Applied Cryptography

Week #1 Extra

Bernardo Portela and Rogério Reis

2023/2024

Important

- Your answers must **always** be accompanied by a justification. Presenting the final result (e.g. the result of a calculation) without the rationale that laid to said result will result in a grade of 0.
- Submit your answers via e-mail to *bernardo.portela@fc.up.pt*, with adequate identification of the group and its members.

Notation:

Note: reverse denotes the function that takes a bit string and produces the reverse bit string. $||$ denotes the concatenation of bit-strings. $\oplus$ denotes the bit-wise XOR operation. x^n is the representation of n times x in sequence, e.g. $0^3 = 000$. $\leftarrow_\$$ denotes generating uniformly random values from a given set.

These notations will be common throughout the proposed exercises during the semester.

Q1: Semantically secure schemes

Consider a (one-time) semantically secure encryption scheme (E, D), with message and ciphertext space $\{0,1\}^n$. We now want to propose an alternative encryption scheme (E', D'). Consider the following alternatives:

1. $E'(k, m) = \mathsf{reverse}(E(k, m))$
2. $E'(k, m) = E(0^n, m)$
3. $E'(k, m) = E(k, m) \,||\, 0$
4. $E'(k, m) = E(k, m) \oplus 1^n$
5. $E'(k, m) = E(k, 0^n)$
6. $E'(k, m) = E(k, m) \,||\, m$
7. $E'((k, k'), (m, m')) = E(k, m) \,||\, E(k', m')$

Question: Which of the encryption schemes E' are also (one-time) semantically secure?

Q2: Shifting the alphabet

Consider the following encryption scheme (E, D), with message and ciphertext space the english alphabet, considering words of size n. The scheme is as follows:

- Generate a key k with n uniform values $[0 \ldots 25]$
- $E(k, m)$ shifts the letters of m according to k, producing c
- $D(k, m)$ takes c and applies the reverse shift according to k

A simple example of how this encryption scheme works:

- $k = \{3, 7, 1, 20, 15, 2\}$
- $m =$ banana
- $c =$ ehoucc

Question: Is the proposed scheme E *perfectly secure*?

Q3: Secret Sharing

Secret sharing is a method for distributing a *secret* by breaking it into *shares*, which are distributed over multiple participants. This is done in such a way that no individual holds enough information about the secret to recover it, but such that when a threshold of participants in the group combine their information, the secret can be retrieved. There are somewhat complex ways to do secret sharing, by representing the secret as points in a polynomial, and using polynomial interpolation to reconstruct it, also known as Shamir Secret Sharing. These are important building blocks for an area of advanced cryptography, also known as **secure computation**.

We will now consider a much simpler way to do it, which is simply to use something that cryptographers love: the XOR ($\oplus$). To exemplify how this can be done, lets do it such that message m is broken into shares m_1, m_2, m_3, and can only be recovered if all shares are gathered.

- $m_1 \leftarrow_\$ \{0,1\}^n$
- $m_2 \leftarrow_\$ \{0,1\}^n$
- $m_3 \leftarrow m \oplus m_1 \oplus m_2$

Observe that, without knowledge of all secrets, all possible values of m are equally probable. However, when all secrets are combined, we can compute $m = m_1 \oplus m_2 \oplus m_3$ and recover the message.

This will be used to distribute a message $m \in \{0,1\}^n$, divided into six secrets, and distributed over three participants $P_1, \ldots, P_3$, such that *no two participants can recover the message*, but *all three participants should be able to recover the message.*

- P_1: (m_1, m_2); P_2: (m_3, m_4); P_3 : ?

The following are alternatives to shares, to be given to P_3

1. (m_5, m_6)
2. (m_3, m_4, m_5, m_6)
3. (m_2, m_3, m_5, m_6)
4. (m_1, m_4, m_5)

Question - P1: Explain which of the proposed alternatives meets the aforementioned criteria.

Question - P2: Propose an alternative distribution of these six secrets over the same three participants, in a way that now allows for any two participants to recover the message.