PSP0201 Week 5 Writeup

Group Name: OraOraOra

Members

ID	Name	Role
1211103141	Muhammad Haikal Afiq Bin Rafingei	Leader
1211103148	Muhamad Izzul Iqbal Bin Ismail	Member
1211103830	Hakeem Bin Aminudin	Member

Day 16: Scripting - Help! Where is Santa?

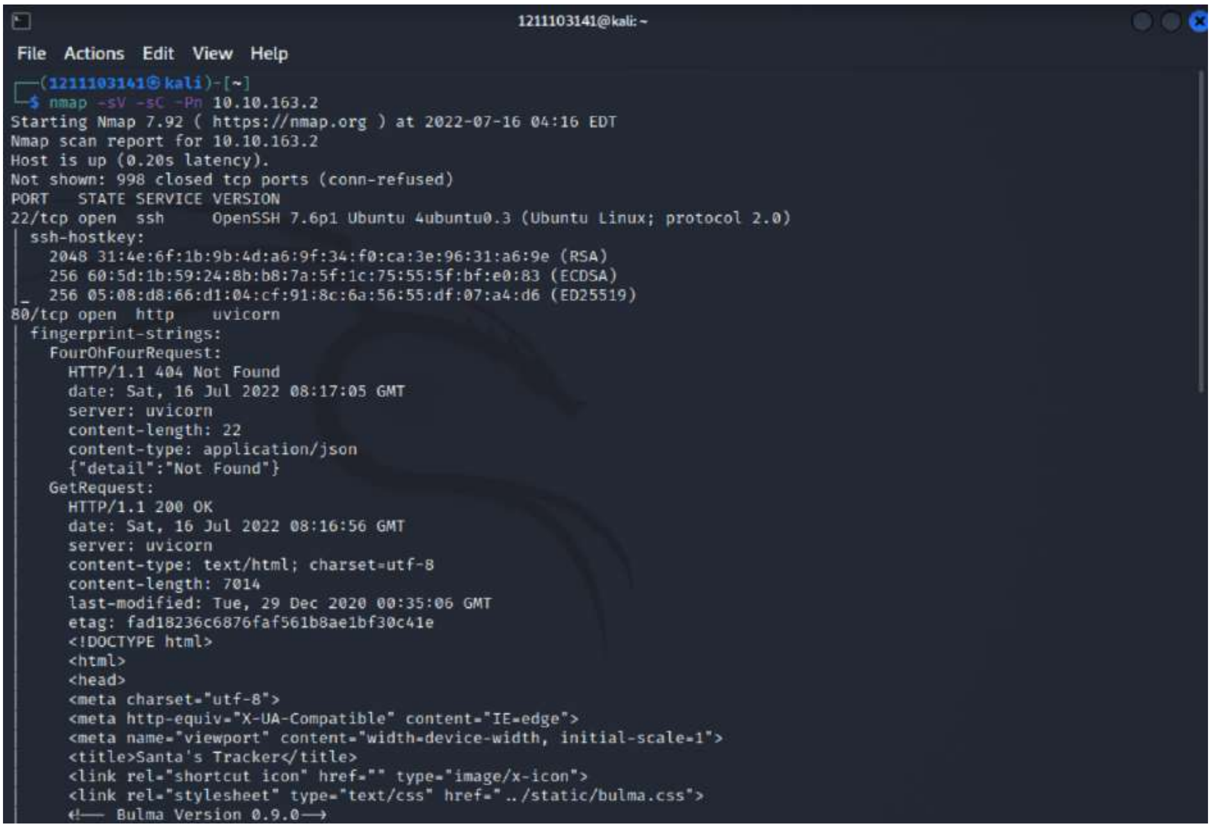
Tools Used: Kali Linux

Solution/Walkthrough:

Question 1

Answer: 80

Scan all available ports on the machine by using nmap. We can see that port 80 is the open http port which is the webserver.

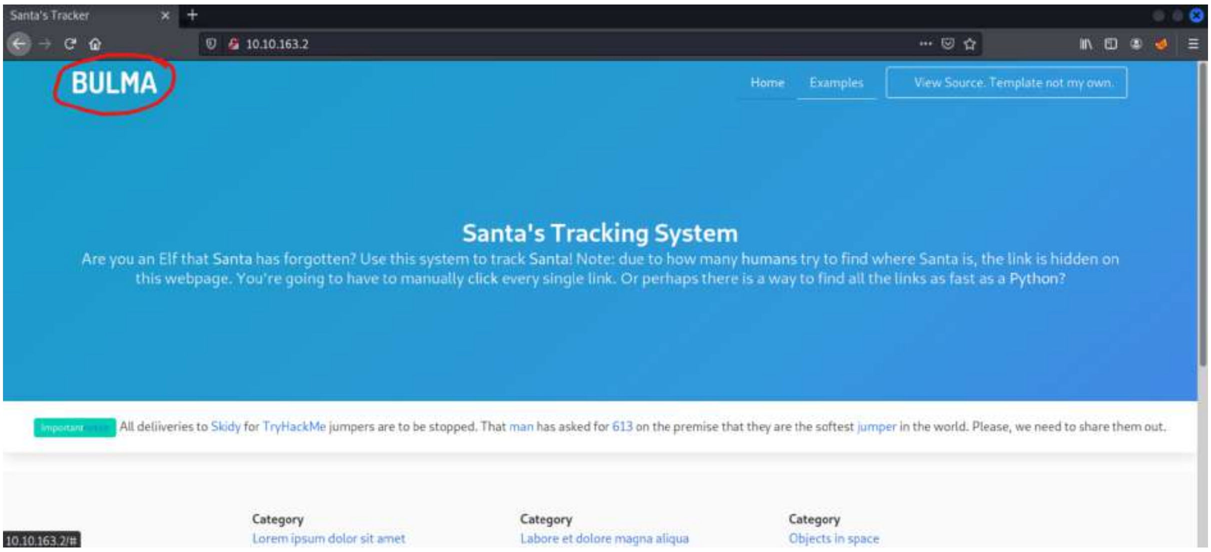


```
┌──(1211103141㉿kali)-[~]
└─$ nmap -sV -sC -Pn 10.10.163.2
Starting Nmap 7.92 ( https://nmap.org ) at 2022-07-16 04:16 EDT
Nmap scan report for 10.10.163.2
Host is up (0.20s latency).
Not shown: 998 closed tcp ports (conn-refused)
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 31:4e:6f:1b:9b:4d:a6:9f:34:f0:ca:3e:96:31:a6:9e (RSA)
|   256 60:5d:1b:59:24:8b:b8:7a:5f:1c:75:55:5f:bf:e0:83 (ECDSA)
|_  256 05:08:d8:66:d1:04:cf:91:8c:6a:56:55:df:07:a4:d6 (ED25519)
80/tcp open  http    uvicorn
| fingerprint-strings:
|   FourOhFourRequest:
|     HTTP/1.1 404 Not Found
|     date: Sat, 16 Jul 2022 08:17:05 GMT
|     server: uvicorn
|     content-length: 22
|     content-type: application/json
|     {"detail":"Not Found"}
|   GetRequest:
|     HTTP/1.1 200 OK
|     date: Sat, 16 Jul 2022 08:16:56 GMT
|     server: uvicorn
|     content-type: text/html; charset=utf-8
|     content-length: 7014
|     last-modified: Tue, 29 Dec 2020 00:35:06 GMT
|     etag: fad18236c6876faf561b8ae1bf30c41e
|     <!DOCTYPE html>
|     <html>
|     <head>
|     <meta charset="utf-8">
|     <meta http-equiv="X-UA-Compatible" content="IE=edge">
|     <meta name="viewport" content="width=device-width, initial-scale=1">
|     <title>Santa's Tracker</title>
|     <link rel="shortcut icon" href="" type="image/x-icon">
|     <link rel="stylesheet" type="text/css" href="../static/bulma.css">
|     <!-- Bulma Version 0.9.0-->
```

Question 2

Answer: BULMA

When you open the website using the ip address, the template name is shown at the top left of the website.



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 deliiveries 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
Lorem ipsum dolor sit amet

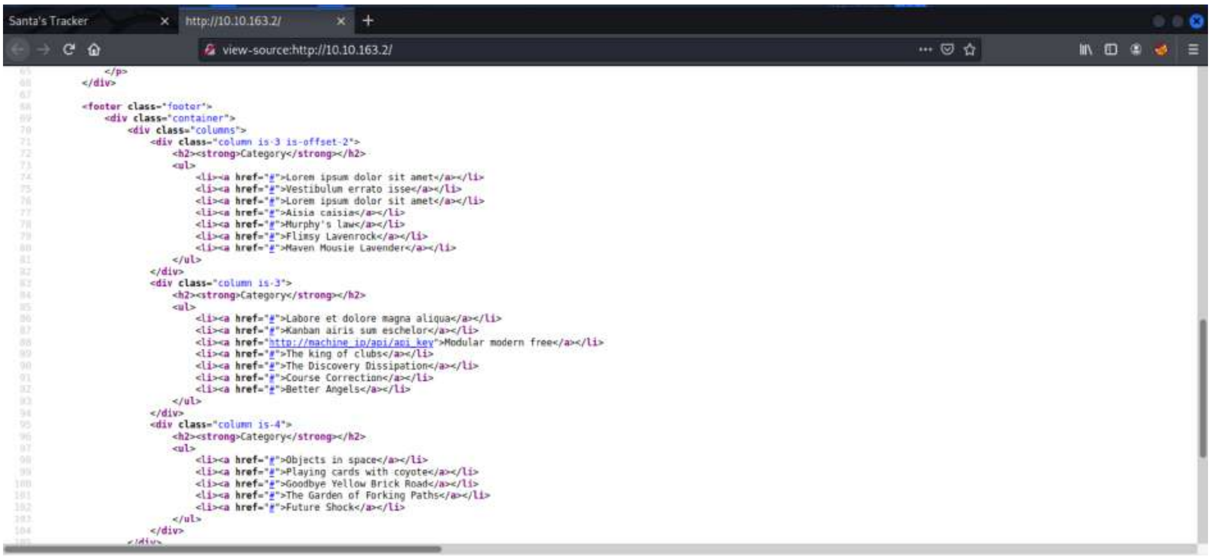
Category
Labore et dolore magna aliqua

Category
Objects in space

Question 3

Answer: /api/

View the page source and find the api directory.

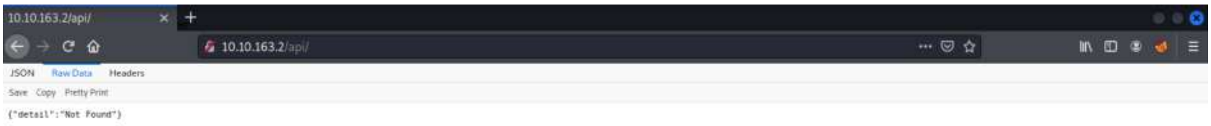


```
        </p>
    </div>

    <footer class="footer">
        <div class="container">
            <div class="columns">
                <div class="column is-3 is-offset-2">
                    <h2><strong>Category</strong></h2>
                    <ul>
                        <li><a href="#">Lorem ipsum dolor sit amet</a></li>
                        <li><a href="#">Vestibulum errato isse</a></li>
                        <li><a href="#">Lorem ipsum dolor sit amet</a></li>
                        <li><a href="#">Aisia caisia</a></li>
                        <li><a href="#">Murphy's law</a></li>
                        <li><a href="#">Flimsy Lavenrock</a></li>
                        <li><a href="#">Maven Mousie Lavender</a></li>
                    </ul>
                </div>
                <div class="column is-3">
                    <h2><strong>Category</strong></h2>
                    <ul>
                        <li><a href="#">Labore et dolore magna aliqua</a></li>
                        <li><a href="#">Kanban airis sum eschelor</a></li>
                        <li><a href="http://machine_ip/api/api_key">Modular modern free</a></li>
                        <li><a href="#">The king of clubs</a></li>
                        <li><a href="#">The Discovery Dissipation</a></li>
                        <li><a href="#">Course Correction</a></li>
                        <li><a href="#">Better Angels</a></li>
                    </ul>
                </div>
                <div class="column is-4">
                    <h2><strong>Category</strong></h2>
                    <ul>
                        <li><a href="#">Objects in space</a></li>
                        <li><a href="#">Playing cards with coyote</a></li>
                        <li><a href="#">Goodbye Yellow Brick Road</a></li>
                        <li><a href="#">The Garden of Forking Paths</a></li>
                        <li><a href="#">Future Shock</a></li>
                    </ul>
                </div>
            </div>
```

Question 4

Answer: {"detail":"Not Found"}

Open the api endpoint without the api key or parameter. Change from JSON to raw data.

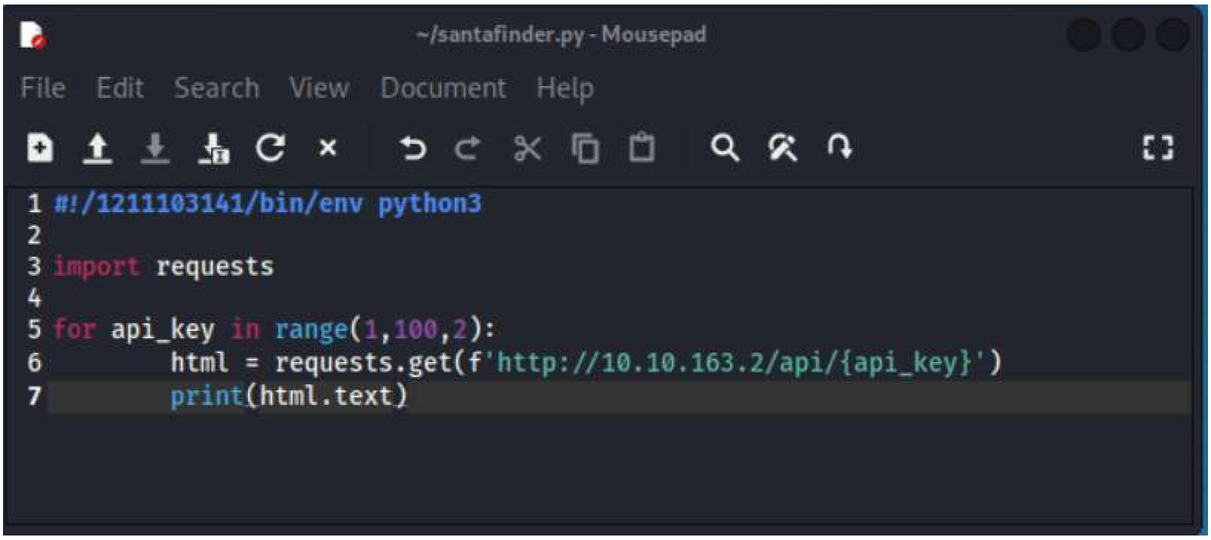


```
{"detail":"Not Found"}
```

Question 5

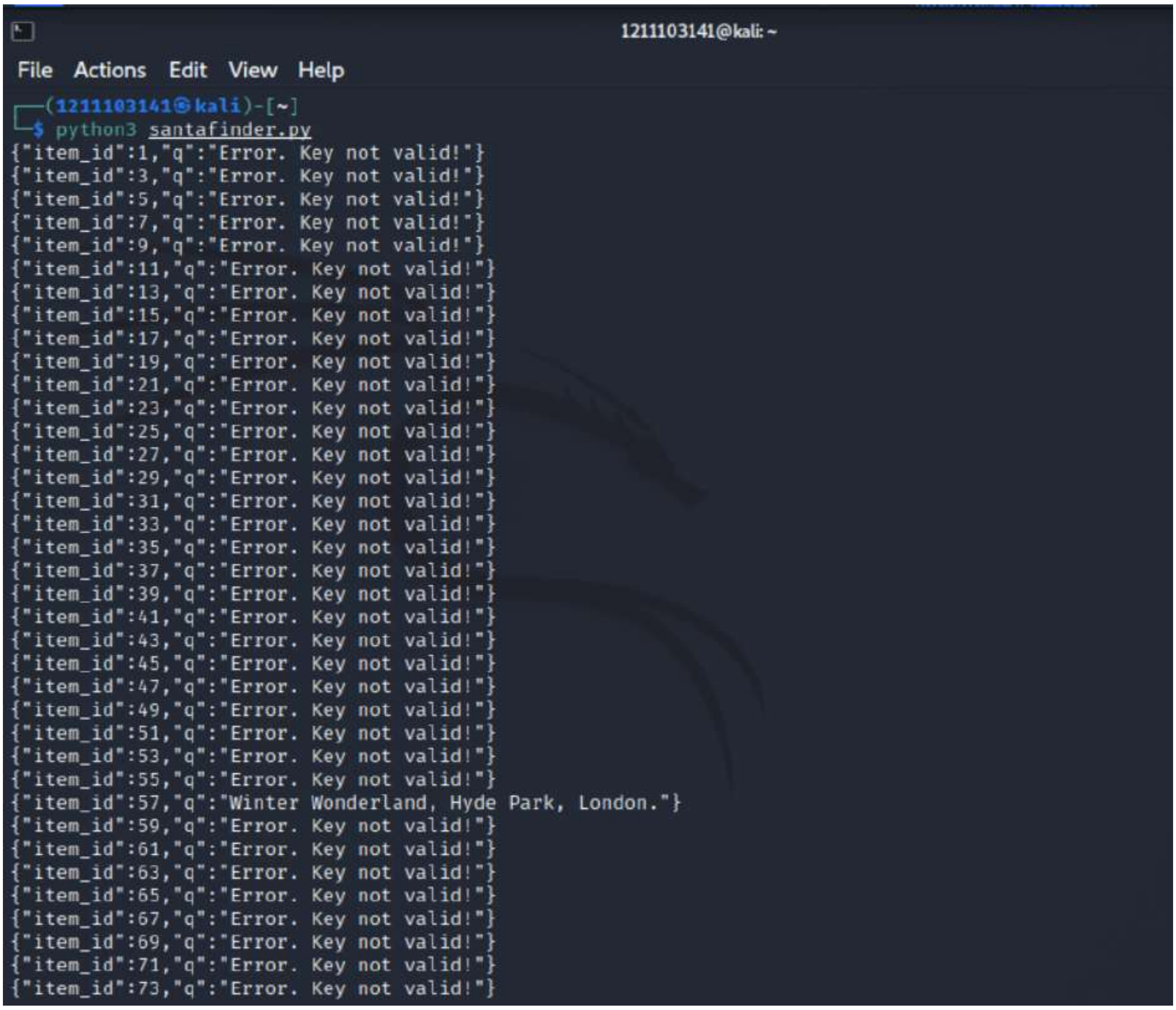
Answer: Winter Wonderland, Hyde Park, London

Make a python code at any text editor that will retrieve data for each odd number api key parameter from the website.



```python
#!/1211103141/bin/env python3

import requests

for api_key in range(1,100,2):
        html = requests.get(f'http://10.10.163.2/api/{api_key}')
        print(html.text)
```

Run the file by using the terminal and you will find Santa's place.

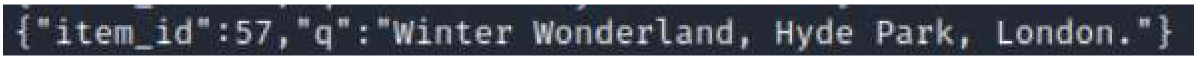


```
┌──(1211103141㉿kali)-[~]
└─$ python3 santafinder.py
{"item_id":1,"q":"Error. Key not valid!"}
{"item_id":3,"q":"Error. Key not valid!"}
{"item_id":5,"q":"Error. Key not valid!"}
{"item_id":7,"q":"Error. Key not valid!"}
{"item_id":9,"q":"Error. Key not valid!"}
{"item_id":11,"q":"Error. Key not valid!"}
{"item_id":13,"q":"Error. Key not valid!"}
{"item_id":15,"q":"Error. Key not valid!"}
{"item_id":17,"q":"Error. Key not valid!"}
{"item_id":19,"q":"Error. Key not valid!"}
{"item_id":21,"q":"Error. Key not valid!"}
{"item_id":23,"q":"Error. Key not valid!"}
{"item_id":25,"q":"Error. Key not valid!"}
{"item_id":27,"q":"Error. Key not valid!"}
{"item_id":29,"q":"Error. Key not valid!"}
{"item_id":31,"q":"Error. Key not valid!"}
{"item_id":33,"q":"Error. Key not valid!"}
{"item_id":35,"q":"Error. Key not valid!"}
{"item_id":37,"q":"Error. Key not valid!"}
{"item_id":39,"q":"Error. Key not valid!"}
{"item_id":41,"q":"Error. Key not valid!"}
{"item_id":43,"q":"Error. Key not valid!"}
{"item_id":45,"q":"Error. Key not valid!"}
{"item_id":47,"q":"Error. Key not valid!"}
{"item_id":49,"q":"Error. Key not valid!"}
{"item_id":51,"q":"Error. Key not valid!"}
{"item_id":53,"q":"Error. Key not valid!"}
{"item_id":55,"q":"Error. Key not valid!"}
{"item_id":57,"q":"Winter Wonderland, Hyde Park, London."}
{"item_id":59,"q":"Error. Key not valid!"}
{"item_id":61,"q":"Error. Key not valid!"}
{"item_id":63,"q":"Error. Key not valid!"}
{"item_id":65,"q":"Error. Key not valid!"}
{"item_id":67,"q":"Error. Key not valid!"}
{"item_id":69,"q":"Error. Key not valid!"}
{"item_id":71,"q":"Error. Key not valid!"}
{"item_id":73,"q":"Error. Key not valid!"}
```

Question 6

Answer: 57

Given with Santa's place



```
{"item_id":57,"q":"Winter Wonderland, Hyde Park, London."}
```

Thought Process/Methodology:

First we use nmap to scan the available web server port. Then, we open the website and find the template used, which is BULMA. Next, we look at the page source to find the api directory. We tried opening the directory without the parameter(api key) and it only showed “detail not found”. We made a python code to easily find the correct api key and Santa’s location, and ran it using the terminal.

Day 17: Reverse Engineering - ReverseELFneering

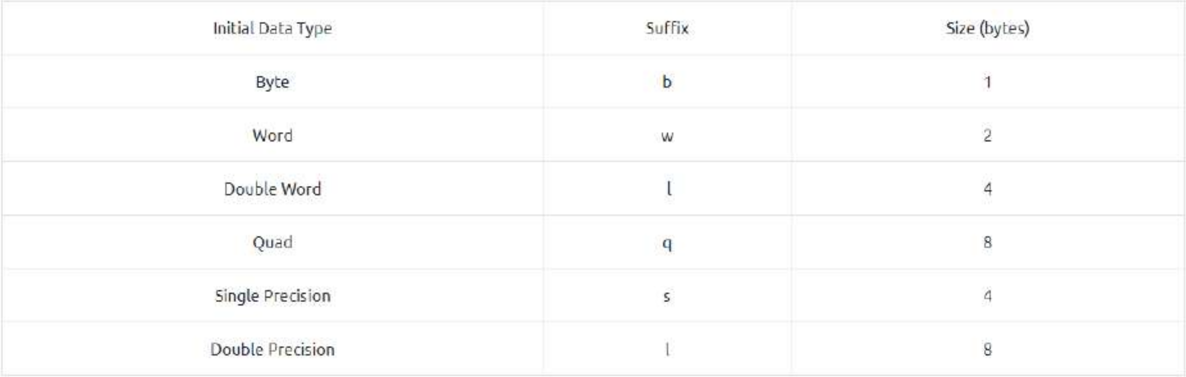
Tools Used: Kali Linux

Solution/Walkthrough:

Question 1

Answer: Byte > 1
Word > 2
Double word > 4
Quad > 8
Single-precision > 4
Double-precision > 8

We enter the data size(bytes) from the data given in Tryhackme

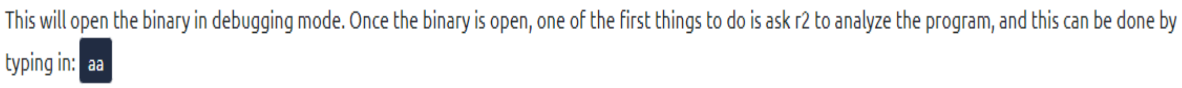


Initial Data Type	Suffix	Size (bytes)
Byte	b	1
Word	w	2
Double Word	l	4
Quad	q	8
Single Precision	s	4
Double Precision	l	8

Question 2

Answer: aa

we use command 'aa' to ask radare 2 to analyse the program

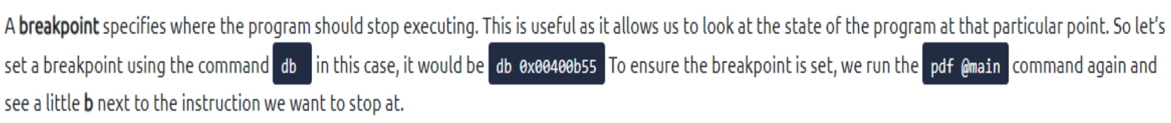


This will open the binary in debugging mode. Once the binary is open, one of the first things to do is ask r2 to analyze the program, and this can be done by typing in: `aa`

Question 3

Answer: db

We use command 'db' to set a breakpoint as it allows to look at the state of the program at particular point

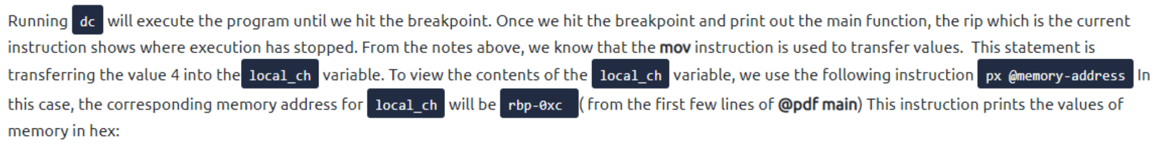


A **breakpoint** specifies where the program should stop executing. This is useful as it allows us to look at the state of the program at that particular point. So let's set a breakpoint using the command `db` in this case, it would be `db 0x00400b55` To ensure the breakpoint is set, we run the `pdf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

Question 4

Answer: dc

To execute the breakpoint, we use command ‘dc’ to execute the program until we hit the breakpoint

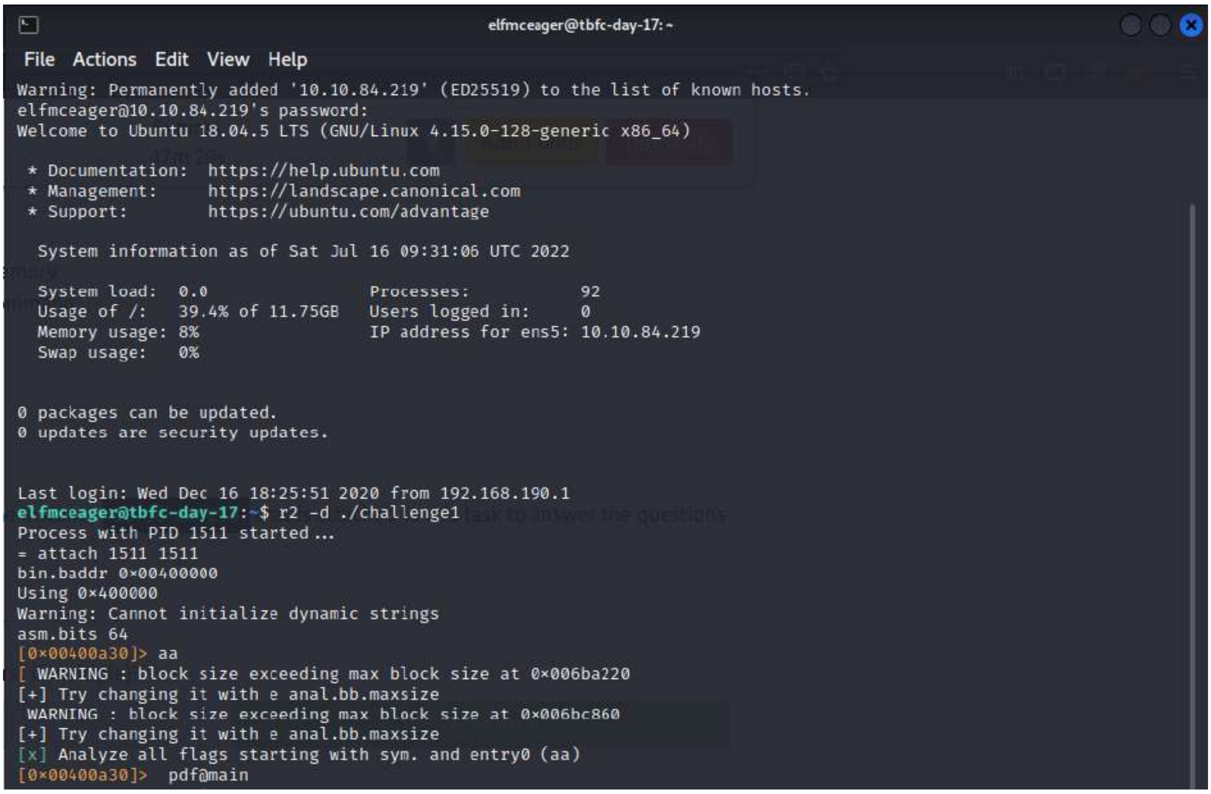


Running `dc` will execute the program until we hit the breakpoint. Once we hit the breakpoint and print out the main function, the rip which is the current instruction shows where execution has stopped. From the notes above, we know that the **mov** instruction is used to transfer values. This statement is transferring the value 4 into the `local_ch` variable. To view the contents of the `local_ch` variable, we use the following instruction `px @memory-address` In this case, the corresponding memory address for `local_ch` will be `rbp-0xc` (from the first few lines of **@pdf main**) This instruction prints the values of memory in hex:

Question 5

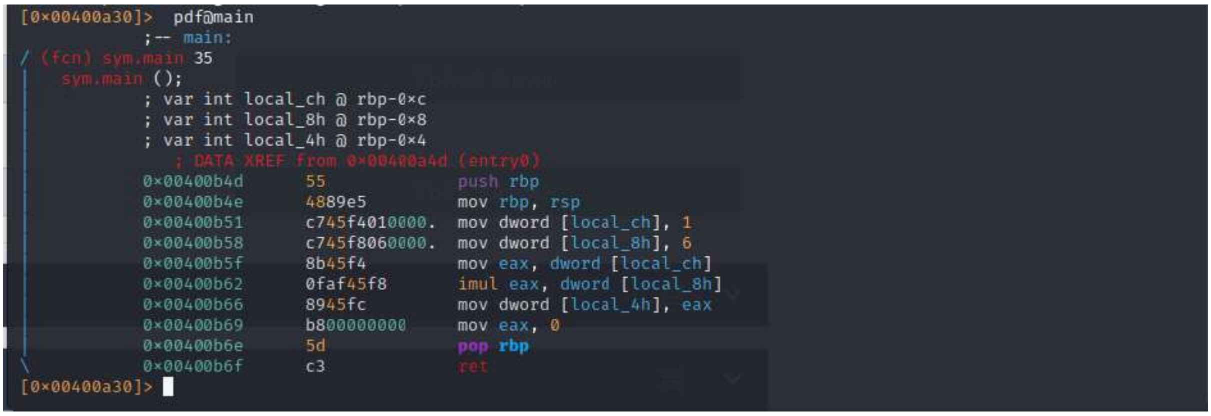
Answer: 1

First launch the radare 2 in the debug mode and launch the challenge 1 file , and start analysis with command ‘aa’ . After that command pdf@main to get to the main.



```
Warning: Permanently added '10.10.84.219' (ED25519) to the list of known hosts.
elfmceager@10.10.84.219's password:
Welcome to Ubuntu 18.04.5 LTS (GNU/Linux 4.15.0-128-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Sat Jul 16 09:31:06 UTC 2022

  System load:  0.0               Processes:           92
  Usage of /:   39.4% of 11.75GB  Users logged in:     0
  Memory usage: 8%                IP address for ens5: 10.10.84.219
  Swap usage:   0%


0 packages can be updated.
0 updates are security updates.


Last login: Wed Dec 16 18:25:51 2020 from 192.168.190.1
elfmceager@tbfc-day-17:~$ r2 -d ./challenge1
Process with PID 1511 started...
= attach 1511 1511
bin.baddr 0×00400000
Using 0×400000
Warning: Cannot initialize dynamic strings
asm.bits 64
[0×00400a30]> aa
[ WARNING : block size exceeding max block size at 0×006ba220
[+] Try changing it with e anal.bb.maxsize
 WARNING : block size exceeding max block size at 0×006bc860
[+] Try changing it with e anal.bb.maxsize
[x] Analyze all flags starting with sym. and entry0 (aa)
[0×00400a30]>  pdf@main
```

You can get local_ch with the data given

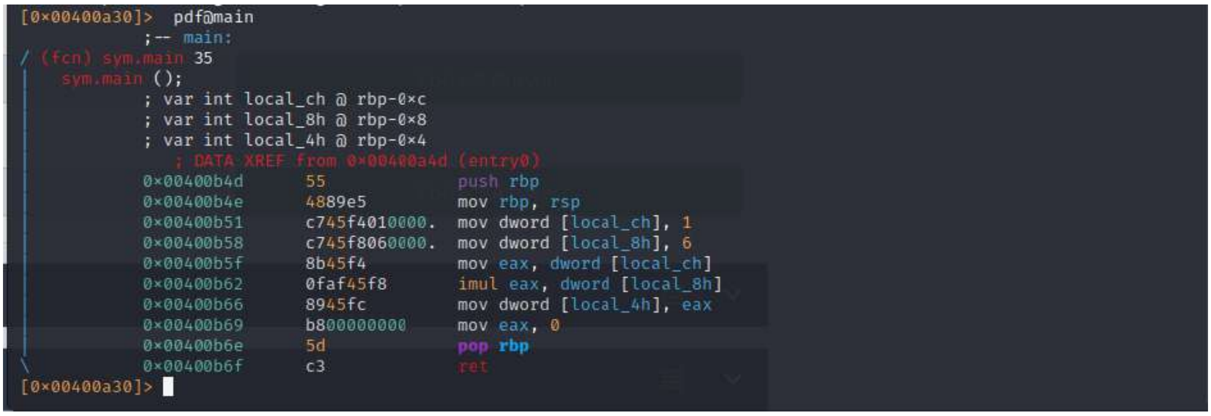


```
[0x00400a30]>  pdf@main
            ;-- main:
/ (fcn) sym.main 35
|   sym.main ();
|           ; var int local_ch @ rbp-0xc
|           ; var int local_8h @ rbp-0x8
|           ; var int local_4h @ rbp-0x4
|              ; DATA XREF from 0x00400a4d (entry0)
|           0x00400b4d      55             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
[0x00400a30]> 
```

Question 6

Answer: 6

Dword [local_8h] = 1x6

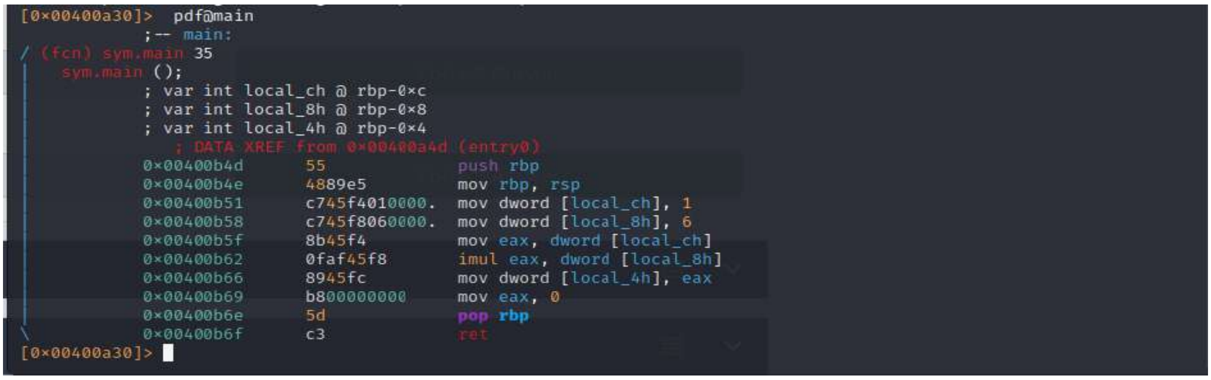


```
[0x00400a30]>  pdf@main
            ;-- main:
/ (fcn) sym.main 35
|   sym.main ();
|           ; var int local_ch @ rbp-0xc
|           ; var int local_8h @ rbp-0x8
|           ; var int local_4h @ rbp-0x4
|              ; DATA XREF from 0x00400a4d (entry0)
|           0x00400b4d      55             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
[0x00400a30]> 
```

Question 7

Answer: 6

Taking the value from eax and then copying it into that other variable



```
[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
[0x00400a30]> █
```

Thought Process/Methodology:

First launch the radare 2 in the debug mode and launch the challenge 1 file , and start analysis with command 'aa' . After that command pdf@main to get to the main. Then You can get local_ch with the data given . To get the value of eax , we multiply 1 with 6 and get answer 6 and for the local_8h , we take the value from eax and then copy it into that other variable.

Day 18: Reverse Engineering - The Bits of Christmas

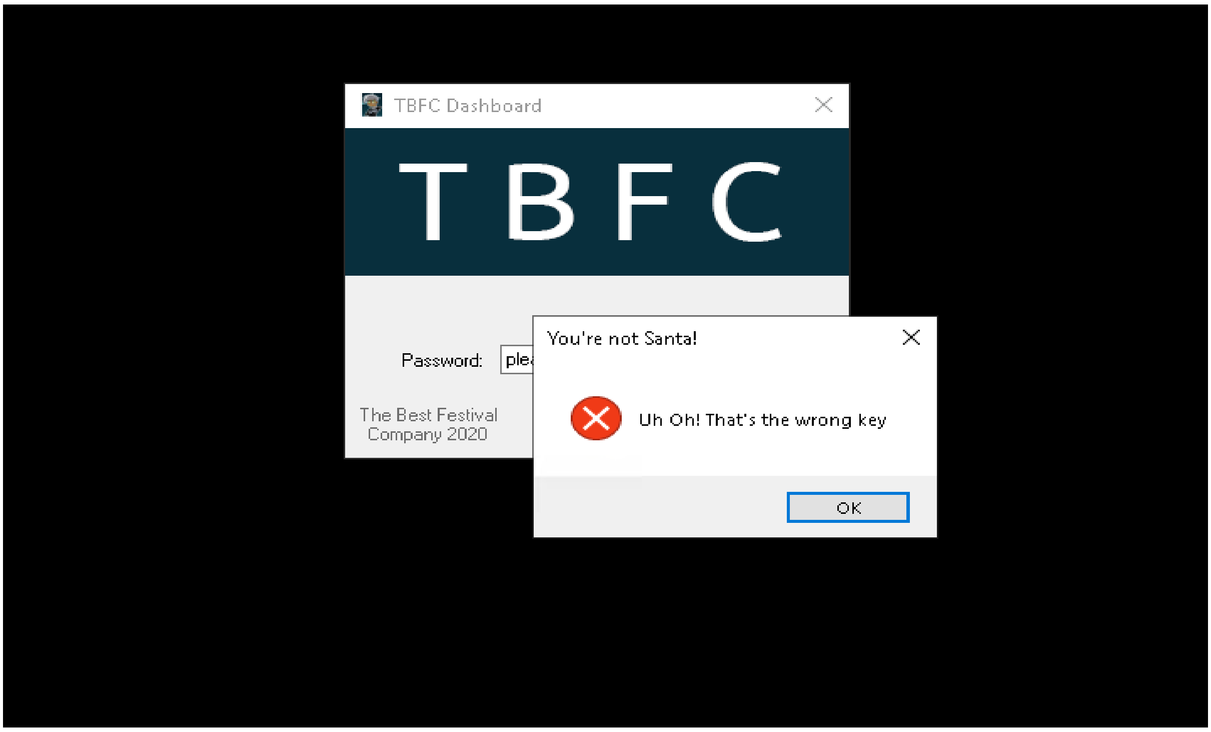
Tools Used: Kali Linux

Solution/Walkthrough:

Question 1

Answer: Uh Oh! That's the wrong key", "You're not Santa!

When you enter the wrong password for TBFC_APP , it will display this



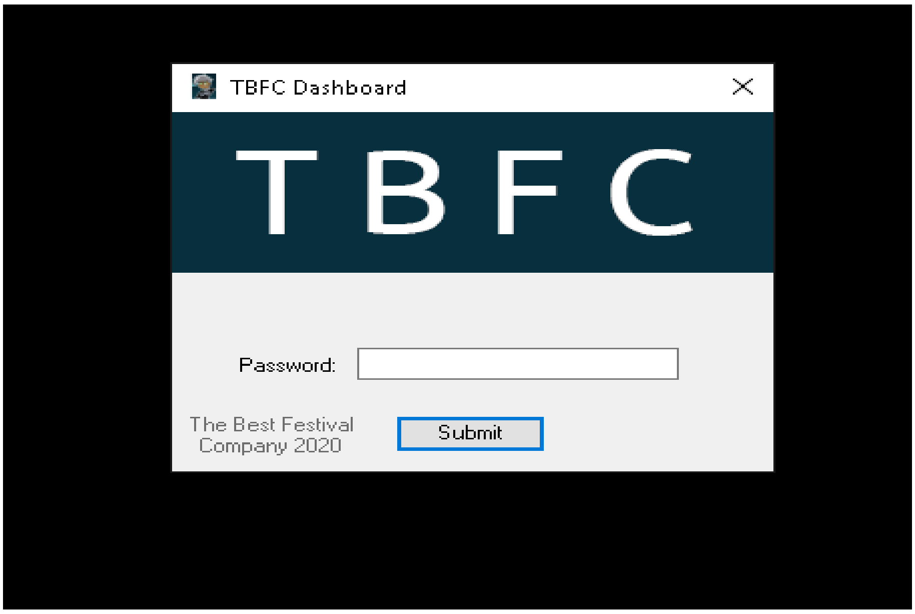
You're not Santa!

Uh Oh! That's the wrong key

Question 2

Answer: The Best Festival Company

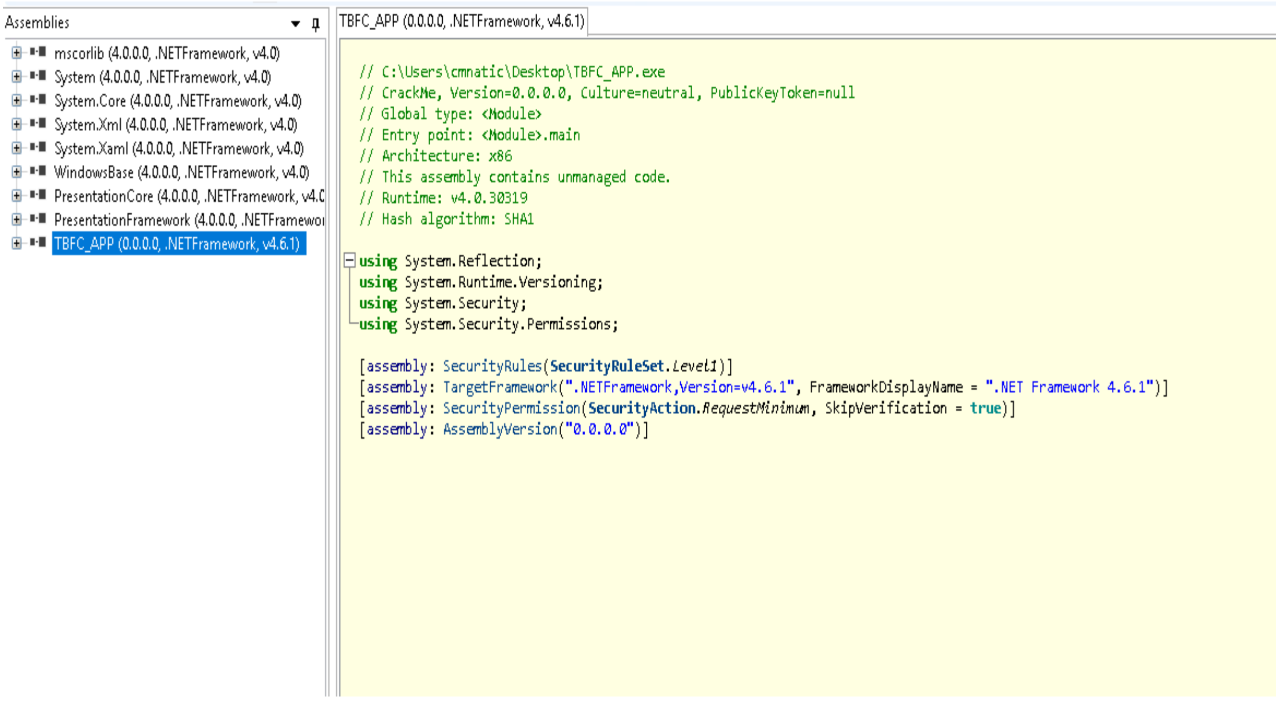
Open the remmina from AttachBox ,then click the TBCF_APP and enter the IP address given. After that it will show the App desktop and click the TBFC_APP to get what TBFC means from the app dashboard.



Question 3

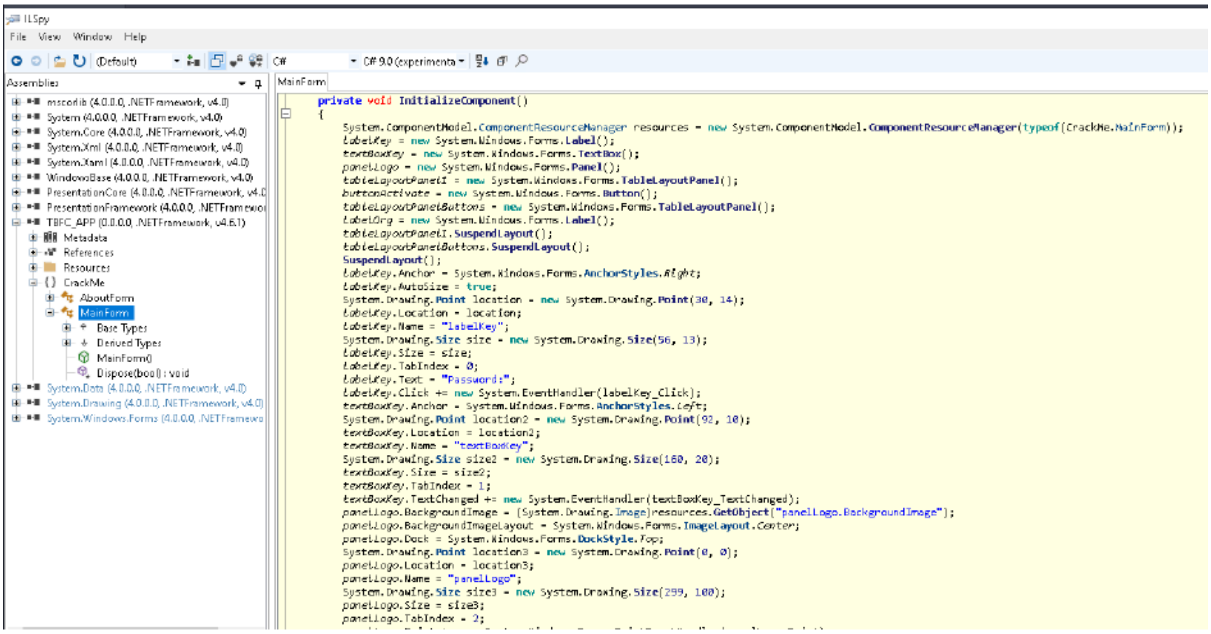
Answer: CrackMe

Open ILSpy app and decompile the TBFC_APP


Assemblies
mscorlib (4.0.0.0, .NETFramework, v4.0)
System (4.0.0.0, .NETFramework, v4.0)
System.Core (4.0.0.0, .NETFramework, v4.0)
System.Xml (4.0.0.0, .NETFramework, v4.0)
System.Xaml (4.0.0.0, .NETFramework, v4.0)
WindowsBase (4.0.0.0, .NETFramework, v4.0)
PresentationCore (4.0.0.0, .NETFramework, v4.0
PresentationFramework (4.0.0.0, .NETFramewo
TBFC_APP (0.0.0.0, .NETFramework, v4.6.1)
TBFC_APP (0.0.0.0, .NETFramework, v4.6.1)

```
// C:\Users\cmnatic\Desktop\TBFC_APP.exe
// CrackMe, Version=0.0.0.0, Culture=neutral, PublicKeyToken=null
// Global type: <Module>
// Entry point: <Module>.main
// Architecture: x86
// This assembly contains unmanaged code.
// Runtime: v4.0.30319
// Hash algorithm: SHA1

using System.Reflection;
using System.Runtime.Versioning;
using System.Security;
using System.Security.Permissions;

[assembly: SecurityRules(SecurityRuleSet.Level1)]
[assembly: TargetFramework(".NETFramework,Version=v4.6.1", FrameworkDisplayName = ".NET Framework 4.6.1")]
[assembly: SecurityPermission(SecurityAction.RequestMinimum, SkipVerification = true)]
[assembly: AssemblyVersion("0.0.0.0")]
```

Then click the plus (+) sign beside TBFC_APP and it will show the module

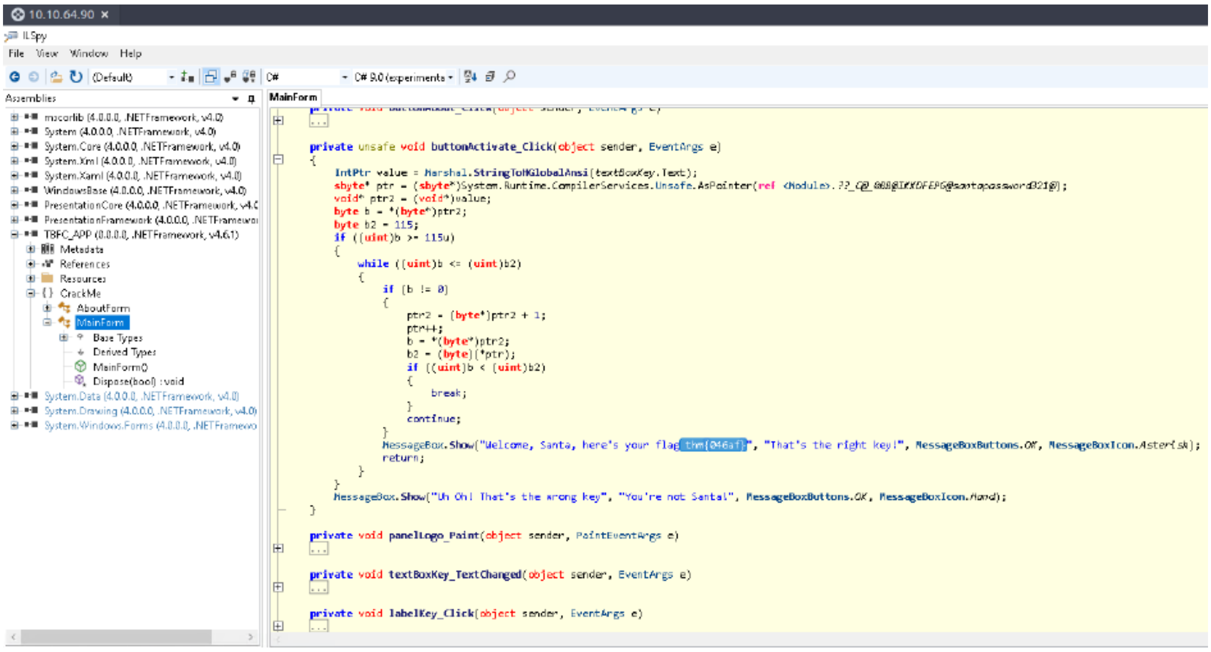

ILSpy
File View Window Help
(Default)
C#
C# 9.0 (experimenta
Assemblies
MainForm
mscorlib (4.0.0.0, .NETFramework, v4.0)
System (4.0.0.0, .NETFramework, v4.0)
System.Core (4.0.0.0, .NETFramework, v4.0)
System.Xml (4.0.0.0, .NETFramework, v4.0)
System.Xaml (4.0.0.0, .NETFramework, v4.0)
WindowsBase (4.0.0.0, .NETFramework, v4.0)
PresentationCore (4.0.0.0, .NETFramework, v4.0
PresentationFramework (4.0.0.0, .NETFramewo
TBFC_APP (0.0.0.0, .NETFramework, v4.6.1)
Metadata
References
Resources
CrackMe
AboutForm
MainForm
Base Types
Derived Types
MainForm()
Dispose(bool) : void
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();
    tableLayoutPanel1 = new System.Windows.Forms.TableLayoutPanel();
    buttonActivate = new System.Windows.Forms.Button();
    tableLayoutPanelButtons = new System.Windows.Forms.TableLayoutPanel();
    labelOrg = new System.Windows.Forms.Label();
    tableLayoutPanel1.SuspendLayout();
    tableLayoutPanelButtons.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;
```

Question 4

Answer: MainForm

Click the (+) sign beside CrackMe module, then it will show you the 2 form and pick the Main form to get into other main file

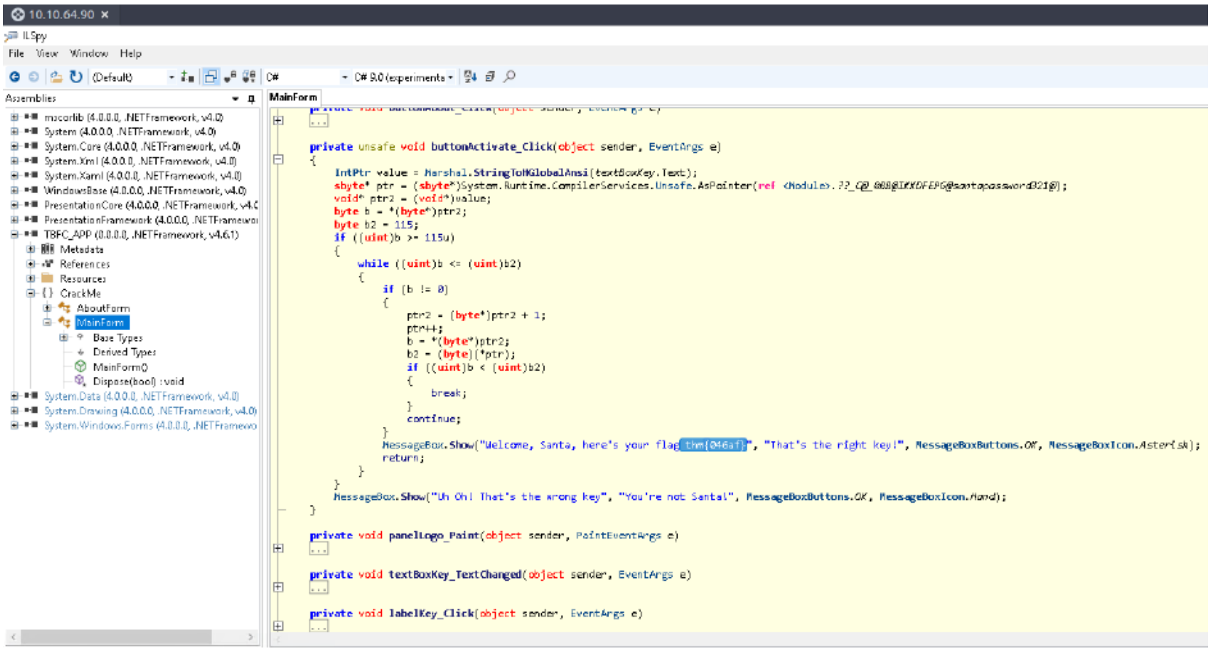

10.10.64.90
ILSpy
File View Window Help
(Default)
C#
C# 9.0 (experiments
Assemblies
mscorlib (4.0.0.0, .NETFramework, v4.0)
System (4.0.0.0, .NETFramework, v4.0)
System.Core (4.0.0.0, .NETFramework, v4.0)
System.Xml (4.0.0.0, .NETFramework, v4.0)
System.Xaml (4.0.0.0, .NETFramework, v4.0)
WindowsBase (4.0.0.0, .NETFramework, v4.0)
PresentationCore (4.0.0.0, .NETFramework, v4.0
PresentationFramework (4.0.0.0, .NETFramewo
TBFC_APP (0.0.0.0, .NETFramework, v4.6.1)
Metadata
References
Resources
CrackMe
AboutForm
MainForm
Base Types
Derived Types
MainForm()
Dispose(bool) : void
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
MainForm

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

private unsafe void buttonActivate_Click(object sender, EventArgs e)
{
    IntPtr value = Marshal.StringToHGlobalAnsi(textBoxKey.Text);
    sbyte* ptr = (sbyte*)System.Runtime.CompilerServices.Unsafe.AsPointer(ref <Module>.??_C@_08BIKKDFEPG@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);
}

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

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

private void labelKey_Click(object sender, EventArgs e)
...
```

Question 5

Answer: buttonActivate_Click

Click the plus(+) sign beside MainForm and it will show you the list of files inside the form , then choose the file that is related to log in to get the login password.

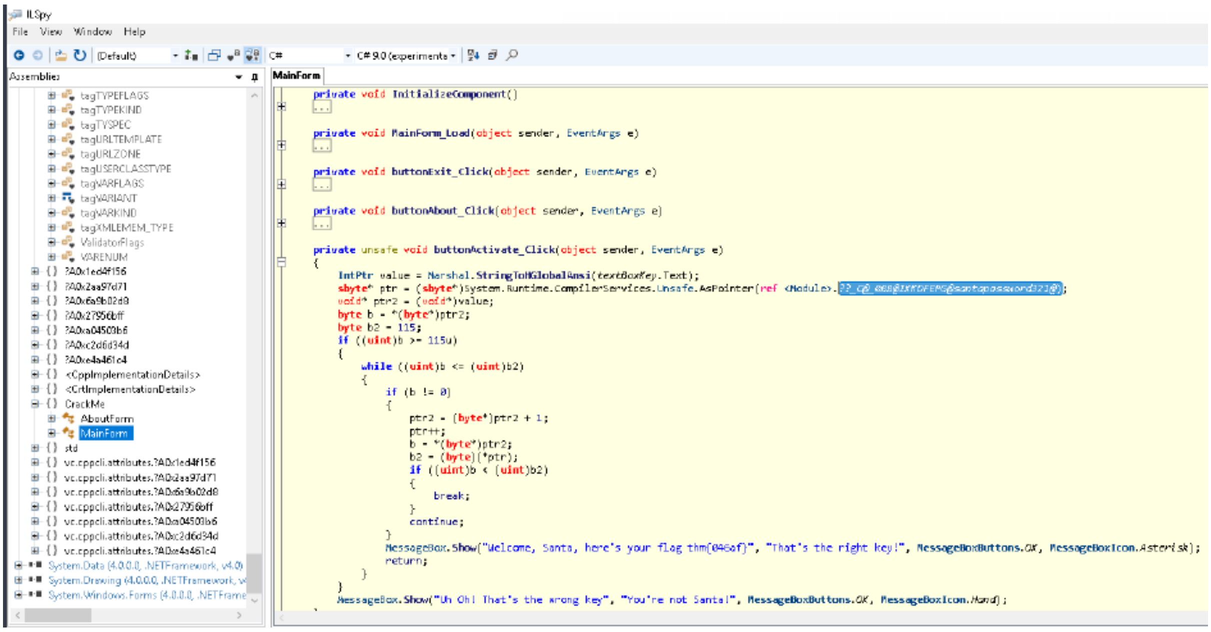


```
private void buttonAbout_Click(object sender, EventArgs e)
...

private unsafe void buttonActivate_Click(object sender, EventArgs e)
{
    IntPtr value = Marshal.StringToHGlobalAnsi(textBoxKey.Text);
    sbyte* ptr = (sbyte*)System.Runtime.CompilerServices.Unsafe.AsPointer(ref <Module>.??_C@_08BIKKDFEPG@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);
}

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

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

private void labelKey_Click(object sender, EventArgs e)
...
```

Question 6

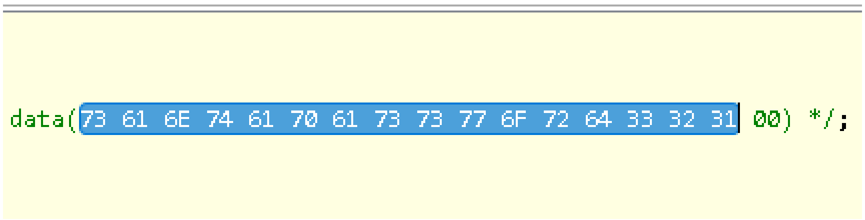
Answer: santapassword321

In the same file as the second question's answer, there is also Santa's password written on the code. Then click the santa's password


ILSpy
File View Window Help
(Default)
C#
C# 9.0 (experiments
Assemblies
MainForm
tagTYPEFLAGS
tagTYPEKIND
tagTYPESPEC
tagURLTEMPLATE
tagURLZONE
tagUSERCLASSTYPE
tagVARFLAGS
tagVARIANT
tagVARKIND
tagXMLEMEM_TYPE
ValidatorFlags
VARENUM
?A0x1ed4f156
?A0x2aa97d71
?A0x6a9b02d8
?A0x27956bff
?A0xa04503b6
?A0xc2d6d34d
?A0xe4a461c4
<CppImplementationDetails>
<CrtImplementationDetails>
CrackMe
AboutForm
MainForm
std
vc.cppcli.attributes.?A0x1ed4f156
vc.cppcli.attributes.?A0x2aa97d71
vc.cppcli.attributes.?A0x6a9b02d8
vc.cppcli.attributes.?A0x27956bff
vc.cppcli.attributes.?A0xa04503b6
vc.cppcli.attributes.?A0xc2d6d34d
vc.cppcli.attributes.?A0xe4a461c4
System.Data (4.0.0.0, .NETFramework, v4.0)
System.Drawing (4.0.0.0, .NETFramework, v
System.Windows.Forms (4.0.0.0, .NETFrame

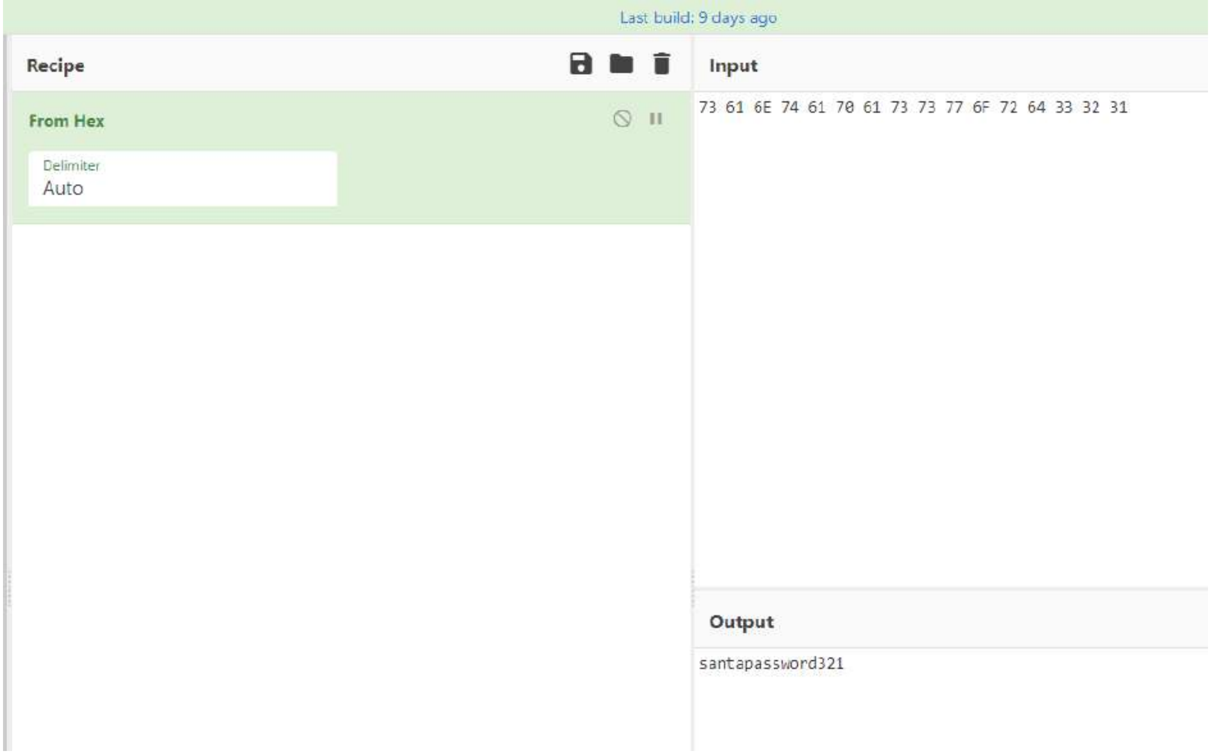
```
private void InitializeComponent()
[...]

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)
{
    IntPtr value = Marshal.StringToHGlobalAnsi(textBoxKey.Text);
    sbyte* ptr = (sbyte*)System.Runtime.CompilerServices.Unsafe.AsPointer(ref <Module>.??_C@_0BB@IKKDFEMG@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);
```

The file that is displayed to us after we click the password is formatted in hexadecimal. We can use an online tool such as Cyberchef to decode that.



```
data(73 61 6E 74 61 70 61 73 73 77 6F 72 64 33 32 31 00) */;
```

Copy the hexadecimal then decode that and And the result given by CyberChef is the same as the password on the previous file. Which that means it is the santa's password is "santapassword321"



Delimiter: Auto

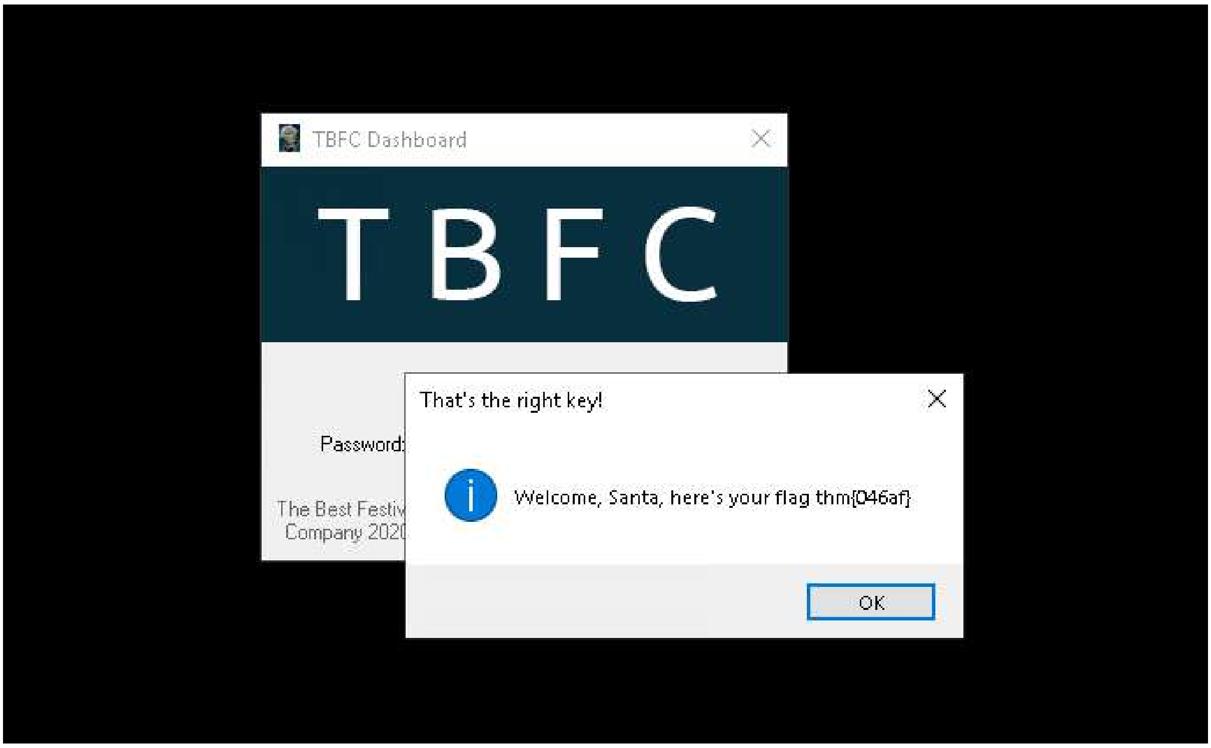
```
73 61 6E 74 61 70 61 73 73 77 6F 72 64 33 32 31
```

```
santapassword321
```

Question 7

Answer: thm{046af}

Login using the password that you just get and you will get the flag



That's the right key!

Welcome, Santa, here's your flag thm{046af}

Thought Process/Methodology

We use the remmina on TryHackMe AttachBox to connect the instance with the following credentials. After we connected to the virtual machine , we could see some of the apps displayed on the desktop. There is Recycle bin, dotPeek, ILSpy, and TBFC_APP which we are going to decompile. In the tutorial above, CMNatics uses ILSpy for decompiling the calculator, so in this case we are going to do the same but with TBFC_APP. To decompile the app, first we open up the ILSpy, then click ‘file’ and select ‘open’. Once the prompt popped up, select the TBFC_APP which is located at the Desktop. TBFC_APP is loaded into ILSpy. After going through all the components on the ILSpy, there are components named ‘CrackMe’ which is a very weird name. We expand the thing, and inside it there is the MainForm() where the main function is. On the TBFC dashboard picture, the login mechanism is using a button to trigger the action and is also declared in the main function. After that, we click a ‘buttonActivate and luckily the flag is hardcoded on the source code. In the same file as the second question’s answer, there is also Santa's password written on the code. The file that is displayed to us after we click the password is formatted in hexadecimal. We can use an online tool such as Cyberchef to decode that. And the result given by CyberChef is the same as the password on the previous file.

Day 19: Web Exploitation - The Naughty or Nice List

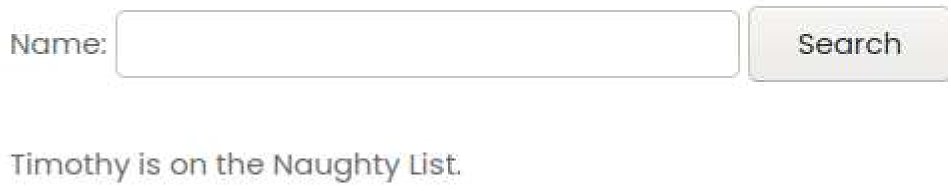
Tools Used: Kali Linux

Solution/Walkthrough:

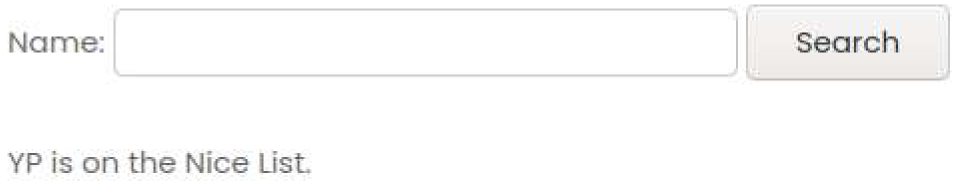
Question 1

Answer: Nice,Naughty, Nice, Naughty, Nice, Naughty

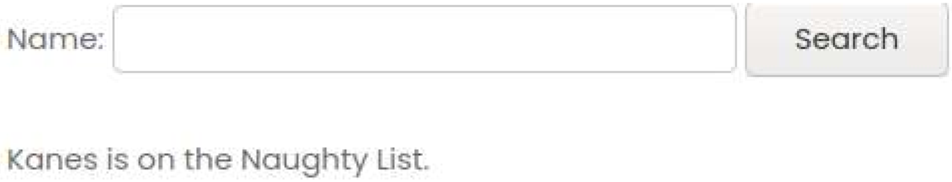
By inputting the names in the check box one by one, we get to know the results.



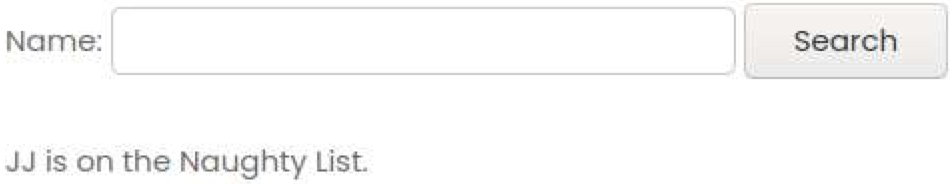
Timothy is on the Naughty List.



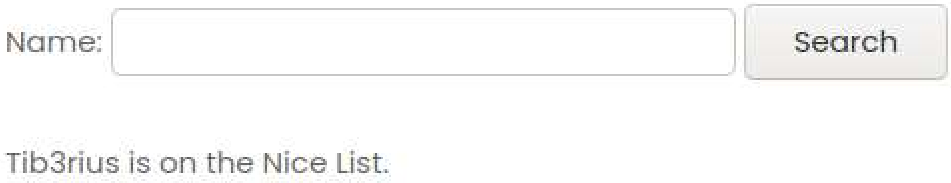
YP is on the Nice List.



Kanes is on the Naughty List.



JJ is on the Naughty List.

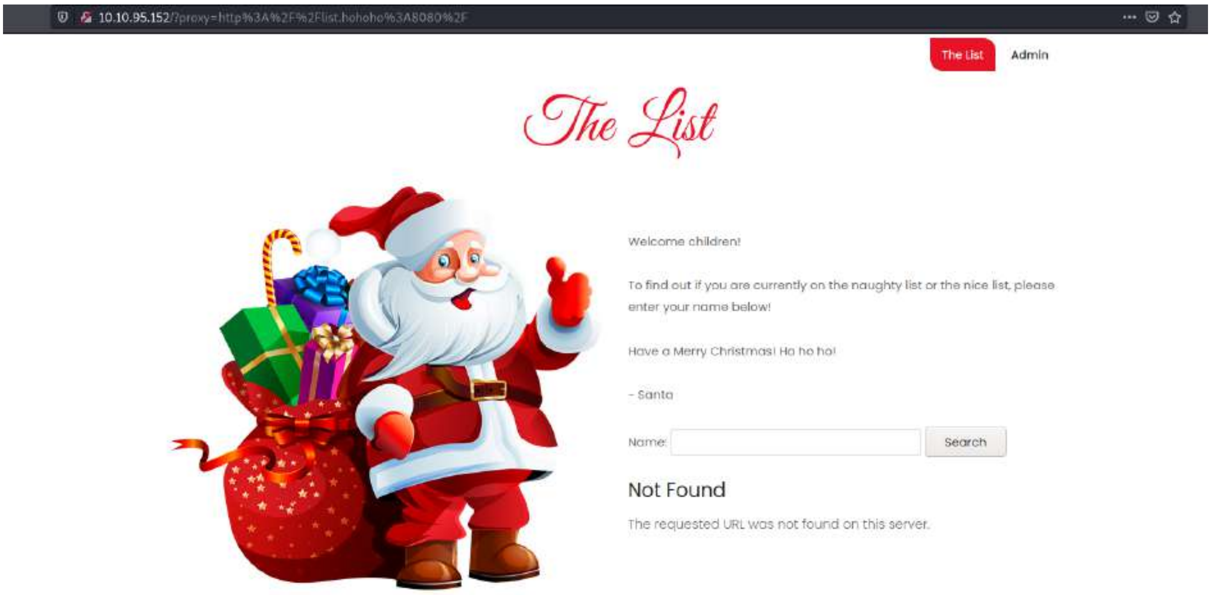


Tib3rius is on the Nice List.

Question 2

Answer: Not Found. The requested URL was not found on this server.

This is because the host does not exist so it cannot find the page.



10.10.95.152/?proxy=http%3A%2F%2Flist.hohoho%3A8080%2F

The List

Welcome children!

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:

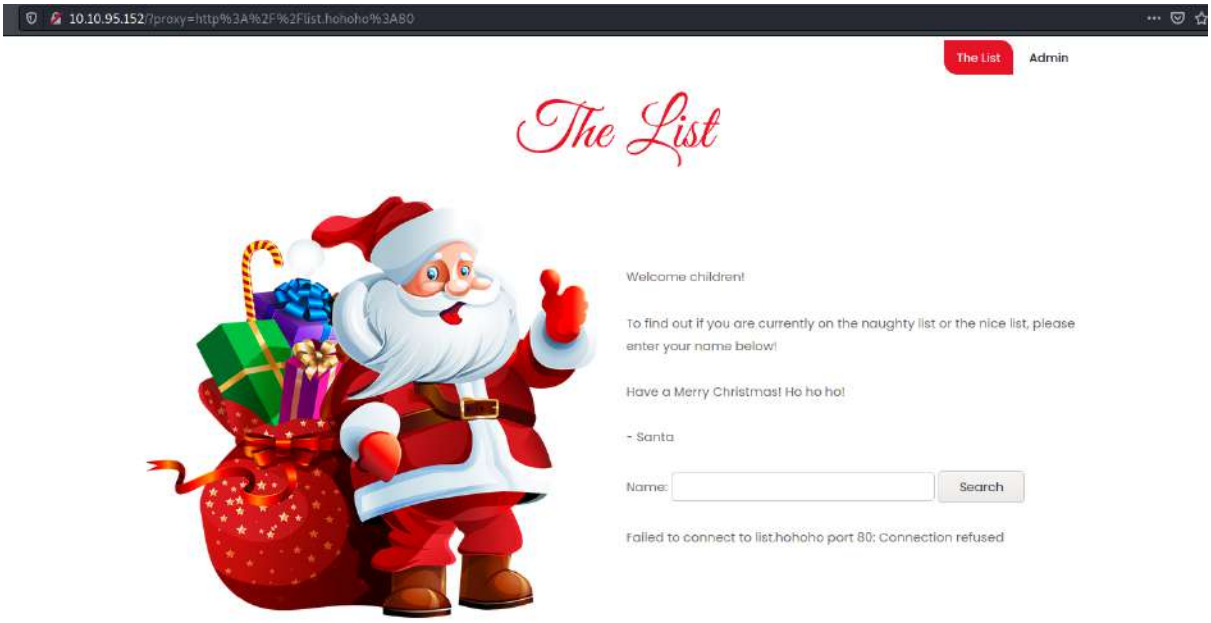
Not Found

The requested URL was not found on this server.

Question 3

Answer: Failed to connect to list.hohoho port 80: Connection refused

The host seems to not exist thus it cannot connect.



10.10.95.152/?proxy=http%3A%2F%2Flist.hohoho%3A80

The List

Welcome children!

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

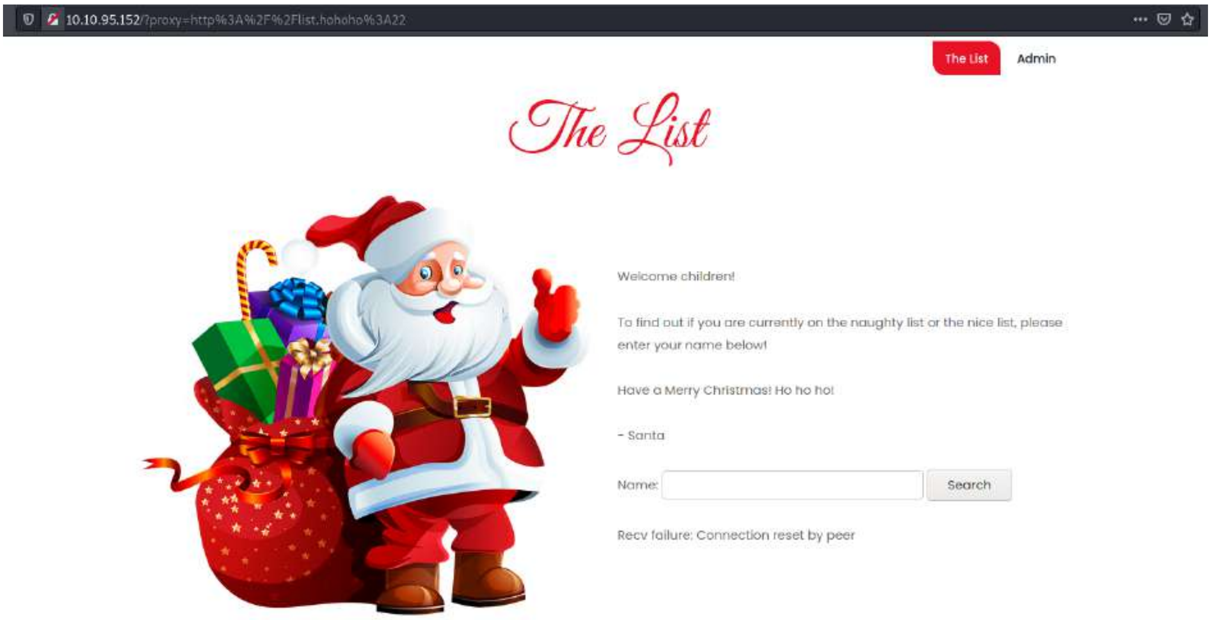
Name:

Failed to connect to list.hohoho port 80: Connection refused

Question 4

Answer: Recv failure: Connection reset by peer

The port exists but it is an SSH server so it did not understand.



10.10.95.152/?proxy=http%3A%2F%2Flist.hohoho%3A22

Welcome children!

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

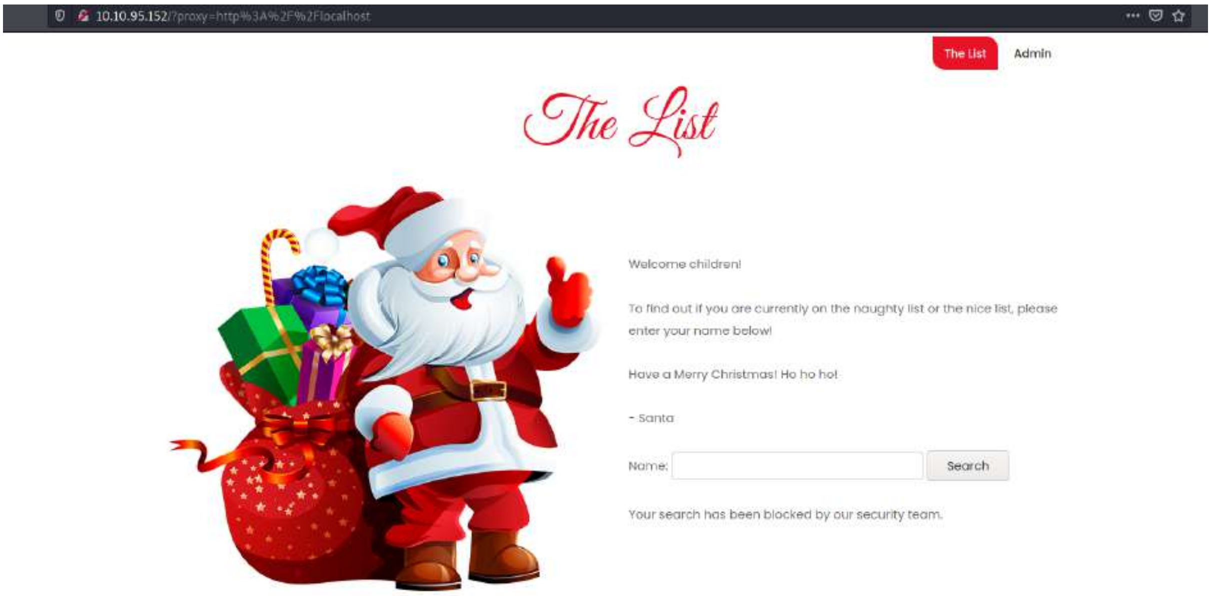
Name: Search

Recv failure: Connection reset by peer

Question 5

Answer: Your search has been blocked by our security team.

The hostname is different, as the only hostname allowed is list.hohoho.



10.10.95.152/?proxy=http%3A%2F%2Flocalhost

Welcome children!

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

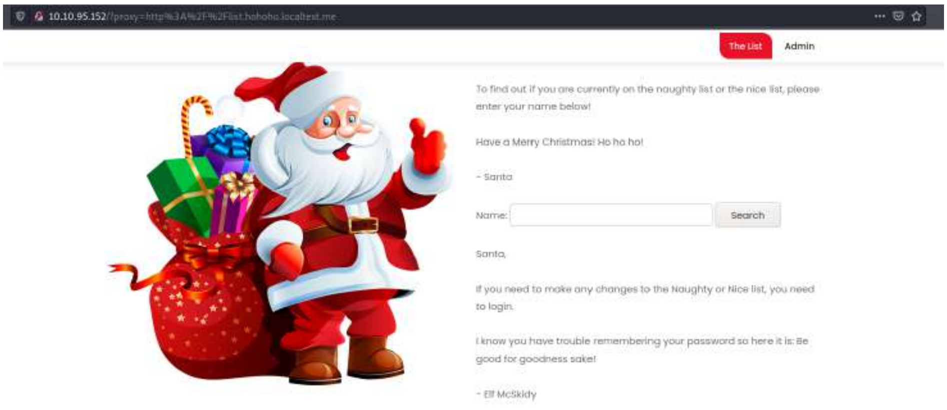
Name: Search

Your search has been blocked by our security team.

Question 6

Answer: Be good for goodness sake!

By going to the localhost.me page, there will be a note left by Elf. The note includes the password needed.



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

Santa,

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

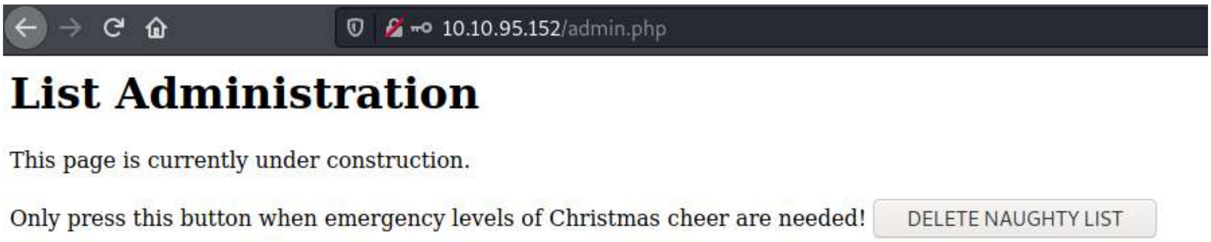
I know you have trouble remembering your password so here it is: Be good for goodness sake!

- Elf McSkidy

Question 7

Answer: THM{EVERYONE_GETS_PRESENTS}

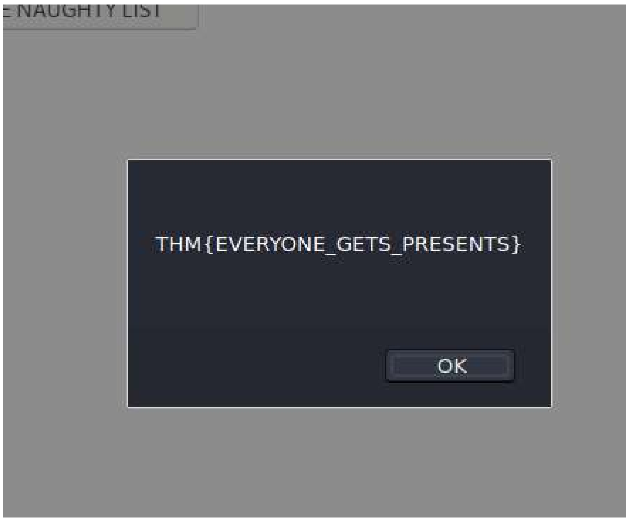
By using the password given, you can now delete all the naughty lists. After deleting, the flag will pop out.



List Administration

This page is currently under construction.

Only press this button when emergency levels of Christmas cheer are needed!



THM{EVERYONE_GETS_PRESENTS}

Thought Process/Methodology

First, we visit the page and insert the name to get the naughty or nice list. We then change the hostname to list.hohoho and it passes through. Now we know that his hostname is allowed by the admin of the page. We then proceed to go to the localhost page to find the santa password. By inputting the username and password in the admin section, we can now alter with the naughty list, deleting it will show the flag needed.

Day 20: Blue Teaming - PowershELlF to the rescue

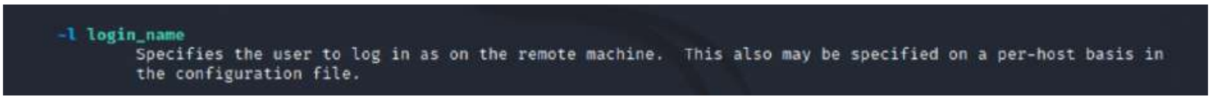
Tools Used: Kali Linux

Solution/Walkthrough:

Question 1

Answer: login name

Open the manpage of ssh and find the parameter -l.

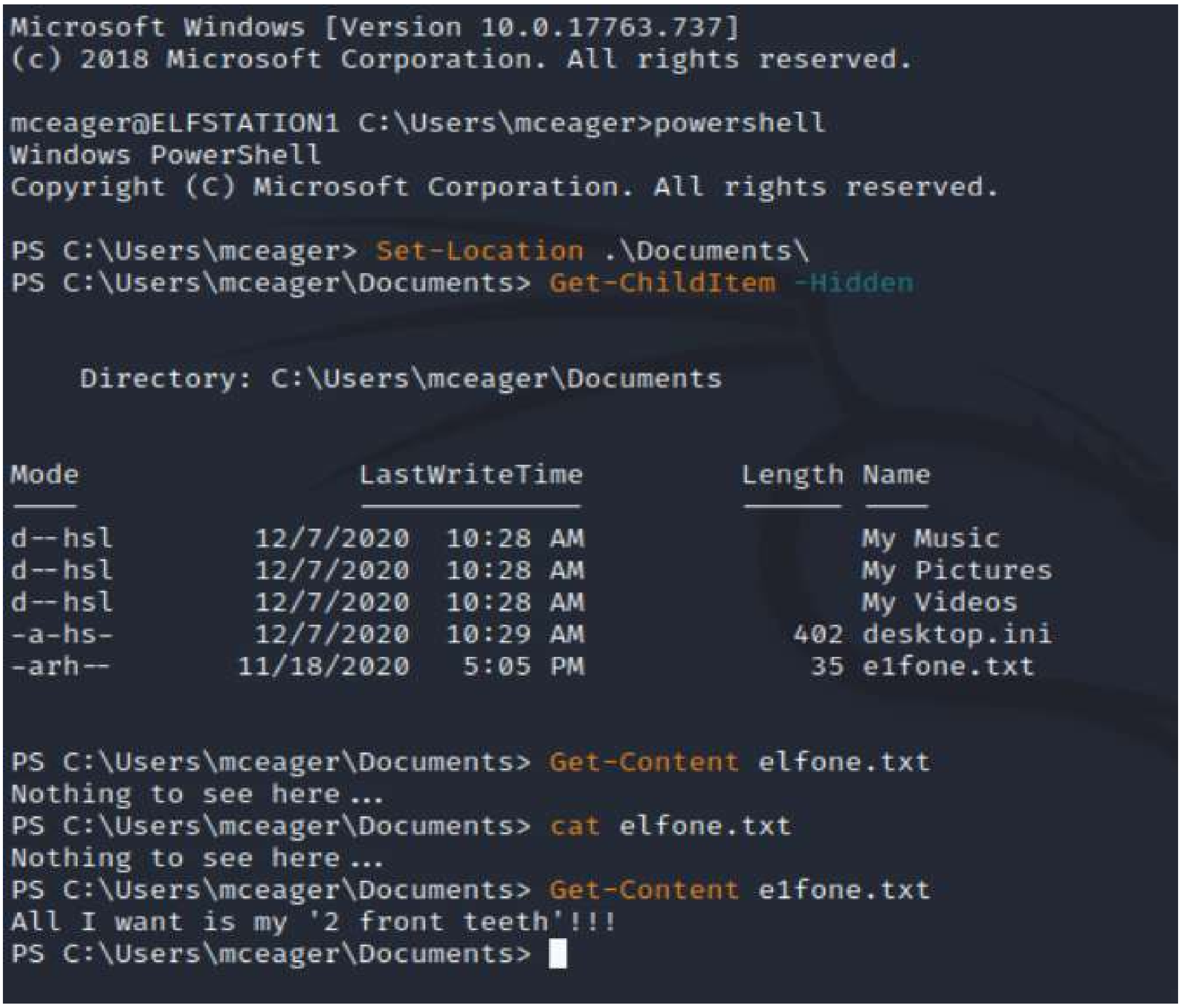


```
    -l login_name
            Specifies the user to log in as on the remote machine.  This also may be specified on a per-host basis in
            the configuration file.
```

Question 2

Answer: 2 front teeth

After you login successfully, find the hidden files by using Get-ChildItem -Hidden. Then, read the content by using Get-Content.

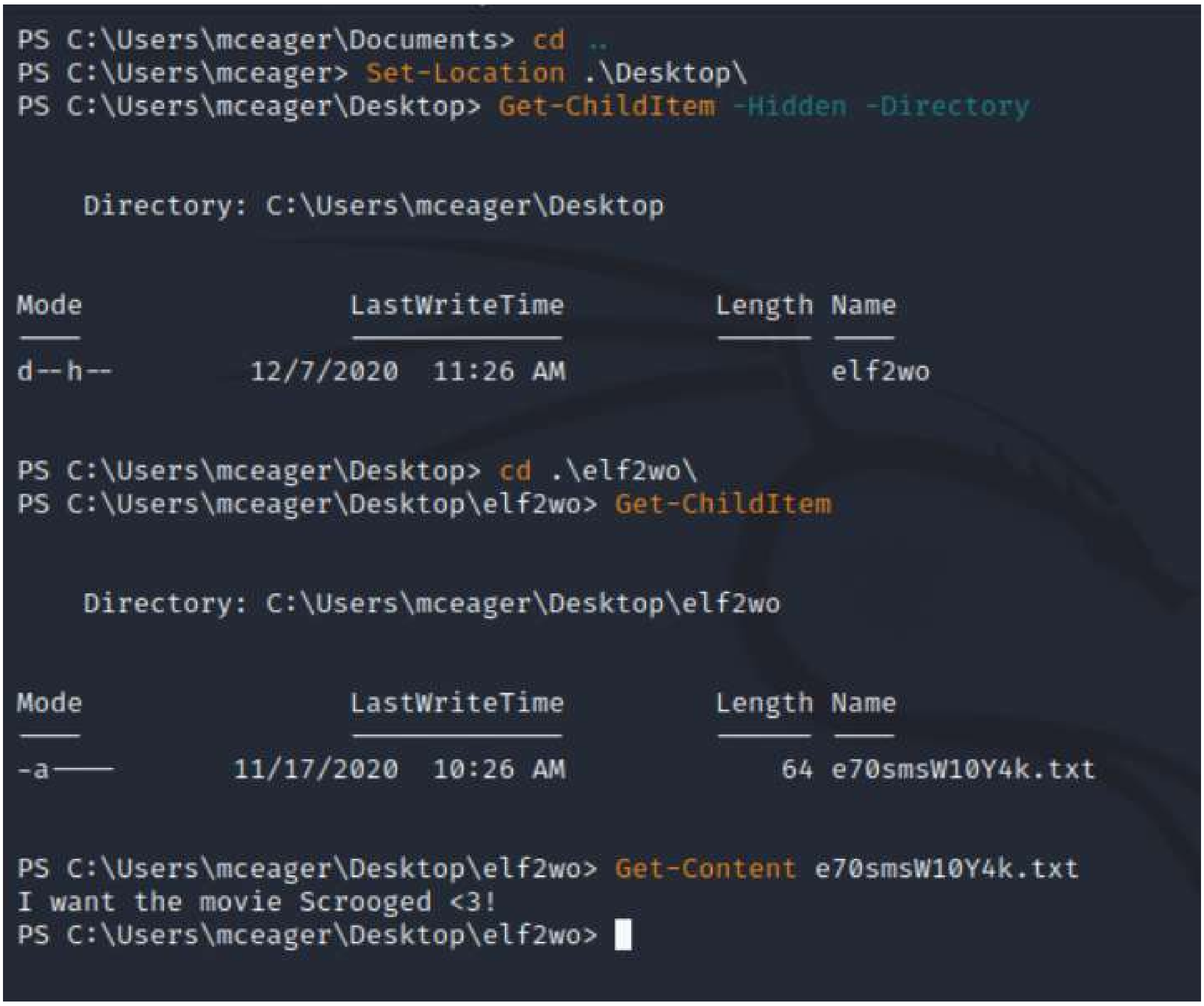


```
Microsoft Windows [Version 10.0.17763.737]
(c) 2018 Microsoft Corporation. All rights reserved.

mceager@ELFSTATION1 C:\Users\mceager>powershell
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

PS C:\Users\mceager> Set-Location .\Documents\
PS C:\Users\mceager\Documents> Get-ChildItem -Hidden


    Directory: C:\Users\mceager\Documents


Mode                LastWriteTime         Length Name
----                -------------         ------ ----
d--hsl        12/7/2020  10:28 AM                My Music
d--hsl        12/7/2020  10:28 AM                My Pictures
d--hsl        12/7/2020  10:28 AM                My Videos
-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> Get-Content elfone.txt
Nothing to see here...
PS C:\Users\mceager\Documents> cat elfone.txt
Nothing to see here...
PS C:\Users\mceager\Documents> Get-Content e1fone.txt
All I want is my '2 front teeth'!!!
PS C:\Users\mceager\Documents> 
```

Question 3

Answer: Scrooged

Change the directory to desktop and find the hidden folder with Get-ChildItem -Hidden -Directory . Change directory to the hidden folder and view file available with Get-ChildItem. Read the file using Get-Content.

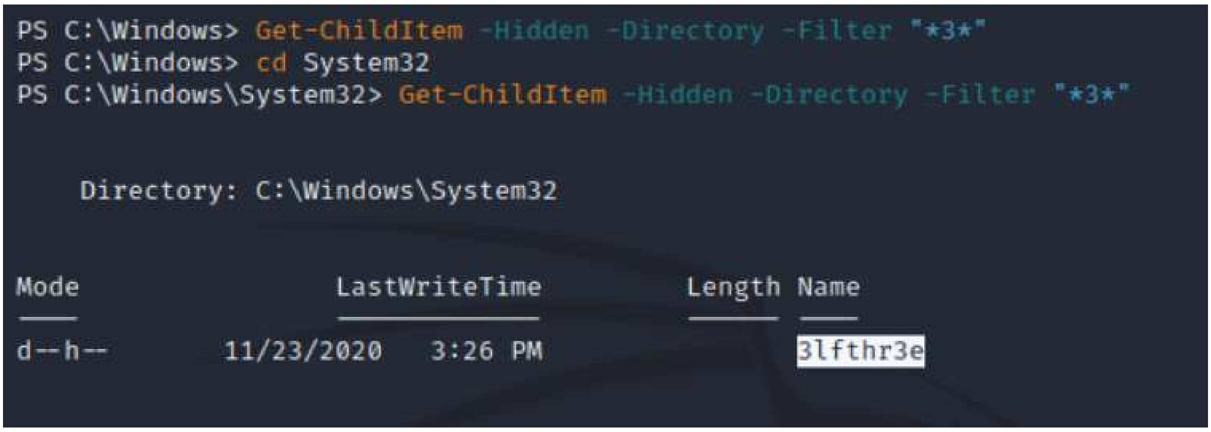


```
PS C:\Users\mceager\Documents> cd ..
PS C:\Users\mceager> Set-Location .\Desktop\
PS C:\Users\mceager\Desktop> Get-ChildItem -Hidden -Directory


    Directory: C:\Users\mceager\Desktop


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d--h--         12/7/2020  11:26 AM                elf2wo


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


    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 4

Answer: 3lfthr3e

Change directory to C:\Windows\System32. Find the hidden folder with Get-ChildItem -Hidden -Directory -Filter “*3*”.

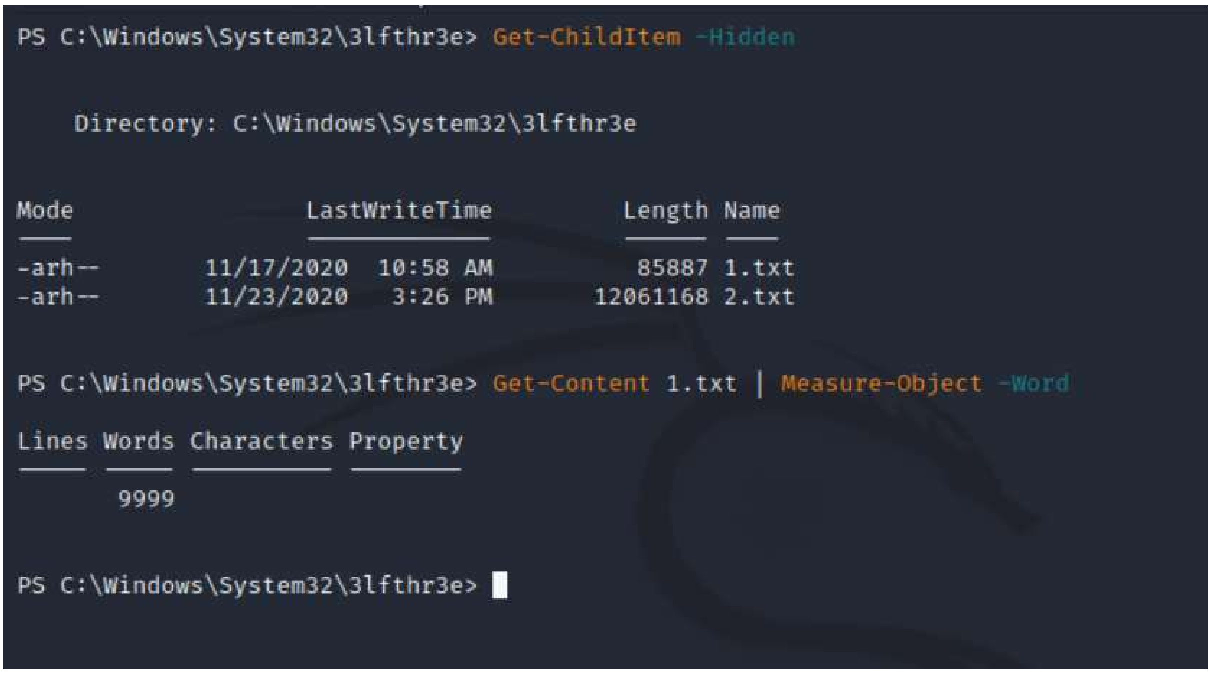


```
PS C:\Windows> Get-ChildItem -Hidden -Directory -Filter "*3*"
PS C:\Windows> cd System32
PS C:\Windows\System32> Get-ChildItem -Hidden -Directory -Filter "*3*"


    Directory: C:\Windows\System32


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

Question 5

Answer: 9999

Find the hidden files by using Get-ChildItem -Hidden then find the number of words on the first text file by using Get-Content 1.txt | Measure-Object -Word.

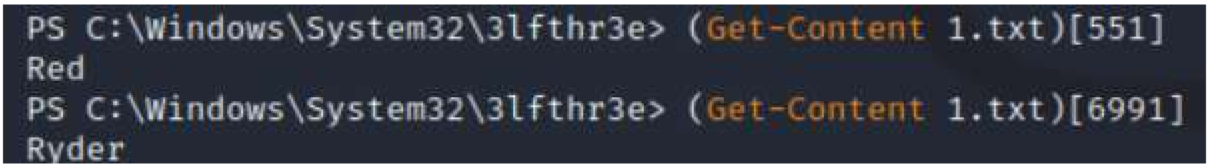


```
PS C:\Windows\System32\3lfthr3e> Get-ChildItem -Hidden


    Directory: C:\Windows\System32\3lfthr3e


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


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

Lines Words Characters Property
----- ----- ---------- --------
       9999


PS C:\Windows\System32\3lfthr3e> █
```

Question 6

Answer: Red Ryder

Use (Get-Content 1.txt)[index number] to find the words from the same file.

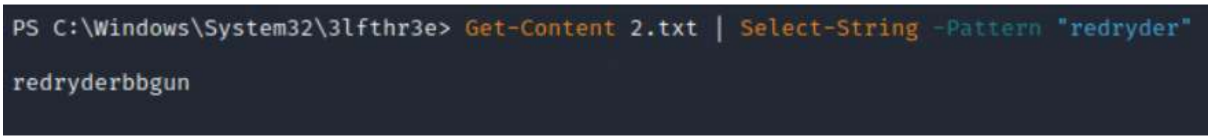


```
PS C:\Windows\System32\3lfthr3e> (Get-Content 1.txt)[551]
Red
PS C:\Windows\System32\3lfthr3e> (Get-Content 1.txt)[6991]
Ryder
```

Question 7

Answer: redryderbbgun

On the same directory, search the word in 2.txt by using Get-Content 2.txt | Select-String -Pattern "redryder".



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

redryderbbgun
```

Thought Process/Methodology

We logged in the remote machine and used multiple cmdlets for our next steps such as Get-Content to read content and Get-ChildItem to list all directories and files. We first find the hidden file e1fone.txt in Documents that says elf 1 wants 2 front teeth. We then found the file e70smsW10Y4k.txt at the hidden folder elf2wo which is located at the Desktop. We also found a hidden folder called 3lfthr3e at \Windows\System32. From there we examined the two hidden text files and found how many words there are and the two important words in the first file. Also, we found what elf 3 wants from the second file.