Lab Exercise - Decrypting A File

The files in my home directory are encrypted. The goal here is to decrypt the files in order to reveal the hidden message in the file.

STEP 1: Opening Encryption File For Directions

For this step, I first had to open up the file in my directory that will instruct me on how to decrypt my targeted file. In this case, the file is **`README.txt`**.

```
analyst@01a9901ea02c:~$ ls
Q1.encrypted  README.txt  caesar
analyst@01a9901ea02c:~$ cat README.txt
Hello,
All of your data has been encrypted. To recover your data, you
will need to solve a cipher. To get started look for a hidden
file in the caesar subdirectory.
```

Here the **`ls`** command is used to list all the files that were in my **`analyst@01a9901ea02c`** directory. Then I used the **`cat`** command to open the targeted file, **`README.txt.`** The content of the file reads:
“All of your data has been encrypted. To recover your data, you will need to solve a cipher. To get started look for a hidden file in the caesar subdirectory.”

STEP 2: Going to Caesar Sub-Directory

For this step, I use the **`cd caesar`** command to navigate to the caesar sub-directory as instructed by the **`README.txt`** file to locate the hidden file for the cipher.

```
analyst@01a9901ea02c:~$ cd caesar
analyst@01a9901ea02c:~/caesar$ ls -a
.  ..  .leftShift3
analyst@01a9901ea02c:~/caesar$ cat leftShift3
```

After that, I then used the **`ls -a`** command to open the files in the directory. The **`-a`** command is used to show all hidden files in the directory, shown as “.” and “..”.

The `cat` command is executed to open the `.leftShift3` file, which contains the instructions to decrypt the target file:

```
analyst@01a9901ea02c:~/caesar$ cat .leftShift3
Lq rughu wr uhfryhu brxu ilohv brx zloo qhhg wr hqwhu wkh
iroorzlqj frppdqg:

rshqvvo dhv-256-fef -sengi2 -d -g -lq T1.hqfubswhg -rxw
T1.uhfryhuhg -n hwwxeuxwh
```

The message in the file is encrypted with the caesar cypher! In order to decrypt it, the following command must be executed:

```
analyst@01a9901ea02c:~/caesar$ cat .leftShift3 | tr
"d-za-cD-ZA-C" "a-zA-Z"
```

The `tr` command translates one set of characters to the next set of characters. In this command, the first set of characters is `d-za-cD-ZA-C` and the second set of characters is `a-zA-Z`.

After executing this command, this was the result:

```
In order to recover your files you will need to enter the
following command:
openssl aes-256-cbc -pbkdf2 -a -d -in Q1.encrypted -out
Q1.recovered -k ettubrute
```

The message was successfully decrypted!

STEP 3 Decrypting the Target File

Lasty, I returned to the home directory by using the `cd ~` command to decrypt the target file.

```
analyst@01a9901ea02c:~/caesar$ cd ~
```

Then I executed the following command to decrypt the target file:

```
analyst@01a9901ea02c:~$ openssl aes-256-cbc -pbkdf2 -a -d -in
Q1.encrypted -out Q1.recovered -k ettubrute
```

The components of the command are as follows:

`openssl` - reverse encryption of file
`aes-256-cbc` - a symmetric cypher that works along with the openssl command
`-pbkdf2` - adds extra security for the key
`-a` - indicates the desire coding for the output
`-d` - decryption
`-in` - input file, where the encrypted message is at
`Q1.encrypted` - the encrypted file
`-out` - output file, indicates where the result of the decrypted file will go to
`Q1.recovered` - where the results of decrypting the file will be stored
`-k` - specifies the password
`ettubrute` - the password

Once executed, the directory will have the new file, **`Q1.recovered`**. Finally, I used the **`ls`** command to list out all the files in the directory, then use the **`cat`** command to open the **`Q1.recovered`** to see the decrypted message.

```
analyst@01a9901ea02c:~$ ls
Q1.encrypted  Q1.recovered  README.txt  caesar
analyst@01a9901ea02c:~$ cat Q1.recovered
If you are able to read this, then you have successfully
decrypted the classic cipher text. You recovered the encryption
key that was used to encrypt this file. Great work!
```

Then I am done! The message was successfully decrypted and is now readable!