UConn, School of Computing
Fall 2024
CSE 3400/CSE 5850: Introduction to Computer and Network Security / Introduction to Cybersecurity

Assignment 2

Instructor: Prof. Ghada Almashaqbeh
Posted: 9/18/2024
Submission deadline: 9/26/2024, 11:59 pm

Note: Solutions **must be typed** (using latex or any other text editor) and must be submitted as a pdf (not word or source latex files).

Problem 1 [45 points]
Let $F : \{0,1\}^n \times \{0,1\}^n \rightarrow \{0,1\}^n$ be a secure PRF, state whether the following constructions are secure PRFs (in all parts k is a long random secret key).

1. $F'_k(x) = F_k(\bar{x}) \parallel F_k(x)$, where each of k and x is of length n bits, and $\bar{x}$ is the bitwise negation of x.

   The PRF is secure. Both $F_k(\bar{x})$ and $F_k(x)$ would output whaht appears as random strings which the attacker would be unable to differentiate from completely random. Even though both use similar inputs, equal of length and opposite, it does not matter because of the randomn appearance of the function's outputs. The randomness would give the attacker no advantage, thus $.5 - .5 = 0$.

2. $F''_k(x) = \big(F_{k_1}(x) \oplus F_{k_2}(x)\big) \parallel x$, where $k = k_1 \parallel k_2$, and each of k_1, k_2, x is of length n bits.

   The PRF is secure. Even though on the surface it appears $k = k_1 \parallel k_2$ may create a vulnerability, it does not as the attacker could still not differentiate the output from completely random. Because of this randomness, the attacker would still have no advantage, thus $.5 - .5 = 0$.

3. $F'''_k(x) = lsb(F_{k_1}(x)) \parallel F_{k_2}(x)$, where $k = k_1 \parallel k_2$, each of k_1, k_2, x is of length n bits, and lsb is the least significant bit.

   This PRF is not secure. As the output string is $F'''_k(x) = lsb(F_{k_1}(x)) \parallel F_{k_2}(x)$, the first bit in the string would always be a 0 or a 1, giving the hacker the ability to differentiate the PRF functions and then a fifty fifty for guessing as the first bit could only be a 0 or a 1. This gives the hacker an advantage of $1 - .5 = .5$ which is non-negligible and means the PRF is not secure.

Note: if the scheme is not a PRF then provide an attack against it and analyze/justify its success probability. If the scheme is a PRF, just provide a convincing argument (formal proofs are not required) and state why the attacker advantage is negligible.

Problem 2 [45 points]
Let $G : \{0,1\}^{n/2} \to \{0,1\}^n$ be a secure PRG, and $F : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$ be a secure PRF. For each of the following encryption constructions, state the decryption algorithm, and then state whether it is a secure encryption scheme against a CPA attacker. (All the following are block ciphers; we encrypt m all at once, and in all parts k is a long random secret key.)

1. Given message $m \in \{0,1\}^n$, choose random string $r \in \{0,1\}^n$, and form an encryption as: let $y = F_k(r)$, $E_k(m) = (RH(y), G(RH(y)) \oplus m)$, where RH is the right half of the string.

   The decryption process involves taking the right half of the ciphertext, running it through the PRG, G, and XORing it with the left half of the output to recover the plaintext. This would allow the attacker to identify it as a non-random string with an advantage of $1 - .5 = .5$ or non-negligible.

2. Given message $m \in \{0,1\}^n$, choose a random string $r \in \{0,1\}^n$ and encrypt m as $E_k(m) = (r, F_k(F_k(r)) \oplus m)$.

   The decryption process would take the left half of the output and then perform the operation $F_k(F_k(r)) \oplus m$. As r would be a completely random string, both the left half and right half would be appear random, thus not giving the attacker any advantage.

3. Given message $m \in \{0,1\}^{3n}$, parse m as $m = m_1 \| m_2 \| m_2$ where $|m_1| = |m_2| = |m_3| = n$, then choose a random $r \in \{0,1\}^n$ and a random $r' \in \{0,1\}^n$ and encrypt m as: $E_k(m) = (r, r', F_k(1^n) \oplus m_1, F_k(r) \oplus m_2, F_k(r') \oplus m_3)$.

   This is nearly secure, however, $F_k(1^n) \oplus m_1$ creates a vulnerability. As $F_k(1^n)$ would feed the same input into F_k, if the attacker sends multiple messages with the same key the attacker would be able to identify this portion of the output thus allowing the attacker to find m_1 as m_1 would simply be the flipped bits of that constant string.

Note: If the scheme is insecure then provide an attack against it and analyze its success probability. If the scheme is secure, just provide a convincing argument (formal security proofs are not required) and state why the attacker advantage is negligible.

Problem 3 [15 points]

- Alice claims that OTP is a deterministic encryption scheme (so it cannot be secure against a CPA attacker) since there is no randomness generation in OTP. Is her claim true? Justify your answer.

  Alice is not correct. OTP is secure against a CPA attacker as long as the key was only used once. If the key was used more than once, the attacker can use information from one text on the other, revealing information about the plaintext.

- Show how to decrypt (or basically invert) using the Feistel network shown in Slide 10, Lecture 4. So given an input $g_k(m)$ that is described in that slide, can you get m back using the same network structure? If yes, how?

  The Feistel network is basically a series of bitwise operations mixed with cutting in halves. All one would need to do to decrypt, or invert, would be to follow the series of operations and simply do the inverse working backwards.

Note: This problem has 5 points bonus.