Chapter 2

Cryptographic Tools

Course Instructor: Dr. Nausheen Shoaib

Symmetric Encryption

- The universal technique for providing confidentiality for transmitted or stored data
- Also referred to as conventional encryption or single-key encryption
- Two requirements for secure use:
  - Need a strong encryption algorithm
  - Sender and receiver must have obtained copies of the secret key in a secure fashion and must keep the key secure

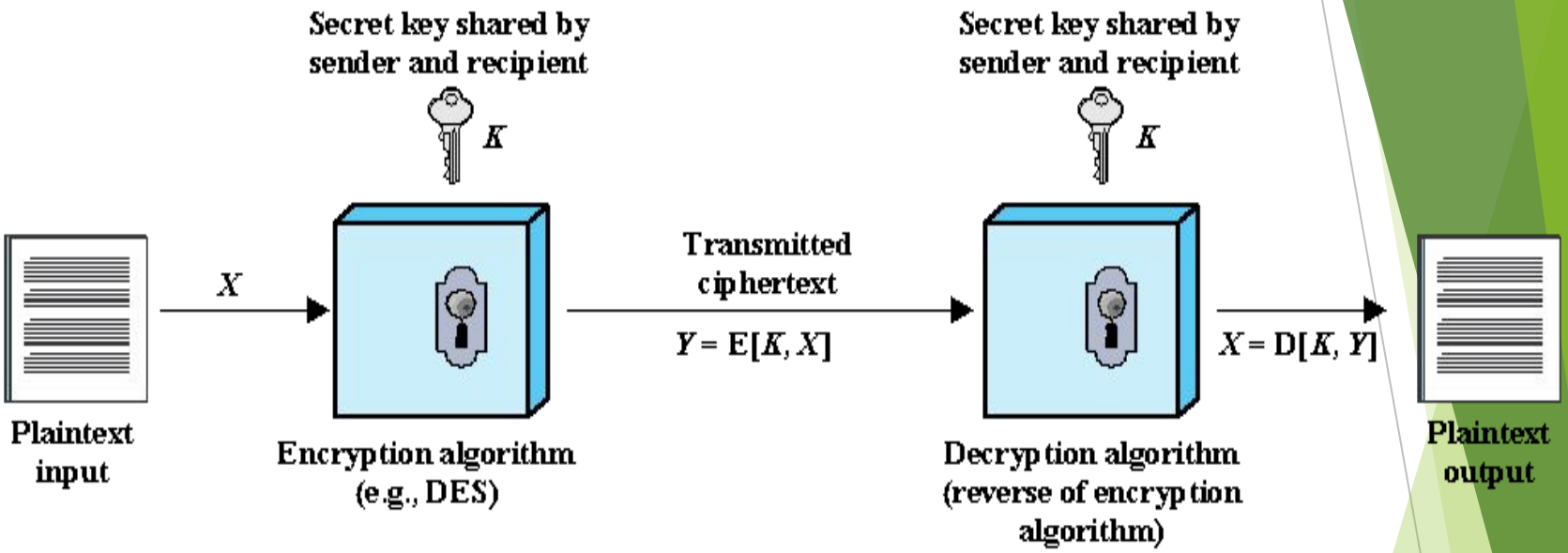


Figure 2.1 Simplified Model of Symmetric Encryption

Attacking Symmetric Encryption

Cryptanalytic Attacks

- Rely on:
  - Nature of the algorithm
  - Some knowledge of the general characteristics of the plaintext
  - Some sample plaintext-ciphertext pairs
- Exploits the characteristics of the algorithm to attempt to deduce a specific plaintext or the key being used
  - If successful all future and past messages encrypted with that key are compromised

Brute-Force Attacks

- Try all possible keys on some ciphertext until an intelligible translation into plaintext is obtained
  - On average half of all possible keys must be tried to achieve success

Table 2.1

	DES	Triple DES	AES
Plaintext block size (bits)	64	64	128
Ciphertext block size (bits)	64	64	128
Key size (bits)	56	112 or 168	128, 192, or 256

DES = Data Encryption Standard
AES = Advanced Encryption Standard

Comparison of Three Popular Symmetric Encryption Algorithms

Data Encryption Standard (DES)

- Until recently was the most widely used encryption scheme
  - FIPS PUB 46
  - Referred to as the Data Encryption Algorithm (DEA)
  - Uses 64 bit plaintext block and 56 bit key to produce a 64 bit ciphertext block
- Strength concerns:
  - Concerns about the algorithm itself
    - DES is the most studied encryption algorithm in existence
  - Concerns about the use of a 56-bit key
    - The speed of commercial off-the-shelf processors makes this key length woefully inadequate

Table 2.2

Key size (bits)	Cipher	Number of Alternative Keys	Time Required at 10^9 decryptions/s	Time Required at 10^{13} decryptions/s
56	DES	$2^{56} \approx 7.2 \times 10^{16}$	2^{55} ns = 1.125 years	1 hour
128	AES	$2^{128} \approx 3.4 \times 10^{38}$	2^{127} ns = 5.3×10^{21} years	5.3×10^{17} years
168	Triple DES	$2^{168} \approx 3.7 \times 10^{50}$	2^{167} ns = 5.8×10^{33} years	5.8×10^{29} years
192	AES	$2^{192} \approx 6.3 \times 10^{57}$	2^{191} ns = 9.8×10^{40} years	9.8×10^{36} years
256	AES	$2^{256} \approx 1.2 \times 10^{77}$	2^{255} ns = 1.8×10^{60} years	1.8×10^{56} years

Average Time Required for Exhaustive Key Search

Triple DES (3DES)

- Repeats basic DES algorithm three times using either two or three unique keys
- First standardized for use in financial applications in ANSI standard X9.17 in 1985
- Attractions:
  - 168-bit key length overcomes the vulnerability to brute-force attack of DES
  - Underlying encryption algorithm is the same as in DES
- Drawbacks:
  - Algorithm is sluggish in software
  - Uses a 64-bit block size

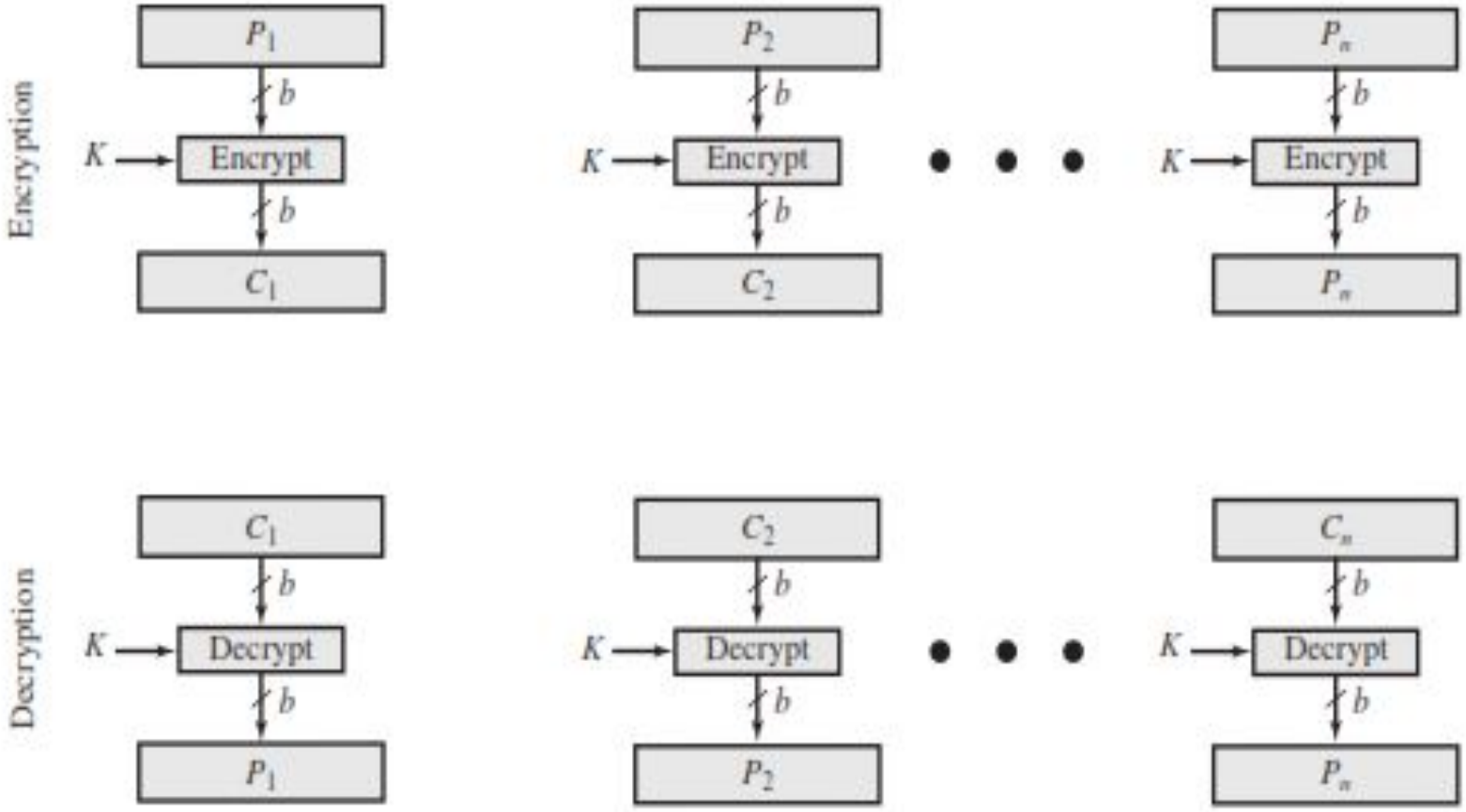
Advanced Encryption Standard (AES)

- **Needed a replacement for 3DES**
  - **3DES was not reasonable for long term use**
- **NIST called for proposals for a new AES in 1997**
  - **Should have a security strength equal to or better than 3DES**
  - **Significantly improved efficiency**
  - **Symmetric block cipher**
  - **128 bit data and 128/192/256 bit keys**
- **Selected Rijndael in November 2001**
  - **Published as**
  - **FIPS 197**

Practical Security Issues

- Typically symmetric encryption is applied to a unit of data larger than a single 64-bit or 128-bit block. E-mail messages, network packets,database records, and other plaintext sources must be broken up into a series of fixed length block for encryption by a symmetric block cipher.
- Electronic codebook (ECB) mode is the simplest approach to multiple-block encryption
  - Each block of plaintext is encrypted using the same key
  - Cryptanalysts may be able to exploit regularities in the plaintext
- Modes of operation
  - Alternative techniques developed to increase the security of symmetric block encryption for large sequences
  - Overcomes the weaknesses of ECBs

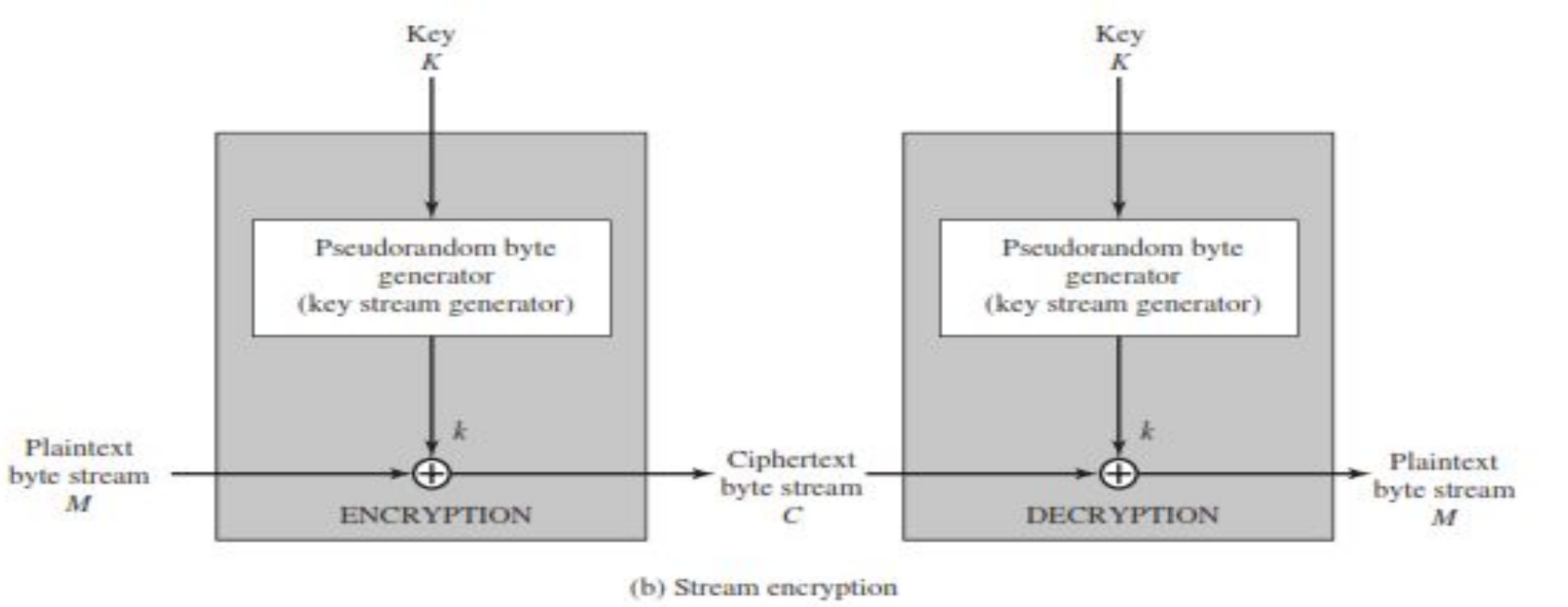
ECB Mode



(a) Block cipher encryption (electronic codebook mode)

Stream Cipher and Block Cipher

- A *block cipher processes the input one block of elements at a time, producing an* output block for each input block. A *stream cipher processes the input elements continuously*, Producing output one element at a time,



(b) Stream encryption

Stream Cipher

key is input to a pseudorandom bit generator that produces a stream of 8-bit numbers that are apparently random. A pseudorandom stream is one that is unpredictable without knowledge of the input key

Stream Cipher vs. Block Cipher

- a stream cipher can be as secure as a block cipher of comparable key length.
- stream ciphers are almost always faster and use far less code than do block ciphers. block cipher is that you can reuse keys.
- For applications that require encryption/decryption of a stream of data, such as over a data communications channel or a browser/Web link, a stream cipher might be the better alternative.
- For applications that deal with blocks of data, such as file transfer, e-mail, and database, block ciphers may be more appropriate.

Block & Stream Ciphers

Block Cipher

- Processes the input one block of elements at a time
- Produces an output block for each input block
- Can reuse keys
- More common

Stream Cipher

- Processes the input elements continuously
- Produces output one element at a time
- Primary advantage is that they are almost always faster and use far less code

Message Authentication and Hash Functions

- Encryption protects against passive attack (eavesdropping). A different requirement is to protect against active attack (falsification of data and transactions). Protection against such attacks is known as message or data authentication.
- The two important aspects are to verify that the contents of the message have not been altered and that the source is authentic.
- verify a message's timeliness (it has not been artificially delayed and replayed) and
- sequence relative to other messages flowing between two parties

Authentication Using Symmetric Encryption

- only the sender and receiver share a key
- if the message includes an error-detection code and a
- sequence number, the receiver is assured that no alterations have been made and that sequencing is proper.
- If the message also includes a timestamp, the receiver is assured that the message has not been delayed.
- in the ECB mode of encryption, if an attacker reorders the blocks of ciphertext, then each block will still decrypt successfully.
- block reordering is still a threat

Message Authentication without Message Encryption

- , an authentication tag is generated and appended to each message for transmission. The message itself is not encrypted
- same message is broadcast to a number of destinations. Two examples are notification to users that the network is now unavailable, and an alarm signal in a control center
- Authentication of a computer program in plaintext is an attractive service. The computer program can be executed without having to decrypt it every time,

Message Authentication Code

- One authentication technique involves the use of a secret key to generate a small block of data, known as a message authentication code, that is appended to the message.

Message Authentication

- Protects against active attacks
- Verifies received message is authentic
  - Contents have not been altered
  - From authentic source
  - Timely and in correct sequence
- Can use conventional encryption
  - Only sender and receiver share a key

Message Authentication Without Confidentiality

- Message encryption by itself does not provide a secure form of authentication
- It is possible to combine authentication and confidentiality in a single algorithm by encrypting a message plus its authentication tag
- Typically message authentication is provided as a separate function from message encryption
- Situations in which message authentication without confidentiality may be preferable include:
  - There are a number of applications in which the same message is broadcast to a number of destinations
  - An exchange in which one side has a heavy load and cannot afford the time to decrypt all incoming messages
  - Authentication of a computer program in plaintext is an attractive service
- Thus, there is a place for both authentication and encryption in meeting security requirements

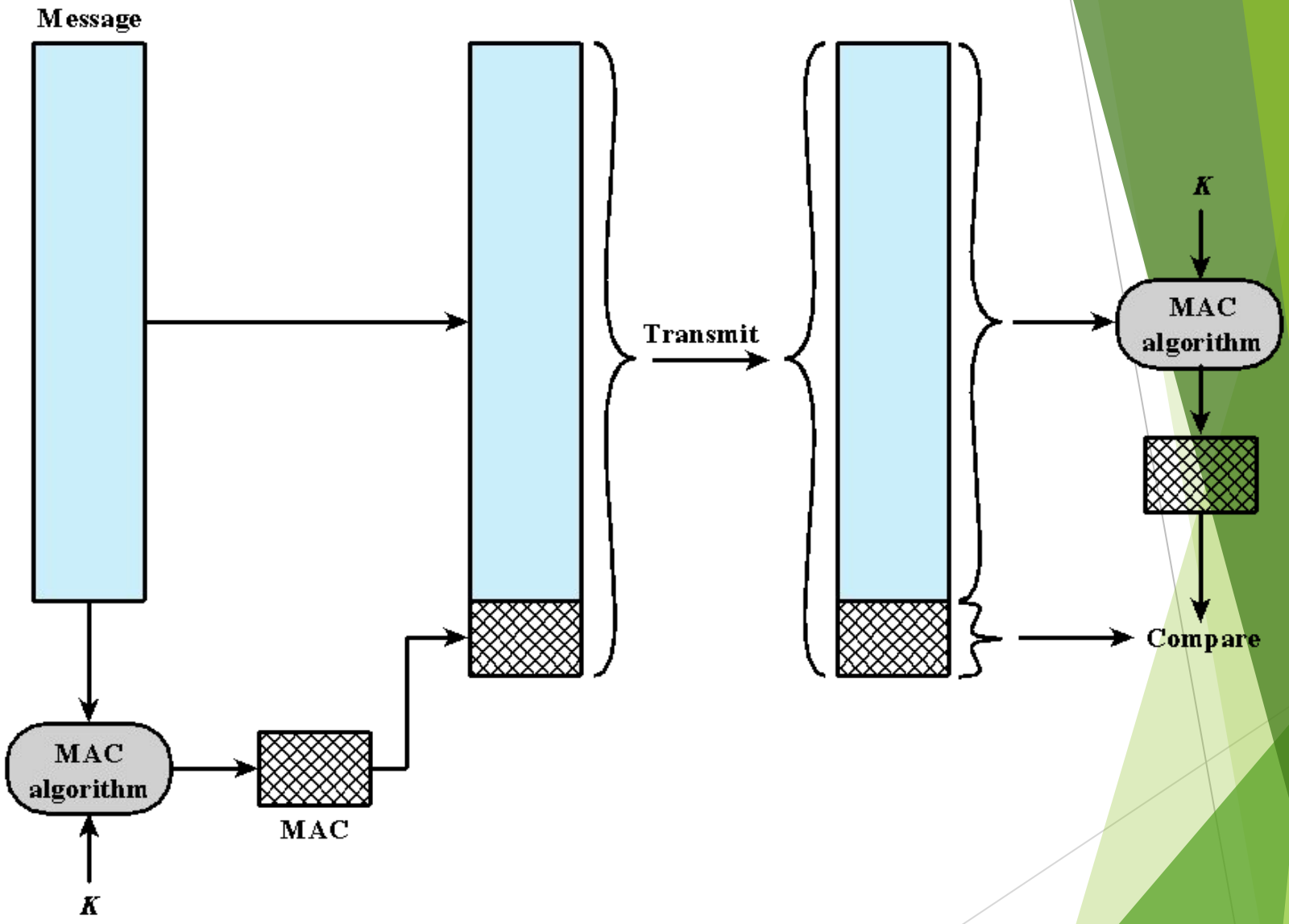


Figure 2.3 Message Authentication Using a Message Authentication Code (MAC).

One way Hash Function

- accepts a variable-size message *M* as input and produces a fixed-size message digest H(*M*) as output.
- the message is padded out to an integer multiple of some fixed length (e.g., 1024 bits).
- length field is a security measure to increase the difficulty for an attacker.

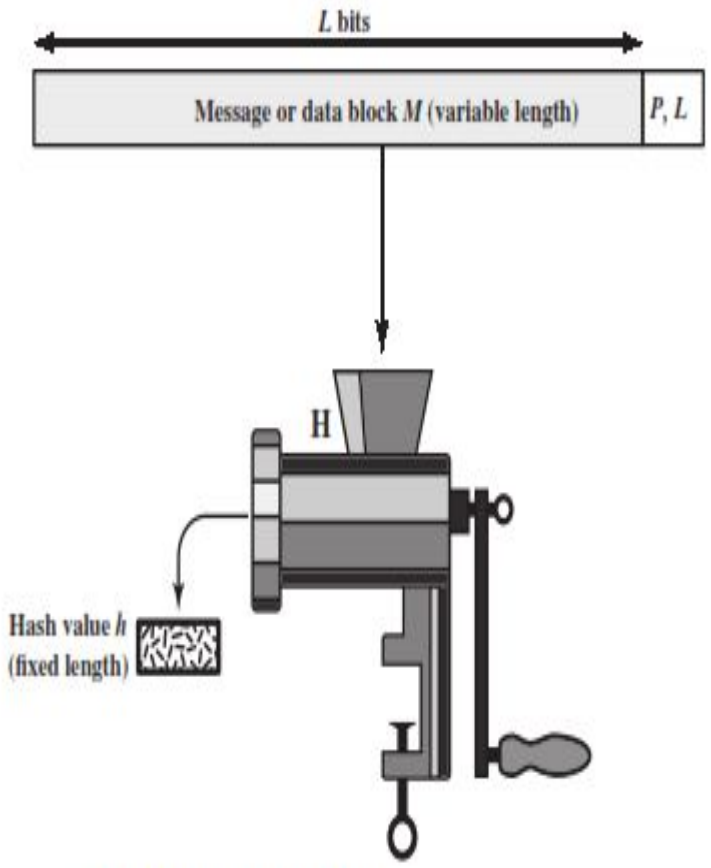
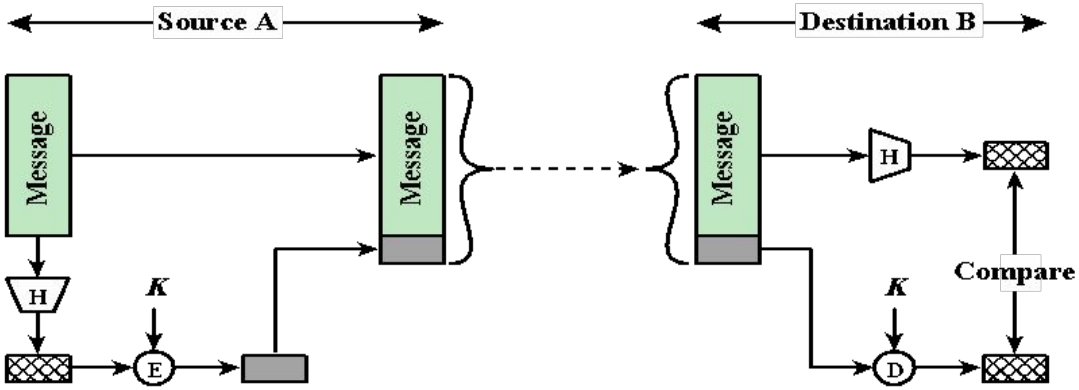
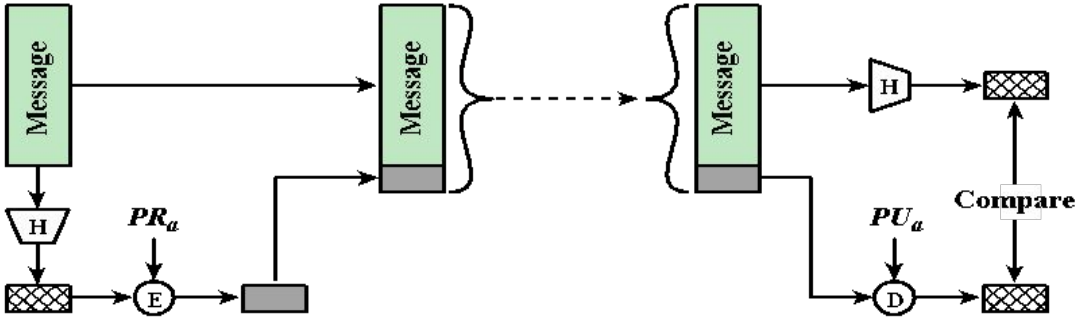


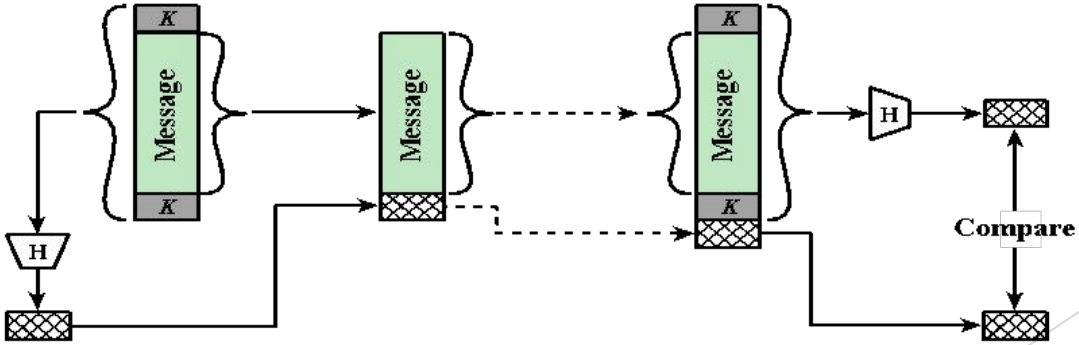
Figure 2.4 Cryptographic Hash Function; $h = H(M)$



(a) Using symmetric encryption



(b) Using public-key encryption



(c) Using secret value

Figure 2.5 Message Authentication Using a One-Way Hash Function.

To be useful for message authentication, a hash function H must have the following properties:

- Can be applied to a block of data of any size
- Produces a fixed-length output
- H(x) is relatively easy to compute for any given x
- One-way or pre-image resistant
  - Computationally infeasible to find x such that H(x) = h
- Computationally infeasible to find $y \neq x$ such that H(y) = H(x)
- Collision resistant or strong collision resistance
  - Computationally infeasible to find any pair (x,y) such that H(x) = H(y)

Security of Hash Functions

There are two approaches to attacking a secure hash function:

- **Cryptanalysis**
  - Exploit logical weaknesses in the algorithm
- **Brute-force attack**
  - Strength of hash function depends solely on the length of the hash code produced by the algorithm

SHA most widely used hash algorithm

Additional secure hash function applications:

- **Passwords**
  - Hash of a password is stored by an operating system
- **Intrusion detection**
  - Store H(F) for each file on a system and secure the hash values

Public-Key Encryption Structure

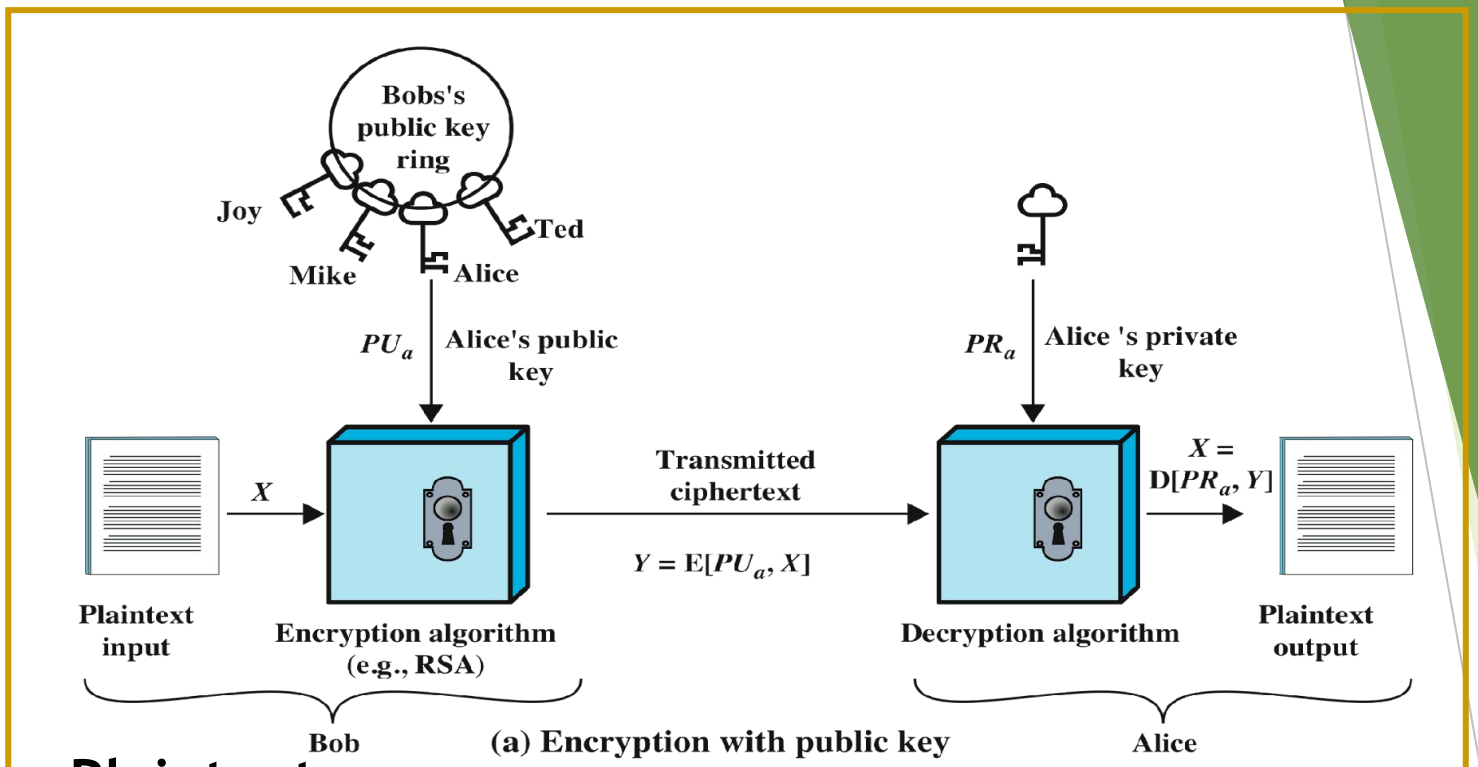
Publicly proposed by Diffie and Hellman in 1976

Based on mathematical functions

Asymmetric

- **Uses two separate keys**
- **Public key and private key**
- **Public key is made public for others to use**

Some form of protocol is needed for distribution



- **Plaintext**
  - Readable message or data that is fed into the algorithm as input
- **Encryption algorithm**
  - Performs transformations on the plaintext
- **Public and private key**
  - Pair of keys, one for encryption, one for decryption
- **Ciphertext**
  - Scrambled message produced as output
- **Decryption key**
  - Produces the original plaintext

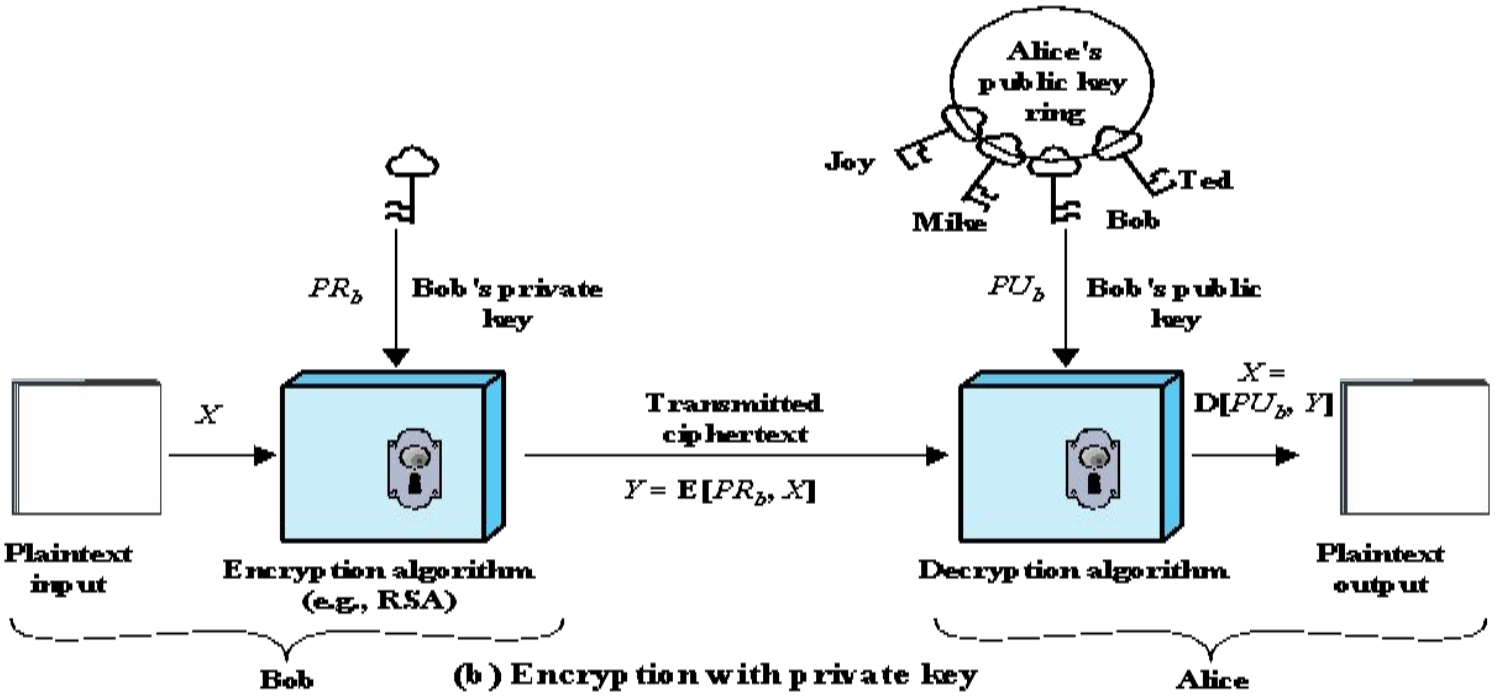


Figure 2.6 Public-Key Cryptography

- User encrypts data using his or her own private key
- Anyone who knows the corresponding public key will be able to decrypt the message

If Bob wishes to send a private message to Alice, Bob encrypts the message using Alice's public key.

4. When Alice receives the message, she decrypts it using her private key. No other recipient can decrypt the message because only Alice knows Alice's private key.

Table 2.3

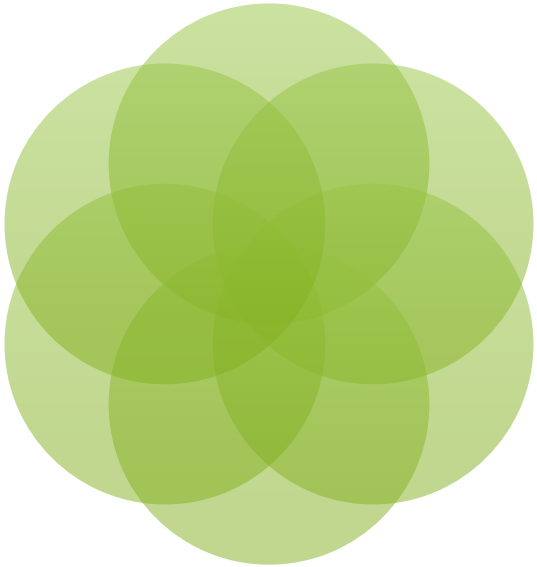
Applications for Public-Key Cryptosystems

Algorithm	Digital Signature	Symmetric Key Distribution	Encryption of Secret Keys
RSA	Yes	Yes	Yes
Diffie-Hellman	No	Yes	No
DSS	Yes	No	No
Elliptic Curve	Yes	Yes	Yes

Requirements for Public-Key Cryptosystems

Computationally easy to create key pairs

Useful if either key can be used for each role



Computationally easy for sender knowing public key to encrypt messages

Computationally infeasible for opponent to otherwise recover original message

Computationally easy for receiver knowing private key to decrypt ciphertext

Computationally infeasible for opponent to determine private key from public key

Asymmetric Encryption Algorithms

RSA (Rivest, Shamir, Adleman)	Developed in 1977	Most widely accepted and implemented approach to public-key encryption	Block cipher in which the plaintext and ciphertext are integers between 0 and *n*-1 for some *n*.
Diffie-Hellman key exchange algorithm	Enables two users to securely reach agreement about a shared secret that can be used as a secret key for subsequent symmetric encryption of messages	Limited to the exchange of the keys	
Digital Signature Standard (DSS)	Provides only a digital signature function with SHA-1	Cannot be used for encryption or key exchange	
Elliptic curve cryptography (ECC)	Security like RSA, but with much smaller keys		