Crypto Models

CS463/ECE424

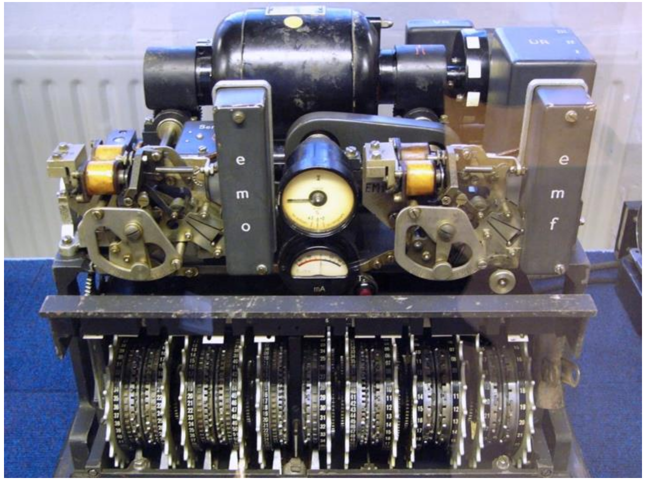
University of Illinois

Outline

Secure Communication & Kerckhoffs' Principle

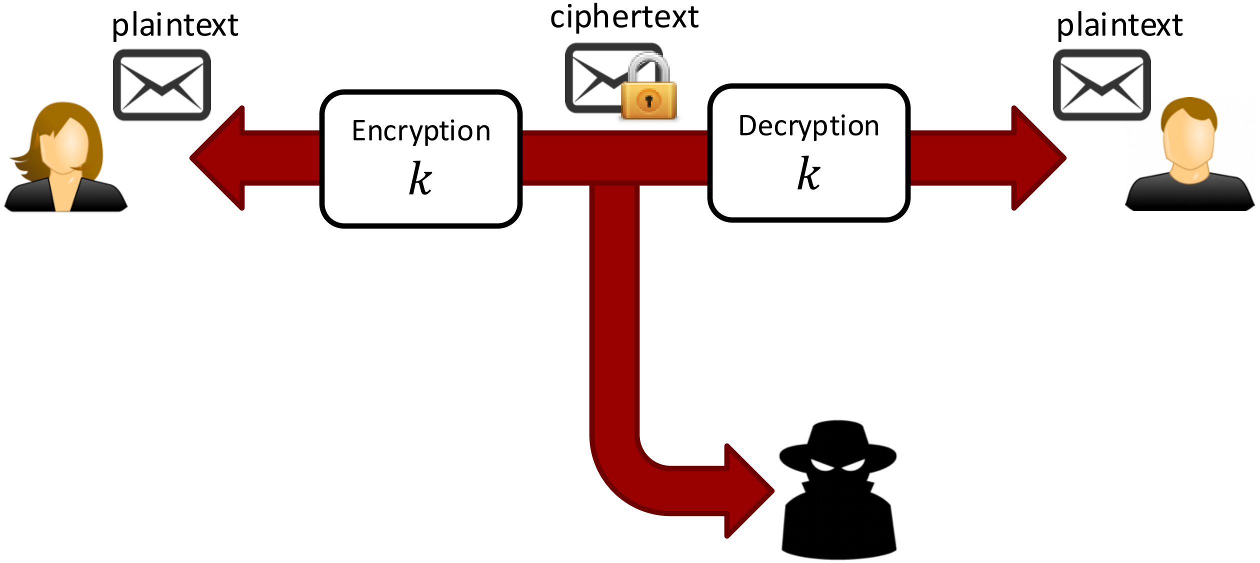
Principles of Modern Cryptography

Perfectly Secret Encryption (One-Time Pad)



Secure Communication

- Private-key (symmetric-key) setting



Components

- Key-generation algorithm: Gen
- Encryption algorithm: Enc
- Decryption algorithm: Dec
- Key: k

- What should we hide?
  - Kerckhoffs' principle: only the key

Kerckhoffs' Principle

"A cryptosystem should be secure even if everything about the system, except the key, is public knowledge."

– **Auguste Kerckhoffs**

- In other words: the security should be based only on the secrecy of the key
- In contrast with the idea of "security by obscurity"

- Q: Why is this a good idea?

Kerckhoffs' Principle

- Easier to exchange a (short) key than maintain secrecy of the algorithms
- If key is leaked, it can be changed easily, whereas changing algorithms is cumbersome.
  - Good practice to change key periodically
- Everyone uses the same algorithms, and different parties can use different keys to communicate

Open Cryptographic Designs?

- Public scrutiny increases confidence in the strength of the algorithms
- Better if “ethical hackers” to reveal flaws
- If cryptosystems are secret, they can be reverse-engineered
- Standards can be established

Recap: Attack Scenarios

- Ciphertext-only attacks
- Known-plaintext attacks
- Chosen-plaintext attacks
- Chosen-ciphertext attacks

Principles of Modern Cryptography

Historical Ciphers

- Caesar's Cipher, ROT-13, Vigenère Cipher
- These and others were all broken
  - E.g., through frequency analysis
- Historically, designing ciphers was more like an art than a science

Principles of Modern Cryptography

1. Formulation of exact definitions
2. Reliance on precise assumptions
3. Rigorous proofs of security

1. Formulation of exact definitions

- Designing cryptosystems
  - What do we want to achieve?
- Using cryptosystems
  - What encryption scheme suffices for an application?
- Studying cryptosystems
  - How to compare two different encryption schemes?

1. Formulation of exact definitions

- Why is this important?

- Example: how do we define secure encryption?

- Definition: ~~*An encryption scheme is* secure *if no adversary can find the secret key when given a ciphertext.*~~

- What about? $Enc(k, m) = m$

1. Formulation of exact definitions

- Example: how do we define secure encryption?

- ~~Definition: *An encryption scheme is secure if no adversary can find the plaintext that corresponds to the ciphertext.*~~

- What if we reveal 90% of the plaintext?

- $Enc(k, "cs463") = "cs46*"$

1. Formulation of exact definitions

- Example: how do we define secure encryption?

- Definition: ~~*An encryption scheme is secure if no adversary can determine any character of the plaintext that corresponds to the ciphertext.*~~

- Suppose we encrypt someone’s salary

- What if the scheme reveals whether that salary is more than USD 100’000?

1. Formulation of exact definitions

- Example: how do we define secure encryption?

- Definition: *An encryption scheme is secure if no adversary can derive any meaningful information about the plaintext from the ciphertext.*

- What is “meaningful”? Is learning part of the plaintext meaningful?

1. Formulation of exact definitions

- Example: how do we define secure encryption?
- Definition: *An encryption scheme is* secure *if no adversary can compute* ***any*** *function of the plaintext from the ciphertext.*
- Close to the “right” definition, but does not specify the attacker model, e.g., adversary’s computing power

2. Reliance on precise assumptions

- Modern cryptographic schemes can only be proved secure under some assumptions

- Security relies on some **hard problems**

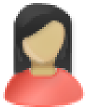
- These problems are ***assumed*** to be hard

Plain RSA

Setup:

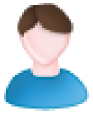
p and q large primes, $N = pq$, $\varphi = (p-1)(q-1)$,
Take e coprime with φ,
$d = e^{-1} \bmod \varphi$, $K' = (N, d)$

Alice



$K \leftarrow (N, e)$

Bob



Message m

$c \leftarrow m^e \bmod N$

Decryption
$m \leftarrow c^d \bmod N$

[RSA76]

2. Reliance on precise assumptions

- Example: RSA
- The security of RSA is based on two assumptions:
  1. Hardness of factoring: Given the modulus N, it is ***difficult*** to find primes p and q such that $N = pq$ (hard to reverse the private key)
  2. RSA assumption: Given the public key (N, e), finding the e^{th} root of an arbitrary number mod N is ***difficult*** (hard to get the plaintext)

(Here difficult means it can’t be done in polynomial time.)

2. Reliance on precise assumptions

- Validity:
  - The more an assumption is studied without being refuted, the more confident we are that it is true
  - We can provide evidence that the assumption is true by showing it is implied by some other (accepted) assumption
  - Assumption needs to be precisely stated to be studied

2. Reliance on precise assumptions

- Comparison of cryptographic schemes:
  - Two schemes A and B have same efficiency, but A depends on an assumption implied by B's assumption
  - Then A is better
  - If the assumptions are incomparable, then we give preference to better studied assumptions

2. Reliance on precise assumptions

- Facilitation of proofs of security:
  - Security proofs for most cryptographic schemes are stated as “the scheme is secure if the assumption is true”
  - This is only meaningful if the assumption is precise

3. Rigorous proofs of security

- Having exact definitions and precise assumptions make rigorous proofs possible

- Modern cryptographic schemes are accompanied with a proof of security

- Without a proof we are left with our intuition, and experience has shown this is disastrous

Perfectly Secret Encryptio

Perfectly Secret Encryption

- We want:
  - An encryption scheme that cannot be broken by an adversary even if they has unlimited computing power and unlimited time.
- Intuition:

observing the ciphertext should give no information about the plaintext, i.e., the *a posteriori* distribution (of the plaintext) is the same as the *a priori* distribution

Perfectly Secret Encryption

- Defintion 1:
  - Message space $\mathcal{M}$ — set of all messages
  - Ciphertext space $\mathcal{C}$— set of all ciphertexts
  - *An encryption scheme (Gen, Enc, Dec) is* perfectly secret *if for every probability distribution over* $\mathcal{M}$*, every message* $m \in \mathcal{M}$*, and every ciphertext* $c \in \mathcal{C}$ *for which* $\Pr[c \in \mathcal{C}] > 0$:

To simplify the presentation, we won't mention these.

$$\underbrace{\Pr[\boldsymbol{M} = m \mid \boldsymbol{C} = c]}_{\text{a posteriori}} = \underbrace{\Pr[\boldsymbol{M} = m]}_{\text{a priori}}$$

a posteriori distribution: the probability that the message was m if the ciphertext is c

a priori distribution: the probability that the message was m

Perfectly-Secret Encryption

- Definition 2 (Equivalent to Def. 1):
  - *An encryption scheme (Gen, Enc, Dec) is* perfectly secret *if for every probability distribution over* $\mathcal{M}$, *every message* $m \in \mathcal{M}$, *and ciphertext* $c \in \mathcal{C}$:

$$\Pr[\boldsymbol{C} = c \mid \boldsymbol{M} = m] = \Pr[\boldsymbol{C} = c]$$

- Definition 3 (Equivalent to Def. 1):
  - *An encryption scheme (Gen, Enc, Dec) is* perfectly secret *if for every probability distribution over* $\mathcal{M}$, *every message* $m_0, m_1 \in \mathcal{M}$, *and ciphertext* $c \in \mathcal{C}$:

$$\Pr[\boldsymbol{C} = c \mid \boldsymbol{M} = m_0] = \Pr[\boldsymbol{C} = c \mid \boldsymbol{M} = m_1]$$

Perfectly Secret Encryption

$$\Pr[\boldsymbol{M} = m \mid \boldsymbol{C} = c] = \Pr[\boldsymbol{M} = m]$$

$$\Longleftrightarrow$$

$$\Pr[\,\boldsymbol{C} = c \mid \boldsymbol{M} = m] = \Pr[\boldsymbol{C} = c]$$

$$\Longleftrightarrow$$

$$\Pr[\,\boldsymbol{C} = c \mid \boldsymbol{M} = m_0] = \Pr[\,\boldsymbol{C} = c \mid \boldsymbol{M} = m_1]$$

The distribution over ciphertext is independent of the plaintext, i.e., the ciphertext contains no information about the plaintext.

Perfectly-Secret Encryption

- Proof (Def. 1 $\Leftrightarrow$ Def. 2):

  $\Longrightarrow$

  Suppose: $\Pr[\boldsymbol{M} = m \mid \boldsymbol{C} = c] = \Pr[\boldsymbol{M} = m]$,

  Now, multiply both sides by $\frac{\Pr[\boldsymbol{C}=c]}{\Pr[\boldsymbol{M}=m]}$:

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

$$\Pr[\boldsymbol{C} = c \mid \boldsymbol{M} = m] = \Pr[\boldsymbol{C} = c]$$ (Bayes' Theorem)

- Simple exercise: $\Longleftarrow$, (Def. 1 $\Leftrightarrow$ Def. 3)

Perfectly Secret Encryption

- Adversarial indistinguishability game:
  1. Adversary $\mathcal{A}$ chooses messages $m_0, m_1 \in \mathcal{M}$.
  2. Gen outputs random key k, and a random bit $b \in \{0,1\}$ is selected. Then ciphertext $c = \mathrm{Enc}_k(m_b)$ is sent to $\mathcal{A}$.
  3. Adversary $\mathcal{A}$ (guesses) outputs bit $b' \in \{0,1\}$.
  4. The output is 1 if $b = b'$, and 0 otherwise. If the output is 1 we say adversary $\mathcal{A}$ is successful.

- Definition 4 (Equivalent to Def. 1):
  – *An encryption scheme (Gen, Enc, Dec) is* perfectly secret *if for **every** adversary* $\mathcal{A}$:

$$\Pr[\mathcal{A} \text{ is successful}] = {}^{1}\!/_{2}$$

One-Time Pad

- Message space $\mathcal{M}$, key space $\mathcal{K}$, ciphertext space $\mathcal{C}$ are $\{0,1\}^l$, for some integer $l > 0$.
- Gen: picks key uniformly at random in $k \in \mathcal{K}$.
- Enc: given key k, message $m \in \mathcal{M}$, output ciphertext $c = m \oplus k$.
- Dec: given key k, ciphertext c $\in \mathcal{C}$, output plaintext $m = c \oplus k$.

One-Time Pad

- Suppose $l = 4$, and Gen outputs $k = 1011_{\ b} = \mathrm{0xB}$
- If the plaintext is $m = \mathrm{0x5} = 0101_b$, then the ciphertext is:

$$c = m \oplus k = 0101_b \oplus 1011_{\ b} = 1110_{\ b} = \mathrm{0xE}$$

- Why is this perfectly secret?
  - Ciphertext: $c = \mathrm{0xE}$, what is the plaintext $m = \mathrm{0xE} \oplus k$?

k	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
m	E	F	C	D	A	B	8	9	6	7	4	5	2	3	0	1

Each of the possibility with probability $^1/_{16}$

One-Time Pad

- Theorem 1: The one-time pad is perfectly secret.

- Proof:
  - Pick some arbitrary distribution of the message space $\mathcal{M}$, and a particular $m \in \mathcal{M}$*, and ciphertext* $c \in \mathcal{C}$. We have:

$$\begin{aligned}\Pr[\,\boldsymbol{C} = c \mid \boldsymbol{M} = m] &= \Pr[\boldsymbol{M} \oplus \boldsymbol{K} = c | \boldsymbol{M} = m] \\ &= \Pr[m \oplus \boldsymbol{K} = c] \\ &= \Pr[\boldsymbol{K} = m \oplus c] = 2^{-l}\end{aligned}$$

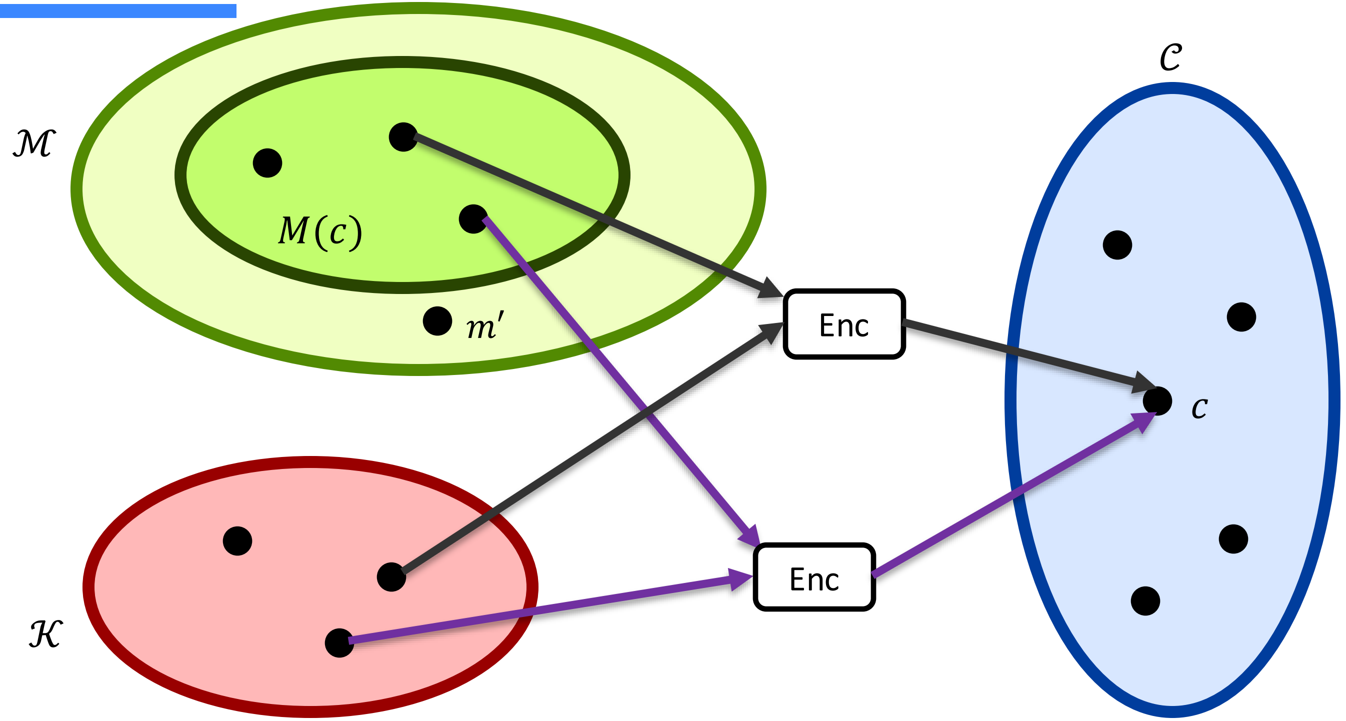
  - So: $\Pr[\,\boldsymbol{C} = c \mid \boldsymbol{M} = m_0] = \Pr[\boldsymbol{C} = c | \,\boldsymbol{M} = m_1]$ (Def. 3), since the above holds for every $m_0, m_1 \in \mathcal{M}$ and $c \in \mathcal{C}$.

One-Time Pad

- What happens if we use the same key to encrypt multiple messages?
  - $c_1 = m_1 \oplus k$, $c_2 = m_2 \oplus k$
  - then $c_1 \oplus c_2 = m_1 \oplus m_2$

- Observation: keys are as long as the messages.
  - Can we have perfect security with shorter keys?

Perfectly Secret Encryption



Observe that $|M(c)| \leq |\mathcal{K}|$, but since $|\mathcal{K}| < |\mathcal{M}|$, there exists $m' \in \mathcal{M} \setminus M(c)$

Perfectly Secret Encryption

- Theorem 2: *Let (Gen, Enc, Dec) be a perfectly secret encryption scheme for some message space $\mathcal{M}$, and with key space $\mathcal{K}$.*
  - *Then:* $|\mathcal{K}| \geq |\mathcal{M}|$
- Proof:
  - Suppose $|\mathcal{K}| < |\mathcal{M}|$, take the uniform distribution over $\mathcal{M}$, and pick any ciphertext $c \in \mathcal{C}$ with $\Pr[\boldsymbol{C} = c] > 0$.
  - Define $M(c)$ to be the set of possible plaintext $m \in \mathcal{M}$ which are valid decryptions of c.
  - Observe: $|M(c)| \leq |\mathcal{K}|$; since $|\mathcal{K}| < |\mathcal{M}|$, $\exists m' \in \mathcal{M} \setminus M(c)$
  - But, $\Pr[\boldsymbol{M} = m' \mid \boldsymbol{C} = c] = 0 \neq \Pr[\boldsymbol{M} = m']$.

Symmetric-Key Encryption

- Schemes used in practice are not perfectly secure, but only computationally secure

- Key space (e.g., 128 bits) is much smaller than plaintext space (i.e., virtually unlimited)
  - Use modes of operations to encrypt arbitrary length messages using block ciphers (which operate on fixed-length chunks)

References

- Jonathan Katz, and Yehuda Lindell. "Introduction to modern cryptography." CRC Press, 2014. Chapters 1 & 2.

Discussion Questions

1. One-Time Pad:
   - What if the key happens to be 0^l?
     - Suppose $m = \text{“hello”},$ what is the ciphertext c?
   - Is it a good idea to change Gen to only pick keys $k \neq 0^l$?
   - Why or why not? Is the scheme still perfectly secret?