Decrypt an encrypted message

Project description

In my home directory, I have multiple files that have been encrypted using the Caesar Cipher encryption. I will search for those files and use the ‘tr’ commands to decrypt them. The tr command will translate a set of characters to another using mapping. Since the Caesar cipher encrypts characters by shifting the chosen characters left or right, e.g. ‘A’ changes to ‘D’.

Exploring content of the directory

I first needed to find the encrypted files. To do this, I need to navigate to my home directory and list all the files. To do this, I used the ‘ls’ command. These are the files currently in the directory.

```
analyst@fb4826e15e1c:~$ pwd
/home/analyst
analyst@fb4826e15e1c:~$ ls
Q1.encrypted  Q1.recovered  README.txt  caesar
analyst@fb4826e15e1c:~$
```

It's been a while since the encrypted files have been decrypted therefore to get additional information, I will open the ‘README.txt’ as it may contain some clues. To open it, I will use the ‘cat’ command. The files are hidden in the ‘caesar’ directory.

```
analyst@7fa57caedf5a:~$ ls
Q1.encrypted  README.txt  caesar
analyst@7fa57caedf5a:~$ cat README.txt
Hello,
All of your data has been encrypted. To recover your data, you will need to so
lve a cipher. To get started look for a hidden file in the caesar subdirectory
.
analyst@7fa57caedf5a:~$
```

Finding the hidden file

I first go to the ‘caesar’ subdirectory of my home directory, to do this, I use the command ‘cd’. For example, ‘cd caesar.’ Next, I use the ‘ls’ command along with the option ‘-a’ to display hidden files. To open the hidden file I use the ‘cat’ command. The hidden file contains scrambled letters due to the use of caesar cipher.

```
analyst@7fa57caedf5a:~$ cd caesar
analyst@7fa57caedf5a:~/caesar$ ls -a
.  ..  .leftShift3
analyst@7fa57caedf5a:~/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 hwwxeu
xwh
analyst@7fa57caedf5a:~/caesar$
```

Decrypting the file

To decrypt the hidden file, I will use this command ‘cat .leftShift3 | tr "d-za-cD-ZA-C" "a-zA-Z"’. This is what the tr command does:

First parameter: "d-za-cD-ZA-C"

- Tells the 'tr' command to map shifted letters back to their original form, effectively shifting each letter by '3'
- 'd-z' means 'd' will be 'a', 'e' will be 'b' and so on
- 'a-c' means 'a' will be 'x' and 'b' will be 'y' and so on
- 'D-ZA-C' is the same thing as above however maps uppercase instead

Second parameter: "a-zA-Z"

- This is a reference point
- Tells the first parameter what to replace each letter with
- Without the reference point, 'tr' wouldn't know how to map the shifted letters back to their original position.

```
analyst@77d804185b83:~/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 ettubr
ute
analyst@77d804185b83:~/caesar$
```

File content revealed

After decrypting the file, we got this command ‘openssl aes-256-cbc -pbkdf2 -a -d -in Q1.encrypted -out Q1.recovered -k ettubrute’, we can use this command to decrypt the file called Q1.encrypted. Here is the file we scrambled letters:

```
analyst@cf851177d93d:~$ cat Q1.encrypted
U2FsdGVkX1/nxHZY2p53/6gRmQ9alkNrVwOwPOgpTeB09rdnvKnydLPQsnOYHjgR
42Mwdv0ye94Im+u100Fl2+Bx3SHjJ7wZjOxA7Jew1x7g3LcRsRnFcFLyfAnn0f3u
xMIH/y+Y4HfVb6NUFueXM43M5Cn/Gz9JqIxpw+tZaajsrtZrsoEwenZEND1Ya6AY
rnVCjCFdTmSVG9EnzGxFT40DOw0yIhEAw5WqfBzjwgNSfz+p44Bnb3jUHsJt38gw
analyst@cf851177d93d:~$
```

This is what the command does:

- OpenSSL reverses the encryption of the file using symmetric cipher
- It uses Advanced encryption standard(AES) with 256 bit key
- Pbkdf2 makes the decryption password harder to guess
- -a encodes the output in base64 for easy readability
- -d means decrypt
- -in specifies the input encrypted file
- -out means output a file
- -k creates a password for the encryption key, the password itself is not the encryption key. To strengthen this key, pbkdf2 is used.

```
If you are able to read this, then you have successfully decrypted the class
ic cipher text. You recovered the encryption key that was used to encrypt th
is file. Great work!
analyst@cf851177d93d:~$
```

Summary

After discovering the hidden files, I extracted their content using the ‘tr’ command. Based on the information in those files, I was able to decrypt the encrypted files using OpenSSL. This was a basic method of decrypting a message in my home directory. Keeping a file that explains how to decrypt another file is a bad practice, a patient and persistent threat actor could eventually figure out the process.