Lecture 2 - Private key encryption

Gidon Rosalki

2025-10-29

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

1 Reminder

Last week we discussed symmetric encryption, and perfect secrecy:

$$\mathbb{P}[M = m | C = c] = \mathbb{P}[M = m]$$

which has the limitations of only considering security for a single message, and that the key must be as long as the message.

2 Computational security

Computational security is that all the information is present, given $Enc_k(m)$ one may completely determine k and m. It should be **computationally infeasible** to retrieve any useful information. Here we have two realistic relaxations compared to last week:

1. Security is preserved only against **computationally bounded** adversaries (e.g. 2000 years using currently technology)
2. Allow such adversaries to succeed with some *negligible* probability (small enough that it will essentially never happen)

2.1 Approaches

2.1.1 Concrete approach

Definition 2.1. *A scheme is (t, ε)-secure if every adversary, running for time at most t succeeds in breaking the scheme with probability at most ε.*

We have some sample parameters of $t = 2^{60}$, which is the order of the number of seconds since the big bang, and $\varepsilon = 2^{-60}$, which is order of occurring once every 100 billion years.
This is very useful in practice, and may be tailored to specific technology. However, in general we would like a notion of security that is essentially independent of the underlying technology.

2.1.2 Asymptotic approach

Definition 2.2. *A scheme is secure if every **probabilistic polynomial-time** (PPT) adversary succeeds in breaking the scheme with only **negligible** probability.*

Definition 2.3 (PPT)**.** *An algorithm A runs in **probabilistic polynomial-time** if there exists a polynomial $p(\cdot)$ such that, for any input $x \in \{0,1\}^*$, and a random tape $r \in \{0,1\}^*$, the computation of $A(x; r)$ terminates within $p(|x|)$ steps*

The security parameter:

- KeyGen takes as input the security parameter 1^n, and outputs $k \in \mathcal{K}_n$
- Keys produced by $KeyGen(1^n)$ should provide security against adversaries whose running time is polynomial in n (so increasing n provides better security)
- $\mathcal{K} = \bigcup_{n \in \mathbb{N}} \mathcal{K}_n,\ \mathcal{M} = \bigcup_{n \in \mathbb{N}} \mathcal{M}_n,\ \mathcal{C} = \bigcup_{n \in \mathbb{N}} \mathcal{C}_n$

Definition 2.4 (Negligible)**.** *A function $f : \mathbb{N} \to \mathbb{R}^+$ is **negligible** if for every polynomial $p(\cdot)$ there exists an N such that $\forall n > N$, it holds that $f(n) < \frac{1}{p(n)}$*

For example, $2^{-n}, 2^{-\sqrt{n}}, 2^{-\log^2(n)}$ are all negligible functions, where $\frac{1}{2}, \frac{1}{\log^2(n)}, \frac{1}{n^5}$ are non negligible.

Theorem 1. *Let $v_1(n), v_2(n)$ be negligible functions. Then, for any positive polynomial $p(n)$, the function $p(n) \cdot (v_1(n) + v_2(n))$ is negligible.*

Proof . A negligible function is $\frac{1}{\hat{p}}$, where $\hat{p}$ is larger than every polynomial. As a result, whatever we put in the numerator, will not impact our result. Therefore, the sum remains a negligible function. Multiplying by a polynomial is like writing $\frac{1}{\frac{\hat{p}}{p}}$, or subtracting in the powers: n^{l-c}, but since l is asymptotically larger than all polynomials, we still have a negligible function. □

So why these choices? "Efficient": PPT, and "negligible": smaller than any inverse polynomial. It is intuitively well-behaved under composition:

$$poly(n) \cdot poly(n) = poly(n)$$

Polynomially many invocations of a PPT algorithm is still a PPT algorithm.

$$poly(n) \cdot negligible(n) = negligible(n)$$

Polynomially many invocations of a PPT algorithm that succeeds with a negligible probability is an algorithm that succeeds with a negligible probability overall.

3 Indistinguishable encryptions

The most basic notion of security for symmetric-key encryption: Encryptions of any two messages should be indistinguishable. The adversary still observes only a single ciphertext.

$$Enc_k(m_0) \approx Enc_k(m_1)$$

This seems weaker compared to perfect secrecy. Perfectly-secure encryption reveals no information, so intuitively, what security does indistinguishable encryptions provide?

Given $\Pi = (KeyGen, Enc, dec)$, and an adversary $\mathcal{A}$, consider the experiment $IND_{\Pi,\mathcal{A}}(n)$, where one of two plaintexts m_0, m_1 is encrypted by the system, and then $\mathcal{A}$ needs to figure out which plaintext it was from the returned ciphertext c. The system is indistinguishable if $\mathcal{A}$ cannot do better than a coin flip.

Definition 3.1 (Indistinguishable encryption)**.** Π *has indistinguishable encryption if for every PPT adversary $\mathcal{A}$ there eixsts a negligible function $v(\cdot)$ such that*

$$\mathbb{P}[IND_{\Pi,\mathcal{A}}(n) = 1] \leq \frac{1}{2} + v(n)$$

where the probability is taken over the random coins used by $\mathcal{A}$, and by the experiment.

Recall the one time pad:

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

Perfectly secure since $\mathbb{P}[M = m | C = c] = \mathbb{P}[M = m]$, but requires the key k to be as long as the message m. Way too long. Can we guarantee computational security with shorter keys?

4 Pseudo-random generator

Our goal is to expand a short, random seed into a long "random looking" value:

$$G : \{0,1\}^l \to \{0,1\}^l$$

"Random looking" means "indistinguishable" from the uniform distribution.

Definition 4.1 (PRG)**.** *Let $G : \{0,1\}^l \to \{0,1\}^l$ be a polynomial-time computable function, and let $l(\cdot)$ be a polynomial such that for any input $s \in \{0,1\}^n$, we have $G(s) \in \{0,1\}^{l(n)}$. Then, G is a* ***pseudorandom generator*** *if the following two conditions hold:*

- *Expansion: $l(n) > n$*
- *Pseudorandomness: For every PPT "distinguisher" $\mathcal{D}$, there exists a negligible function $v(\cdot)$ such that*

$$\left| \mathbb{P}_{s \leftarrow \{0,1\}^n}[\mathcal{D}(G(s)) = 1] - \mathbb{P}_{r \leftarrow \{0,1\}^{l(n)}}[\mathcal{D}(r) = 1] \right| \leq v(n)$$

The notation $x \leftarrow \{0,1\}^m$ denotes that x is sampled from the **uniform distribution** over $\{0,1\}^m$ (so each value is obtained with the probability $\frac{1}{2^m}$)

4.1 Do PRGs even exist?

If so, then how difficult is it to construct a PRG? Recall two properties:

- Expansion: $|G(s)| > |s|$
- Pseudorandomness: For every PPT $\mathcal{D}$, there exists a negligible $v(\cdot)$ such that:

$$\left|\mathbb{P}_{s\leftarrow\{0,1\}^n}\left[\mathcal{D}(G(s))=1\right]-\mathbb{P}_{r\leftarrow\{0,1\}^{l(n)}}\left[\mathcal{D}(r)=1\right]\right|\leq v(n)$$

Let us try. Consider the following candidates that expand a seed $s = s_1 \dots s_n \in \{0,1\}^n$ by a single bit. Let us define

$$G(s) = s0$$

Is it distinguishable from a truly random string? Yes. A truly random string may finish in 1, whereas this may not.

How about

$$G(s) = s_1 \dots s_n s_1$$

It is distinguishable, since we can just check if we begin with the same bit as with which we started.

Finally:

$$G(s) = s_1 \dots s_n z \ : \ z = s_1 \oplus \dots \oplus s_n$$

This is also distinguishable, since we can just check if the final bit is the xor of the bits before it.

The existence of any PRG implies $P \neq NP$. Constructions are known based on various computational assumptions. We have not created a PRG that may be proven to be such, since that would then prove that $P \neq NP$.

Theorem 2. *Let there be $G : \{0,1\}^n \to \{0,1\}^{2n}$. There exists $\mathcal{D}$ (not computationally efficient) such that*

$$\mathbb{P}_{s\leftarrow\{0,1\}^n}\left[\mathcal{D}(G(s)=1)\right]-\mathbb{P}_{r\leftarrow\{0,1\}^{2n}}\left[\mathcal{D}(r)=1\right]>\frac{1}{2}$$

Proof . Where $|z| = 2n$,

$$D(z) = Im(G) = \left\{l \in \{0,1\}^{2n} \,|\exists s \mid G(s) = l\right\}$$

We are asking if it is true that $z \in Im(G)$. If so, we will return 1, and otherwise 0. This distinguisher works since

$$\mathbb{P}\left[D(G(s))=1\right]=1$$
$$\mathbb{P}\left[D(r)=1\right]=\frac{|Im(G)|}{2^{2n}}\leq\frac{1}{2^n}$$

□

Useful fact: All efficiently-testable statistical properties of the uniform distribution are preserved by the output of any PRG. For example: If G is a PRG, then there exists a negligible function $v(\cdot)$ such that

$$\mathbb{P}_{s\leftarrow\{0,1\}^n}\left[\text{Fraction of 1s in } G\left(s<\frac{1}{4}\right)\right]\leq v(n)$$

5 PRG-based OTP

Let us assume that there exists PRGs. Let G be a PRG with expansion $l(n)$. $\mathcal{K}_n = \{0,1\}^n$, but $\mathcal{M}_n = \mathcal{C}_n = \{0,1\}^{l(n)}$. $KeyGen(1^n)$ samples $k \leftarrow \{0,1\}^n$. $Enc_k(m) = m \oplus G(k)$, and $Dec_k(c) = c \oplus G(k)$.

So, given k, we generate $G(k)$, where $|G(k)| > |k|$, and then $c = G(k) \oplus m$.

Theorem 3. *If G is a PRG, then the scheme has indistinguishable encryptions.*

Proof . It's not perfect, the key is smaller than the messages, and we proved that the key must be the same length as the messages. We shall prove by **reduction**:

- Given an adversary $\mathcal{A}$, for the encryption scheme, construct a distinguisher $\mathcal{D}$ for the PRG
- $\mathcal{D}$ internally emulates $\mathcal{A}$
- $\mathcal{D}$'s efficiency, and advantage are Polynomially related to $\mathcal{A}$'s

So in short, if G is a PRG, then Π has indistinguishable encryptions. We will prove by contradiction, by assuming that Π is not IE, and therefore G is not a PRG (but we know this to be false, and thus have a contradiction).

Behold, the actual proof: Let us assume that there exists a PPT adversary $\mathcal{A}$ and a polynomial $pp(n)$ such that

$$\mathbb{P}\left[IND_{\Pi,\mathcal{A}}(n)=1\right] \geq \frac{1}{2}+\frac{1}{p(n)}$$

for infinitely many ns.

We will show that there exists a PPT distinguisher $\mathcal{D}$ and a polynomial $q(n)$, such that

$$\left|\mathbb{P}_{s\leftarrow\{0,1\}^n}\left[\mathcal{D}(G(s))=1\right]-\mathbb{P}_{r\leftarrow\{0,1\}^{l(n)}}\left[\mathcal{D}(r)=1\right]\right| \geq \frac{1}{q(n)}$$

for infinitely many ns. Or in short, if there exists the adversary, then we can use it to construct the distinguisher. The distinguisher $\mathcal{D}$, on input z invokes $\mathcal{A}$, and obtains (m_0, m_1). It samples $b \leftarrow \{0,1\}$, and let $b' = \mathcal{A}(z \oplus m_b)$. It then outputs 1 **if and only if** $b' = b$.
From here we get 2 cases:

- Case 1: $z \leftarrow \{0,1\}^{l(n)}$. $\mathcal{A}$'s view is independent of b, and so

$$\mathbb{P}_{z\leftarrow\{0,1\}^{l(n)}}\left[\mathcal{D}(z)=1\right]=\frac{1}{2}$$

- Case 2: $z = G(k)$, where $k \leftarrow \{0,1\}^n$. $\mathcal{A}$'s view is identical to the experiment $IND_{\Pi,\mathcal{A}}$ and so it is equivalent to trying to find if they are distinguishable:

$$\mathbb{P}_{k\leftarrow\{0,1\}^n}\left[\mathcal{D}(G(k))=1\right]=\mathbb{P}\left[IND_{\Pi,\mathcal{A}}(n)=1\right] \geq \frac{1}{2}+\frac{1}{p(n)}$$

So overall we constructed a PPT distinguisher $\mathcal{D}$ such that

$$\left|\mathbb{P}_{s\leftarrow\{0,1\}^n}\left[\mathcal{D}(G(s))=1\right]-\mathbb{P}_{r\leftarrow\{0,1\}^{l(n)}}\left[\mathcal{D}(r)=1\right]\right| \geq \frac{1}{p(n)}$$

which contradicts the theorem that G is a PRG. □

We have made significant progress, but we still have the problem that each key may only be used once.

6 Indistinguishable encryptions revisited

So far, this has enabled it to be infeasible to distinguish between $Enc_k(m_0)$ and $Enc_k(m_1)$, but can we learn information of m from $Enc_k(m)$

Theorem 4 (Toy theorem)**.** *Let Π have indistinguishable encryptions. Then, for any PPT adversary $\mathcal{B}$, there exists a negligible function $v(\cdot)$ such that*

$$\mathbb{P}\left[\mathcal{B}(1^n, Enc_k(m)) = LSB(m)\right] \leq \frac{1}{2}+v(n)$$

where $m \leftarrow \{0,1\}^{l(n)}$ is sampled uniformly

Proof by reduction. Assume a contradiction that there exists a PPT adversary $\mathcal{B}$ and a polynomial $p(n)$ such that

$$\mathbb{P}\left[\mathcal{B}(1^n, Enc_k(m)) = LSB(m)\right] \leq \frac{1}{2}+\frac{1}{p(n)}$$

for infinitely many ns. We then show that there exists a PPT adversary $\mathcal{A}$ and a polynomial $q(n)$ such that

$$\mathbb{P}\left[IND_{\Pi,\mathcal{A}}(n)=1\right] > \frac{1}{2}+\frac{1}{q(n)}$$

for infinitely many ns.

Behold the proof: For each $\sigma \in \{0,1\}$, let $I_\sigma \subset \{0,1\}^l$ be the set of messages whose LSB is σ. We will create the adversary $\mathcal{A}$, which on input 1^n will sample $m_0 \leftarrow I_0$, and $m_1 \leftarrow I_1$ uniformly and independently. On input c^*, it will output $b' = B(1^n, c^*)$. In short, $\mathcal{A}$ creates 2 messages, receives the encrypted form of one of them, and gives it to $\mathcal{B}$. If $\mathcal{B}$ correctly assumes the LSB, then we win, if not, we fail.

$$\begin{aligned}
\mathbb{P}\left[IND_{\Pi,\mathcal{A}}(n)=1\right] &= \mathbb{P}\left[\mathcal{B}(1^n, Enc_k(m_b)) = b\right] \\
&= \frac{1}{2}\mathbb{P}_{m_0\leftarrow I_0}\left[\mathcal{B}(1^n, Enc_k(m_0)) = 0\right] + \frac{1}{2}\mathbb{P}_{m_1\leftarrow I_1}\left[\mathcal{B}(1^n, Enc_k(m_1)) = 1\right] \\
&= \mathbb{P}_{m\leftarrow\{0,1\}^l}\left[\mathcal{B}(1^n, Enc_k(m)) = LSB(m)\right] \geq \frac{1}{2}+\frac{1}{p(n)}
\end{aligned}$$

□

6.1 Semantic security

Goldwasser-Micali in 1982: "Whatever" can be computed efficiently, given the ciphertext, can essentially be computed efficiently without the ciphertext.

Definition 6.1 (Semantically secure)**.** Π *is* ***semantically secure*** *if for every adversary* $\mathcal{A}$ *there exists a PPT "simulator"* $\mathcal{S}$ *such that for every efficiently sampleable plaintext distribution* $M = \{M_n\}_{n\in\mathbb{N}}$*, and all polynomial-time computable functions* f *and* h*, there exists a negligible function* $v(\cdot)$ *such that*

$$\left|\mathbb{P}\left[\mathcal{A}\left(1^n, Enc_k(m), h(m)\right) = f(m)\right] - \mathbb{P}\left[\mathcal{S}\left(1^n, h(m)\right) = f(m)\right]\right| \leq v(n)$$

where $k \leftarrow KeyGen\left(1^n\right)$ *and* $m \leftarrow M_n$

Or in other words, whatever you can learn from the encryption, can also be efficiently learnt *without* the encryption, or most simply, the ciphertext teaches us **nothing**.

Theorem 5. Π *is* ***semantically secure if and only if*** *it has* ***indistinguishable encryption***

Why do we need both notions? Well, semantic security explains "what security means", where indistinguishability of encryption is "easier with which to work". Since they are equivalent, we can use IE, to show semantic security.

6.2 One way functions

Definition 6.2. *A polynomial-time computable function* $f : \{0,1\}^* \to \{0,1\}^*$ *is* ***one way*** *if for any PPT* A

$$\mathbb{P}_{y\leftarrow f(U_n)}\left[A\left(1^n, y\right) \in f^{-1}(y)\right] \leq negligible(y)$$

In short, easy to compute, but hard to invert on a random image.

Informal theorem: One way functions are the basis for foundational cryptography. They are **complete** for private key cryptography (PRG $\Leftrightarrow$ OWF, as in, one can make PRGs from one way functions, and vice versa)

Some recommended reading: J. Katz and Y. Lindell. Introduction to Modern Cryptography. Chapter 3 (Private-Key Encryption): 3.0 – 3.3