41900 – Fundamentals of Security
Project – 1 (Week 4 – Week 8)

Secret Key encryption: Find the Key

OpenSSL provides an API called EVP, which is a high-level interface to cryptographic functions. Although OpenSSL also has direct interfaces for each individual encryption algorithm, the EVP library provides a common interface for various encryption algorithms. To ask EVP to use a specific algorithm, we simply need to pass our choice to the EVP interface. A sample code is given in http://www.openssl.org/docs/crypto/EVP_EncryptInit.html. Please get yourself familiar with this program, and then complete this project.

You are given a plaintext and a ciphertext, and you know that *aes-128-cbc* is used to generate the ciphertext from the plaintext, and you also know that the numbers in the IV are all zeros (not the ASCII character '0'). Another clue that you have learned is that the key used to encrypt this plaintext is an English word shorter than 16 characters; the word that can be found from a typical English dictionary. Since the word has less than 16 characters (i.e. 128 bits), space characters (hexadecimal value 0x20) are appended to the end of the word to form a key of 128 bits. Your goal is to write a program to find out this key. English word list is also provided along with the program structure in the Project-1.zip file. The plaintext and ciphertext is the following:

Plaintext (total 21 characters): `'This is a top secret.'`

Ciphertext (in hex format):
`2075386b75eed8b4f2b4a9c9b76967d072fe22daca7b8f5a56d16ce6ee483b59`

Expected Result:

If the C program has been completed correctly, you should be able to locate the word used as the key to encrypt the plain text. The program must:

1. Scan through the entire words.txt file.
2. Use each word to encrypt the plaintext.
3. Compare the encrypted text with the ciphertext bit by bit.
4. Find a match, based on which find the word used.
5. Display the found word, its ciphertext next to the ciphertext being searched for.
6. The length of the ciphertext.

41900 – Fundamentals of Security
Project – 1 (Week 4 – Week 8)

Important Notes

Note 1: If you choose to store the plaintext message in a file, and feed the file to your program, you need to check whether the file length is 21. Some editors may add a special character to the end of the file. If that happens, you can use a hex editor tool to remove the special character.

Note 2: In this project, you are supposed to complete the given program to invoke the crypto library. No credit will be given if you simply use the openssl commands to do this project.

Note 3: To compile your code, follow the comments provided in the Skeleton code. You must edit/complete the areas where the comments start with ' /* ' and ending with ' */ '.

Note 4: Instructions to install openssl developer libraries. The instructions have also been provided in the Project-1.zip file.

Note 5: Information regarding the program file will be discussed during the tutorial session by your respective Tutors. Any questions regarding the assignment must be conveyed during the tutorial session or emailed to the tutor (with the Subject Coordinator CC'd)