

My Project on Using Linux commands to break Caesar cipher and Decrypt files

Maria Audu

Scenario

In this scenario, all of the files in the home directory have been encrypted. Use Linux commands to break the Caesar cipher and decrypt the files to read the hidden messages they contain.

First, explore the contents of the home directory and read the contents of a file. **Next**, find a hidden file and decrypt the Caesar cipher it contains. **Finally**, decrypt the encrypted data file to recover my data and reveal the hidden message.

Task 1. Read the contents of a file

In this task, explore the contents of my home directory and read the contents of a file to get further instructions.

1. Use the **ls** command to list the files in the current working directory.

The command to execute this step:

ls /home/analyst

Two files, Q1.encrypted and README.txt, and a subdirectory, caesar, are listed:

Q1.encrypted README.txt caesar

The README.txt file contains an important message with instructions i need to follow.

2. Use the **cat** command to list the contents of the README.txt file.

The command to execute this step:

cat README.txt

This will display the following output:

Hello,

All of my data has been encrypted. To recover my data, I will need to solve a cipher. To get started I will look for a hidden file in the caesar subdirectory.

The message in the README.txt file advises that the caesar subdirectory contains a hidden file.

In the next task, find the hidden file and solve the Caesar cipher that protects it. The file contains instructions on how to recover my data.

I completed this task and discovered the encrypted files in my home directory.

Task 2. Find a hidden file

In this task, find a hidden file in my home directory and decrypt the Caesar cipher it contains. This task will enable me to complete the next task.

1. First, use the **cd** command to change to the caesar subdirectory of your home directory:

cd caesar

2. Use the **ls -a** command to list all files, including hidden files, in my home directory.

The command to execute this step:

ls -a

This will display the following output:

. .. .leftShift3

Hidden files in Linux can be identified by their name starting with a period (.).

3. Use the **cat** command to list the contents of the .leftShift3 file.

The command to execute this step:

cat .leftShift3

The message in the .leftShift3 file appears to be scrambled. This is because the data has been encrypted using a Caesar cipher. This cipher can be solved by shifting each alphabet character to the left or right by a fixed number of spaces. In this example, the shift is three letters to the left. Thus "d" stands for "a", and "e" stands for "b".

4. Decrypt the Caesar cipher in the .leftshift3 file by using the following command:

cat .leftShift3 | tr "d-za-cD-ZA-C" "a-zA-Z"

The tr command translates text from one set of characters to another, using a mapping. The first parameter to the tr command represents the input set of characters, and the second represents the output set of characters. Hence, if you provide parameters "abcd" and "pqrs", and the input string to the tr command is "ac", the output string will be "pr".

This will display the following output:

In order to recover my files I will need to enter the following command:

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

In this case, the command **tr "d-za-cD-ZA-C" "a-zA-Z"** translates all the lowercase and uppercase letters in the alphabet back to their original position. The first character set, indicated by "d-za-cD-ZA-C", is translated to the second character set, which is "a-zA-Z".

The output provides the command I need to solve the next task!

5. Now, return to my home directory before completing the next task:

cd ~

I completed this task. I identified the hidden file in my home directory and decrypted the Caesar cipher contained in the hidden file.

Task 3. Decrypt a file

Now that I have solved the Caesar cipher, in this task I need to use the command revealed in `.leftshift3` to decrypt a file and recover my data so I can read the message it contains.

1. Use the exact command revealed in the previous task to decrypt the encrypted file:

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

In this instance, the `openssl` command reverses the encryption of the file with a secure symmetric cipher, as indicated by AES-256-CBC. The `-pbkdf2` option is used to add extra security to the key, and `-a` indicates the desired encoding for the output. The `-d` indicates decrypting, while `-in` specifies the input file and `-out` specifies the output file. The `-k` specifies the password, which in this example is `ettubrute`.

2. Use the `ls` command to list the contents of my current working directory again.

The command to complete this step:

`ls`

The new file `Q1.recovered` in the directory listing is the decrypted file and contains a message.

3. I will use the `cat` command to list the contents of the `Q1.recovered` file.

The command to execute this step:

`cat Q1.recovered`

This will display the following output:

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!

I completed this task by decrypting the `Q1.encrypted` file, recovering your data, and reading the message in the `Q1.recovered` file.

I now have practical experience in using basic Linux Bash shell commands to

- list hidden files,
- decrypt a Caesar cipher, and
- decrypt an encrypted file.