Lab 8 Notes - Encryption

Step 0: preparation

1. Install `openssl` by running this (or another command per your package manager)

```
sudo dnf install openssl
sudo apt install openssl
```

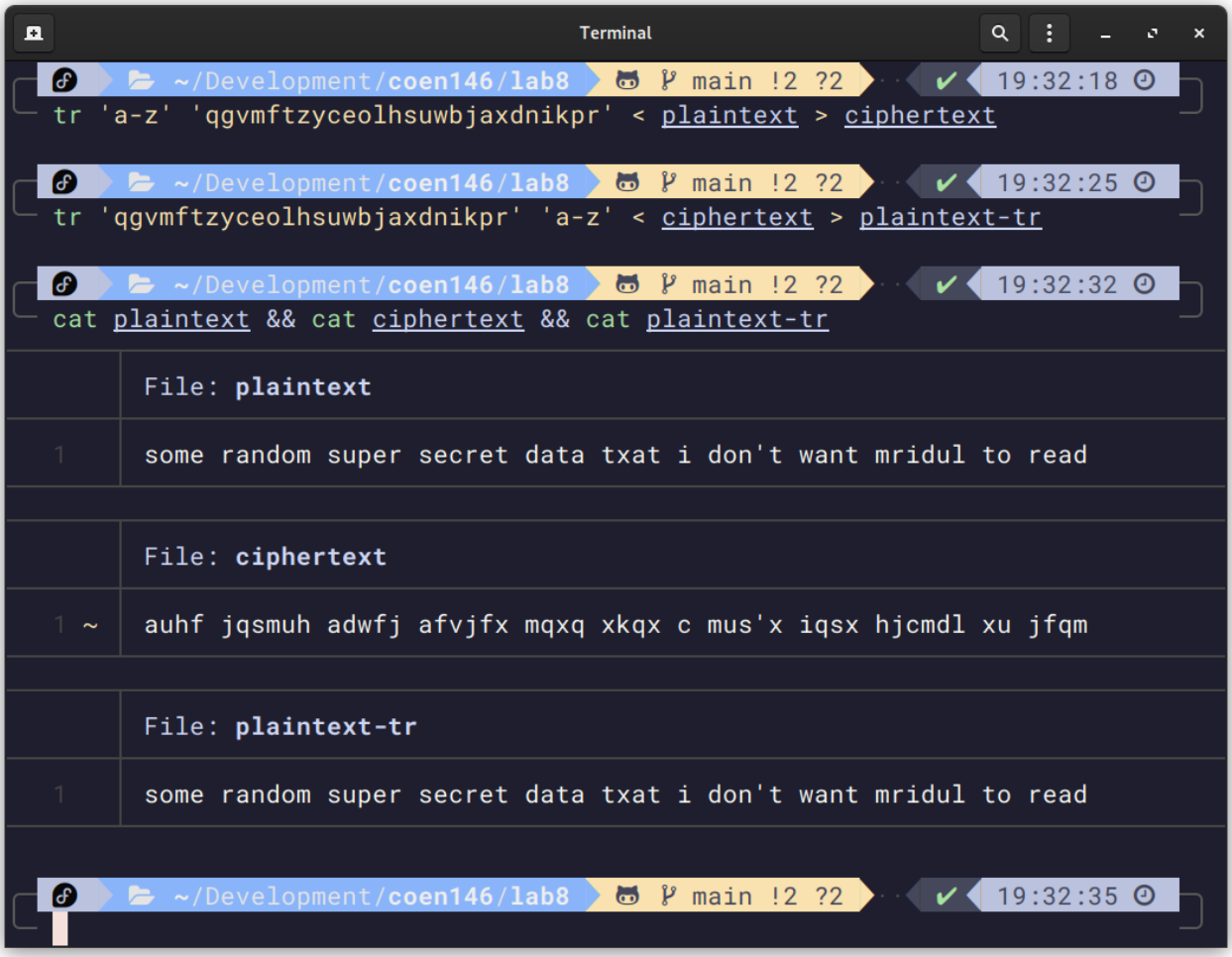
2. Create a plaintext file that we'll use for the rest of the project to encrypt

```
echo "some random super secret data txat i don't want mridul to read" >
plaintext
```

Step 1: tr tool

After creating a plain text file we learned to use `tr` tool with a random key to encrypt and decrypt the files. `tr` command, basically, swaps the characters in the file with provided key, replacing all instances of `s` with `a`, for example.

```
tr 'a-z' 'qgvmftzyceolhsuwbjaxdnikpr' < plaintext > ciphertext-tr
tr 'qgvmftzyceolhsuwbjaxdnikpr' 'a-z' < ciphertext-tr > plaintext-tr
```



Step 2: openssl and aes

Using openssl we encrypted our text file with a 128-bit key using AES algorithm.

```
openssl enc -aes-128-ecb -e -in plaintext -out ciphertext-aes -k
00112233445566778899AABBCCDDEEFF
% openssl enc -aes-128-ecb -d -in ciphertext-aes -out plaintxt-aes -k
00112233445566778899AABBCCDDEEFF
```

Step 3: openssl rsa tool and key encryption

This method allows us to use a public key and a password to decrypt the file. A private key is needed to encrypt the file but can also be used to decrypt it. We need to create a password that is used for decryption along with the public key when we create the private key.

```
openssl genrsa -aes128 -out privatekey 1024
openssl rsa -in privatekey -pubout > publickey
openssl rsautl -encrypt -inkey publickey -pubin -in plaintext -out
ciphertext-rsa
openssl rsautl -decrypt -inkey privatekey -in ciphertext-rsa -out
plaintext-rsa
```

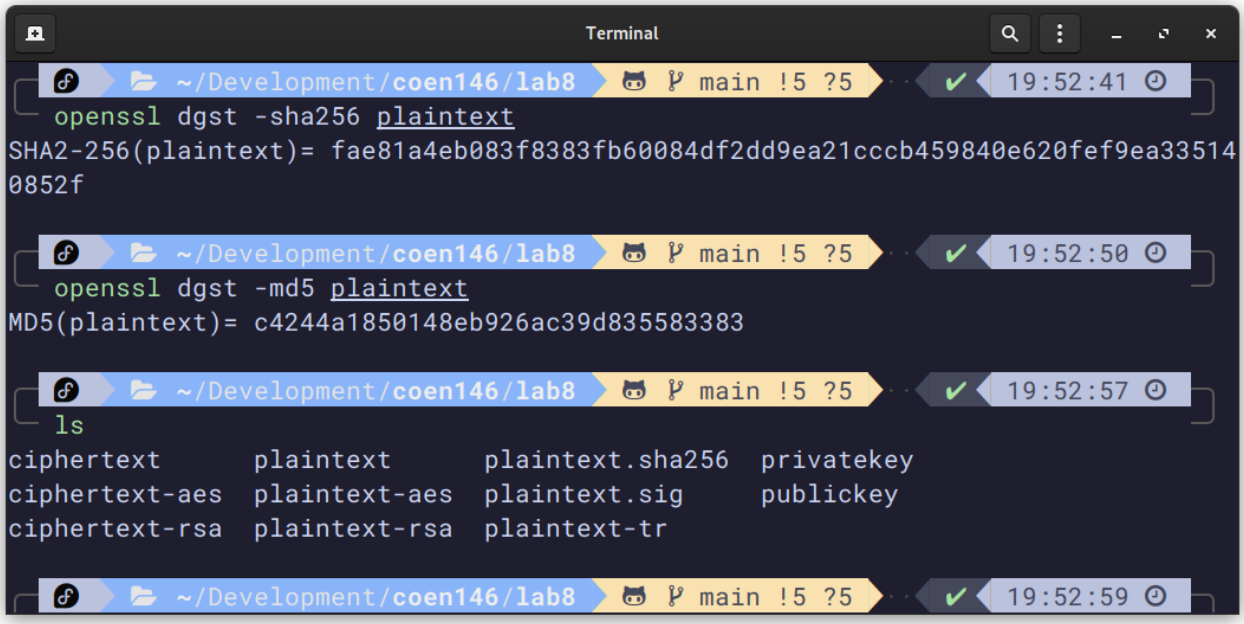
Step 4: digital signature using rsa

We can use the following commands to generate message hash, sign it, and verify the signature.

Step 5: calculating the hash

OpenSSL can be also used to calculate the hash

```
openssl dgst -sha256 plaintext
openssl dgst -md5 plaintext
```


Terminal
~/Development/coen146/lab8 main !5 ?5 19:52:41
openssl dgst -sha256 plaintext
SHA2-256(plaintext)= fae81a4eb083f8383fb60084df2dd9ea21cccb459840e620fef9ea335140852f
~/Development/coen146/lab8 main !5 ?5 19:52:50
openssl dgst -md5 plaintext
MD5(plaintext)= c4244a1850148eb926ac39d835583383
~/Development/coen146/lab8 main !5 ?5 19:52:57
ls
ciphertext plaintext plaintext.sha256 privatekey
ciphertext-aes plaintext-aes plaintext.sig publickey
ciphertext-rsa plaintext-rsa plaintext-tr
~/Development/coen146/lab8 main !5 ?5 19:52:59