CSC 591 Cryptography 10/1/2018

Lecture Block-Cipher

Lecturer: Alessandra Scafuro *Scribe: James Wheeler*

PRP and Block-Cipher

In this lecture, expanding from the concept of PRF, we defined a Pseudo-Random Permutation (PRP) and the idea of block-ciphers. Block ciphers can be created using PRPs.

Definition

A Pseudo-Random Permutation (PRP) is a keyed permutation F and is invertible. There is also a 1-to-1 mapping. A PRP consists of F and F^{-1} such that:

$$F_k(x) = y$$

$$F_k^{-1}(y) = x$$

Scheme: Feistel Transform

If you have a PRF, you can transform it to a PRP by using a Feistel transform. Examples of real- world PRPs are 3DES, AES, and Chacha20. One iteration or round of Feistel transform is as follows:

If you have message $m = m_L||m_R$, where $length(m_L) = length(m_R)$ and PRF F_k, then one round of the Feistel transform in the forward direction gives c:

$$c_L = m_R$$

$$c_R = F_k(m_R) \oplus m_L$$

$$c = c_L||c_R$$

The inverse of this Feistel transform is done as follows:

$$m_R = c_L$$

$$m_L = F_k^{-1}(c_L) \oplus c_R$$

$$m = m_L||m_R$$

For multiple rounds, the output of round $r - 1$ becomes the input for the next round r.

Security Proof: Feistel Transform

One Round

It is intuitive to see this is not secure enough with running one Feistel iteration. If the input is

$$m = m_L||m_R$$

you know the output is

$$c = m_R||c_R$$

so with input $x = L_1||R_1$, we can build a distinguisher $D(y)$ that operates as follows :

1. parse $y = L_2||R_2$
2. if $L_2 = R_1$, output 1, else output 0

We note that if $y \leftarrow F(m)$, where $F(\cdot)$ is the above Feistel transform then,

$$Pr[D(y) \rightarrow 1] = 1$$

hence NOT a PRP

Two Rounds

What is this? Using two rounds is not as intuitive that it is not secure enough. Assume input $x_1 = 0^n||R_1$ and $x2 = 1^n||R_1$, then we construct the distinguisher $D(y)$ as follows :

1. parse $y_1 = L_1||R_2$ and $y_2 = L_2||R_3$
2. $L_2 \oplus 1^n = L_1$, output 1, else output 0

We note that if $y \leftarrow F(F(m))$, where $F(\cdot)$ is the Feistel transform then,

$$Pr[D(y) \rightarrow 1] = 1$$

hence NOT a PRP

Theorem: If F is a PRF, the three rounds of Feistal is a PRP

At least three rounds is pseudorandom and secure. This is why DES (Data Encryption Standard) became 3DES (uses 3 rounds). DES was later deemed insecure due to its 56b key.

Scheme: Block-cipher Modes

We now describe different block cipher modes of operation. A mode of operation is a way of encrypting arbitrary-length messages using a block-cipher.

Electronic Code Book (ECB)

This is the simplest mode of operation. Given a plain-text $m = m_1||m_2$, the ciphertext is obtained by encrypting each block m_1 and m_2 separately. And encryption here is a direct application of the PRP $F_k(\cdot)$ as $c =< F_k(m_1), F_k(M_@) >$.
Given $m = m_1||m_2$ and PRP F_k

$$c_1 = F_k(m_1)$$
$$c_2 = F_k(m_2)$$
$$c = c_1||c_2$$

This is not CPA secure and it is trivial to attack.

Output Feedback (OFB)

In this mode of operation, we use a block cipher to generate a pseudorandom stream that is then XORed with the message. An IV is chosen and a pseudorandom stream is generated. Following this each block of the plain text is XORed with the appropriate block of the stream.
Given $m = m_1||m_2...||m_n$, PRF F_k, and Initialization vector $IV = (c_0)$

$$c_1 = m_1 \oplus F_k(c_0)$$
$$c_2 = m_2 \oplus F_k(c_1 \oplus m_1)$$
$$...$$
$$c_n = m_n \oplus F_k(c_{n-1} \oplus m_{n-1})$$
$$c = c_0||c_1...||c_n$$

This is good for parallel encryption and no need of inversion.

Cipher Block Chaining (CBC)

In this mode of operation, a random initial vector is first chosen. Then, the first ciphertext block is generated by applying the pseudorandom permutation to $IV \oplus m_1$. The remainder of the ciphertext is obtained by XORing the i-th ciphertext block with the $(i+1)$-th plaintext block.
Given $m = m_1||m_2...||m_n$, PRP F_k, and IV c_0

$$c_1 = F_k(m_1 \oplus c_0)$$
$$c_2 = F_k(m_2 \oplus c_1)$$
$$...$$
$$c_n = F_k(m_n \oplus c_{n-1})$$
$$c = c_0||c_1...||c_n$$

Decryption is sequential and needs inversion.
Chained CBC is when the output from one CBC cipher is used as the IV for the next CBC cipher. In the next section we analyse the security of chained CBC.

Security Proof: Chained CBC

We show that there exists a PPT adversary A_{CPA} that wins with probability 1
Assume we have $A_{CPA}(1^n)$

Training Phase:

The adversary queries an oracle to receive the encryption of message m_1 as follows :

$$m_1 \rightarrow (IV, c_1)$$

Challenge Phase:

The adversary now constructs two messages m_0 and m_1 as follws :

$$m_0 = IV \oplus x \oplus c_1$$

$$m_1 = frank$$

The adversary sends the two messages to the challenger and receives a ciphertext back from the challenger as c_b

$$challenge(m_0, m_1) \rightarrow c^*$$

Decision Phase:

if $c^* = c_1$, then output 0, else output 1

Analysis:

Let the output be *out*
if $out = 0$, then

$$c^* = Enc(m_0)$$

$$c^* = F_k(m_0 \oplus c_1)$$

$$c^* = F_k((IV \oplus x \oplus c_1) \oplus c_1)$$

$$c^* = F_k(IV \oplus x)$$

$$c^* = c_1$$

$$A_{CPA} \rightarrow 0$$

$$Pr[A_{CPA} \rightarrow 0 | out = 0] = 1$$

if $out = 1$, then

$$c^* = Enc(m_1)$$

$$c^* = F_k(m_1 \oplus c_1)$$

$$c^* = F_k(frank \oplus c_1)$$

$$c^* \neq c_1$$

$$A_{CPA} \to 1$$

$$Pr[A_{CPA} \to 1 | out = 1] = 1$$

hence Chained CBC is NOT CPA secure, since the adversary can always guess which message was encrypted by the challenger.