Homework 1

Due Date: By the end of Wednesday, 2/5/2025.

Submit your answers to Problem 1 in PDF in HW1. Submit your answers to other questions in HW1-test in HuskyCT. The grade of the last attempt is the assignment grade. Write down how you get the answers in a document for reference.

OpenSSL is a library that includes many operations needed in crypto systems/protocols. It also provides an application that performs the operations in a terminal, through command line arguments or in an interactive environment. We experiment with the software in Problems 2, 3, and 4.

OpenSSL is installed in the virtual machine we are using for projects. You can also install OpenSSL on your local computer. Git on Windows 10 includes openssl. The command is openssl. The version used for preparing solution is 3.0.15.

1. Almost all recently issued credit cards are EMV cards, which have a chip that can store data in a much more secure way than magnetic strips. There are two authentication methods when using an EMV card: Chip and PIN or Chip and Signature. Do research and compare the two methods. What might be the reasons for countries/companies to adopt Chip and PIN, and what might be the reasons for adopting Chip and Signature, especially in the United States?

2. OpenSSL Encryption.

   a) "data.enc" is a file that contains the ciphertext encrypted with AES-256-CBC. The pass phrase is "CSE4400-Spring2025" (without quotation marks). The key derivation option is "-pbkdf2". Use OpenSSL to decrypt the file and save the plaintext into "data.txt". What is the command line you use to decrypt the file? What is the number near the end of the plaintext?

   b) "data.enc" is a binary file that some applications do not handle very well. Alternatively, OpenSSL can save the ciphertext in base64 format, which can be viewed with text editors. Encrypt the plaintext file with the parameters listed below and save the ciphertext in base64 format in file "data.asc".

   ```
   Algorithm: AES-128-CBC
   Key (-K): ebf01882090b2e6cf9b7e61b0ae5e3fb (in hex)
   Initialization vector: 5b2f79fbd76b65de872f5abfb8b5375e (in hex)
   Output encoding: -base64
   ```

   Note that openssl options are case sensitive. For example, -p and -P are different. Some options are given below. Find out other options in the manual.

3. OpenSSL Hash.
   a) What is the SHA256 hash value of the file “data.enc”? The last 4 hex digits are c62d.
   b) What is the SHA256 hash value of “data.txt”? The last 4 hex digits are 172c.
   c) What is the MD5 hash of “data.asc”? The last 4 hex digits are 94fd.

4. OpenSSL Public-key. We experiment with public key and digital signature in this problem
   a) “private.pem” is an RSA private key file. We can extract the public key from it. Use OpenSSL to save the public key in “public.pem”.
   b) “small.rsa.enc” is a file encrypted with “public.pem”. Decrypt it with the private key and save the plaintext in “small.txt”.
   c) Use the digt command to sign “data.enc” and then verify the signature. The hash algorithm is SHA256. The sign command looks like

```
openssl dgst -sign private.pem …
```

Extra

OpenSSL (v1.1.1) encryption manual page

https://www.openssl.org/docs/man1.1.1/man1/openssl-enc.html

Salt

Discussions about salting in openssl.

https://security.stackexchange.com/questions/20628/where-is-the-salt-on-the-openssl-aes-encryption

OpenSSL RSA

Sign and verify using OpenSSL
https://pagefault.blog/2019/04/22/how-to-sign-and-verify-using-openssl
If you sign the same file twice using the default padding, does signature change?
What do you see if you verify the signature with a different file?

Encryption tools

Although we learned some features in OpenSSL, GunPG or PGP might be better options for encrypting files. At least, GnuPG (GPG) and PGP files are in standard format. Here is an example of online discussions on OpenSSL vs GPG.

https://stackoverflow.com/questions/28247821/openssl-vs-gpg-for-encrypting-off-site-backups