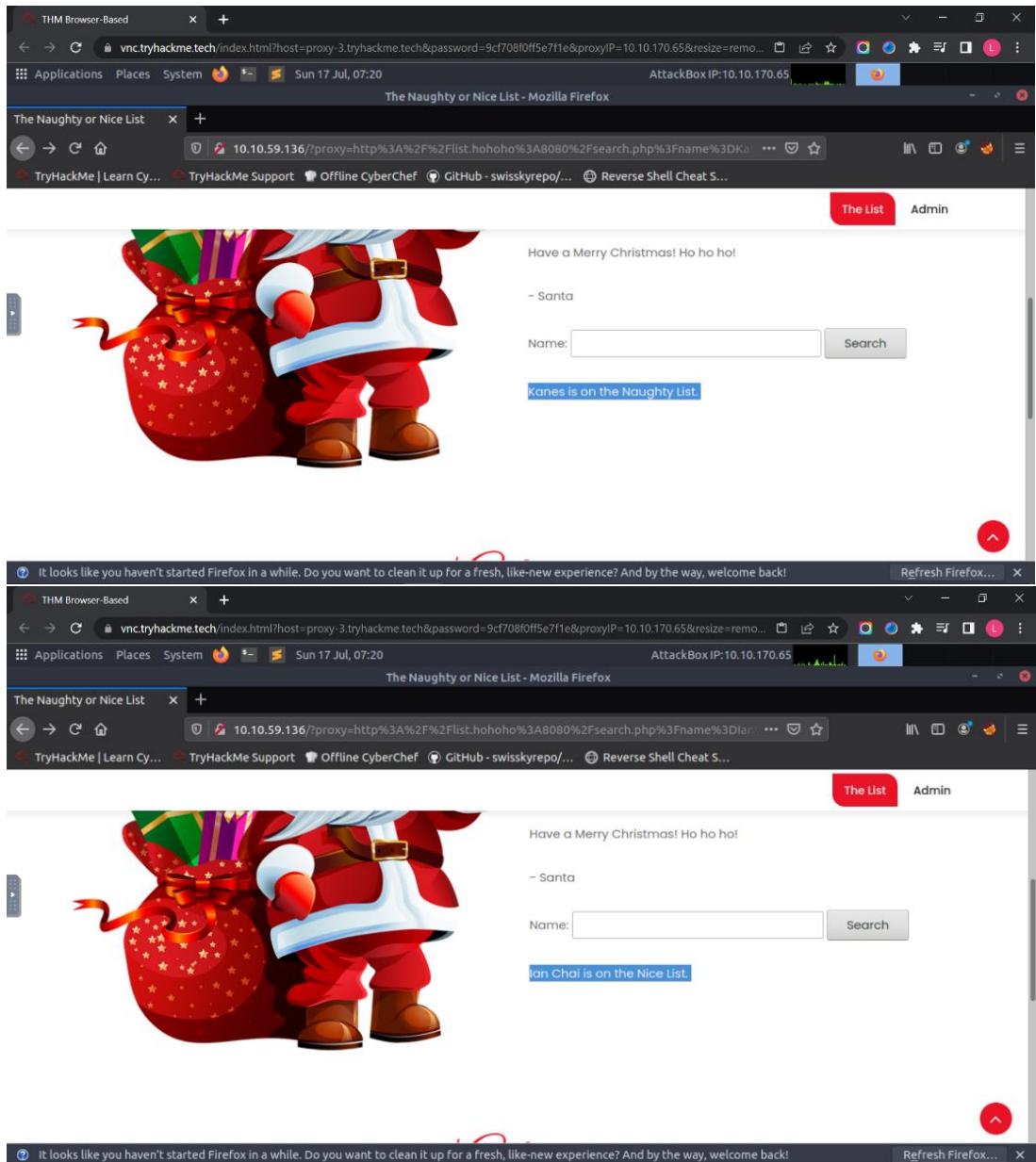
- Santa

Name: Search

Tib3rius is on the Nice List.

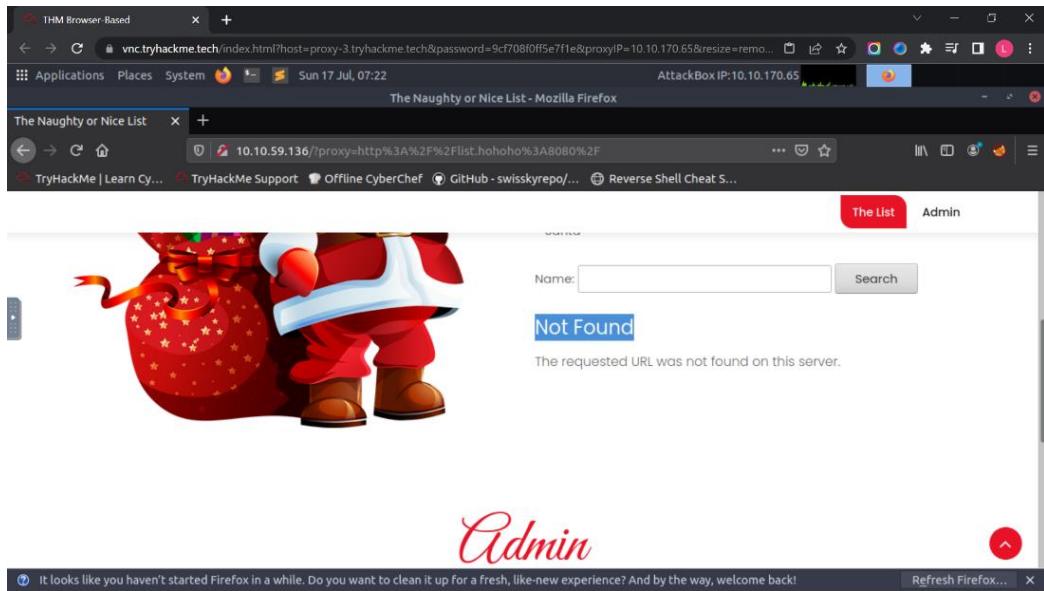
It looks like you haven't started Firefox in a while. Do you want to clean it up for a fresh, like-new experience? And by the way, welcome back!

Refresh Firefox...



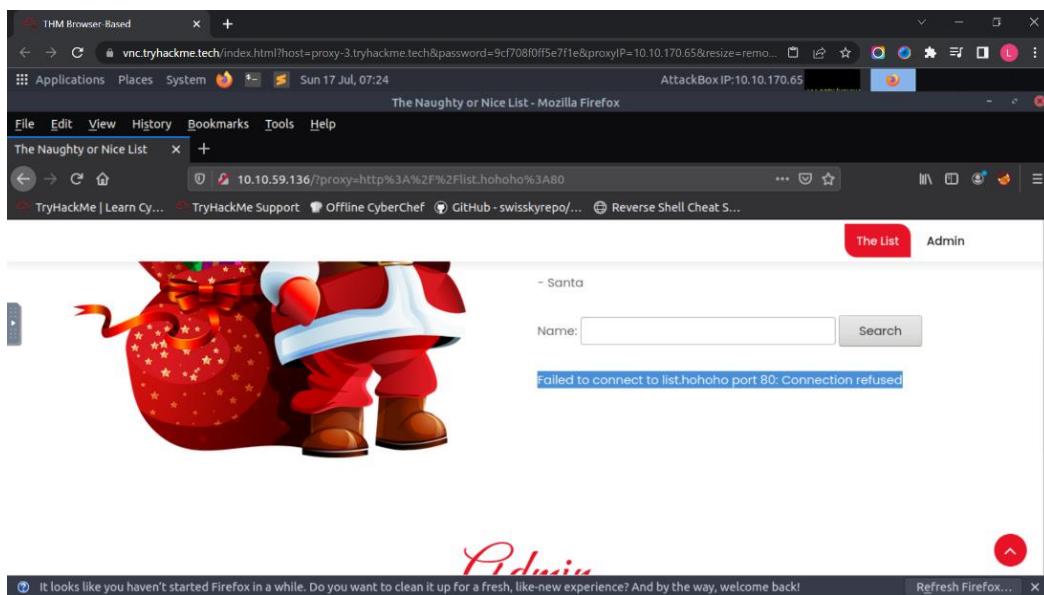
Question 2

Add "?proxy=http%3A%2F%2Flist.hohoho%3A8080%2F" behind the "http://IP address" and get the display of "Not Found. The requested URL was not found on this server."



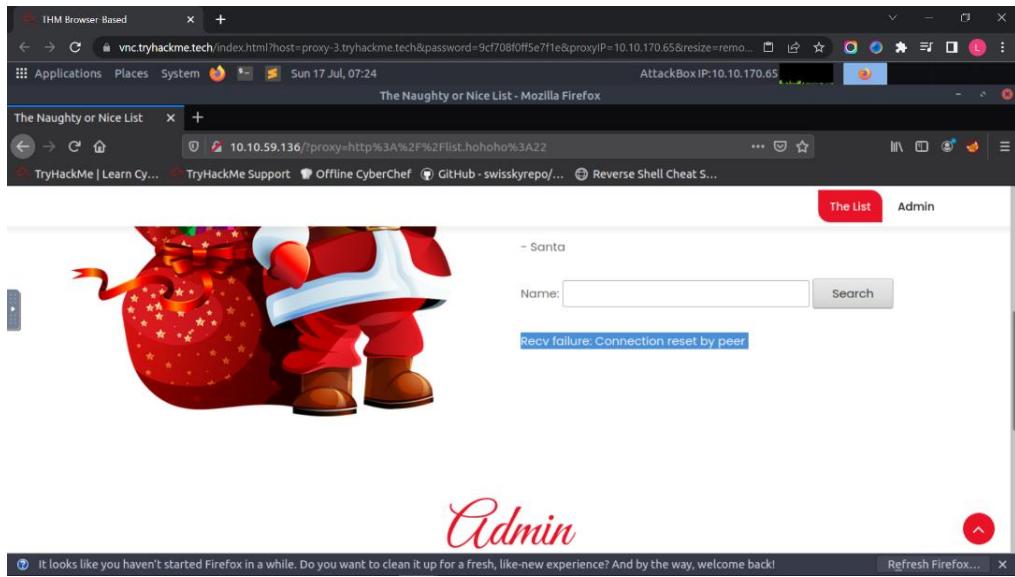
Question 3

Add "?proxy=http%3A%2F%2Flist.hohoho%3A80" behind the "http://IP address" and get the display of "Failed to connect to list.hohoho port 80: Connection refused".



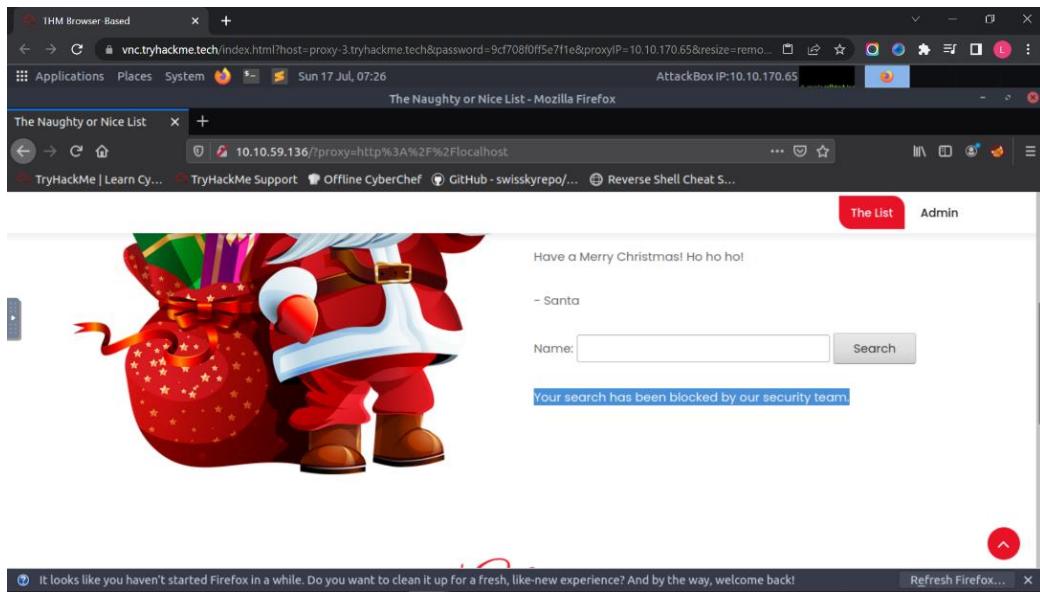
Question 4

Add "?proxy=http%3A%2F%2Flist.hohoho%3A22" behind the "http://IP address" and get the display of "Recv failure: Connection reset by peer".



Question 5

Add "/?proxy=http%3A%2F%2Flocalhost" behind the "http://IP address" and get the display of "Your search has been blocked by our security team.".



Question 6

First, we need to prove that the address is the same 127.0.0.1 by using the terminal. We type "host localtest.me", "host leong.localtest.me" and "host anything.localtest.me" to make sure all of the output is the same address 127.0.0.1. Next, we add "/?proxy=http%3A%2F%2Flist.hohoho.localtest.me" behind the "http://IP address" and get the password of Santa.

```

root@ip-10-10-170-65:~# host localhost.me
localhost.me has address 127.0.0.1
localhost.me has IPv6 address ::1
root@ip-10-10-170-65:~# host leong.localtest.me
leong.localtest.me has address 127.0.0.1
leong.localtest.me has IPv6 address ::1
root@ip-10-10-170-65:~# host anything.localtest.me
anything.localtest.me has address 127.0.0.1
anything.localtest.me has IPv6 address ::1
root@ip-10-10-170-65:~#

```


The Naughty or Nice List - Mozilla Firefox

The Naughty or Nice List

Name: Search

Santa,

If you need to make any changes to the Naughty or Nice list, you need to login.

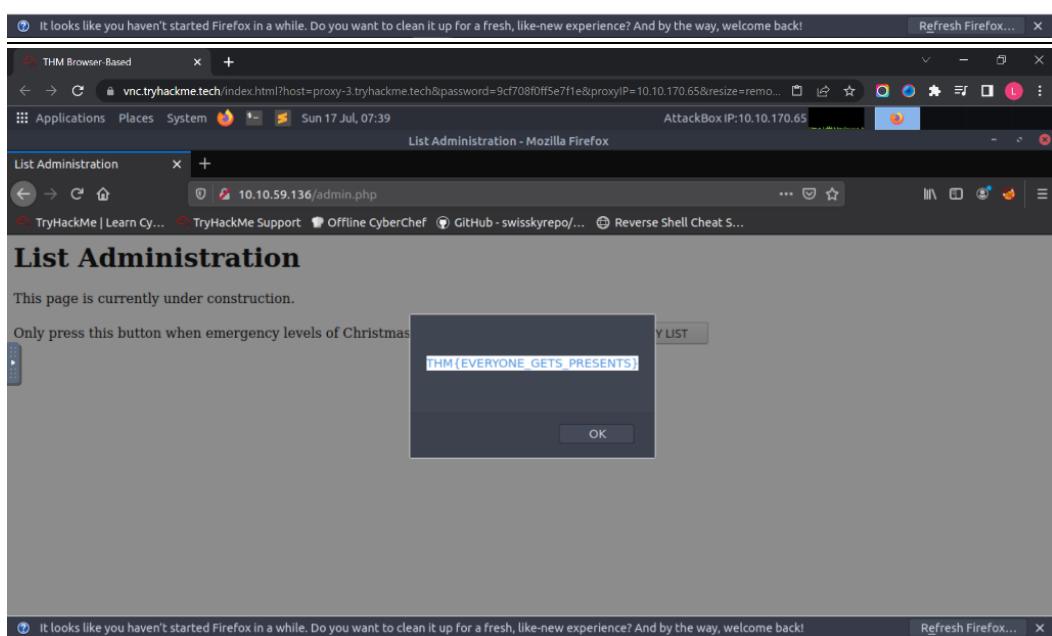
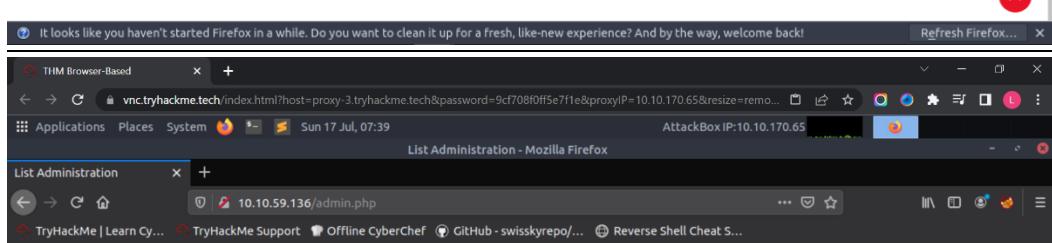
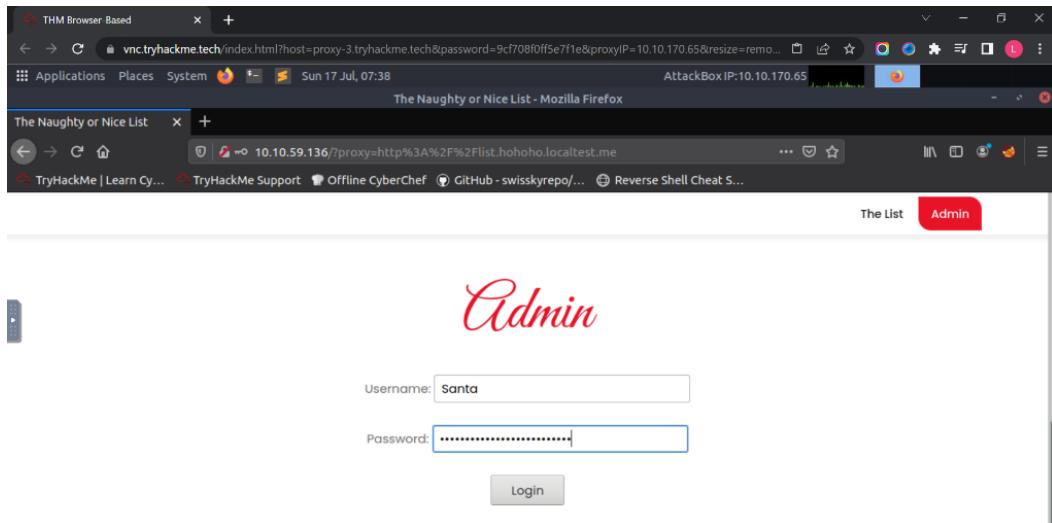
I know you have trouble remembering your password so here it is: Be good for goodness sake!

- Elf McSkidy

It looks like you haven't started Firefox in a while. Do you want to clean it up for a fresh, like-new experience? And by the way, welcome back!

Question 7

First, we type the username “Santa” and the password “Be good for goodness sake!” to access to the next webpage. We will see the List Administration which mentions “This page is currently under construction.” . After that, we click on the button “DELETE NAUGHTY LIST” and get the flag.



Thought Process/Methodology:

First, we access to URL “http://IP address”. Next, we need to prove that the address is the same 127.0.0.1 by using the terminal. We type “host localtest.me”, “host leong.localtest.me” and “host anything.localtest.me” to make sure all of the output is the same address 127.0.0.1. Besides, we add “/?proxy=http%3A%2F%2Flist.hohoho.localtest.me” behind the “http://IP address” and get the password of Santa. First, we type the username “Santa” and the password “Be good for goodness sake!” to access to the next webpage. We will see the List Administration which mentions “This page is currently under construction.” . Lastly, we click on the button “DELETE NAUGHTY LIST” and get the flag.

Day 20: Blue Teaming - Powershell to the rescue

Tools used: Terminal and PowerShell

Solution/walkthrough:

Question 1

The parameter -l is used for the login name.

-i **identity_file** A file from which the [identity key](#) (private key) for [public key authentication](#) is read.

-J [user@]host[:port] Connect to the target host by first making a ssh connection to the pjump host[(:/iam/jump-host) and then establishing a [TCP forwarding](#) to the ultimate destination from there.

-l login_name Specifies the user to log in as on the remote machine.

-p port Port to connect to on the remote host.

-q Quiet mode.

-V Display the version number.

-v Verbose mode.

-X Enables X11 forwarding.

A little history

SSH replaced several older commands and protocols in Unix and Linux the 1990s. These include [telnet](#), [rlogin](#), and [rsh](#).

SSH runs at TCP/IP port 22. This is right between [ftp](#) and telnet, which are 20 years older.

Question 2

First, we type “ssh -l mceager IP address” and the password “r0ckStar!” in the terminal. Next, we type the command “cd Documents” and we are in the documents file. We type the command “Get-ChildItem -File -Hidden” to find all the files and hidden items and type the command “ls” to find out those files which are not hidden. We type the command “Get-Content elfone.txt” to check whatever

got anything inside “elfone.txt”. Lastly, we type the command “cat e1fone.txt” from the hidden file and we will get the content from Elf 1.

```
THM Browser-Based x +
← → C vnc.tryhackme.tech/index.html?host=proxy-2.tryhackme.tech&password=83b31dd568ff072&proxyIP=10.10.205.162&resize=re... Applications Places System Sun 17 Jul, 08:47 AttackBox IP:10.10.205.162
File Edit View Search Terminal Help
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

PS C:\Users\mceager\Documents> ls
Directory: C:\Users\mceager\Documents

Mode LastWriteTime Length Name
---- ----- ------
-a--- 11/23/2020 12:06 PM 22 elfone.txt

PS C:\Users\mceager\Documents> Get-Content e1fone.txt
Nothing to see here...
PS C:\Users\mceager\Documents> cat e1fone.txt
All I want is my '2 front teeth'!!!
PS C:\Users\mceager\Documents>
```

Question 3

First, we need to change the file to Desktop by using the command “cd ..” and “Set-Location Desktop”. Next, we type the command “ls -Hidden -Directory” to find out the directory of the hidden items. We need to get in the file “elf2wo” by using the command “cd elf2wo” and use comment “Get-ChildItem” to list out the file inside “elf2wo” which is “e70smsW10Y4K.txt”. Lastly, we use the command “cat e70smsW10Y4K.txt” to get the content from this file.

```

THM Browser-Based
vnc.tryhackme.tech/index.html?host=proxy-2.tryhackme.tech&password=83b31ddf568ff072&proxyIP=10.10.205.162&resize=re...
Sun 17 Jul, 09:03
AttackBox IP:10.10.205.162
c:\windows\system32\cmd.exe - powershell

File Edit View Search Terminal Help
root@lp-10-10-205-162:~# ssh -l mceager 10.10.27.160
mceager@10.10.27.160's password:
PS C:\Users\mceager\Documents> Get-ChildItem -File -Hidden

Directory: C:\Users\mceager\Documents
.....
PS C:\Users\mceager> Set-Location Desktop
PS C:\Users\mceager\Desktop> ls
PS C:\Users\mceager\Desktop> ls -Hidden

Directory: C:\Users\mceager\Desktop
Mode LastWriteTime Length Name
---- ----- ------
d--h-- 12/7/2020 11:26 AM 282 elftwo
-a-hs- 12/7/2020 10:29 AM 282 desktop.inl

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

Directory: C:\Users\mceager\Desktop
Mode LastWriteTime Length Name
---- ----- ------
d--h-- 12/7/2020 11:26 AM 282 elftwo

PS C:\Users\mceager> Set-Location Desktop
PS C:\Users\mceager\Desktop> ls -Hidden -Directory

Directory: C:\Users\mceager\Desktop
Mode LastWriteTime Length Name
---- ----- ------
d--h-- 12/7/2020 11:26 AM 282 elftwo
-a-hs- 12/7/2020 10:29 AM 282 desktop.inl

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

Directory: C:\Users\mceager\Desktop
Mode LastWriteTime Length Name
---- ----- ------
d--h-- 12/7/2020 11:26 AM 282 elftwo
-a-hs- 12/7/2020 10:29 AM 282 desktop.inl

PS C:\Users\mceager\Desktop> cd elftwo
PS C:\Users\mceager\Desktop\elftwo> Get-ChildItem

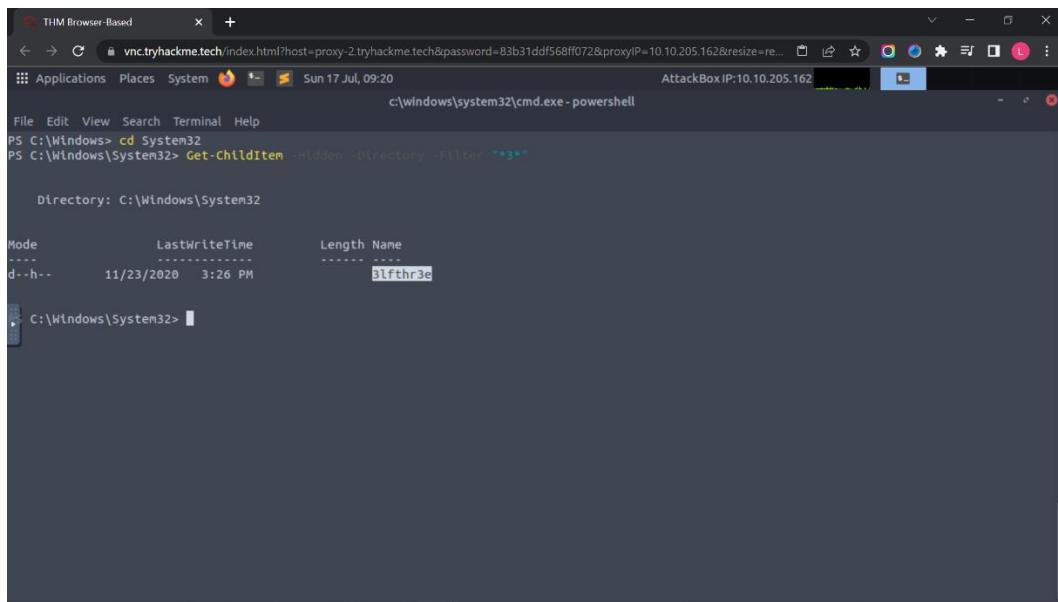
Directory: C:\Users\mceager\Desktop\elftwo
Mode LastWriteTime Length Name
---- ----- ------
-a---- 11/17/2020 10:26 AM 64 e70smsW10Y4k.txt

PS C:\Users\mceager\Desktop\elftwo> cat e70smsW10Y4k.txt
I want the movie Scrooged <3!
PS C:\Users\mceager\Desktop\elftwo>

```

Question 4

First, we need to back to Windows and get to the files name “System32” by using the command “cd Windows” and “cd System32”. Lastly, we type the command “Get-ChildItem -Hidden -Directory -Filter “*3***” to get the directory of the hidden item named with only the integer 3.



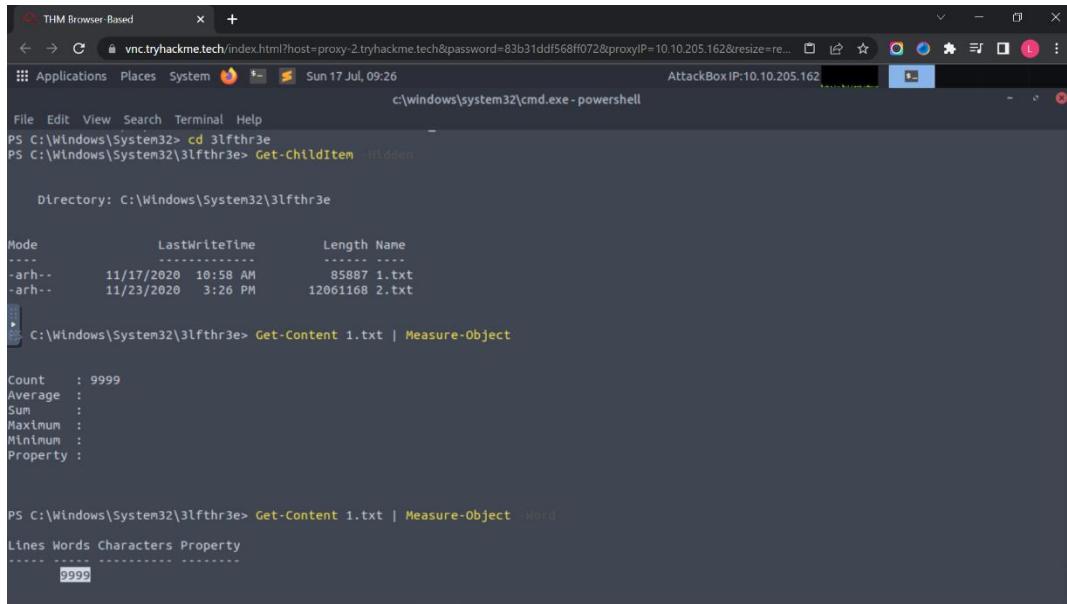
```
vnc.tryhackme.tech/index.html?host=proxy-2.tryhackme.tech&password=83b31ddf568ff072&proxyIP=10.10.205.162&resize=re... Sun 17 Jul, 09:20 c:\windows\system32\cmd.exe - powershell
File Edit View Search Terminal Help
PS C:\Windows> cd System32
PS C:\Windows\System32> Get-ChildItem -Hidden -Directory -Filter "3*"
Directory: C:\Windows\System32

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

C:\Windows\System32>
```

Question 5

First, we need to find the hidden item in “3lfthr3e” by using the command “Get-ChildItem -Hidden”. Next, we type the command “Get-Content 1.txt | Measure-Object” to know the words that the first file contain.



```
vnc.tryhackme.tech/index.html?host=proxy-2.tryhackme.tech&password=83b31ddf568ff072&proxyIP=10.10.205.162&resize=re... Sun 17 Jul, 09:26 c:\windows\system32\cmd.exe - powershell
File Edit View Search Terminal Help
PS C:\Windows\System32> cd 3lfthr3e
PS C:\Windows\System32\3lfthr3e> Get-ChildItem -Hidden
Directory: C:\Windows\System32\3lfthr3e

Mode                LastWriteTime      Length Name
----                -----          ---- 
-арх--       11/17/2020 10:58 AM     85887 1.txt
-арх--       11/23/2020 3:26 PM    12061168 2.txt

PS C:\Windows\System32\3lfthr3e> Get-Content 1.txt | Measure-Object
Count : 9999
Average :
Sum :
Maximum :
Minimum :
Property :

PS C:\Windows\System32\3lfthr3e> Get-Content 1.txt | Measure-Object -Word
Lines Words Characters Property
----- ----- ----- -----
9999
```

Question 6

We type the command “Get-Content 1.txt | Select-Object -Index 551,6991” to find out the two words in the index 551 and 6991 in the first file.

```
vnc.tryhackme.tech/index.html?host=proxy-2.tryhackme.tech&password=83b31dd568ff072&proxyIP=10.10.205.162&resize=re... Sun 17 Jul, 09:31 AttackBoxIP:10.10.205.162 c:\windows\system32\cmd.exe - powershell
File Edit View Search Terminal Help
PS C:\Windows\System32\3lfthr3e> Get-Content 1.txt | Select-Object
Red
Ryder
PS C:\Windows\System32\3lfthr3e>
```

Question 7

We use the command “Get-Content 2.txt | Select-String -Pattern “redryder” to get the full answer in the second file.

```
vnc.tryhackme.tech/index.html?host=proxy-2.tryhackme.tech&password=83b31dd568ff072&proxyIP=10.10.205.162&resize=re... Sun 17 Jul, 09:36 AttackBoxIP:10.10.205.162 c:\windows\system32\cmd.exe - powershell
File Edit View Search Terminal Help
PS C:\Windows\System32\3lfthr3e> Get-Content 2.txt | Select-String -Pattern "redryder"
redryderbbgun
PS C:\Windows\System32\3lfthr3e>
```

Thought Process/Methodology:

Firstly, we type “ssh -l mceager IP address” and the password “r0ckStar!” in the terminal. Next, we type the command “cd Documents” and we are in the documents file. We type the command “Get-ChildItem -File -Hidden” to find all the files and hidden items and type the command “ls” to find out those files which are not hidden. We type the command “Get-Content elfone.txt” to check whatever got anything inside “elfone.txt”. To conclude, we type the command “cat e1fone.txt” from the hidden file and we will get the content from Elf 1. Secondly, we need to change the file to Desktop by using the command “cd ..” and “Set-Location Desktop”. Next, we type the command “ls -Hidden -Directory” to find out the directory of the hidden items. We need to get in the file “elf2wo” by using the command “cd elf2wo” and use command “Get-ChildItem” to list out the file inside “elf2wo”

which is “e70smsW10Y4K.txt”. In a nutshell, we use the command “cat e70smsW10Y4K.txt” to get the content from this file. Thirdly, we need to back to Windows and get to the files name “System32” by using the command “cd Windows” and “cd System32”. Lastly, we type the command “Get-ChildItem -Hidden -Directory -Filter “*3*”” to get the directory of the hidden item named with only the integer 3. After that, we need to find the hidden item in “3lfthr3e” by using the command “Get-ChildItem -Hidden”. Next, we type the command “Get-Content 1.txt | Measure-Object” to know the words that the first file contain. We type the command “Get-Content 1.txt | Select-Object -Index 551,6991” to find out the two words in the index 551 and 6991 in the first file for question 6. We use the command “Get-Content 2.txt | Select-String -Pattern “redryder” to get the full answer in the second file for question 7.