1 Goal

In this lab, students will learn the idea of information hiding and explore the basic usage of OpenSSL.

2 Description

1. Download `alice.bmp` from `http://cs.unh.edu/~dxu/cs780/lab1/alice.bmp`.
2. Open the image by a hex editor, write your name inside the image, and save the new image as `alice-new.bmp`.
3. Compare the new image with the original image. Can you see any differences?
4. Generate digest for the original image and the new image using SHA-256. Compare the two digests.
5. Generate a RSA public and private key pair (2048 bit) using OpenSSL.
6. Generate a SHA-256 digest of new image signed by your private key.
7. Implement the encryption and decryption of standard TEA algorithm as two programs: `tea-enc.c` and `tea-dec.c`. The encryption program should be invoked from command-line by `tea-enc key plaintext` and the output file name is “ciphertext”. The decryption program is invoked by `tea-dec key ciphertext` and the output file name is “plaintext”.
8. Use your name as the encryption key to encrypt `alice-new.bmp`.
9. (BONUS) Implement CBC mode TEA encryption and decryption, and use it to encrypt `alice-new.bmp`.

3 Submission

Please pack the following files into a .zip file and submit to Canvas.

1. A lab report (.pdf) including the description and screenshot of every step.
2. `alice-new.bmp`, your public key, the signed SHA256 digest of `alice-new.bmp`.
3. `tea-enc.c`, `tea-dec.c`, your key, the ciphertext.
4. (BONUS) `tea-enc-cbc.c`, `tea-dec-cbc.c`, your key, the ciphertext.

4 Resource

- BMP file format: `https://en.wikipedia.org/wiki/BMP_file_format`
- Hex editor: Emacs hexl-mode
  `https://www.gnu.org/software/emacs/manual/html_node/emacs/Editing-Binary-Files.html`
  Or other tools you like.
- OpenSSL Tutorial: `https://wiki.openssl.org/index.php/Command_Line_Utilities`
  Commands: `openssl genrsa ...`, `openssl dgst ...`
- TEA: `https://en.wikipedia.org/wiki/Tiny_Encryption_Algorithm`