C++ JPEG Encryption User Documentation

Mark Watson, CS 306 Project, May 5th, 2010

1. Installation

Requirements: To install, one needs to have a *nix environment with GNU `make` and GCC installed and configured. It may very well be possible to compile in a Visual Studio environment, however that has not yet been tested.

Installation is simple:

```
% cd source_code
% make
g++ -g -c main.cpp
g++ -g -c encryptjpeg.cpp
g++ -g -c accessjpeg.cpp
g++ -g -c aesencrypt.cpp
g++ -g main.o encryptjpeg.o accessjpeg.o aesencrypt.o -o image_encrypt
rm *.o
chmod +x+x+x image_encrypt
% cp ./image_encrypt /usr/bin/
```

2. Running

The basic idea of the encryption program is to encrypt JPEGS. To test it out you need a JPEG and a plain text password. It's important to note that the longer the password the better. Plain text passwords are simply extended to fill up a key space, and so more digits means more security.

To encrypt an image simply type something like:

```
% image_encrypt -e input.jpg output.secure
Please enter a key: test
```

The -e flag denotes encryption, and the output file is stored in a secure way, not readable by anyone except someone with a key and this program. Once given a key, the file will be encrypted and stored into 'output.secure'.

Decryption follows a similar pattern:

```
% image_encrypt -d output.secure output.jpg
Please enter a key: test
```

In this case we chose a secure file as the input, and told the software to decrypt it using the '-d' flag. It is important to enter the same key that was used to encrypt it, otherwise the program will put garbage into the output file.

3. Compatibility and Additional Test Cases

There are a few sample JPEG files in the source directory that can be used to test the program. The output of the encryption is not listed here, as the files are too large. Currently the program only supports features described in part 2. It should work with a variety of JPEG files. Other image formats are not supported.