Lecture 1

Gidon Rosalki

2025-10-22

Notice: If you find any mistakes, please open an issue at `https://github.com/robomarvin1501/notes_intro_to_crypto`

1 Course overview

This is the second year of this format, before this it was the same name, but different format.

1.1 What is cryptography?

Cryptography is an ancient art, that for many years focused mainly on secret communication. For as long as humans have been communicating, we have wanted to be able to communicate in ways that hides the contents from people who are not meant to know it. The main consumers were military, and intelligence. It generally relied on creativity, and personal skill. From 500BC until the 20th century, there was a complete cycle of design → break → repair → break → repair. We will focus on how one **breaks** this cycle. We will focus on modern cryptography, which underwent a radical change in the 20th century, where it became a science, and covers much more than secret communication. It is now consumed by everyone, and relies on rigorous models, definitions, and proofs.

So, to answer our question: The scientific study of techniques for designing systems that withstand adversarial behaviour.

1.2 Course objectives

We want to introduce the basic paradigms, principles of cryptography, and explore a variety of cryptographic tools and systems. We will learn how to reason about their security, and how to use them correctly. By the end of this course we will be educated crypto consumers, and know why it is dangerous to assume you are a "crypto expert" (spoiler, you're really really not. Do not **ever** roll your own crypto), and be able to learn more about cryptography on our own.

Tentative structure:

1. Weeks 1 - 5: Private key cryptography
2. Weeks 6 - 10: Public key cryptography
3. Weeks 11 - 13: Zero knowledge proofs and secret computation

We are recommended to read

- J. Katz, and Y. Lindell's *Introduction to Modern Cryptography*
- O. Goldreich *Foundations of Cryptography - Volume 1: Basic tools*
- O. Goldreich *Foundations of Cryptography - Volume 2: Basic applications*
- Coursera's *Cryptography* course by Professor Jonathan Katz

There will be somewhere between 3 and 5 homeworks, depending on how bothered the lecturer can be, and our final grade will be made up of 10% the $n - 1$ best homeworks, and 90% final exam.

2 Symmetric key encryption

Let there be Alice, and Bob, located in different places, that want to communicate secretly. Eve will observe their communications. Our assumption is that Alice, and Bob, share a secret key, that is not known to Eve. This key is used for both encryption, and decryption. This key is some collection of bits, which may be used as described above. Let us formalise these concepts: An encryption system includes three algorithms: *KeyGen*, *Enc*, *Dec*. Let there be the key space $\mathcal{K}$, plaintext / message space $\mathcal{M}$, and ciphertext space $\mathcal{C}$.

- The key generation algorithm KeyGen outputs a key $k \in \mathcal{K}$
- The encryption algorithm Enc takes a key $k \in \mathcal{K}$, and a plaintext $m \in \mathcal{M}$, and outputs a ciphertext $c \in \mathcal{C}$

- Decryption algorithm Dec takes a key $k \in \mathcal{K}$, and a ciphertext $c \in \mathcal{C}$, and outputs a plaintext $m \in \mathcal{M}$

$$\begin{aligned} k &\leftarrow KeyGen() \\ c &\leftarrow Enc_k\left(m\right) \\ m &= Dec_k\left(c\right) \end{aligned}$$

In this course $\leftarrow$ indicates randomised generation, and $=$ indicates deterministic generation.

2.1 Correctness

An encryption system is defined as correct if

$$\forall k \in \mathcal{K}, m \in \mathcal{M}\ Dec_k\left(Enc_k\left(m\right)\right) = m$$

Kerckhoff's principle: All of KeyGen, Enc, and Dec are publicly known, and the only secret is the key k. A crypto system for whom the only security is the secrecy of the algorithms is not secure.

2.2 Caesar Cipher

Let there be:

- KeyGen uniformly samples $k \leftarrow \{0, \ldots, 25\}$
- $M = \{a, \ldots, z\}^l$ and $C = \{A, \ldots, Z\}^l$
- Enc shifts each letter k positions forward (wrapping around from z to a)
- Dec performs the same wrapping shift, but backwards

This is **not** a secure cipher (shocking, I know). Why? There are only 26 possible keys. An important part of good ciphers is that $|K|$ must **not** allow an exhaustive search.

2.3 Substitution cipher

Let there be:

- KeyGen uniformly samples a permutation k over $\{a, \ldots, z\}$
- $M = \{a, \ldots, z\}^l$ and $C = \{A, \ldots, Z\}^l$
- Enc applies the permutation k to each letter
- Dec applies the inverse permutation k^{-1}

This is not secure either (shocking, I know). Despite there being many more keys (26!), this is particularly susceptible to frequency analysis, where we use statistical patterns of the frequencies of different letters in the source language.

2.4 Vigenere cipher

Let there be:

- KeyGen uniformly samples $k = k_0 \ldots k_{t-1} \leftarrow \{0, \ldots, 25\}^t$
- $M = \{a, \ldots, z\}^l$ and $C = \{A, \ldots, Z\}^l$
- Enc shifts the ith letter $k_{i \bmod t}$ positions forward
- Dec applies the inverse shift

Not secure, it is trickier, but since the key is repeated, we can figure out the length of the key, and establish the different parts of the key through frequency analysis once more.

3 Historical ciphers

There is a fascinating history of interesting, creative (and now broken) ideas. It was particularly influenced by world history (e.g. the cryptanalysis of the German enigma in World War 2). Creating a crypto system is very very hard. Trying to do so will probably result in one that is easily broken.

4 Basic principles of modern cryptography

Analysing the security pf a cryptographic system involves:

1. Formalising a precise definition of security (security = computational ability × type of attack × notion of "break")
2. Stating the underlying assumptions: Others will attempt to validate (or invalidate) your assumptions
3. Proving that the definition is satisfied given the assumptions. Despite this, schemes can still be broken.

There are a few attacks on encryption schemes:

- Known ciphertext attack: Eve may observe a challenge ciphertext c^*
- Known plaintext attack: Eve learns pairs $(m, Enc_k(m))$, and then observes a challenge ciphertext c^*
- Chosen plaintext attack (CPA): Eve learns airs $(m, Enc_k(m))$ for messages m of her choice, then observes a challenge ciphertext c^*
- Chosen ciphertext attack: (CCA) Eve learns pairs $(m, Enc_k(m))$, for messages m of her choice, and pairs $(c, Dec_k(c))$ for ciphertexts c of her choice, and then observes a challenge ciphertext $c^* \neq c$

So, what does it mean to break an encryption scheme? Does it mean recovering the key? Recovering the plaintext? Recovering part of the plaintext? Not really any of these. Breaking an encryption scheme means learning anything "meaningful" about the plaintext? So, how do we define "meaningful"? We'll come back to that.
We shall characterise an adversary's computational abilities as follows:

- Typically (not always) run in probabilistic polynomial time (PPT)
- Sometimes, we will say computationally unbounded

5 Perfect secrecy

Let (K:eyGen, Eng, Dec) be a symmetric key encryption scheme. Alice and Bob share a key $k \leftarrow KeyGen()$. Eve knows an a-priori distribution M. Informally, perfect secrecy is that the ciphertext c does not reveal *any* information about the plaintext m.

Definition 5.1 (Perfect secrecy)**.** *A symmetric key encryption scheme $\Pi = (KeyGen, Enc, Dec)$ is* ***perfectly secret*** *if for every distribution over M, and for every $m \in \mathcal{M}$, and for every $c \in \mathcal{C}$ it holds that*

$$Pr\left[M = m | C = c\right] = Pr\left[M = m\right]$$

That is, the probability that some plaintext is the plaintext given the ciphertext, is the same as the probability that some plaintext is the plaintext, with no priors.

Consider a die. If I throw a die, the probability of guessing its result is $\frac{1}{6}$. The encryption system is perfect, if given an encrypted form of what number was thrown, the probability of knowing what number was thrown is still $\frac{1}{6}$.

Lemma 1. *A symmetric key encryption scheme Π is perfectly secret* ***if and only if*** *for every distribution over M, for every $m \in \mathcal{M}$, and for every $c \in \mathcal{C}$, it holds that*

$$Pr\left[C = c | M = m\right] = Pr\left[C = c\right]$$

I.e. the probability of a specific message encoding to a specific ciphertext is the same for every message in the world.

. Let there be a distribution over $M, m \in \mathcal{M}$, and $c \in \mathcal{C}$. Let us assume that

$$Pr\left[C = c | M = m\right] = Pr\left[C = c\right]$$

therefore

$$\begin{aligned} Pr\left[M = m | C = c\right] &= \frac{Pr\left[C = c | M = m\right] \cdot P\left[M = m\right]}{Pr\left[C = c\right]} \\ &= Pr\left[M = m\right] \end{aligned}$$

The other direction is the exact same thing, uses bayes theorem, but swapping M and C. □

Lemma 2. *A symmetric key encryption scheme π is perfectly secret* ***if and only if*** *for every distribution over $\Updownarrow$, for every $m_0, m_1 \in \Updownarrow$ and for every $c \in \rfloor$ it holds that*

$$pr\left[c = c | m = m_0\right] = pr\left[c = c | m = m_1\right]$$

. □

Theorem 1. *The shift and substitution ciphers are* ***not perfectly secret*** *for plaintexts of length $l > 1$.*

. Shift cipher:

$$Pr\left[C = "AB" | M = "ab"\right] = \frac{1}{26} \neq 0 = Pr\left[C = "AB" | M = "aa"\right]$$

□

6 One time pad

Created by Turing.

- $\mathcal{K} = \mathcal{M} = \mathcal{C} = \{0,1\}^l$
- KeyGen uniformly samples $k \leftarrow \{0,1\}^l$
- $Enc_k(m) = m \oplus k$
- $Dec_k(c) = c \oplus k$

This is correct since:

$$\forall k \in \mathcal{K},\ m \in \mathcal{M}\ Dec_k\left(Enc_k\left[m\right]\right) = Dec_k\left[m \oplus k\right] = m \oplus k \oplus k = m$$

Theorem 2 (Perfect secrecy)**.** *The one time pad is perfectly secret for plaintexts of any length l*

. Let us fix $m_0, m_1 \in \mathcal{M}$, and $c \in \mathcal{C}$. We will prove that

$$Pr\left[C = c | M = m_0\right] = Pr\left[C = c | M = m_1\right]$$

For each $b \in \{0,1\}$ it holds that

$$\begin{aligned} Pr\left[C = c | M = m_b\right] &= Pr\left[M \oplus K = c | M = m_b\right] \\ &= Pr\left[m_b \oplus K = c\right] \\ &= Pr\left[K = c \oplus m_b\right] \\ &= \frac{1}{2^l} \end{aligned}$$

This is true for every m_0, m_1, and so is generalised, as is required. □

6.1 Limitations of the one time pad

Keys have to be as long as the plaintexts, and so are very long. Additionally, there is "Two time" insecurity. Given $c = Enc_k(m)$ and $c' = Enc_k(m')$, we can learn $c \oplus c' = m \oplus m'$. There is an additional insecurity against known plaintext attacks. From m and $c = Enc_k(m)$ we can recover $k = m \oplus c$.

Theorem 3. *Let Π be a symmetric encryption scheme, with key space $\mathcal{K}$, and message space $\mathcal{M}$. If Π is perfectly secret, then $|\mathcal{K}| \geq |\mathcal{M}|$*

. Let us assume that $|\mathcal{K}| < |\mathcal{M}|$, and then we will show that the scheme is not perfectly secret. Let M be the uniform distribution over $\mathcal{M}$, and fix some $m \in \mathcal{M}$. Let us also fix some $c \in \mathcal{C}$, which is a possible encryption of m. Let

$$\mathcal{M}(c) \overset{def}{=} \left\{\hat{m} : \hat{m} = Dec_{\hat{k}}(c) \text{ for some } \hat{k} \in \mathcal{K}\right\}$$

Then $|\mathcal{M}(c)| \leq |\mathcal{K}|$. Thus, the assumption that $|\mathcal{K}| < |\mathcal{M}|$ implies that $|\mathcal{M}(c)| < |\mathcal{M}|$. In particular, there exists some $m^* \in \mathcal{M} : m^* \notin \mathcal{M}(c)$. This implies that

$$Pr\left[M = m^* : C = c\right] = 0 \neq \frac{1}{|\mathcal{M}|} = Pr\left[M = m^*\right]$$

and so the scheme is not perfectly secret. □

6.2 Characterising perfect secrecy

Theorem 4 (Shannon's theorem)**.** *Let Π be a symmetric key encryption scheme for which $|\mathcal{K}| = |\mathcal{M}| = |\mathcal{C}|$. Π is perfectly secret* ***if and only if*** *the following two conditions hold:*

1. *Every $k \in \mathcal{K}$ chosen by KeyGen is chosen with a probability of $\frac{1}{|\mathcal{K}|}$*
2. *For every $m \in \mathcal{M}$, and $c \in \mathcal{C}$, there exists exactly one $k \in \mathcal{K}$ such that $Enc_k(m)$ outputs c*

7 Tutorial

Behold: Another definition of perfect secrecy:

Exercise 1 (Perfect secrecy)**.** *For every encryption system Π, that has perfect secrecy: For every distribution M on the plaintext space $\mathcal{M}$, and for every 2 ciphertexts $c_0, c_1 \in \mathcal{C}$, it is true that*

$$Pr\left[C = c_0\right] = Pr\left[C = c_1\right]$$

Solution. This is incorrect. Let

$$KeyGen : k \leftarrow \{0,1\}^l$$

Let there be an additional bin $b = \begin{cases} 0, & \text{with probability } \frac{1}{3} \\ 1, & \text{with probability } \frac{2}{3} \end{cases}$ $Enc\left(k, m\right) : c = k \oplus m \| b$ here the double line indicates appending

$$Dec\left(c, k\right) : c' \oplus k = m \text{ where } c' = c \text{ without } b$$

So, as we can see here, it does not hold that $Pr\left[C = c_0\right] = Pr\left[C = c_1\right]$, since $\frac{2}{3}$rds of the ciphertexts will end with 1, and $\frac{1}{3}$ will end with 0. □

Exercise 2. *Given Π that has perfect secrecy, distribution M on $\mathcal{M}$. Let there be 2 messages $m_0, m_1 \in \mathcal{M}$. For every $c \in \mathcal{C}$:*

$$Pr\left[C = c | M = m_0\right] = Pr\left[C = c | M = m_1\right]$$

Solution. Correct: By using Bayes law:

$$Pr\left[C = c | M = m_0\right] = Pr\left[C = c\right]$$
$$Pr\left[C = c | M = m_1\right] = Pr\left[C = c\right]$$

INCOMPLETE As required □

Exercise 3. *Let us define $\hat{OTP}$, which is the same as OTP, but the keygen is defined as follows*

$$KeyGen : k \leftarrow \{0,1\}^l \setminus \left\{0^l\right\}$$

Is this secure?

Solution. No. Let there be $M \leftarrow \{0,1\}^l$, and so

$$Pr\left[M = m | C = m\right] = 0 \neq \frac{1}{|M|} = Pr\left[M = m\right]$$

when $M \leftarrow \{0,1\}^l$ □