Decrypt an Encrypted Message

Project Description

Learned about cryptography and how encryption and decryption can be used to secure information online. Also introduced to the Caesar cipher, one of the earliest cryptographic algorithms used to protect people's privacy.

As a security analyst, it's important that you understand the role of encryption to secure data online and that you're familiar with the right security controls to do so.

Read the contents of a file

Using the `ls` command to view the files and folders in your current directory.

We get to know that:

- `Q1.encrypted` – an encrypted file
- `README.txt` – a text file with instructions
- `caesar` – a subdirectory that you'll need to explore

Use the `cat` command to display the contents of the `README.txt` file:

```
analyst@d62da4824eae:~$ pwd
/home/analyst
analyst@d62da4824eae:~$ ls
Q1.encrypted  README.txt  caesar
analyst@d62da4824eae:~$ cat README.txt
Hello,
All of your data has been encrypted. To recover your data, you will need t
o solve a cipher. To get started look for a hidden file in the caesar subd
irectory.
analyst@d62da4824eae:~$
```

Find a hidden file

We first access the caesar subdirectory and find the list of hidden files, we get to know hat the there is only file that starts with (.) which identify the hidden file named .leftShift3. We read the contents of the hidden file using cat command but found that the text was encrypted. We see a message that appears jumbled because it's encrypted using a **Caesar cipher** with a **left shift of 3**.

```
analyst@d62da4824eae:~$ cd caesar
analyst@d62da4824eae:~/caesar$  ls -a
.  ..  .leftShift3
analyst@d62da4824eae:~/caesar$ cat .leftShift3
Lq rughu wr uhfryhu brxu ilohv brx zloo qhhg wr hqwhu wkh iroorzlqj frppdq
g:

rshqvvo dhv-256-fef -sengi2 -d -g -lq T1.hqfubswhg -rxw T1.uhfryhuhg -n hw
wxeuxwh
analyst@d62da4824eae:~/caesar$ cat .leftShift3 | tr "d-za-cD-ZA-C" "a-zA-Z
"
In order to recover your files you will need to enter the following comman
d:

openssl aes-256-cbc -pbkdf2 -a -d -in Q1.encrypted -out Q1.recovered -k et
tubrute
analyst@d62da4824eae:~/caesar$ cd ~
analyst@d62da4824eae:~$
```

Use the `tr` command to shift the letters 3 places to the left:

Explanation

- `tr` translates characters from one set to another
- The first set (`d-za-cD-ZA-C`) maps the encrypted letters.
- The second set (`a-zA-Z`) is the normal alphabet.
- This effectively reverses the Caesar cipher with a left shift of 3.

Before moving to the next task, return to your home directory using cd ~ command.

Decrypt the Encrypted File and Recover the Message

Using the exact command obtained from the previous task to decrypt the `Q1.encrypted` file and create a new file with the decrypted content:

openssl aes-256-cbc -pbkdf2 -a -d -in Q1.encrypted -out Q1.recovered -k ettubrute

Explanation of Command Components:

- `openssl` – Utility for performing encryption and decryption.
- `aes-256-cbc` – Specifies the encryption algorithm (AES with a 256-bit key in CBC mode).
- `-pbkdf2` – Applies a key derivation function for improved security.
- `-a` – Uses base64 encoding for input/output.
- `-d` – Tells OpenSSL to decrypt.
- `-in Q1.encrypted` – Indicates the encrypted input file.
- `-out Q1.recovered` – Specifies the output file where the decrypted data will be saved.
- `-k ettubrute` – Provides the password for decryption.

```
analyst@d62da4824eae:~/caesar$ cd ~
analyst@d62da4824eae:~$ openssl aes-256-cbc -pbkdf2 -a -d -in Q1.encrypted
 -out Q1.recovered -k ettubrute
analyst@d62da4824eae:~$ ls
Q1.encrypted  Q1.recovered  README.txt  caesar
analyst@d62da4824eae:~$ cat Q1.recovered
If you are able to read this, then you have successfully decrypted the cla
ssic cipher text. You recovered the encryption key that was used to encryp
t this file. Great work!
analyst@d62da4824eae:~$
```

Verify the recovered file by ls and should see `Q1.recovered` in the output, indicating successful decryption. Display the contents of the recovered file to read the hidden message by cat `Q1.recovered`