Worksheet 1 - CPA

October 4, 2022

1 Useful definitions

1.1 Perfect Secrecy (Shannon secrecy)

Suppose the adversary knows how an encryption scheme ($\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec}$) is implemented.

The perfect secrecy game for the encryption scheme works as follows:

- The adversary chooses two messages $m_0 \neq m_1$. Send m_0, m_1 to the game master.
- The game master (who knows the secret key k) chooses m_0 or m_1 at random. In other words, it choose a random bit b. Then, the game master computes the challenge $c^* = \mathsf{Enc}_k(m_b)$ and sends c^* to the adversary.
- The adversary guesses if m_0 or m_1 was encrypted: it outputs a bit b' that represents its guess.
- The adversary **wins** if $b' = b$.

The encryption scheme is **perfectly secure** if the adversary wins the game with probability exactly $\frac{1}{2}$.

1.2 Chosen Plaintext Attack (CPA)

The adversary attempts to break an encryption scheme. In CPA, the adversary has a method of encrypting messages: the adversary ask for messages m to be encrypted and see the ciphertext $\mathsf{Enc}(m)$[1]. Given this power, the adversary **breaks CPA security** if it can win the perfect secrecy game. However, it **cannot** choose m_0, m_1 to be any message that is already encrypted.

[1] In cryptography, we say that the adversary has access to **an encryption oracle**

Exercise 1. Consider an encryption scheme where the first bit of a message is equal to the last bit of its ciphertext.

This encryption scheme is not perfectly secure - show an attack.

Exercise 2. Consider an encryption scheme that works as follows: The plaintext messages m have length $2n$. The secret key k has length n. To encrypt, for each bit m_i, compute $m_i \oplus k_{\lfloor i/2 \rfloor}$. In other words, XOR 1st two bits of m with the 1st bit of k, the 2nd two bits of m with the 2nd bit of k, etc.

This is not a perfectly-secure encryption scheme: show an attack.

Exercise 3. The following “encryption scheme” is *not* secure. Let k be a n bit key. To encrypt an n-bit plaintext m, output ciphertext $c = k \oplus m$. We use the same key k to encrypt every n-bit plaintext message. (The symbol $\oplus$ is the bitwise XOR; recall that $a \oplus a \oplus b = b$.)

1. Write down the decryption algorithm.

2. Present an attack that proves that this scheme is not CPA secure.