1 May 2024

Cryptography I: One-Time Pads and One-Way Functions

Shannon and Statistically Secure Encryption

An encryption scheme consists of 3 parts, each executed by a (possibly randomized) poly-time algorithm

key set

- Key generation: $\text{Gen}()$ outputs $k \in \mathcal{K}$

message set ciphertexts

- Encryption: $\text{Enc}: \mathcal{K} \times \mathcal{M} \longrightarrow \mathcal{C}$

  $\text{Enc}(k, m)$ is the encryption of message m using key k.

- Decryption $\text{Dec}: \mathcal{K} \times \mathcal{C} \longrightarrow \mathcal{M}$

  $\text{Dec}(k, c)$ is the decryption of ciphertext c using key k.

$\text{Dec}(k, \cdot)$ inverts $\text{Enc}(k, \cdot)$ for every k.

$$\text{Dec}(k, \text{Enc}(k, m)) = m \quad \forall k, m$$

Shannon security: $\forall m_0, m_1 \in \mathcal{M} \quad \forall c \in \mathcal{C}$

$$\Pr_{k \leftarrow \text{Gen}}[\text{Enc}(k, m_0) = c] = \Pr_{k \leftarrow \text{Gen}}[\text{Enc}(k, m_1) = c]$$

An attacker who has complete knowledge of Gen, Enc, Dec, and intercepts c but has no knowledge of k learns nothing about m.

A <u>one-time pad</u> is an encryption scheme with these characteristics.

1. $|\mathcal{M}| = |\mathcal{C}| = |\mathcal{K}|$ ← all 3 sets same size
2. $\forall$ key $k \in \mathcal{K}$, $\mathrm{Enc}(k,\cdot)$ and $\mathrm{Dec}(k,\cdot)$ are inverse bijections between $\mathcal{M}$ and $\mathcal{C}$.
3. $\forall$ message $m \in \mathcal{M}$, $\mathrm{Enc}(\cdot, m)$ is a bijection between $\mathcal{K}$ and $\mathcal{C}$.
4. Gen samples uniformly from $\mathcal{K}$.

<u>Ex. 1.</u> $\mathcal{M} = \{0,1\}^n = \mathcal{K} = \mathcal{C}$

bitwise XOR

$$\mathrm{Enc}(k, m) = m \oplus k = \mathrm{Dec}(k, m)$$

<u>Ex. 2.</u> $\mathcal{M} = \mathcal{K} = \mathcal{C} = \mathbb{Z}/(N)$ ← integers modulo N

$$\mathrm{Enc}(k, m) = m + k \qquad \mathrm{Dec}(k, m) = m - k$$

Problem with the one-time pad: requires a <u>huge</u> secret key.

<u>Theorem.</u> (Shannon) Any encryption scheme satisfying Shannon security must have $|\mathcal{K}| \geq |\mathcal{M}|$.

Progress on cryptography with smaller (more practical) key sizes needed new ideas: security against <u>computationally bounded</u> attackers.

(Cannot gain any useful information from intercepted messages without running exponential-time algorithms.)

One-Way Functions: A one-way function $f: \{0,1\}^* \to \{0,1\}^*$ is:

- Easy to compute: $f(x)$ computed by a (rand.) algorithm in $\mathrm{poly}(|x|)$ time.
- Hard to invert: for any attacker using rand. algorithm $\mathcal{A}$ that runs in poly time, if
  - sample $x \in \{0,1\}^n$ at random
  - send $f(x)$ to attacker
  - attacker computes $x' = \mathcal{A}(0^n, f(x))$

  don't worry about that!

  then $\Pr(f(x') = f(x))$ is negligible.

  "negligible" means $\ll \frac{1}{n^c}$ for every $c < \infty$.

Ex. 1. Multiplication.

input $x \in \{0,1\}^n$ is a pair (a,b) of binary numbers > 1.

$$a, b \in \{2, 3, \ldots, 2^{n/2} - 1\}.$$

$$f(x) = a \cdot b$$

To invert f one must solve integer factorization.

$2^{O(n^{1/3})}$ on a classical computer.