Assignment 2

Hyoungshick Kim

September 24, 2017

1 Finding the encryption key

This is the second programming practice to find a key to encrypt information using a symmetric cipher.

- Prof. Kim invented a new encryption algorithm called `DES-AES` by combining `DES` and `AES`: During encryption, the plaintext m is first encrypted with a 64 bits[1] key k_1 using `DES` in ECB mode, and then encrypted again with a 128 bits key k_2 using `AES-128` in CBC mode (Just use all 0's for the initialization vector; that is, 16 null characters (0x00)).
- Your goal is to write a program to implement a key-recovery attack on the `DES-AES` algorithm: Given a plaintext-ciphertext pair (m, c) such that $c = \texttt{AES-128}_{k_2}(\texttt{DES}_{k_1}(m))$ with some unknown keys k_1 and k_2, your program must output k_1 and k_2. To save your computational effort, Prof. Kim used only the MD5 hash values of passwords in the file[2] at `https://seclab.skku.edu/wp-content/uploads/2017/09/passwords.txt` for k_1 (the first 64 bits of the MD5 hash value generated from a password p_1) and k_2 (the MD5 hash value itself generated from a password p_2).
- The input file consists of two lines as follows"

```
[plaintext]
[ciphertext]
```

- That is, given the input file ("`PlaintextCiphertext.txt`") including a plaintext-ciphertext pair, your program must create the output file ("`keys.txt`") including the password p_1 used for the `DES` key k_1 and the password p_2 used for the `AES-128` key k_2).

```
[Input file: PlaintextCiphertext.txt]
SKKU is the top university in the world
CBMsz223gfHe6AH6I+IIEjpXxjFlupBrGYZ8CDYYr9WJj4j0cMuL8uAA/Yxr9pNK

[Output file: keys.txt]
coders
piewtf
```

[1] A 64-bit key is used here, of which every eighth bit is ignored, giving an actual key size of 56 bits.
[2] This file contains MD5 hashes (32 hexadecimal numbers) in form '`[hash password]`' at each line.

- In the input file ("`PlaintextCiphertext.txt`"), `UTF-8` and `Base64` are used to encode the plaintext and the ciphertext, respectively (If you don't have any idea about encoding scheme, please visit the website: `http://www.base64decode.org/`).
- Padding: Simply pad 0's to fill in the last block. If the last block doesn't have 16 bytes, repeatedly put zero (null) characters (0x00) until you have 16 bytes.
- We use a maximum of 100 kilobytes of plaintext for `DES-AES`.
- You will be judged by (1) the correctness of the passwords in the output file ("`keys.txt`") created by your program, and (2) the actual running time of the program and (3) the well-written document to explain your source code and your algorithm to crack the encryption algorithm.
- Your code should be written in ANSI C. We will use the GNU compiler (i.e., `gcc`) to compile your source code.
- To implement this program, you should also use the OpenSSL library (`https://www.openssl.org/`) for cryptographic operations that you need. OpenSSL is a software library that provides a full-featured cryptographic toolkit as well as an implementation of SSL. For compilation and installation, you can refer to this page (`https://wiki.openssl.org/index.php/Compilation_and_Installation`). Please see the web page (`http://www.firmcodes.com/how-do-aes-128-bit-cbc-mode-encryption-c-programming-code-openssl/`) if you want to know how to use OpenSSL for encryption.
- Please upload your source code (c files), instructions to illustrate how your source code works, document to explain your code and the performance analysis to iCampus. Submit your assignment by midnight, Sunday October 8; **late submissions are allowed with a penalty. Each 24 hours (or part thereof) late will cost you 20%.**
- **Your assignments must be your own original work.** We will use a tool to check for plagiarism in assignments.