Thoth Challenge 1

> *"Threatens to send them the photograph. And she will do it. I know that she will do it. You do not know her, but she has a soul of steel. She has the face of the most beautiful of women, and the mind of the most resolute of men. Rather than I should marry another woman, there are no lengths to which she would not go--none."*

-"A Scandal in Bohemia," Decoded at: 11:12 am

Steps

1. Our group started with the initial clue on Moodle at 11:06 am. We used an online text replacer to change each "One" to a '1' and each "Zero" to a '0'. We were left with the following text:

```
10001101010100101000001000001110011110010111100101110110110010111100100100000110111111011100100000011000101100110111000
01011100110100011011101011100110001011010101101110101110011001001101000100000111000011011111110010111010001000000111000
01100000110000011100000010101110101111001111001011110010011101001000001110001111000111010101110110111101011010110001010
11100001100001111001111100110111010010000011010011101010110010111110101100111111101011001101101011111011111011001101011
11010111100110110000100010100001010100111111001100100000110001111011111110101111001011100111100101010110001000001110100
11010001100101010000011000011100011111010011101011100001110110001000001110101111001111001011110010110111011000011101101
11001010100000110000111011101100100010000011100001100001111001111100111110111110111111100101100100010000011011011110101
11100111110100010000011000101100101010000011001001100101110001111100101111001111000011101001100101110010001000001110100
11011110100000110110011011111100111110100111011100100000111000011100101101111111000011001011110010110110011110010101110
01000001010100110100011001010100000111010111100111100101111001011011101100001110110111001010100000111011111000011110011
01000001100101110111011000111110010111100111100001110100110010111001000100000110111111011101100011110010101000001100001
11011101100100010000011101001101000110010101000001110000110000111100111110011111011111011111110010110010001000001110100
11101111101001110001111001010100000111010111100111101001110111011001110100000111011011010011100111110010111011101100101
11100101100101010000011000111101001111000011010001100101111001001000001110111110100111101001101000010000011101001101000
11001010100000110011011011111101100110110011011111110111110100111011101100111010000011010111100101111100101110100100000
0100010110001111110011100010110010111100101110011111010011011111110010110110101000100001010
```

2. Our next step was to decode the binary. We pasted the string above into our Program 1 (binary decoder), which returned the following text:

```
7-bit: FTP server on 138.47.157.24 port 8008
user: qqjvzk
pass: ijezgzfkwlkkfa

Of course, the actual username and password must be decrypted to login properly. The
username was encrypted once and the password twice using vigenere cipher with the
following key: "cyberstorm"

7-bit with control chars: FTP server on 138.47.157.24 port 8008[LF]user: qqjvzk[LF]pass:
ijezgzfkwlkkfa[LF][LF]Of course, the actual username and
password must be decrypted to login properly. The username was encrypted once and the
password twice using vigenere cipher with the following key: "cyberstorm"[LF]
```

3. Our next step was to apply the information in the above text. We ran the username once and the password twice through our Program 2 (Vigenère) with the key "cyberstorm"

which returned the following data:

```
Username: osiris
Password: encryptiongods
```

4. We then pasted the IP, port, and login information into a Python script which we had prepared beforehand to quickly find hidden messages. This program returned nothing. After connecting to the server with a graphical FTP program, we realized why: the folders on the server were hidden. With a little manual browsing, we realized that /.secretstorage/.folder2/.howaboutonemore had a bunch of suspicious files. Feeding that location into the FOLDER constant of the linked script, we got the following output which revealed the secret answer. We cannot reproduce the exact output this program gave us because, as of the time of writing this document, we cannot connect to the FTP server. However, its output was very similar to that of our Program 3 (Covert FTP).

   Our prepared script was designed to find messages as quickly as possible. However, we did not account for hidden files. If we were to attempt this challenge again, modifying line 166 to read `ftp.dir(“-a”, files.append)` and to exclude . and .. from recursive searching would possibly result in a time improvement of 2 minutes.

Contributions

# Brendan Guillory
Contributed to the code which eventually solved the challenge. Helped decode the initial "ZeroOne" to real binary and found the folder which housed the encoded message.

# Cameron Robertson
Reviewed the previous programs in preparation for this challenge. Talked with the team to understand the programs before class. Attempted to convert the txt. file to binary and ran into issues.

# Christian Evans
Got the .txt file saved, placed the text into the binary decoder and translated to the message found in Step 1 by myself; however, Brendan had already long since completed this step, currently having moved on to connecting to the FTP server. Swapped over to pair programming if he needed to bounce ideas.
Also provided a table to team Horus, with the help of Dr. Timofeyev.

# Cody Woessner
Reviewed code. Translated the original file from text to binary then my virtual machine crashed and we had it decoded by the time it restarted.

# Drew Young
Helped with setting up for the challenge and organizing my group to prepare for the challenge. After running the programs, I ran into an issue with my programs running with python 2 and not 3. I had to go back and uninstall python 2 and run the programs again. At this point my group member Brendan was already on the last step and completed the challenge.

# Frankie Lavall
Looked over the txt file and decided to run the binary decoder program. Then i looked over at Brendan's computer and helped him decode because he had already solved what I was trying to solve.

# Tristen Barton
Added to the binary decoder program to change the long string of "OneZero…" to ones and zeros, but Brendan already did it with a website. Then I tried to decode the binary string but I only got the IP, port number, and username. It did not fully decode. I also helped implement the decoding feature of the 2nd program.