



# Network Programming

Ung Văn Giàu  
Email: [giau.ung@eiu.edu.vn](mailto:giau.ung@eiu.edu.vn)



# Protecting Data: Encryption

# Content

- Introduction
- Terminology
- Encryption scheme security
- Symmetric encryption: DES, 3DES, AES
- Asymmetric encryption: DH, RSA
- Piracy protection
- Hash function: MD, SHA

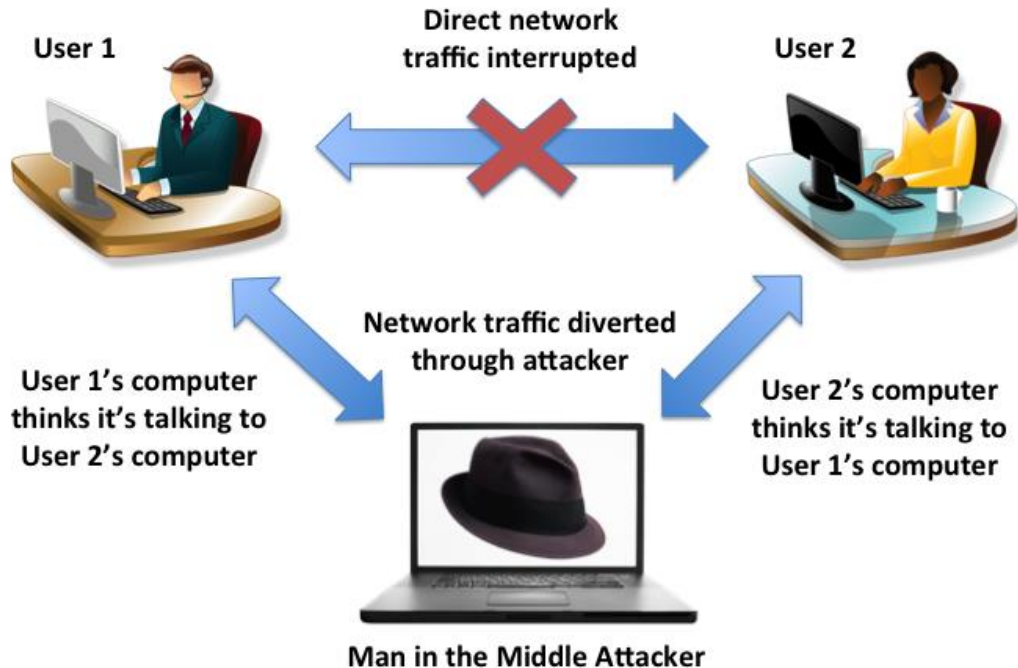
# 1. Introduction

Eavesdropping attack



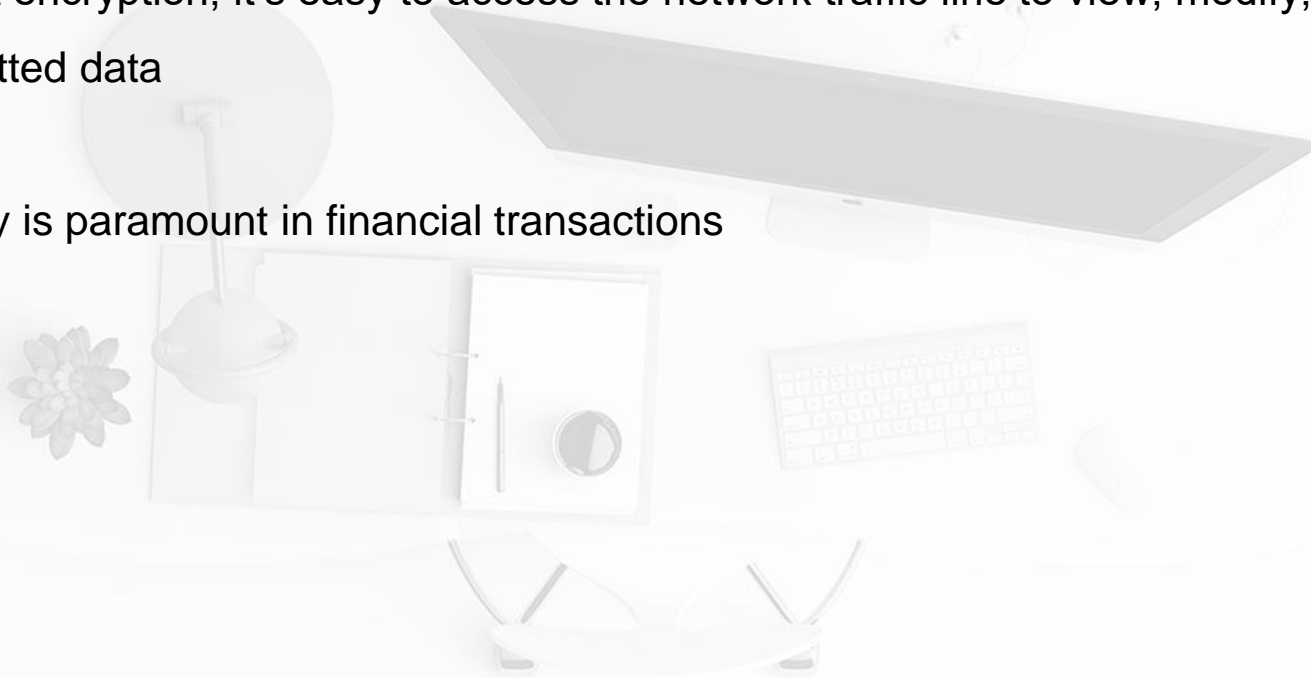
# 1. Introduction

## Man-in-the-middle attack



# 1. Introduction

- Without encryption, it's easy to access the network traffic line to view, modify,... transmitted data
- Security is paramount in financial transactions



# 1. Introduction

## Protecting Data

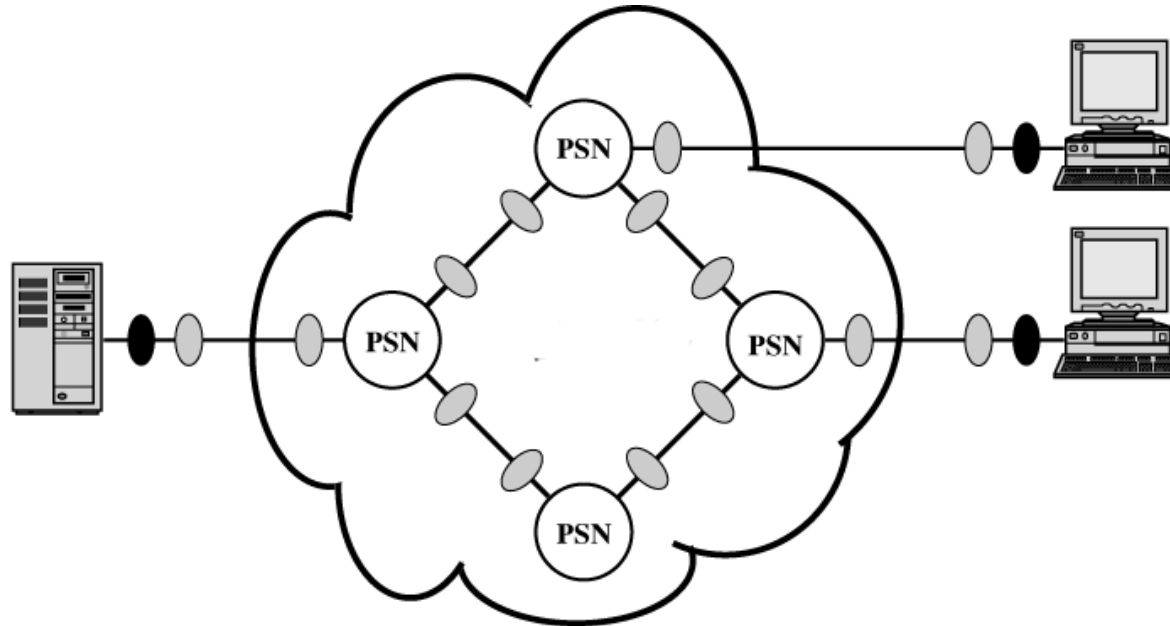
- The most effective and efficient solution against cyber security threats is **encryption**.
- To encrypt, we need to define:
  - What data needs to be encrypted?
  - Where needs to be encrypted?

Two fundamental alternatives:

- Link encryption
- End-to-end encryption

# 1. Introduction

Protecting Data



● End-to-end encryption

PSN: Packet-switching node

○ Link encryption



## 2. Terminology

- **Plain text:** data is unencrypted
- **Cipher text:** data is encrypted
- **Key:** a piece of data is used to convert plain text into cipher text or vice versa
- **Cryptographic algorithm, or cipher:** a prescribed algorithm for converting plain text into cipher text and back again, using a key
- **Strength:** the measure of the difficulty to convert cipher text to plain text without the key

# 3. Encryption scheme security

- **Unconditional security**

Cipher text doesn't contain enough information to decrypt

- **Computation security**

Meet one or both the following criteria:

- The **cost** of breaking the cipher **exceeds** the **value** of the encrypted information
- The **time** required to break the cipher **exceeds** the useful **lifetime** of the information

# 4. Cryptanalysis

- Decrypt encrypted data without knowing the secret key
- Two general approaches:
  - **Brute-force attack**  
Try every possible key
  - **Non-brute-force attack** (cryptanalytic attack)
    - Exploit the disadvantages of the algorithm
    - Based on the general characteristics of the plaintext or some sample plaintext-ciphertext pairs

# Average Brute-force attack time

Key Size (bit)	No. Of Alternative Keys	Time required at 1 decryption/ $\mu$ s	Time required at $10^6$ decryption/ $\mu$ s)
32	$2^{32} = 4,3 \times 10^9$	$2^{31} \mu\text{s} = 35,8 \text{ minutes}$	2,15 ms
56	$2^{56} = 7,2 \times 10^{16}$	$2^{55} \mu\text{s} = 1142 \text{ years}$	10,01 hours
128	$2^{128} = 3,4 \times 10^{38}$	$2^{127} \mu\text{s} = 5,4 \times 10^{24} \text{ years}$	$5,4 \times 10^{18} \text{ years}$
168	$2^{168} = 3,7 \times 10^{50}$	$2^{167} \mu\text{s} = 5,9 \times 10^{36} \text{ years}$	$5,9 \times 10^{30} \text{ years}$
26 characters (permutation)	$26! = 4 \times 10^{26}$	$2 \times 10^{26} \mu\text{s} = 6,4 \times 10^{12} \text{ years}$	$6,4 \times 10^6 \text{ years}$

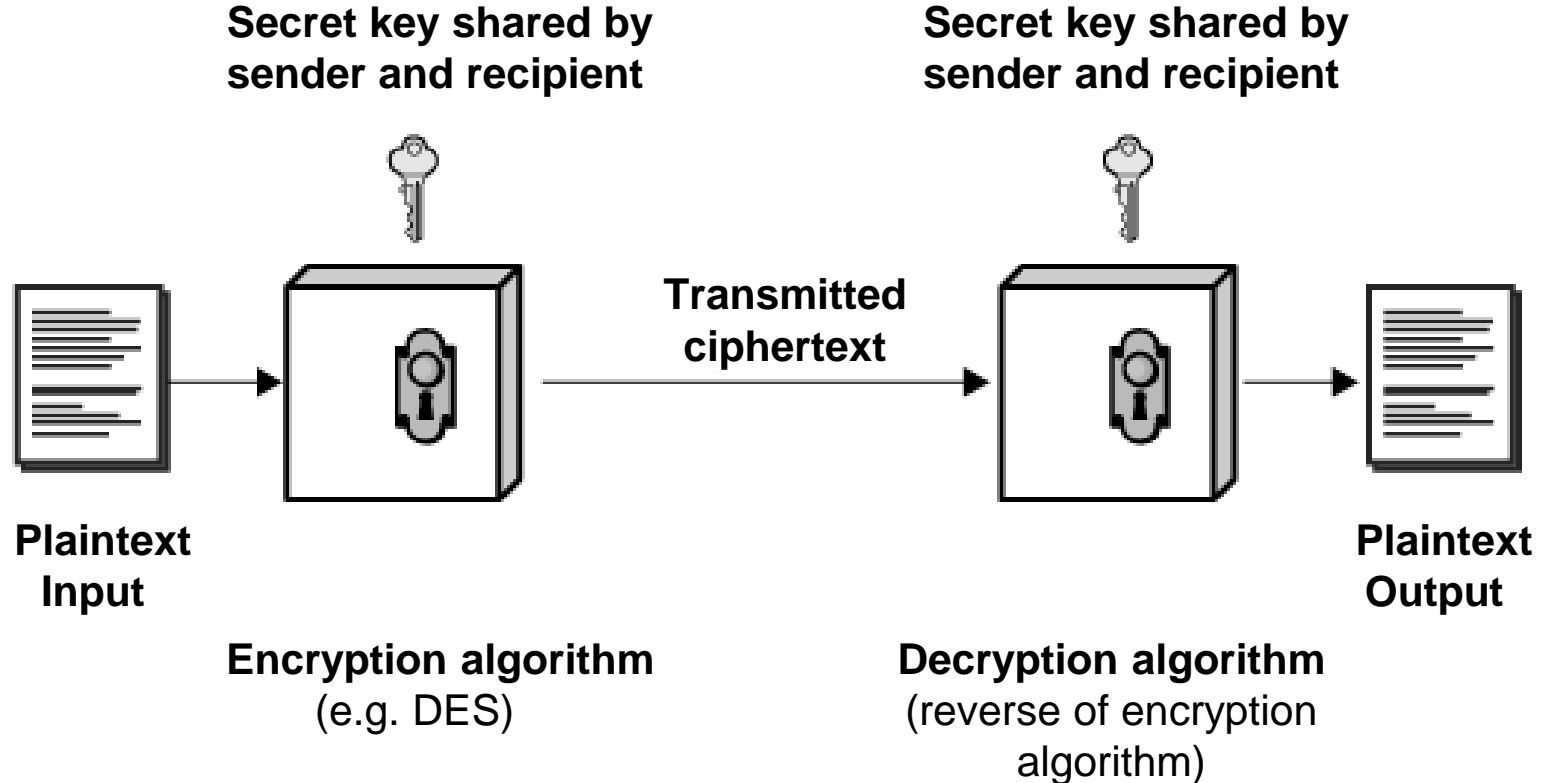
DES key length: 56 bit

AES key length: 128+ bit

3DES key length: 168 bit

Age of the universe:  $\sim 10^{10}$  years

# 5. Symmetric encryption Model



# 5. Symmetric encryption Model

- Comprise 5 ingredients:
  - Plaintext
  - Encryption algorithm
  - Secret key
  - Ciphertext
  - Decryption algorithm
- **Security level depends on the secret of the key**, not on the secret of the algorithm

# 5. Symmetric encryption Model

How to manage secret keys?

- The **disadvantage** of symmetric encryption

How to distribute the key to the broadcast parties?

→ The system is often insecure due to poor management of the distribution of secret keys

- Two ways to distribute key:
  - Manual distribution
  - Automatic distribution by key distribution center

# 5.1. Data Encryption Standard

DES

- Published as an official standard in 1977
- It has been highly influential in the advancement of cryptography
- Referred as the **DEA** (Data Encryption Algorithm)
- Block sizes: 64 bits
- Key sizes: 56 bits



# 5.1. Data Encryption Standard

DES-cracking

- Key length: 56 bit  $\rightarrow 2^{56} = 7,2 \times 10^{16}$  possible keys
- High computational speed can break:
  - 1997: 70.000 computers broke in 96 days
  - 1998: Electronic Frontier Foundation (EFF) broke in 56 hours (< 3 days)
  - 1999: 100.000 broke in 22 hours and 15 minutes
  - 2016: 8 GTX 1080 Ti GPUs broke in a average of under 2 days
  - 2017: using a rainbow table can break in 25 seconds

→ Need more secure algorithms

# 5.2. Triple Data Encryption Algorithm

Triple DES / 3 DES / TDES / TDEA

- First published in 1995
- Derived from DES
- Block sizes: 64 bits
- Key sizes: 168 bits
- This is the best public cryptanalysis



# 5.2. Triple Data Encryption Algorithm

Triple DES / 3 DES / TDES / TDEA

- Apply the DES cipher algorithm 3 times to each data block
- Use 3 keys ( $K_1, K_2, K_3$ )
  - **Encryption:**  $C = EK_3[DK_2[EK_1[p]]]$
  - **Decryption:**  $p = DK_1[EK_2[DK_3[C]]]$
- The actual key length is 168 bits
- Why 3 times?  
To avoid a collision attack

# 5.3. Advanced Encryption Standard

AES


- Published as an new standard in 2001
- Known by the original name Rijndael (Rijmen + Daemen)
- **More secure and faster than 3DES**
- Block sizes: 128 bits
- Key sizes: 128/192/256 bits
- This is the best public cryptanalysis

# 5.4. Library

- **DES Class**
  - **Namespace:** System.Security.Cryptography
  - Represents the base class for DES algorithm
- **TripleDES Class**
  - Namespace: System.Security.Cryptography
  - Represents the base class for 3 DES algorithm
- **Aes Class**
  - Namespace: System.Security.Cryptography
  - Represents the abstract base class for AES algorithm

# Exercise

Write a program to encrypt and decrypt a string using AES



A screenshot of a graphical user interface (GUI) window titled "AES". The window has a standard Windows-style title bar with minimize, maximize, and close buttons. The main area of the window is divided into three horizontal sections. Each section consists of a text label on the left and a corresponding input field on the right. The first section is labeled "Plain Text" and has an empty input field. The second section is labeled "Cipher Text" and has an empty input field. The third section is labeled "Plain Text" and has an empty input field. To the right of the input fields, there are two buttons: "Encrypt" and "Decrypt". The "Encrypt" button is positioned to the right of the first "Plain Text" input field, and the "Decrypt" button is positioned to the right of the "Cipher Text" input field.

Label	Input Field	Action Button
Plain Text	<input type="text"/>	Encrypt
Cipher Text	<input type="text"/>	Decrypt
Plain Text	<input type="text"/>	

# Guideline

## Encryption Phase

- **Step 1:** Create an Aes object with the specified key and IV
- **Step 2:** Create an encryptor to perform the stream transform
- **Step 3:** Create the streams used for encryption
- **Step 4:** Write all data to the stream and encrypt
- **Step 5:** Get the encrypted bytes from the memory stream

# Guideline

## Decryption Phase

- **Step 1:** Create an Aes object with the specified key and IV
- **Step 2:** Create an encryptor to perform the stream transform
- **Step 3:** Create the streams used for decryption
- **Step 4:** Read the decrypted bytes and decrypt
- **Step 5:** Get the string



# 6. Asymmetric encryption

- The **disadvantages** of symmetric encryption:
  - **Key distribution problem**
    - ✓ Hard to guarantee sharing without revealing the secret key
    - ✓ The key distribution center may be hacked
  - **Not suitable for digital signatures**
- The scheme was published by Whitfield Diffie and Martin Hellman in 1976
  - To address the limitations of the symmetric encryption
  - To complement symmetric encryption

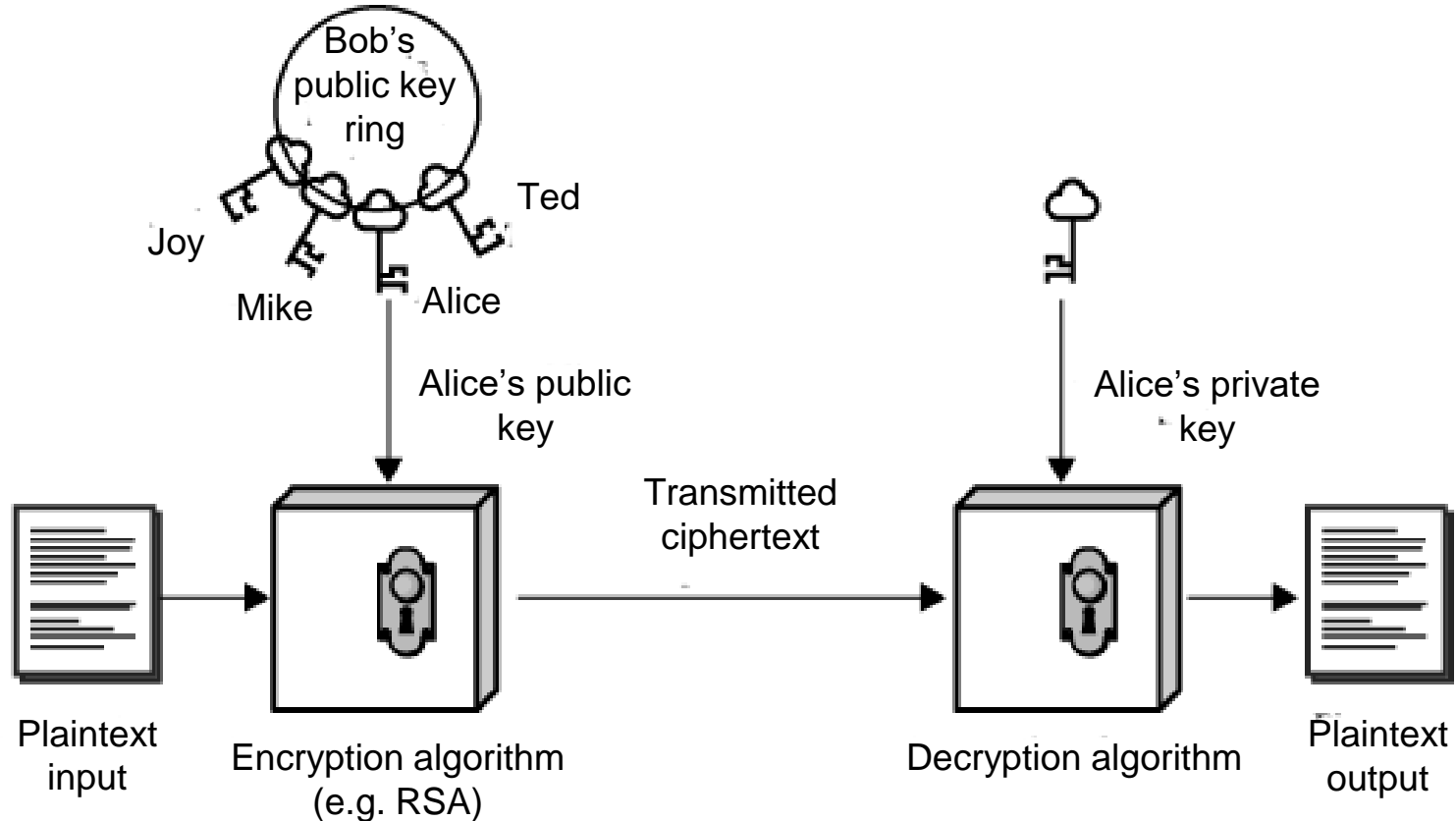
# 6. Asymmetric encryption

- Known as **Public-key cryptography**
- Use paired keys:
  - **public key**
    - ✓ known to all
    - ✓ used to encrypt messages and verify signatures
  - **private key**
    - ✓ controlled solely by the owner
    - ✓ used to decrypt the messages and sign (create) the digital signature

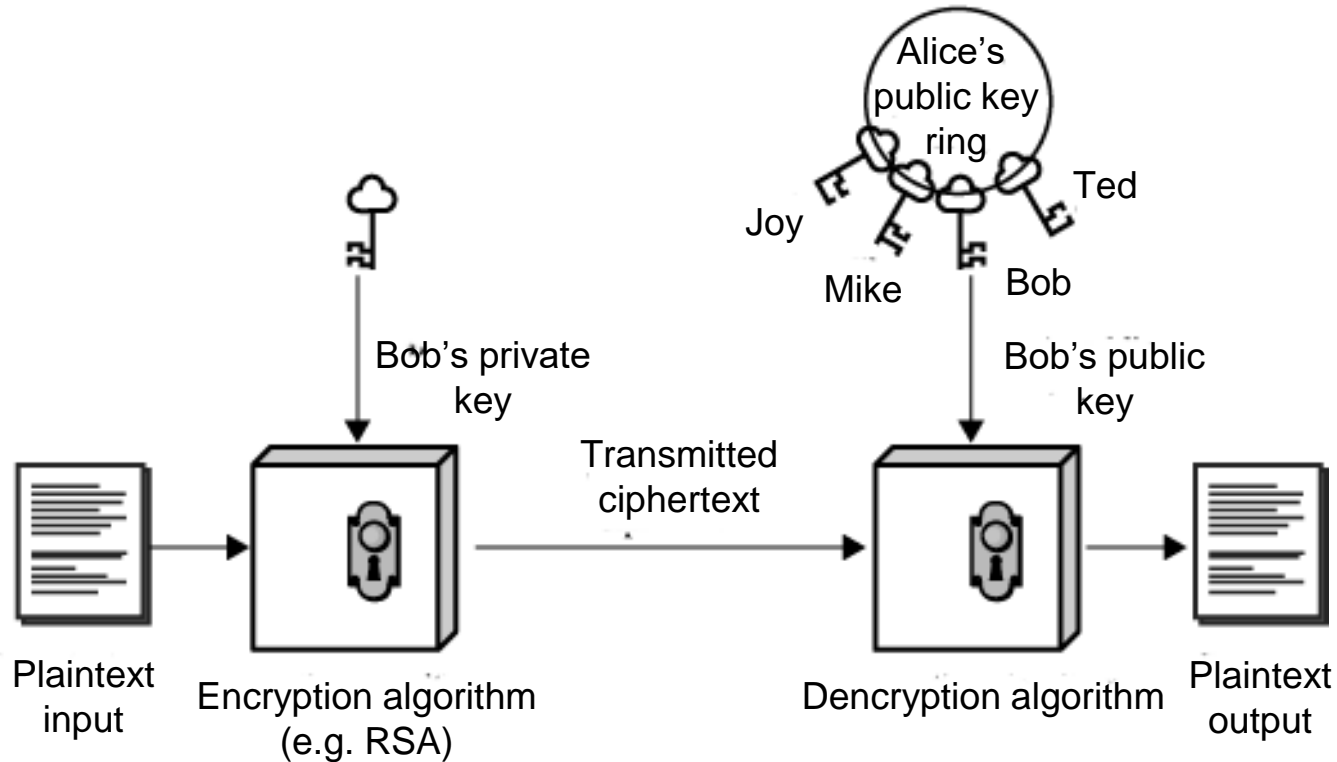
# 6.1. Public-key encryption application

- Classified into 3 types:
  - **Encryption and Decryption**  
ensure the confidentiality of information
  - **Digital signature**  
verify the authenticity of digital messages or documents
  - **Key exchange**  
share session key
- Some algorithms are suitable for all 3 types; others can only be used for 1 or 2 types

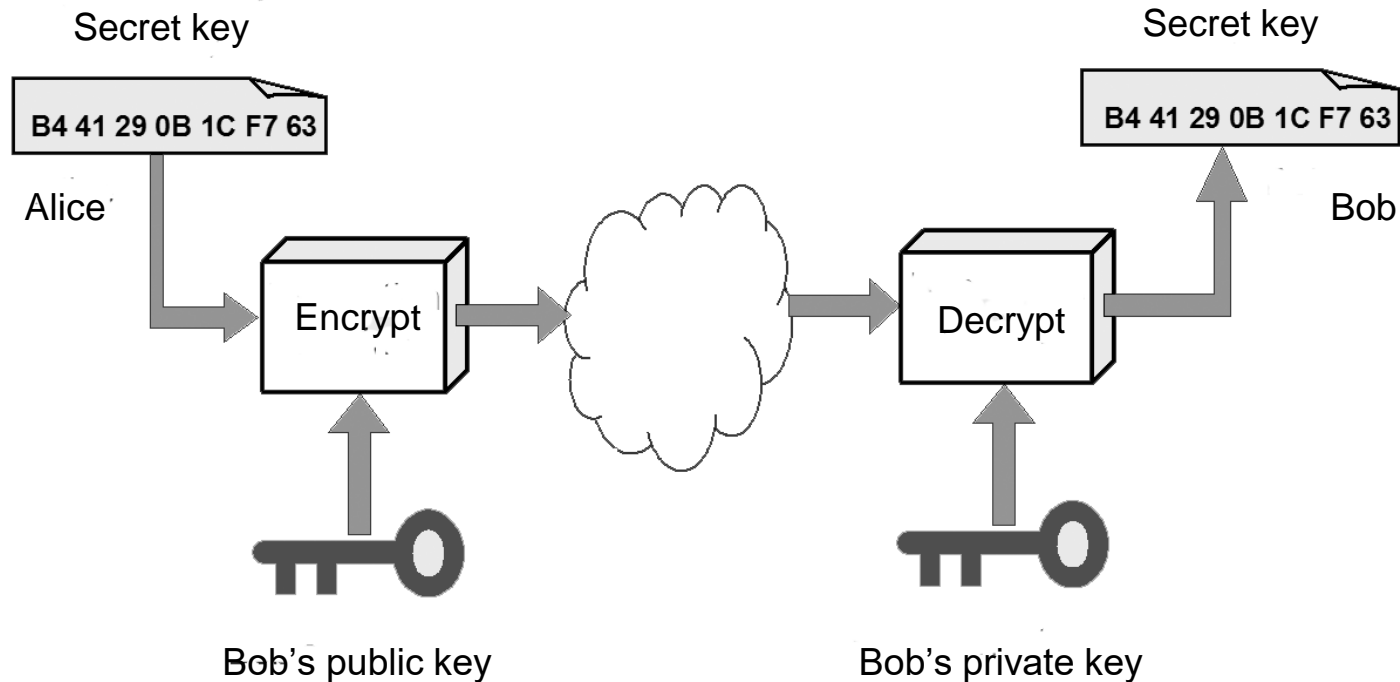
# 6.1a. Public-key cryptography



## 6.1b. Authentication using public-key cryptography



# 6.1c. Key exchange



## 6.2. RSA cryptosystem

- Published in 1977 by Ron **R**ivest, Adi **S**hamir and Len **A**dleman (MIT)
- One of the first public-key cryptosystems and is widely used
- Key sizes **1,024** to **4,096** bits
- **Secure** because the factorization cost of a large integer is huge
- RSA is a **relatively slow** algorithm → it is less commonly used to directly encrypt user data

# 6.3. Combination of two cryptosystem

Secure protocol uses both:

- **Symmetric cryptography**

- secure data exchange via network
- fast processing speed

- **Asymmetric cryptography**

- establish a connection between two network entities
- establish a symmetric key



# 6.4. Library

## **RSA Class**

- Namespace: System.Security.Cryptography
- Represents the base class for RSA algorithm

## **RSACryptoServiceProvider Class**

- Namespace: System.Security.Cryptography
- Performs asymmetric encryption and decryption using the implementation of the RSA algorithm

# Exercise

Write a program to encrypt and decrypt a string using RSA



# Guideline

## Encryption Phase

- **Step 1:** Create an RSA object to generate public and private key data
- **Step 2:** Pass the data to ENCRYPT, the public key information



# Guideline

## Decryption Phase

- **Step 1:** Create an RSA object to generate public and private key data
- **Step 2:** Pass the data to DECRYPT, the private key information
- **Step 3:** Display the plaintext



# 7. Piracy protection

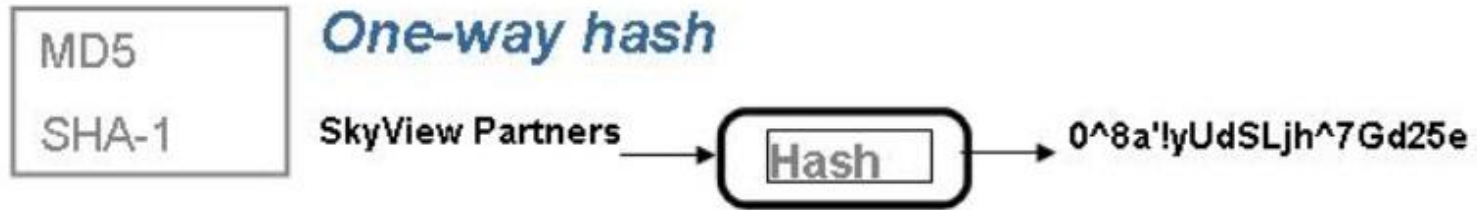
- The most common form of software piracy is a CD-R with the license code scribbled on the front
- The only real way to guarantee that the same license code cannot be used on multiple machines is to **track these codes from a central server**

# 7. Piracy protection

- **A common way to generate license codes** is to choose two large random number  $a$ ,  $b$ . A key  $c$  is valid if:  $(c - a) \bmod b = 0$
- Hackers can also use programs to enter licens key automatically → Software closes after 3 failed attempts and delete itself after 100 times

# 8. Hash functions

- Used to **map data** of arbitrary size to **fixed-size values**
- A **one-way function** which is practically infeasible to invert



- A input bit is changed → at least half the number of result bits will be changed

# 8. Hash functions

- MD (Message Digest) functions
  - MD2, MD4, MD5: 128 bits
  - MD6: 512 bits
- SHA(Secure Hash Algorithm)
  - SHA-1: 160 bits
  - SHA-224: 224 bits
  - SHA-256: 256 bits
  - SHA-512: 512 bits
  - SHA-3: arbitrary



# 9. Message authentication code

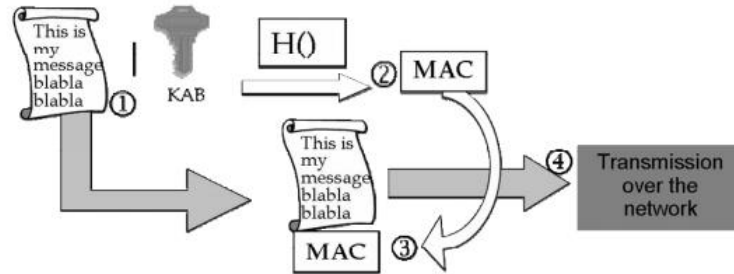
MAC

- A short piece of information used to authenticate a message
  - **Authentication:** confirm that the message came from the stated sender
  - **Integrity:** confirm that the message has not been modified
- Use hash functions, symmetric or asymmetric keys

# 9. Message authentication code

MAC

Source A



Receiver B

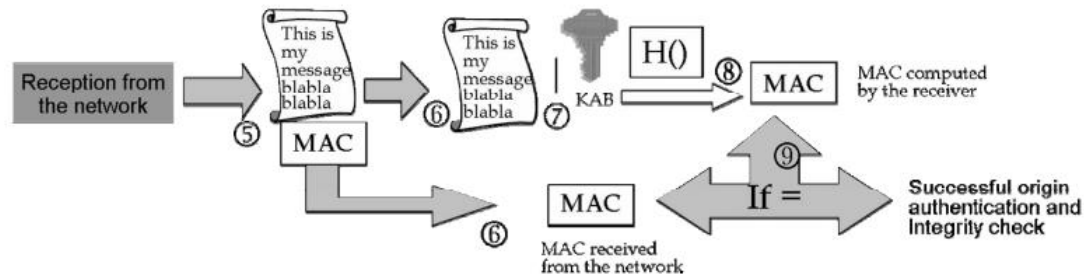
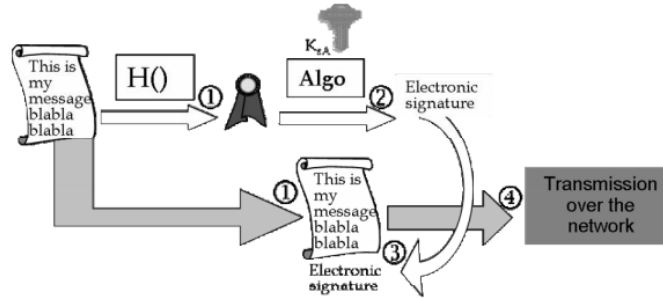


Figure 3.1. Generation and verification of a MAC (symmetric cryptography)

# 9. Message authentication code

MAC

Source A



Receiver B

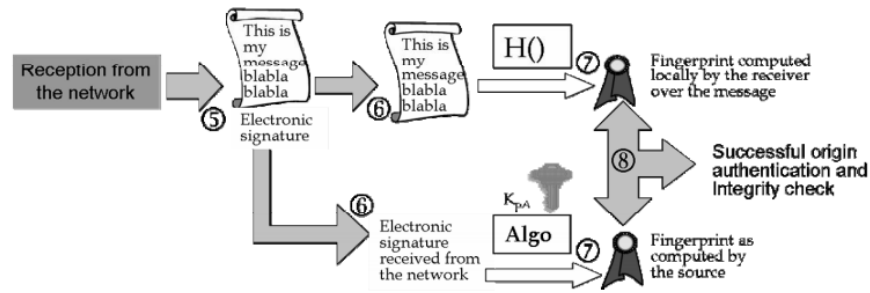


Figure 3.2. Generation and verification of an electronic signature (asymmetric cryptography)

# Types of Encryption

DES

TripleDES

AES

RC5

## Symmetric Keys

- Encryption and decryption use the **same key**.

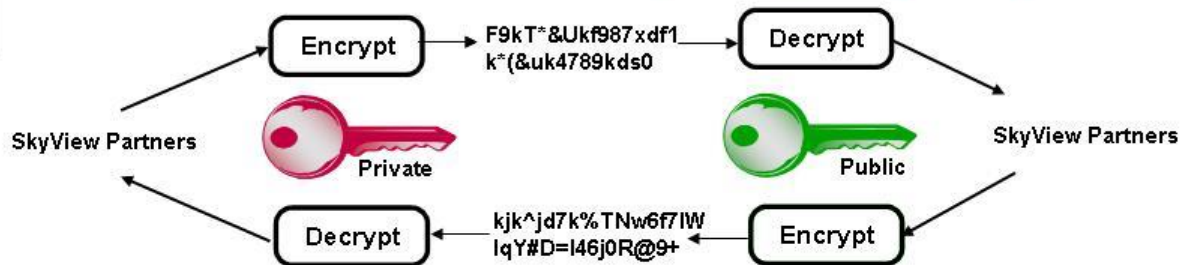


RSA

Elliptic  
Curve

## Asymmetric keys

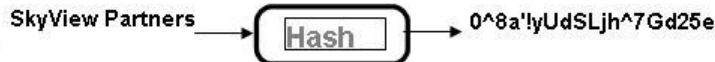
- Encryption and decryption use different keys, a **public key** and a **private key**.



MD5

SHA-1

## One-way hash





**Q&A**