CS381 Exercise 6

Name: Zhang Yupeng

Student ID: 5130309468

1. Design a method of padding for DES, so that you can encrypt an 80-bit plaintext into 80-bit ciphertext in ECB mode. Can we do this with AES?

Solution:

Block cipher algorithms like AES and Triple DES in Electronic Code Book (ECB) and Cipher Block Chaining (CBC) mode require their input to be an exact multiple of the block size. If the plaintext to be encrypted is not an exact multiple, we need to pad before encrypting by adding a padding string. When decrypting, the receiving party needs to know how to remove the padding in an unambiguous manner.

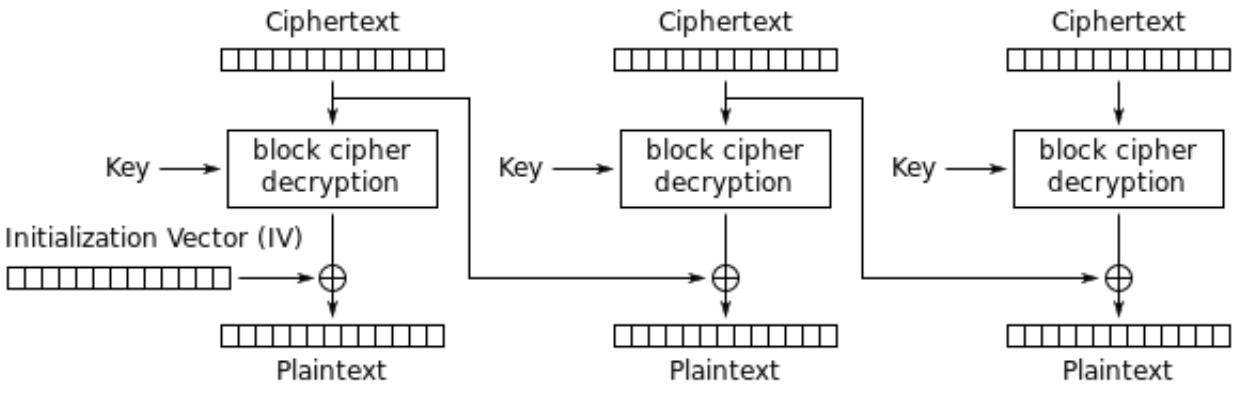
To solve the padding problem, we can pad with bytes all of the same value as the number of padding bytes. That is to pad the input with a padding string of between 1 and 8 bytes to make the total length an exact multiple of 8 bytes. The value of each byte of the padding string is set to the number of bytes added - i.e. 8 bytes of value 0x08, 7 bytes of value 0x07, ..., 2 bytes of 0x02, or one byte of value 0x01.

We can do this with AES as long as we change the padding string's length in order to match the AES's key's length.

2. Prove that in CBC mode, an error in ciphertext affects only 2 plaintext blocks;

Proof:

The following figure is the process of CBC mode decryption:



Cipher Block Chaining (CBC) mode decryption

By the picture we can see that mathematical formula for CBC decryption is:

$$P_i = D_k(C_i) \oplus C_{i-1}$$

So that the error occur in ciphertext block can only affect that plaintext block and the one immediately following it, but none after that.

3. Is pseudo-random number generator a one-way function, and why?

Solution:

Yes， a pseudo-random number generator is a one-way function.

A distribution D_n is considered pseudorandom if it is indistinguishable from the uniform distribution U_n. Formally, for any PPT (probabilistic polynomial time) algorithm A and polynomial function $p(n)$:

$$|Pr_{x\in D_n}[A(x) = 1] - Pr_{x\in U_n}[A(x) = 1]| < 1/p(n)$$

And a pseudorandom generator is a function $G : 0, 1^l \to 0, 1^m$ where $l < m$ and

1. G can be computed in time polynomial in l,
2. $G(x)$ is pseudorandom, where x is uniformly distributed.

A one-way function is a function $f : X \to Y$ such that $f(x)$ can be computed in time polynomial in $|x|$, but for any polynomial randomized algorithm A that attempts to inverse f:

$$Pr[f(A(f(x))) = f(x)] < 1/p(|x|)$$

where $p(\cdot)$ is any polynomial function.

Suppose a pseudorandom number generator $G : 0, 1^l \to 0, 1^m$ is not a one-way function.

Then there exists a polynomial algorithm A and a constant c such that:

$$Pr[f(A(f(x))) = f(x)] \geq |x| - c$$

Let C be an algorithm that returns 1 if $f(A(y)) = y$ and returns 0 otherwise. Then:

$$\begin{aligned} & Pr_{x\in U_l}[C(f(x)) = 1] \\ = & \sum_{y\in\{0,1\}^m} Pr_{x\in U_l}[f(x) = y]Pr[C(f(x)) = 1|f(x) = y] \\ \geq & \sum_{y\in\{0,1\}^m} 2^{-l}Pr[C(y) = 1] \\ \geq & \sum_{y\in\{0,1\}^m} 2 \cdot 2^{-m}Pr[C(y) = 1] \\ = & 2Pr_{y\in U_m}[C(y) = 1] \end{aligned}$$

Let D_m denote the range of f.

$$|Pr_{y \in D_m}[C(y) = 1] - Pr_{y \in U_m}[C(y) = 1]| \geq \frac{1}{2}|x|^{-c}$$

Thus, f is not a pseudorandom generator, which leads to a contradiction. Hence, a pseudorandom generator is a one-way function.