National Technical University of Athens
School of Electrical and Computer Engineering

Computational Cryptography
2nd Set of Exercises

Deadline for submission: December 18, 2024

Exercise 1. Let $n = pq$ be a Blum integer, and let $y \in QR(n)$. Prove that the principal square root of $y \pmod n$ (i.e., the square root of y that is also a quadratic residue) is given by the formula $x \equiv y^{((p-1)(q-1)+4)/8} \pmod n$.

Exercise 2. Consider the variation of $\mathsf{DES\text{-}X}$ with 2 keys k_1, k_2, where the encryption of a plaintext M is performed as follows:

$$Enc_{k_1,k_2}(M) = E_{k_1}(M \oplus k_2),$$

where E is the encryption function of DES.
Does the above system provide more security than the classical DES? Assume that the adversary has the ability for Known-Plaintext Attack (KPA) (possesses enough plaintext-ciphertext pairs).

Exercise 3.

1. A DES key k is weak if the function DES_k is an involution. Find 4 weak keys for DES.

   Note: For a finite set S, a one-to-one and onto function $f : S \to S$ is an involution if $f(f(x)) = x$ for all $x \in S$.

2. A DES key k is semi-weak if it is not weak and there exists a key k' such that:

$$\mathsf{DES}_k^{-1} = \mathsf{DES}_{k'}$$

   Find 4 semi-weak keys for DES.

$$\mathsf{DES}_k^{-1} = \mathsf{DES}_{k'}$$

Exercise 4. Let the encryption of a message n blocks: $x = x_1||\ldots||x_n$ by a cipher E in CBC mode be denoted as $y = y_1||\ldots||y_n$, where y is the corresponding ciphertext.

1. Show that information can be extracted in the case of collisions (i.e., $y_i = y_j$ for $i \neq j$).
2. What is the probability of collision for a block size of 64 bits?
3. For what value of n is the attack useful?

Exercise 5. Given an oracle AES_k that can take binary strings and produce encryptions based on the AES cipher using the secret key k.

1. Describe an algorithm to determine the block size used by the oracle.
2. Describe an algorithm to determine if the oracle uses ECB mode.
3. Describe an algorithm to decrypt any message generated by AES_k in ECB mode. For this purpose, you can use AES_k to produce encryptions of messages of your choice. (Hint: Exploit the fact that you can learn the block size.) What is the complexity of your algorithm if the block size is l bits?

Exercise 6. Examine the RC4 pseudo-random number generator. Prove that the second byte (key) of the output is equal to 0 with a probability approximately equal to 2^{-7}. Begin by showing that if, after the Key Scheduling Algorithm (KSA) phase, it holds for the permutation array P that P[2] = 0 and P[1] ≠ 2, then the second byte of the output is equal to 0 with probability 1.

Exercise 7. Let $F : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$ be a pseudo-random function. Examine the following functions in terms of their pseudorandomness:

1. $F_1(k, x) = F(k, x)||0$
2. $F_2(k, x) = F(k, x) \oplus x$
3. $F_3(k, x) = F(k, x \oplus 1^n)$
4. $F_4(k, x) = F(k, x)||F(k, F(k, x))$

Exercise 8. Consider the Blum-Blum-Shub (BBS) pseudo-random bit generator with a Blum integer $n = pq$.

(a) Determine the period of the generator as a function of n and s_0. Explain why $\gcd(p-1, q-1)$ should be small.

(b) "Safe primes" are special prime numbers of the form $p = 2p' + 1$ where p' is also prime. We call a SafeSafe prime a safe prime $p = 2p' + 1$ for which p' is also a safe prime and $p'' \equiv 1 \pmod 4$, where $p'' = (p'-1)/2$. What is the **maximum** period of the generator in the case where both p and q are SafeSafe primes? Provide a proof for your claim.

Exercise 9. (Programming complement of the previous exercise)

Constructing a Blum integer $n = pq$ with "SafeSafe" primes p, q, each having 20 binary digits as defined in question (b) of exercise 8, we will simulate the BBS generator by choosing s_0 to maximize its period.

(a) Write a program that constructs the generator appropriately (i.e., finds a "smartly" chosen s_0) for specific p, q that you will select according to the conditions mentioned above.

(b) Extend the above program to simulate the generator and experimentally verify its theoretically calculated period.

Exercise 10.

1. Let $\mathsf{H} : \{0,1\}^* \to \{0,1\}^n$ be a hash function, which, when given input $m = x \oplus w$, produces output $\mathsf{H}(m) = \mathsf{H}(x) \oplus \mathsf{H}(w)$. Examine H in terms of the difficulty of finding collisions.
2. Let H be a hash function $\mathsf{H}(x) = \mathsf{H}_1(x)||\mathsf{H}_2(x)||\mathsf{H}_3(x)$ where at least one of $\mathsf{H}_1, \mathsf{H}_2, \mathsf{H}_3$ is collision-free. Is H also collision-free?

Exercise 11.

Given a hash function $\mathsf{H}_1 : \{0,1\}^{2n} \to \{0,1\}^n$. This function is used in a Merkle tree of height h with input being a binary sequence $x_0 x_1 \ldots x_{2^h}$ where each x_i is a binary sequence of size n bits. Through successive applications of H_1, the Merkle tree can be considered as a hash function H that compresses strings of size $n2^h$ into strings of size n. Show that if H_1 has difficulty in finding collisions, then H also has difficulty in finding collisions.

Exercise 12.

- Given a cryptosystem $\mathcal{CS}$ and an adversary $\mathcal{A}$ that can recover the key from a ciphertext of $\mathcal{CS}$ with non-negligible probability. Prove that $\mathcal{CS}$ does not provide CPA security.
- Given a cryptosystem $\mathcal{CS}$ that encrypts all messages using the CBC mode. However, instead of choosing a new IV each time, $\mathcal{CS}$ increments the previous IV by 1. In other words, for the i-th message: $IV_i \leftarrow IV_{i-1} + 1$. Show how an adversary can win the CPA game for $\mathcal{CS}$ with non-negligible probability.
- Show that the Output Feedback (OFB) encryption mode does not provide CCA security.

In all exercises, we use "$\oplus$" to denote XOR and "$||$" for concatenation.

Short instructions: (a) Try to solve the exercises on your own, (b) Discuss with your fellow students, (c) Search for ideas on the internet — in this order and after dedicating enough time to each stage! In any case, the answers must be *strictly individual*. You may be asked to briefly present some of your solutions.

Good luck!