Cryptography

Jack

Last Updated: January 31, 2025

[WIP] I will be compiling my notes on cryptography here.

Contents

§1 Introduction

Various notes from chapter 1 of [KL08].

Definition 1.1 (Kerchoffs' Principle)

Kerchoffs' Principle states that the security of a cryptographic system must not depend on the secrecy of the cipher.

In other words, a cipher should be secure even if the algorithm is public.

A few simple attack scenarios:

- **Ciphertext-only attack**: The adversary observes ciphertexts and must try to determine the plaintext.
- **Known-plaintext attack**: The adversary can learn one or more plaintexts encrypted under the same key. They must try to determine the plaintext of a different ciphertext they have not seen before.
- **Chosen-plaintext attac**k: The adversary can learn encryptions of plaintexts of its choice. They must try to determine the plaintext of a different ciphertext like before.
- **Chosen-ciphertext attac**k: The adversary can *additionally* learn the plaintexts of ciphertexts of its choice. Their goal is the same as above.

Note that the first two are **passive** while the last two are **active**.

Exercise 1.2. The first two scenarios are quite realistic. Can you think of some examples?

Exercise 1.3. Think of, or research, real world scenarios of the latter two attack scenarios.

Historical cryptographic ciphers are weak by modern standards, but they give us a few important lessons:

- **Sufficient key space principle**: it is a necessary condition to have a large key space (the domain from which keys are chosen). Otherwise, we can brute force all the keys.
- **Designing secure ciphers is hard!**: there were many ciphers such as the Vigenere cipher that were insecure (for example, to cryptanalysis). It is the goal of modern cryptography to rigorously define and prove security.

Modern cryptography comes with a few principles:

- **Clear, rigorous definitions**.

- **Clearly stated assumptions; the more minimal assumptions the better**.

- **Rigorous proofs of security with respect to principles 1 and 2**.

Example 1.4

To take an example, we give an idea of just how hard it is to rigorously define a *secure encryption scheme*.

- **Try 1: Secure if no adversary can find the key**

  But what if the adversary simply learns the plaintext?

- **Try 2: Secure if no adversary can learn the plaintext**

  But what if the adversary learns 50% of the plaintext? Or the length of the plaintext? Is this definition clear enough? What "percentage" is okay to learn?

- **Try 3: Secure if the adversary cannot determine *any* character of the plaintext**

  What if our plaintext is an integer like our salary and the adversary learns the *range* of our salary? Surely this isn't what we wanted with an "encryption" scheme.

- **Try 4: Secure if the adversary cannot derive any *meaningful* information from the ciphertext**

  Close, but no cigar. What exactly does meaningful *mean*? Our encryption scheme could be used in multiple different contexts and "meaningful" could have different meanings in each. For a *definition*, this attempt is not enough.

- **Try 5: Secure if the adversary cannot compute *any* function of the plaintext from the ciphertext**

  This is a rigorous definition: we have replaced "meaningful" with a more meaningful term.

Remark 1.5. Albeit, we often allow a carve out for the length of a message to be learned. Can you think of a way to avoid this?

Remark 1.6. For most of these notes, we will consider adversaries that are **efficient**: that is, those that run in polynomial time.

§2 Perfectly-Secret Encryption

Various notes from chapter 2 of [KL08].

Definition 2.1 (Perfectly Secret Encryption Scheme)

Even an adversary with *unbounded* computational power cannot break a **perfectly secret encryption scheme**.

First, we must formally define an encryption scheme.

Definition 2.2 (Encryption Scheme)

Consists of three algorithms:

- Gen, which outputs a key k according to a distribution. The key space is denoted by $\mathcal{K}$ and is finite.
- $\mathsf{Enc}(k, m)$, which encrypts m under k. The space of possible ciphertexts is denoted by $\mathcal{C}$.
- $\mathsf{Dec}(k, m)$, which decrypts m under k.

And also a message space $\mathcal{M}$ where $|\mathcal{M}| > 1$.

Definition 2.3 (Perfectly Correct)

For all $k \in \mathcal{K}$, $m \in \mathcal{M}$, if $c \leftarrow \mathsf{Enc}_k(m)$ then $\mathsf{Dec}_k(c) = m$ with probability 1.

Unless stated otherwise, we will be working with perfectly correct encryption schemes.

References

[KL08] Jonathan Katz and Yehuda Lindell. *Introduction to Modern Cryptography, Second Edition*. Chapman & Hall/CRC, 1st edition, 2008.