Pham Hoang Long Do
NetID: 2995060

Homework 2

CSE 3400
Ghada Almashaqbeh

Problem 1

Let $F : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$ be a PRF, state whether the following constructions are PRFs (in all parts k is a long random secret key).

1. $F'_k(x) = (x \oplus k) \parallel F_k(x) \parallel F_k(x+1)$, where each of k and x is of length n bits.
   This encryption scheme has the following properties:

   (a) The second part ($F_k(x)$) is a secure PRF since key is private and a long random string.

   (b) The third part ($F_k(x+1)$) is also a secure PRF with the same reason like above.

   However, the first part of this encryption scheme includes the key within its algorithm. An attacker could follow this part of this encryption scheme. He can query a message to the oracle and receive the ciphertext back. The algorithm that the attacker can use after getting the ciphertext back:

   (a) Extract the last n-bits of the ciphertext. This is where he gets $x \oplus k$. Call this part Ω.

   (b) He can $\Omega \oplus x = k$ to get the key.
   Hence, it's **not secure**.

2. $F''_k(x) = F_{k_1}(x) \oplus F_{k_2}(x)$, where $k = k_1 \parallel k_2$, and each of k_1, k_2, x is of length n bits. This is a secure PRF for the following reason:

   (a) F is a secure PRF. k_1 and k_2 are random secret. Therefore, $F_{k_1}(x)$ and $F_{k_2}(x)$ are secure PRF and provide pseudo random outputs.

   (b) If we Xor two pseudo random outputs together then it's also a pseudo random output.

   Therefore, $F''_k(x)$ is a **secure PRF**.

3. $F'''_k(x) = k_1 \parallel F_{k_2}(x)$, where $k = k_1 \parallel k_2$ and each of k_1, k_2, x is of length n bits. This is a **secure PRF** for the following reason:

   (a) The attacker knows how k is constructed.

   (b) The attacker can find out what is k_1 by extract the last n bits of the output of the $F'''_k(x)$ (This functions outputs $2n$ bits).

   (c) The attacker, now, knows about the first half part of the key. However, he still doesn't know the second part of the key. Therefore, $F_{k_2}(x)$ is still a PRF.

Problem 2

1. Given message $m \in \{0,1\}^n$, choose random string $r \in \{0,1\}^{n/2}$, and form an encryption as: let $y = G(r)$, $E_k(m) = (y, F_k(y) \oplus m)$.
   The decryption schem is as follow: $D_k(c) = (y, F_k(y) \oplus c)$. This is **a secure encryption** scheme since $E_k(m)$ constructed by 2 part:

   (a) The first part y is PRG which will output random pseudo so it's secure.

   (b) The second part involves a PRG within a PRF so the output of $F_k(y)$ will be pseudo random.

   (c) Xor that output with the message will produce another pseudo random string.

   Therefore, this encryption scheme is secure.

2. Given message $m \in \{0,1\}^n$, choose a random string $r \in \{0,1\}^n$ and encrypt m as $E_k(m) = (r, lsb(F_k(r)) \parallel (F_k(r) \oplus m))$ where lsb is the least significant bit.
   Lets assign the output of $E_k(m) = (c_0, c_1)$ where $c_0 = r$ and $c_1 = lsb(F_k(r)) \parallel F_k(r) \oplus m))$. The decryption algorithm is constructed as follow. On c_1, extract the lsb, take the rest XOR with $F_k(r)$ to recover the message.
   This is **not a secure** encryption scheme since the attacker can use an advantage given in the part: $lsb(F_k(r))$. He can queries two message $m_0 = 0000...000000$ and $m_1 = 1111...111111$. Having the knowledge of the $lsb(F_k(r))$. He can XOR that bit with the lsb of m^* to tell which one is m_0 and m_1. This give the attacker 100% chance to distinguish which one is m_1 and which one is m_0.
3. Given message $m \in \{0,1\}^{3n}$, parse m as $m = m_1 \parallel m_2 \parallel m_2$ where $|m_1| = |m_2| = |m_3| = n$, then choose a random $r \in \{0,1\}^{n/2}$ and $r' \in \{0,1\}^n$ and encrypt m as: $E_k(m) = (r, r', G(r) \oplus m_1, F_k(r') \oplus m_2, F_k(r'+1) \oplus m_3)$.
   This is **not a secure** encryption scheme since the algorithm includes r and $r^{'}$ within the algorithm. In addition, The algorithm G is public, so the attacker can evaluate $G(r)$. He can use this to XOR with the third part of the algorithm $G(r) \oplus m$ to obtain m_1 which is the last n bits of the original message. The CPA attacker can tell what m_b is base on this flaw of the encryption algorithm. The successful rate of this attacker is 100%.

Problem 3

- Since deterministic encryption scheme always produce the same output given the same input. If the CTO attacker observes two indetical ciphertexts, he can infers that they came from the same plaintext. He can build a large dictionary of plaintext/ciphertext pair, and keep eavesdroping the channel to get a partial or maybe even full message. It's just the matter of time for this type of attack. Therefore, it is **not secure**.
- This is **still secure**. At the first time when the attacker queries m_0 and m_1 to the oracle. The oracle uses two different values of r, and the attacker receives c_0 and c_1. When the oracle challenges the attacker with c_b he uses different r than the previous one. Therefore, c_b is indistinguishable from c_0 and c_1. The output of $F_k(r) \oplus m$ is 1 bit, so he has an equal advantage of 50% to guess this is the output of the PRF and from the true random function. Hence, his advantage is $0.5 - 0,5 = 0\%$.