Introduction to Cyber Security 2020
Project: CBC Encryption

Submit by July 31, 2020

You should implement your solution in Python (if you plan on using special modules in Python then ask us before using them, to make sure that we allow it). **For each function, you should also write a short description documentation (as a comment) of the algorithm that you implemented.**

Submission Instructions

Will be provided.

Reminder

A hex string is a string composed of hexadecimal characters `0`, . . . , `F`. Every ASCII character is represented by 2 hexadecimal characters. For example, the NULL character which is 0 in ASCII is represented by the hex string `00`, the characters 'a',. . . ,'z', which are 97,. . . ,122 in ASCII, are represented by the hex strings `61`,. . . ,`7A` respectively. Sometimes, to make sure that we are talking about a hex string we add the '0x' in the beginning of it ('\x' is another common way), but it is not necessary.

Whenever you expect a 'hex string' input to your function then you should expect a string of the above format. For instance, the hex string "`68656C6C6F`" represents the ASCII string "hello", and the hex string "`68656C6C6F2C20776F726C64`" represents the ASCII string "hello, world".

CBC Encryption Mode

Recall that CBC (cipher block chaining) is an encryption mode for block cipher. Let E be a block cipher which receives a 16 byte key k and a 16 byte plaintext m, and outputs the 16 byte ciphertext $E(k, m)$.

CBC encryption receives as inputs a message composed of n blocks, each of length 16 bytes: $m_1, \ldots, m_n$. It also receives as input a 16 byte known initialization vector IV.

Given this input, CBC encryption produces a ciphertext of $n+1$ blocks, each of length 16 bytes:

- $c_0 = IV$
- $c_i = E(k, m_i) \oplus c_{i-1}$, for $1 \leq i \leq n$.

Question 1

We will implement CBC with AES as the block cipher.

Write a function which receives three inputs: a key k, an integer n, and a string of $n+1$ blocks of 16 bytes. The function should output the n block CBC decryption of this ciphertext.

Question 2

Suppose that bit number j of block c_i of the ciphertext got flipped (namely, if the original value of the bit was 0 then it changed to 1, and if the original value of the bit was 1 then it changed to 0). Convince yourself that the decryption process will decrypt all blocks correctly, except for blocks i and $i+1$. The decryption of block i will be completely random, and the decryption of block $i+1$ will be correct, except for bit j in this block that will be flipped.

Write a function which receives three inputs:

- A key k
- An integer n
- A string of $n+1$ blocks of 16 bytes. This string was generated in the following way:
  - Each of the n plaintext blocks m_i, was generated by choosing a random byte and repeating it 16 times.
  - The plaintext message $m_1, \ldots, m_n$ was encrypted using the key k in CBC mode. The result is $c_0, \ldots, c_n$.
  - A random bit in one of the blocks $c_1, \ldots, c_{n-1}$ was flipped.
  - The resulting $n+1$ blocks are the input given to the function.

Your function should output the original value of the block whose encryption was completely corrupted.