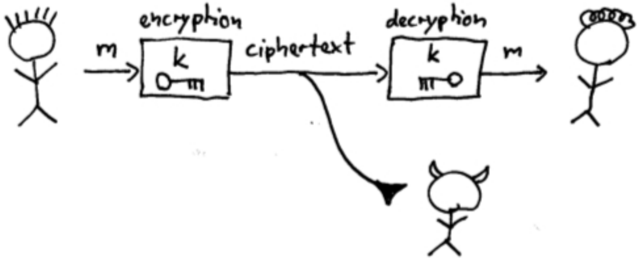
Katz, Lindell
Introduction to Modern Cryptrography
Slides Chapter 1

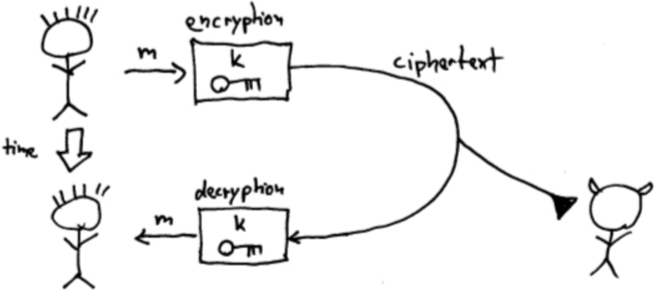
Markus Bläser, Saarland University

Private-Key Encryption



Two parties communicate securely sharing a common key.

Private-Key Encryption (2)



A user stores data securely over time.

Elements of private-key encryption

- message space $\mathcal{M}$
- key space $\mathcal{K}$
- ciphertext space $\mathcal{C}$
- key-generating algorithm Gen
- encryption algorithm Enc
- decryption algorithm Dec

Elements of private-key encryption (2)

1. Gen is a probabilistic algorithm that outputs a key $k \in \mathcal{K}$ chosen according to some distribution.
2. Enc takes a key $k \in \mathcal{K}$ and a message $m \in \mathcal{M}$ and outputs a ciphertext $\mathsf{Enc}_k(m) \in \mathcal{C}$.
3. Dec takes a key $k \in \mathcal{K}$ and a ciphertext $c \in \mathcal{C}$ and outputs a plaintext $\mathsf{Dec}_k(m) \in \mathcal{M}$.

$$\forall k \in \mathrm{im}(\mathsf{Gen})\, \forall m \in \mathcal{M} : \quad \mathsf{Dec}_k(\mathsf{Enc}_k(m)) = m$$

Kerckhoffs' principle

Kerckhoffs' principle

"The cipher method must not be required to be secret, and it must be able to fall into the hands of the enemy without inconvenience."

- easier just to keep a short key secret
- easier only to change the key in case of exposition of secret information
- scalability

Shift cipher (Caesar's chiper)

- Key k is a number between $0, \ldots, 25$.
- Every letter of a message is shifted by k.
- Caeser used a fixed key 3.

Formally:

1. $\mathcal{K} = \{0, \ldots, 25\}$
2. $\mathcal{M} = \mathcal{C} = \{0, \ldots, 25\}^*$
3. Gen outputs a key uniformly at random
4. $\mathsf{Enc}_k(m_1 \ldots m_\ell) = c_1 \ldots c_\ell$, $c_i = m_i + k \mod 26$
5. $\mathsf{Dec}_k(c_1 \ldots c_\ell) = t_1 \ldots t_\ell$, $t_i = c_i - k \mod 26$

Mono-alphabetic substitution cipher

- Keys k are permutations of $\{0, \ldots, 25\}$
- $\mathsf{Enc}_k(m_1 \ldots m_\ell) = k(m_1) \ldots k(m_\ell)$
- $\mathsf{Dec}_k(c_1 \ldots c_\ell) = k^{-1}(c_1) \ldots k^{-1}(c_\ell)$

Vigenère cipher (poly-alphabetic cipher)

"Shift cipher with different keys"

- Key $k \in \{0, \ldots, 25\}^t$ for some t
- $\mathsf{Enc}_k(m_1 \ldots m_\ell) = c_1 \ldots c_\ell$, $c_i = m_i + k_{j(i)} \mod 26$
- $\mathsf{Dec}_k(c_1 \ldots m_\ell) = t_1 \ldots t_\ell$, $t_i = c_i - k_{j(i)} \mod 26$

$$\text{index } j(i) = \begin{cases} i \mod 26 & \text{if } t \not| \, i \\ t & \text{otherwise} \end{cases}$$

Principle 1—Formal Definitions

Formal definitions give clear descriptions of

- threats
- security guarantees

and offer a way to mathematically analyse and compare cryptographic schemes

Example

What does secure encryption mean?

- It should be impossible for an attacker to recover the key.
- It should be impossible for an attacker to recover the entire plaintext.
- It should be impossible for an attacker to recover any character of the plaintext.
- A ciphertext should not leak no (additional) information about the plaintext.

  Mathematical definition of "additional information" is needed.
  $\longrightarrow$ *probability theory*

Example (2)

What is a threat?

- ciphertext-only attack
- known-plaintext attack
- chosen-plaintext attack
- chosen-ciphertext attack

Principle 2—Precise Assumptions

- Most modern cryptographic constructions cannot be proven secure unconditionally.
- This would require resolving questions from computational complexity, like “$P = NP$?”
- Therefore, security proofs rely on assumptions, like “hardness of factoring”

Mathematically precise assumptions allow:

- validation of assumption
- comparison of schemes (by comparing assumptions)
- understanding the necessity of assumptions

Principle 3—Proofs of security

- Relative to the assumptions made and
- relative to the definitions

no attacker will succeed in breaking the scheme

Problems:

- assumptions might be broken
- attacks might not have been modelled

Nevertheless: Formal approach to cryptography has revolutionized the field!