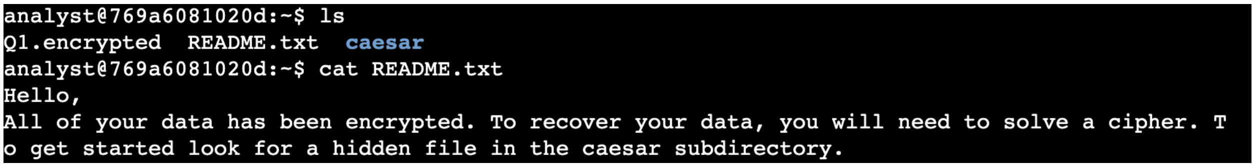
DECRYPTING DATA USING CIPHER AND OPENSSL

Project Description

The task was to recover encrypted data by working through a series of steps that involved exploring directories, reading files, decoding a classical cipher, and finally decrypting the target file using OpenSSL. This exercise combined file system navigation, cryptography, and Linux command-line tools to restore the file to its original state.

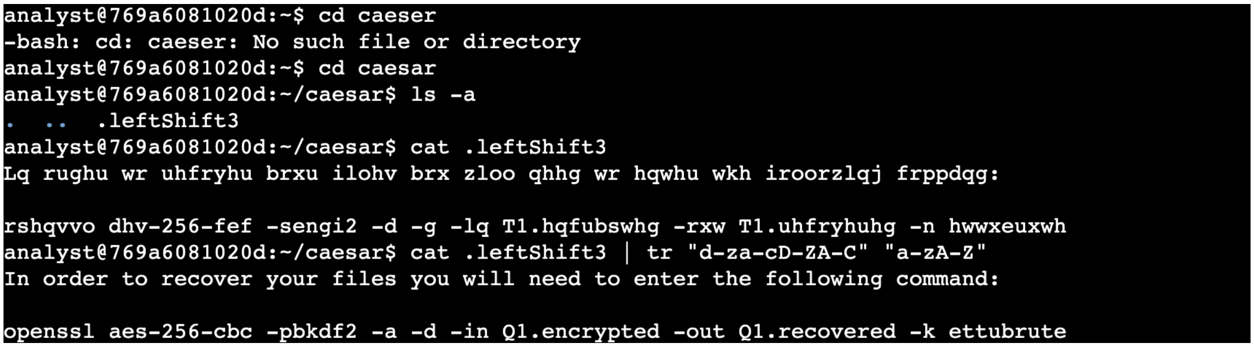
Steps Performed

I started by listing the contents of the working directory, where I found three items: Q1.encrypted, README.txt, and a folder named “caesar”. Reading the README.txt file revealed instructions that all the data had been encrypted and that I needed to solve a cipher. The note directed me to look for a hidden file inside the “caesar” directory.



```
analyst@769a6081020d:~$ ls
Q1.encrypted  README.txt  caesar
analyst@769a6081020d:~$ cat README.txt
Hello,
All of your data has been encrypted. To recover your data, you will need to solve a cipher. T
o get started look for a hidden file in the caesar subdirectory.
```

Next, I navigated into the “caesar” subdirectory and used “ls -a” to reveal hidden files. This uncovered “.leftShift3”, which contained text encoded with a classical Caesar cipher. By using the “tr” command to shift the alphabet three places to the left, I successfully translated the ciphertext into plaintext instructions.

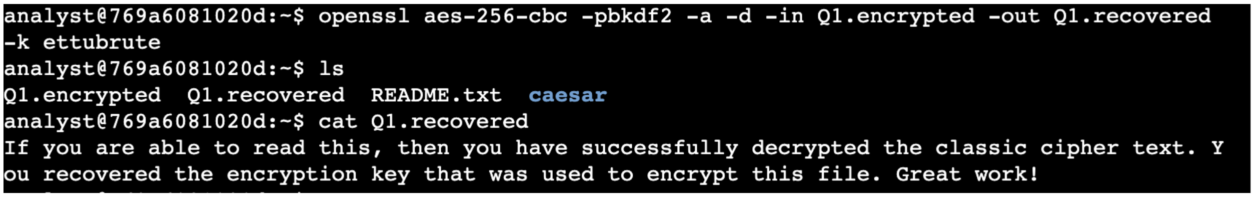


```
analyst@769a6081020d:~$ cd caeser
-bash: cd: caeser: No such file or directory
analyst@769a6081020d:~$ cd caesar
analyst@769a6081020d:~/caesar$ ls -a
.  ..  .leftShift3
analyst@769a6081020d:~/caesar$ cat .leftShift3
Lq rughu wr uhfryhu brxu ilohv brx zloo qhhg wr hqwhu wkh iroorzlqj frppdqg:

rshqvvo dhv-256-fef -sengi2 -d -g -lq T1.hqfubswhg -rxw T1.uhfryhuhg -n hwwxeuxwh
analyst@769a6081020d:~/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 command:

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

The decoded message provided the exact “openssl” command and decryption key required to restore the encrypted file. Using these details, I ran the OpenSSL command, which decrypted Q1.encrypted and produced a new output file, Q1.recovered.



```
analyst@769a6081020d:~$ openssl aes-256-cbc -pbkdf2 -a -d -in Q1.encrypted -out Q1.recovered
-k ettubrute
analyst@769a6081020d:~$ ls
Q1.encrypted  Q1.recovered  README.txt  caesar
analyst@769a6081020d:~$ cat Q1.recovered
If you are able to read this, then you have successfully decrypted the classic cipher text. Y
ou recovered the encryption key that was used to encrypt this file. Great work!
```

Finally, I opened Q1.recovered and confirmed that it contained a readable confirmation message. This verified that the decryption was successful and that the file had been restored to its original state.

DECRYPTING DATA USING CIPHER AND OPENSSL

Summary

In this project, I practiced listing directories, reading file contents, applying Linux commands to decode a Caesar cipher, and using OpenSSL to decrypt a file. These steps demonstrated how classical cryptography and modern encryption tools can be combined to securely protect and successfully recover data.