Introduction to Cryptology

Lecture 4

Announcements

- HW1 due today
- HW2 up on course webpage. Due on 2/12.
- Readings up on course webpage (by 2/10)
- In-class group exercise on 2/10.

Agenda

- Last time:
  - Definition of SKE (2.1)
  - Definition of perfect secrecy (2.1)

- This time:
  - Equivalent formulations (2.1)
  - Constructions of perfectly secret schemes (2.2)
  - Limitations of perfect secrecy (2.3)

Definition of Perfect Secrecy

- An encryption scheme (Gen, Enc, Dec) over a message space $\boldsymbol{M}$ is perfectly secret if for every probability distribution over $\boldsymbol{M}$, every message $m \in \boldsymbol{M}$, and every ciphertext $c \in \boldsymbol{C}$ for which $\Pr[C = c] > 0$:
$$\Pr[M = m \mid C = c] = \Pr[M = m].$$

An Equivalent Formulation

- Lemma: An encryption scheme (Gen, Enc, Dec) over a message space $\boldsymbol{M}$ is perfectly secret if and only if for every probability distribution over $\boldsymbol{M}$, every message $m \in \boldsymbol{M}$, and every ciphertext $c \in \boldsymbol{C}$:

$$\Pr[C = c \mid M = m] = \Pr[C = c].$$

Proof

Proof: $\rightarrow$

- To prove: If an encryption scheme is perfectly secret then

"for every probability distribution over $\boldsymbol{M}$, every message $m \in \boldsymbol{M}$, and every ciphertext $c \in \boldsymbol{C}$:

$$\Pr[C = c \,|M = m] = \Pr[C = c]\,.$$"

Proof (cont'd)

- Fix some probability distribution over $\boldsymbol{M}$, some message $m \in \boldsymbol{M}$, and some ciphertext $c \in \boldsymbol{C}$.
- By perfect secrecy we have that

$$\Pr[M = m \mid C = c] = \Pr[M = m].$$

- By Bayes' Theorem we have that:

$$\Pr[M = m \mid C = c] = \frac{\Pr[C = c \mid M = m] \cdot \Pr[M = m]}{\Pr[C = c]} = \Pr[M = m].$$

- Rearranging terms we have:

$$\Pr[C = c \mid M = m] = \Pr[C = c].$$

Perfect Indistinguishability

- Lemma: An encryption scheme (Gen, Enc, Dec) over a message space M is perfectly secret if and only if for every probability distribution over M, every $m_0, m_1 \in M$, and every ciphertext $c \in C$: $\Pr[C = c \mid M = m_0] = \Pr[C = c \mid M = m_1]$.

Proof (Preliminaries)

- Let $F, E_1, \ldots, E_n$ be events such that $\Pr[E_1 \vee \cdots \vee E_n] = 1$ and $\Pr\left[E_i \wedge E_j\right] = 0$ for all $i \neq j$.
- The E_i partition the space of all possible events so that with probability 1 exactly one of the events E_i occurs. Then

$$\Pr[F] = \sum_{i=1}^{n} \Pr[F \wedge E_i]$$

Proof Preliminaries

- We will use the above in the following way:
- For each $m_i \in M$, E_{m_i} is the event that $M = m_i$.
- F is the event that $C = c$.
- Note $\Pr\left[E_{m_1} \vee \cdots \vee E_{m_n}\right] = 1$ and $\Pr\left[E_{m_i} \wedge E_{m_j}\right] = 0$ for all $i \neq j$.
- So we have:
  - $\Pr[C = c] = \sum_{m \in M} \Pr[C = c \wedge M = m]$

$$= \sum_{m \in M} \Pr[C = c | M = m] \cdot \Pr[M = m]$$

Proof

Proof:$\rightarrow$

Assume the encryption scheme is perfectly secret. Fix messages $m_0, m_1 \in M$ and ciphertext $c \in C$.

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

Proof

Proof $\leftarrow$

- Assume that for every probability distribution over M, every $m_0, m_1 \in M$, and every ciphertext $c \in C$ for which $\Pr[C = c] > 0$:

  $$\Pr[C = c \mid M = m_0] = \Pr[C = c \mid M = m_1].$$

- Fix some distribution over M, and arbitrary $m_0 \in M$ and $c \in C$.
- Define $p = \Pr[C = c \mid M = m_0]$.
- Note that for all m:
  $\Pr[C = c \mid M = m] = \Pr[C = c \mid M = m_0] = p$.

Proof

- $\Pr[C = c] = \sum_{m \in M} \Pr[C = c \wedge M = m]$

$$= \sum_{m \in M} \Pr[C = c | M = m] \cdot \Pr[M = m]$$

$$= \sum_{m \in M} p \cdot \Pr[M = m]$$

$$= p \cdot \sum_{m \in M} \Pr[M = m]$$

$$= p$$

$$= \Pr[C = c \mid M = m_0]$$

Since m was arbitrary, we have shown that
$\Pr[C = c] = \Pr[C = c \mid M = m]$ for all $c \in C, m \in M$.

So we conclude that the scheme is perfectly secret.

The One-Time Pad (Vernam's Cipher)

- In 1917, Vernam patented a cipher now called the one-time pad that obtains perfect secrecy.
- There was no proof of this fact at the time.
- 25 years later, Shannon introduced the notion of perfect secrecy and demonstrated that the one-time pad achieves this level of security.

The One-Time Pad Scheme

1. Fix an integer $\ell > 0$. Then the message space M, key space K, and ciphertext space C are all equal to $\{0,1\}^\ell$.
2. The key-generation algorithm Gen works by choosing a string from $K = \{0,1\}^\ell$ according to the uniform distribution.
3. Encryption Enc works as follows: given a key $k \in \{0,1\}^\ell$, and a message $m \in \{0,1\}^\ell$,output $c \coloneqq k \oplus m$.
4. Decryption Dec works as follows: given a key $k \in \{0,1\}^\ell$, and a ciphertext $c \in \{0,1\}^\ell$, output $m \coloneqq k \oplus c$.

Security of OTP

Theorem: The one-time pad encryption scheme is perfectly secure.

Proof

Proof: Fix some distribution over M and fix an arbitrary $m \in M$ and $c \in C$. For one-time pad:

$$\Pr[C = c |\ M = m] = \Pr[M \oplus K = c \mid M = m]$$

$$= \Pr[m \oplus K = c] = \Pr[K = m \oplus c] = \frac{1}{2^\ell}$$

Since this holds for all distributions and all m, we have that for every probability distribution over M, every $m_0, m_1 \in M$ and every $c \in C$

$$\Pr[C = c \mid M = m_0] = \frac{1}{2^\ell} = \Pr[C = c | M = m_1]$$

Drawbacks of OTP

- Key length is the same as the message length.
  - For every bit communicated over a public channel, a bit must be shared privately.
  - We will see this is not just a problem with the OTP scheme, but an inherent problem in perfectly secret encryption schemes.
- Key can only be used once.
  - You will see in the homework that this is also an inherent problem.

Some Examples

- Is the following scheme perfectly secret?
- Message space $\boldsymbol{M} = \{0,1,\ldots,n-1\}$. Key space $\boldsymbol{K} = \{0,1,\ldots,n-1\}$.
- $\mathrm{Gen}()$ chooses a key k at random from $\boldsymbol{K}$.
- $\mathrm{Enc}_k(m)$ returns $m + k$.
- $Dec_k(c)$ returns $c - k$.

Some Examples

- Is the following scheme perfectly secret?
- Message space $\boldsymbol{M} = \{0,1, \ldots, n-1\}$. Key space $\boldsymbol{K} = \{0,1, \ldots, n-1\}$.
- $\mathrm{Gen}()$ chooses a key k at random from $\boldsymbol{K}$.
- $\mathrm{Enc}_k(m)$ returns $m + k \; mod \; n$.
- $Dec_k(c)$ returns $c - k \; mod \; n$.