CS 171: Problem Set 1

Due Date: February 1, 2024 at 8.59pm via Gradescope

1. Substitute and Shift cipher (10 points)

Consider a modification of the substitution cipher, where instead of applying only the substitution, we first apply a substitution and then apply a shift cipher on the substituted values. Give a formal description of this scheme and show how to break the substitute and shift cipher.

2. Double Vigenère cipher (10 points)

Consider the double Vigenère cipher, where we choose two keys k_1, k_2 with different periods $t_1, t_2 \in [t_{max}]$, and then encrypt a message in the following way: first encrypt the message using the Vigenère cipher with k_1 to obtain an intermediate ciphertext c', and then encrypt c' using the Vigenère cipher with k_2 to obtain the final ciphertext c. Show how to break the double Vigenère cipher.

3. Chosen Plaintext Attacks (10 points)

Assume the adversary has the ability to obtain ciphertexts for arbitrary plaintexts, i.e., it can choose a message and receive an encryption of the message without knowing the secret key. Attacks that leverage this information are known as *chosen plaintext attacks*. Using chosen plaintext attacks, show how to learn the secret key for the shift, substitution, and Vigenère ciphers. Your attacks should each use only a single plaintext. What is the smallest plaintext length that suffices for each attack?

For the Vigenère cipher, please consider two cases: (a) when the period t is known; (b) when the period t is unknown but an upper bound t_{max} on t is known. For the latter case, an asymptotic analysis of the required plaintext length suffices, i.e., an upper bound on the required plaintext length.

4. (optional) Exploring Kerckhoff's Principle

In the lectures, we covered Kerckhoff's principle: cryptographic systems should be designed such that the adversary knows everything except the secret key. We will see how this principle manifests in the real world through a small exercise.

`privacytools.io` maintains a crowdsourced list of tools that adhere to high security and privacy standards. In particular, the following link[1] lists file encryption software. Select at least one of these tools and:

(a) Try out the tool. Observe that encrypted data is unintelligible and that the key payload is random bits.

(b) Confirm that the codebase is open source, i.e., all critical code is publicly auditable.

[1] `https://www.privacytools.io/secure-file-encryption`

(c) Assert that standardized cryptography are used, i.e., the encryption algorithm itself is publicly known. List the cryptographic protocols used and how they are standardized (e.g., AES standardized by NIST).

(d) Observe that a lost encryption key means lost data: there is no way to recover encrypted data without the encryption key. This is in contrast to password recovery, which is effectively key recovery because the underlying keys are derived from the password.