FISF130020: Introduction to Computer Science

Lecture 9. Cryptograph

Hui Xu

xuh@fudan.edu.cn

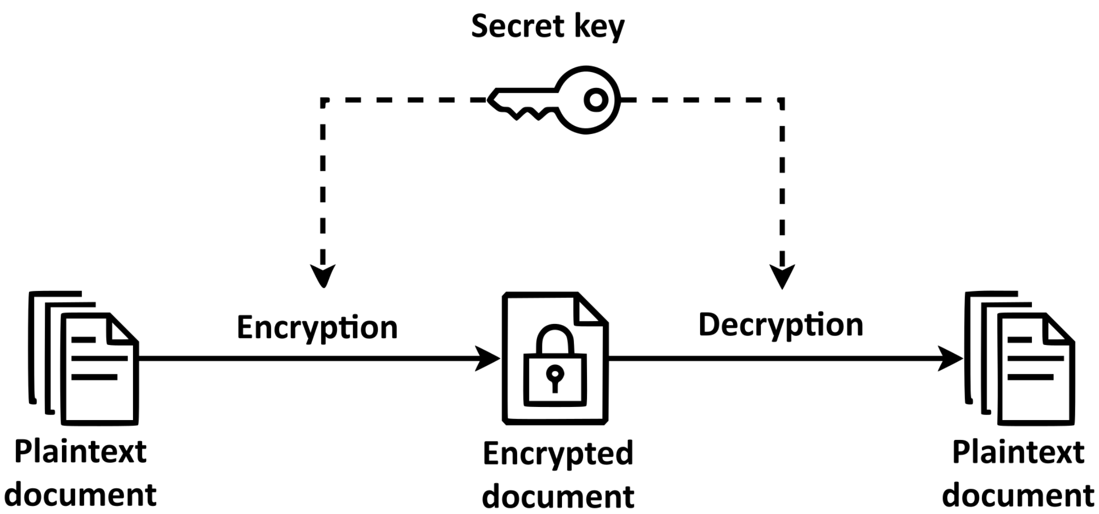
Outline

- 1. Symmetric Encryption
- 2. Asymmetric Encryption
- 3. Hash Function
- 3. In-class Practice

1. Symmetric Encryption

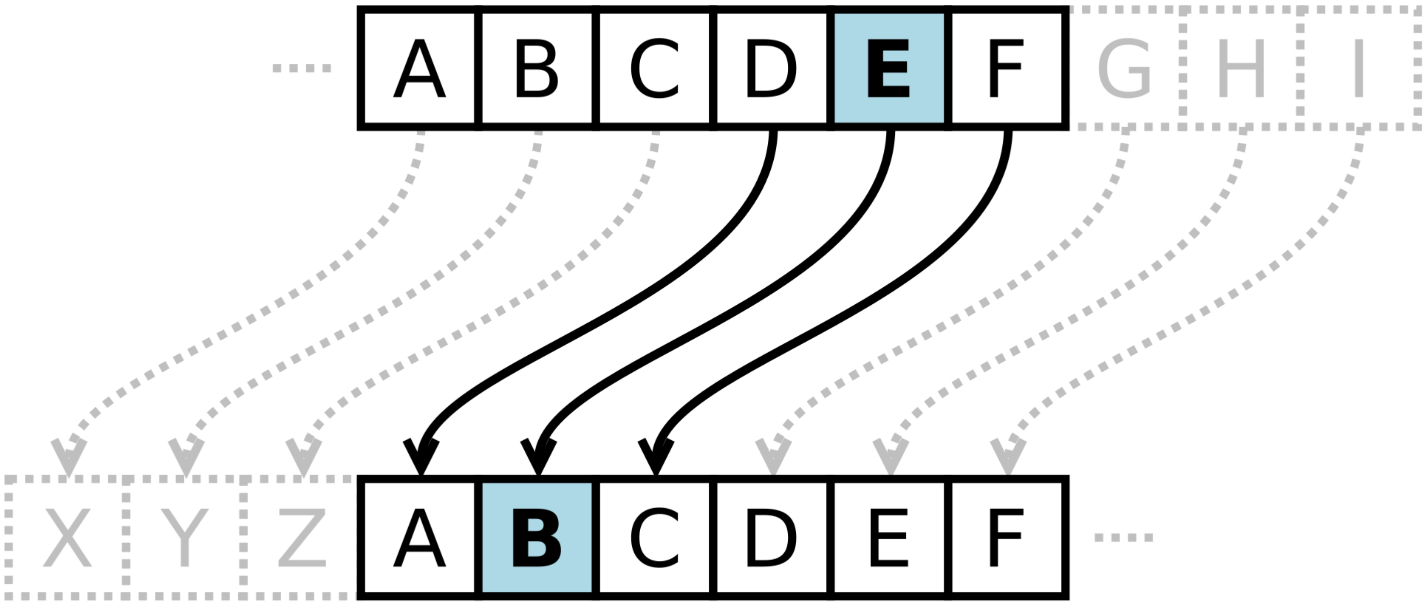
Problem: Confidentiality

- How to design a system that ensures information is not leaked to malicious users during network transfer?
- Symmetric encryption: encryption and decryption via the same key



Caesar Cipher

- Each letter in the plaintext is replaced by another letter based on a specific mapping rule.
- For example, using a left shift of 3.



Plaintext: THE QUICK BROWN FOX JUMPS OVER THE LAZY DOG
Ciphertext: QEB NRFZH YOLTK CLU GRJMP LSBO QEB IXWV ALD

Scytale

- A transposition cipher where a strip of text is wrapped around a rod of a specific width.
- The message is revealed by reading the characters in rows along the length of the rod.

Sample Cipher: Encryption

$$\text{CipherText} = \text{PlainText} \oplus \text{Key}_0 \boxplus \text{Key}_1$$

$\oplus$: exclusive or

$\boxplus$: plus

$\boxminus$: minus

PlainText: 8-bit

Key: a 16-bit key with two parts [Key_0, Key_1], each of which is 8-bit

e.g., PlainText = 'A', Key="0000111111110000",

$$\text{CipherText} = 01000001 \oplus 00001111 \boxplus 11110000$$

$$\text{CipherText} = 01001110 \boxplus 11110000$$

$$\text{CipherText} = 00111110$$

Sample Cipher: Decryption

$$\text{PlainText} = \text{CipherText} \boxminus \text{Key}_1 \oplus \text{Key}_0$$

$$\text{PlainText} = 00111110 \boxminus 11110000 \oplus 00001111$$

$$\text{PlainText} = 01001110 \oplus 00001111$$

$$\text{PlainText} = 01000001$$

Guess The Bits of The Key

- Given two pairs of plaintext and ciphertext,
  - $PlainText_1$ = 01000001, $CipherText_1$ =00111110
  - $PlainText_2$ = 01000010, $CipherText_2$ =00111101
- Guess bit value of the Key:

$$CipherText = 01000001 \oplus Key_0 \boxplus Key_1 = 00111110$$
$$CipherText = 01000010 \oplus Key_0 \boxplus Key_1 = 00111101$$

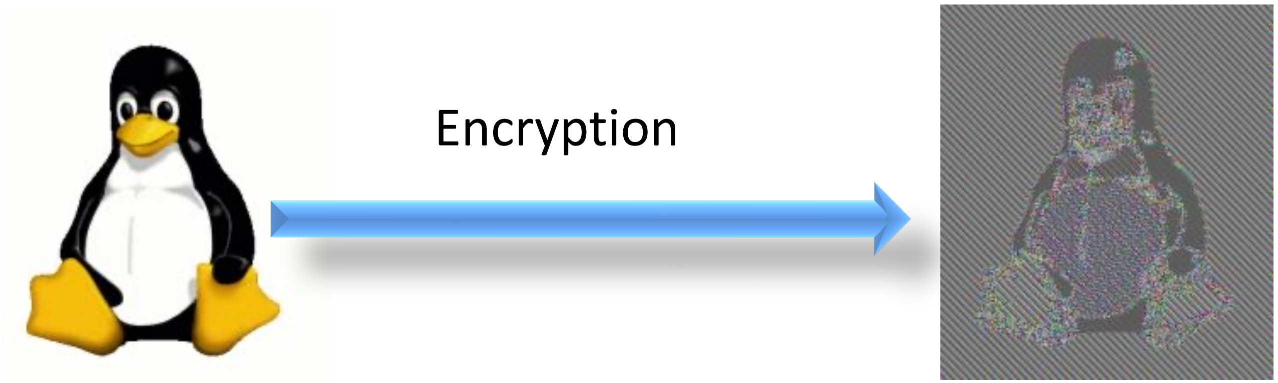
$$\Downarrow$$

$$01000001 \oplus Key_0 \boxplus Key_1 = 01000010 \oplus Key_0 \boxplus Key_1 \boxplus 00000001$$
$$01000001 \oplus Key_0 = 01000010 \oplus Key_0 \boxplus 00000001$$

$$\Downarrow$$

$$Key_0 = ******11$$

Information Leakage



Enigma Machine

- A cipher device used by Germany during World War II.
- Ultimately hacked by Allied forces and impacted the war’s outcome.

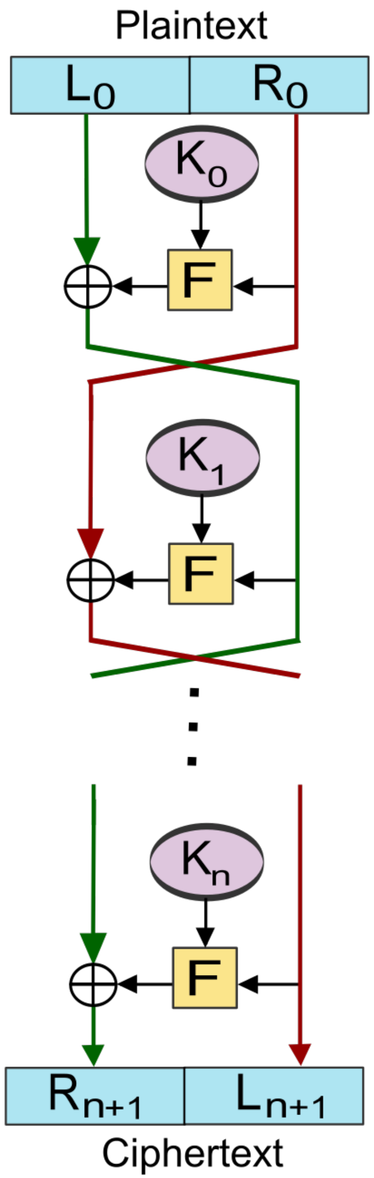
Kerckhoff Principle

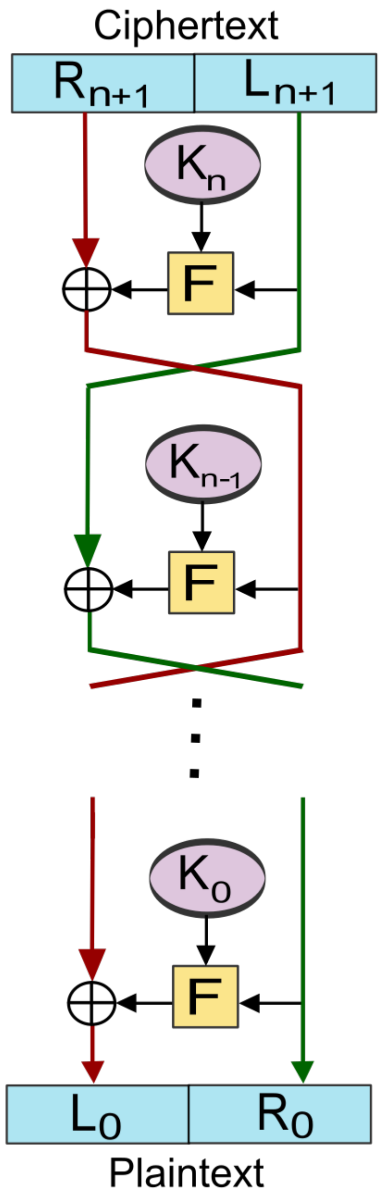
A cryptosystem should be secure even if everything about the system, except the key, is public knowledge.

Commonly-used Ciphers

- DES
  - Released in 1977, FIPS PUB 46, key space of 2^{56}.
  - In 1998, DES was cracked within three days using a device costing $250,000.
- 3DES
  - NIST SP 800-67 (1998), FIPS PUB 46-3 (1999)
  - In 1999, NIST designated 3-DES as a transitional encryption standard.
  - The 3DES algorithm can continue to be used in sensitive government information systems in the United States until 2030.
- AES
  - Released in 2001, FIPS PUB 197.

Feistel Network used by Ciphers





2. Asymmetric Encryption

Problem

- How to negotiate keys without pre-shared knowledge?
- The key should not be accessible or known by unauthorized parties.

Diffie-Hellman Key Exchange

Discrete logarithm problem: $g^x = y \bmod n$

$\Rightarrow x = ?$

Alice

Shared info

$p = 23$ (prime)

$g = 5$

Bob

Alice:

Generate private key: $prk_A = 4$

Calculate public key:

$puk_A = g^{prkA} \bmod p$

$= 5^4 \bmod 23$

$= 4$

Calculate Session key:

$sk_{AB} = puk_B^{prkA}$

$= 10^4 \bmod 23$

$= 18$

Bob:

Generate private key: $prk_B = 3$

Calculate public key:

$puk_B = g^{prkB} \bmod p$

$= 5^3 \bmod 23$

$= 10$

Calculate Session key:

$sk_{AB} = puk_A^{prkB}$

$= 4^3 \bmod 23$

$= 18$

MITM Attack

Alice

Generate private key: $prk_A = 4$

Calculate public key:
$puk_A = g^{prkA} \bmod p$
$= 5^4 \bmod 23$
$= 4$

$puk_B = 2$

Calculate Session key:
$sk_{AB} = puk_B^{pkA}$
$= 2^4 \bmod 23$
$= 16$

Carol

$prk_c = 2$

$puk_c = g^{pkc} \bmod p$
$= 5^2 \bmod 23 = 2$

Session key:
$sk_{AC} = puk_A^{prkc}$
$= 4^2 \bmod 23$
$= 16$

Session key:
$sk_{BC} = puk_B^{prkc}$
$= 10^2 \bmod 23$
$= 8$

Bob

Generate private key: $prk_B = 3$

Calculate public key:
$puk_B = g^{prkB} \bmod p$
$= 5^3 \bmod 23$
$= 10$

$puk_A = 2$

Calculate Session key:
$sk_{AB} = puk_A^{pkB}$
$= 2^3 \bmod 23$
$= 8$

Problem

- Anyone can send a message to A, and only A can decrypt it.
- A can send a message to anyone, and the recipient can prove that A is the sender of the message.

RSA Algorithm

- In 1977, Rivest, Shamir, and Adleman invented it.
- Security: Based on the integer factorization problem.
  - As long as the key length is sufficiently long (2048 bit), information encrypted with RSA is practically unbreakable.
  - On December 12, 2009, the number RSA-768 (768 bits, 232 digits) was successfully factored.

RSA: Key Generation

- Select two distinct large prime numbers: p, q
  - Compute $n = p * q$
  - Compute $\phi(n) = (p - 1)(q - 1)$
- Generate the public key:
  - Select an integer e such that $1 < e < \phi(n)$ and $\gcd(e, \phi(n)) = 1$
  - The public key is: (e, n)
- Compute $d = e^{-1} \bmod \phi(n)$;
  - Use the extended Euclidean algorithm.
  - The private key is: d

Example：
$p=13, q=7$
$n=13*7=91$
$\phi(n)=(13-1)*(7-1)=72$
$e=11$
Publick Key：(11, 91)
$d=11^{-1} \bmod 72=59$，
Private Key：(59, 91)

Calculate Private Key with Extended Euclidean Algo.

$72 = 11*6 + 6$
$11 = 6*1 + 5$
$6 = 5*1 + 1$

$1 = 6 - 5$
$1 = 6 - (11-6) = 2*6 - 11$
$1 = 2*(72-11*6) - 11 = 2*72 - 11*13$

$d = -13 \bmod 72 = 59$

RSA: Encryption/Decryption

Encryption:

CipherText=PlainText^{e} mod n

Decryption:

Plaintext=CipherText^{d} mod n

Example:

Public Key: (11, 91)

Private Key： 59

PlainText: 65

Encryption: 65^{11} mod 91 = 39

Decryption： 39^{59} mod 91=65

Proof: $m^{ed} \equiv m \bmod n$

m^{ed}

$= m^{1+k\phi(n)}$

$= m*(m^{\phi(n)})^k$

According to Euler's theorem: if m and n are coprime, $m^{\phi(n)} \bmod n = 1$

m^{ed}

$\equiv m*(1)^k \bmod n$

$\equiv m \bmod n$

RSA: Digital Signature

Digital Signature:

sign = $PlainText^d \bmod n$

Signature Verification:

PlainText = $sign^e \bmod n$

Example:

Public Key: (11, 91)

Private Key： 59

PlainText: 5

Sign: $5^{59} \bmod 91=73$

Verify: $73^{11} \bmod 91=5$

Security Issue:

Assuming $5^d \bmod 91=73$, $25^d=?$

3. Hash Function

Problem

- How to prevent data forgery or the data from being tampered?

Requirements for a HASH Function

- **Practicality:**
  - H(x) = y can be applied to data xx of any size.
  - For any input x, the result y of H(x) = y is of fixed length.
  - The computational cost of H(x) = y is low, such as linear complexity.
- **One-way function**:
  - Given the result y, it is infeasible to compute x such that H(x) = y.
- **Collision resistance:**
  - Weak collision resistance: Given any x1, it is infeasible to find x2 such that H(x1) = H(x2).
  - Strong collision resistance: It is infeasible to find any pair (x1,x2) such that H(x1) = H(x2).

Birthday Attack

- Assuming a class has n students, what is the probability that at least two students share the same birthday?
  - $1 - \frac{365!}{(365-n)! \times 365^n} = 70.63\%$
    - ✓ n = 2, p = 0.2%
    - ✓ n = 10, p =11.7%
    - ✓ n = 20, p =41.1%
    - ✓ n = 40, p =89.1%
    - ✓ n = 50, p =97%

Sample Hash Function

- Let's define the following hash function for a message M = $a_1, \ldots, a_n$

$$Hash(M) = \left(\sum_{i=1}^{t} a_i\right) \bmod n$$

- Calculate the hash value for m = 70, 117, 100, 97, 110, and n = 256
- Is the hash function safe? i.e., meet the requirements.
  - One-way function?
    - no, because we can easily find a message that produce the same hash value
  - Collision resistance? no

Commonly-used Hash Functions

- MD5:
  - Designed by Ron Rivest in 1991
  - 128-bit (16-byte) hash value, 64 rounds of computation
  - Security: Considered insecure, as collisions can be found within seconds
- SHA-1:
  - Defined in FIPS 180 (1993), based on the design of MD4
  - 160-bit hash length, 80 rounds
  - In 2005, Xiaoyun Wang demonstrated that finding two different inputs with the same hash value in SHA-1 has a complexity of 2^{69}-2^{80}.
  - Enhanced versions include SHA-2 (FIPS 180-3, 2002) and SHA-3 (2007)

3. In-class Practice

In-class Practice

- Given p = 3, q = 11, e = 7,
  1) calculate the private key
  2) use the private key to encrypt M=5
  3) decrypt the previous encrypted data with the public key
- Analyze the security of the following hash function where M = $a_1, \ldots, a_n$

$$Hash(M) = \left(\sum_{i=1}^{t} a_i^2\right) \bmod n$$