



Chapter 2

Cryptographic Tools

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



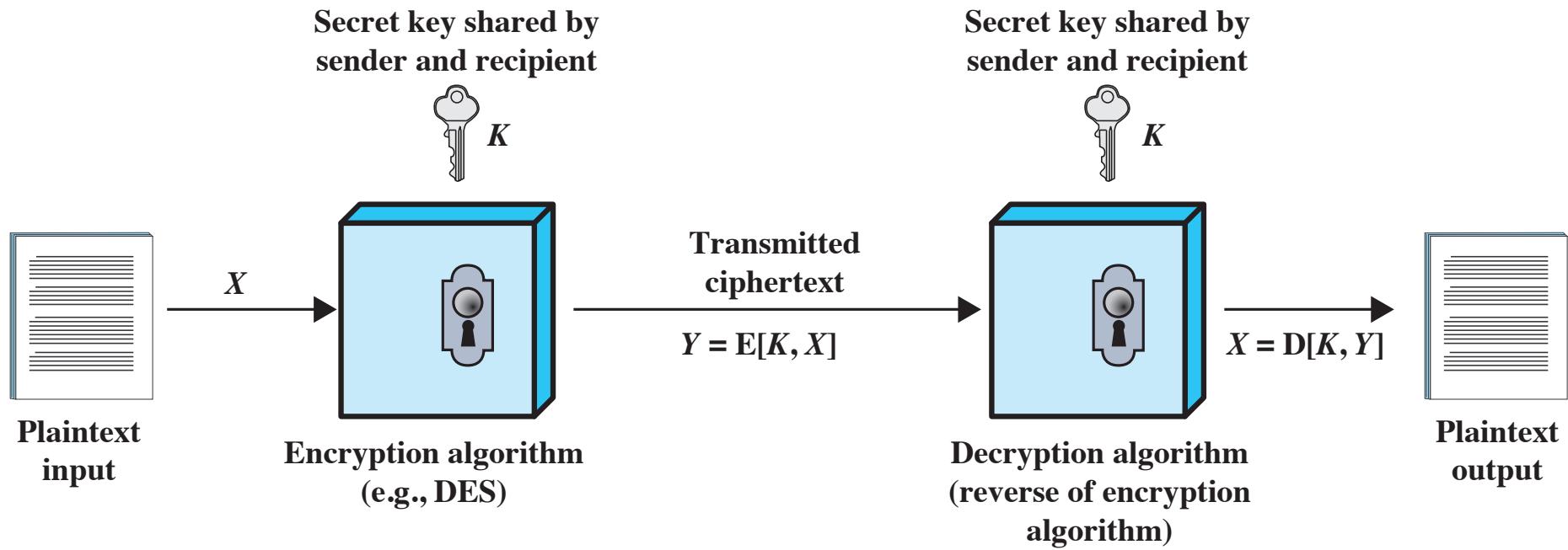


Figure 2.1 Simplified Model of Symmetric Encryption

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 Attack

- 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

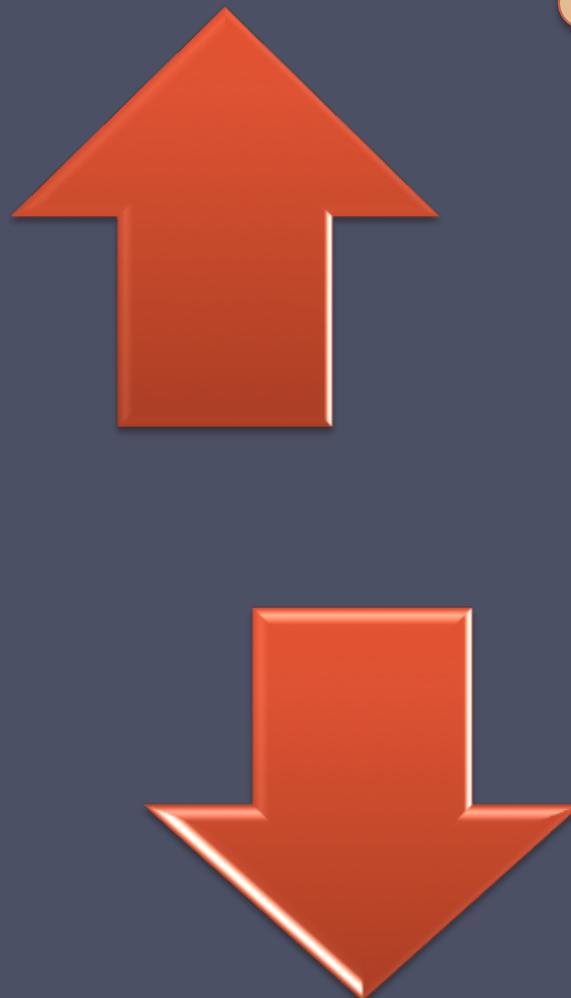
	DES	Triple DES	AES
Plaintext block size (bits)	64	64	128
Ciphertext block size (bits)	64	64	128
Key size (bits)	56	112 or 168	128, 192, or 256

DES = Data Encryption Standard

AES = Advanced Encryption Standard

Comparison of Three Popular Symmetric Encryption Algorithms

Data Encryption Standard (DES)



- 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 algorithm
 - DES is the most studied encryption algorithm in existence
 - Use of 56-bit key
 - Electronic Frontier Foundation (EFF) announced in July 1998 that it had broken a DES encryption

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	$2^{56} \approx 7.2 \times 10^{16}$	$2^{55} \text{ ns} = 1.125 \text{ years}$	1 hour
128	AES	$2^{128} \approx 3.4 \times 10^{38}$	$2^{127} \text{ ns} = 5.3 \times 10^{21}$ years	$5.3 \times 10^{17} \text{ years}$
168	Triple DES	$2^{168} \approx 3.7 \times 10^{50}$	$2^{167} \text{ ns} = 5.8 \times 10^{33}$ years	$5.8 \times 10^{29} \text{ years}$
192	AES	$2^{192} \approx 6.3 \times 10^{57}$	$2^{191} \text{ ns} = 9.8 \times 10^{40}$ years	$9.8 \times 10^{36} \text{ years}$
256	AES	$2^{256} \approx 1.2 \times 10^{77}$	$2^{255} \text{ ns} = 1.8 \times 10^{60}$ years	$1.8 \times 10^{56} \text{ years}$

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

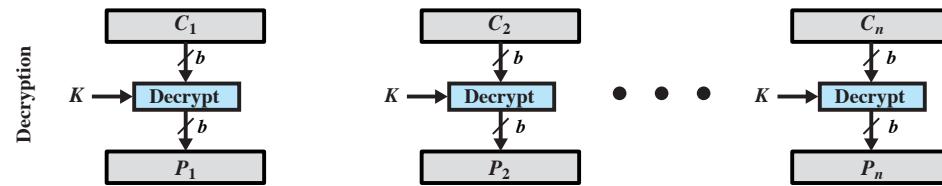
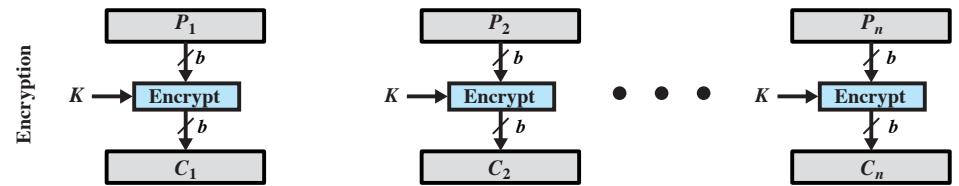
Selected Rijndael in November 2001

Published as FIPS 197

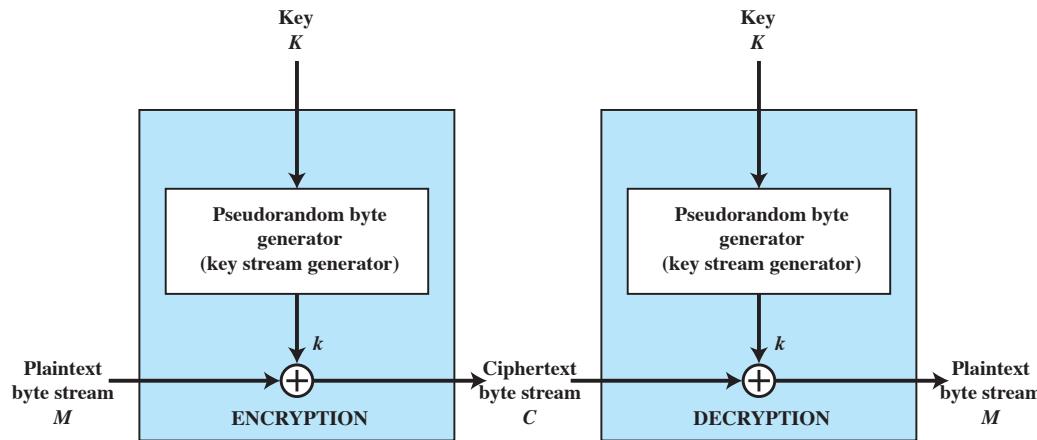
Practical Security Issues

- Typically symmetric encryption is applied to a unit of data larger than a single 64-bit or 128-bit block
- 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 ECB





(a) Block cipher encryption (electronic codebook mode)



(b) Stream encryption

Figure 2.2 Types of Symmetric Encryption





Block & Stream Ciphers

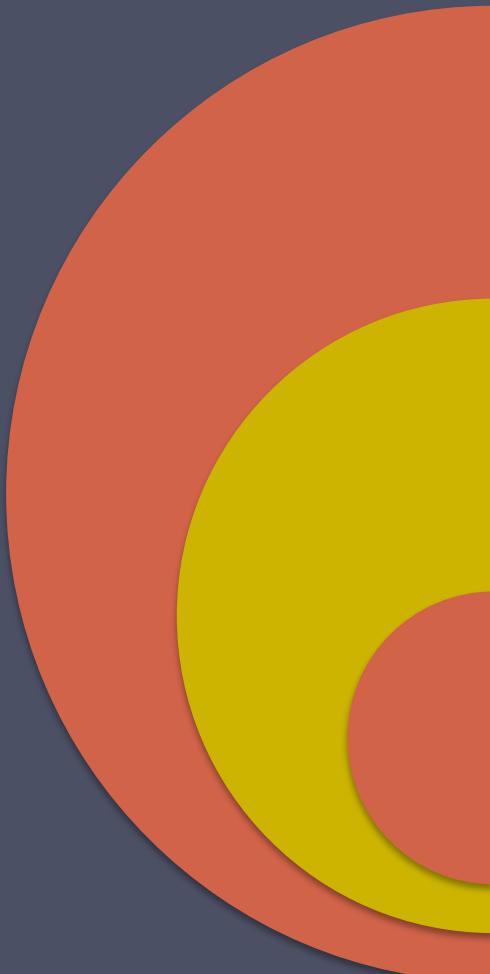
Block Cipher

- 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
- Encrypts plaintext one byte at a time
- Pseudorandom stream is one that is unpredictable without knowledge of the input key

Message Authentication



Protects against
active attacks

Verifies received
message is
authentic

Can use
conventional
encryption

- Contents have not been altered
- From authentic source
- Timely and in correct sequence

- Only sender & receiver share a key

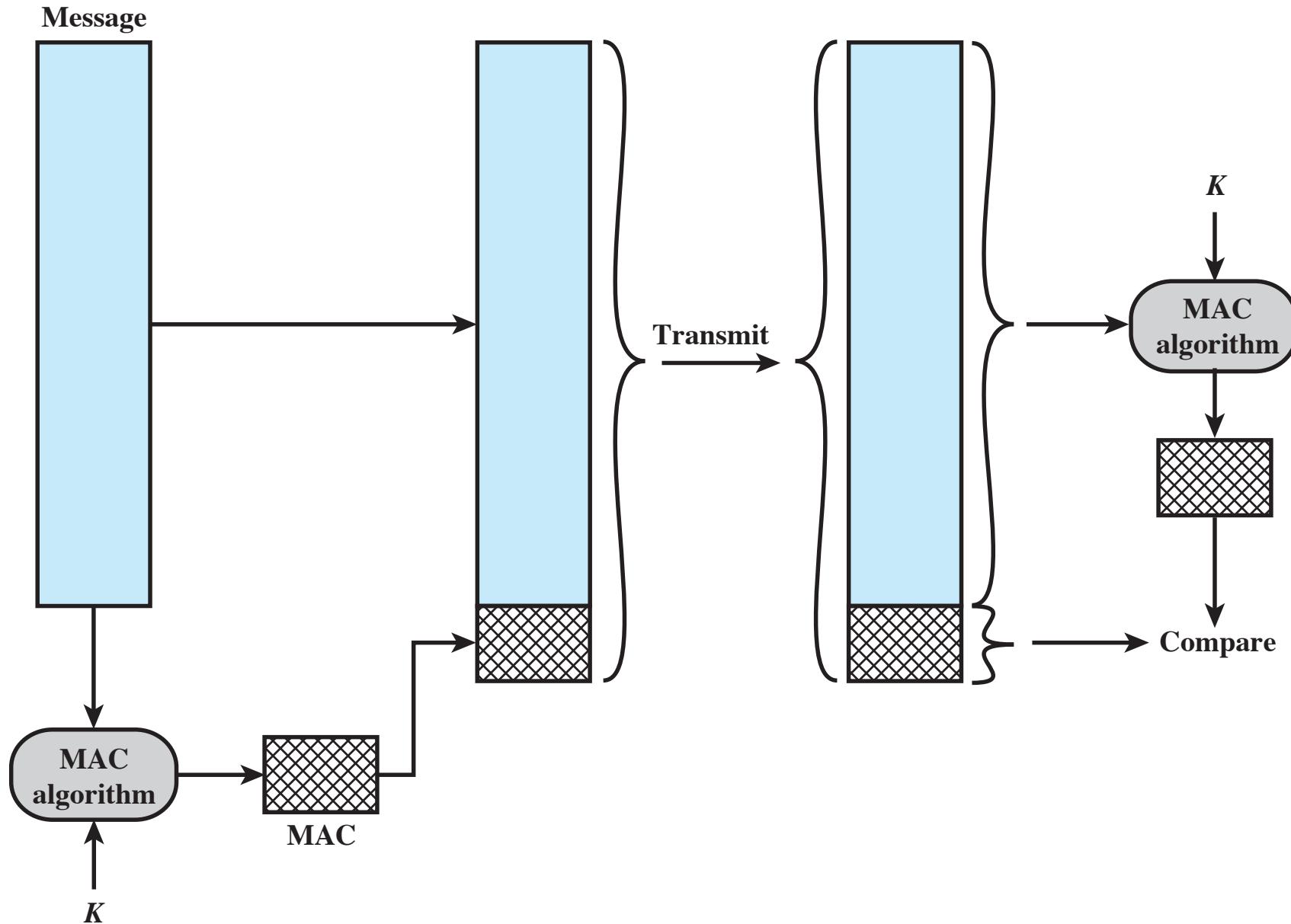
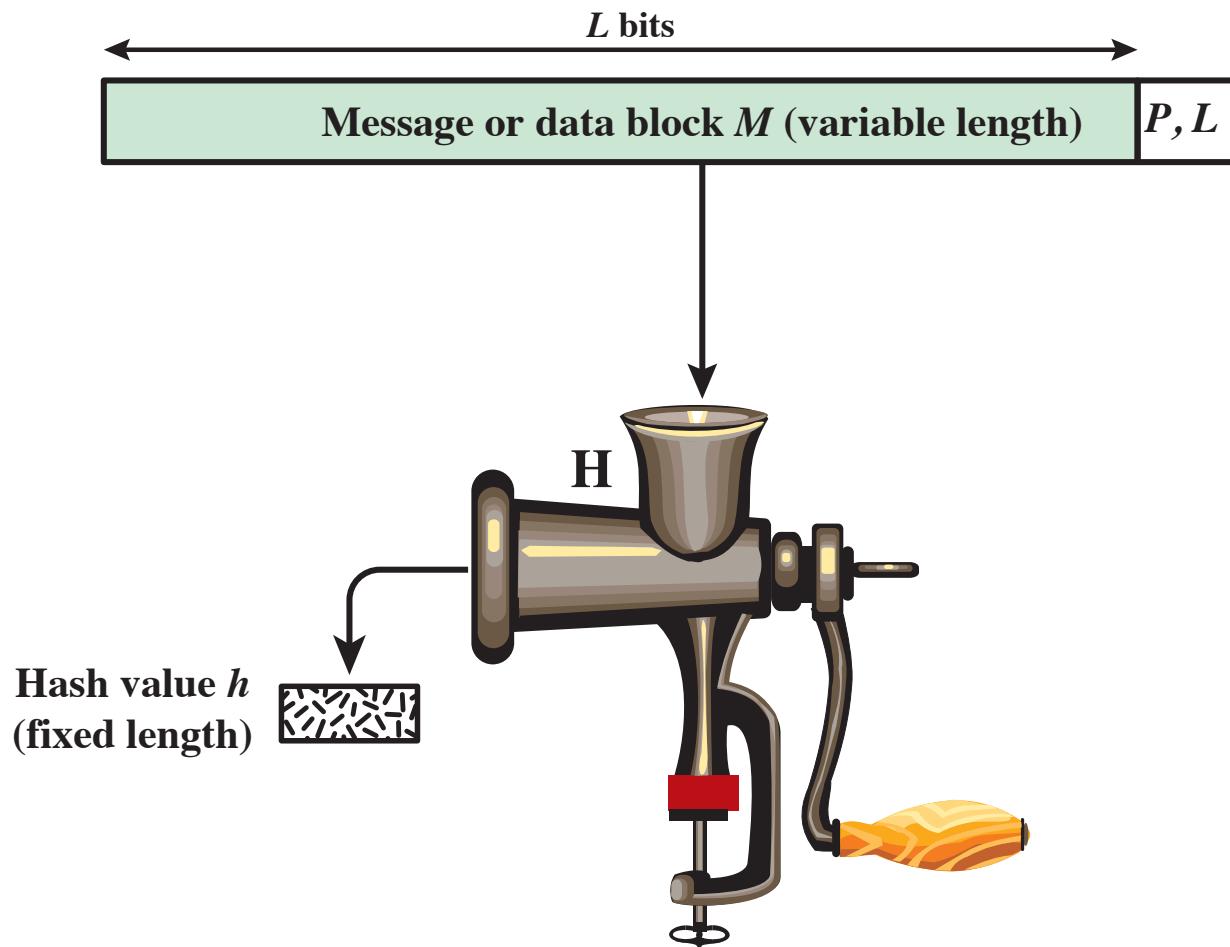


Figure 2.3 Message Authentication Using a Message Authentication Code (MAC).



P, L = padding plus length field

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

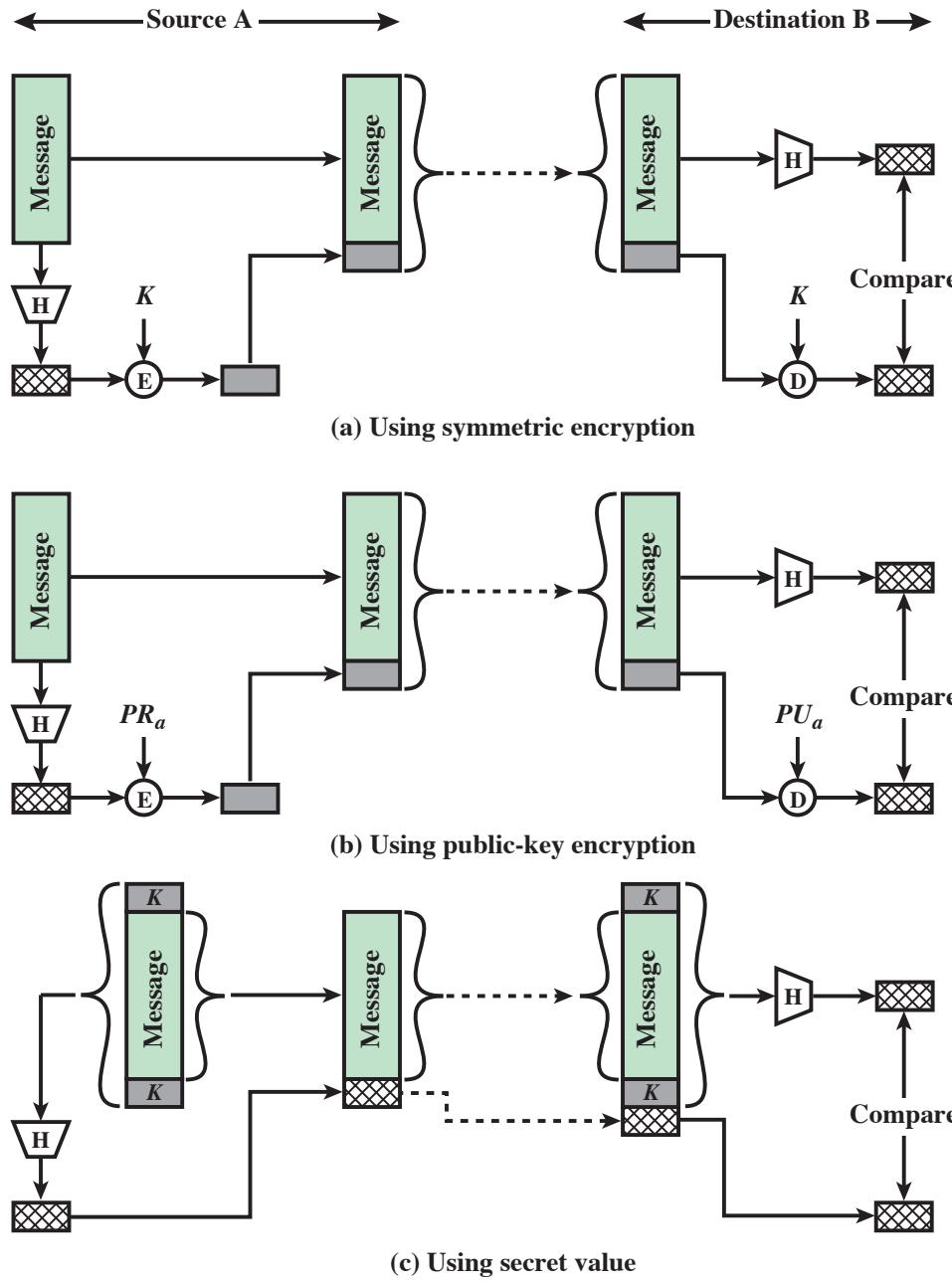
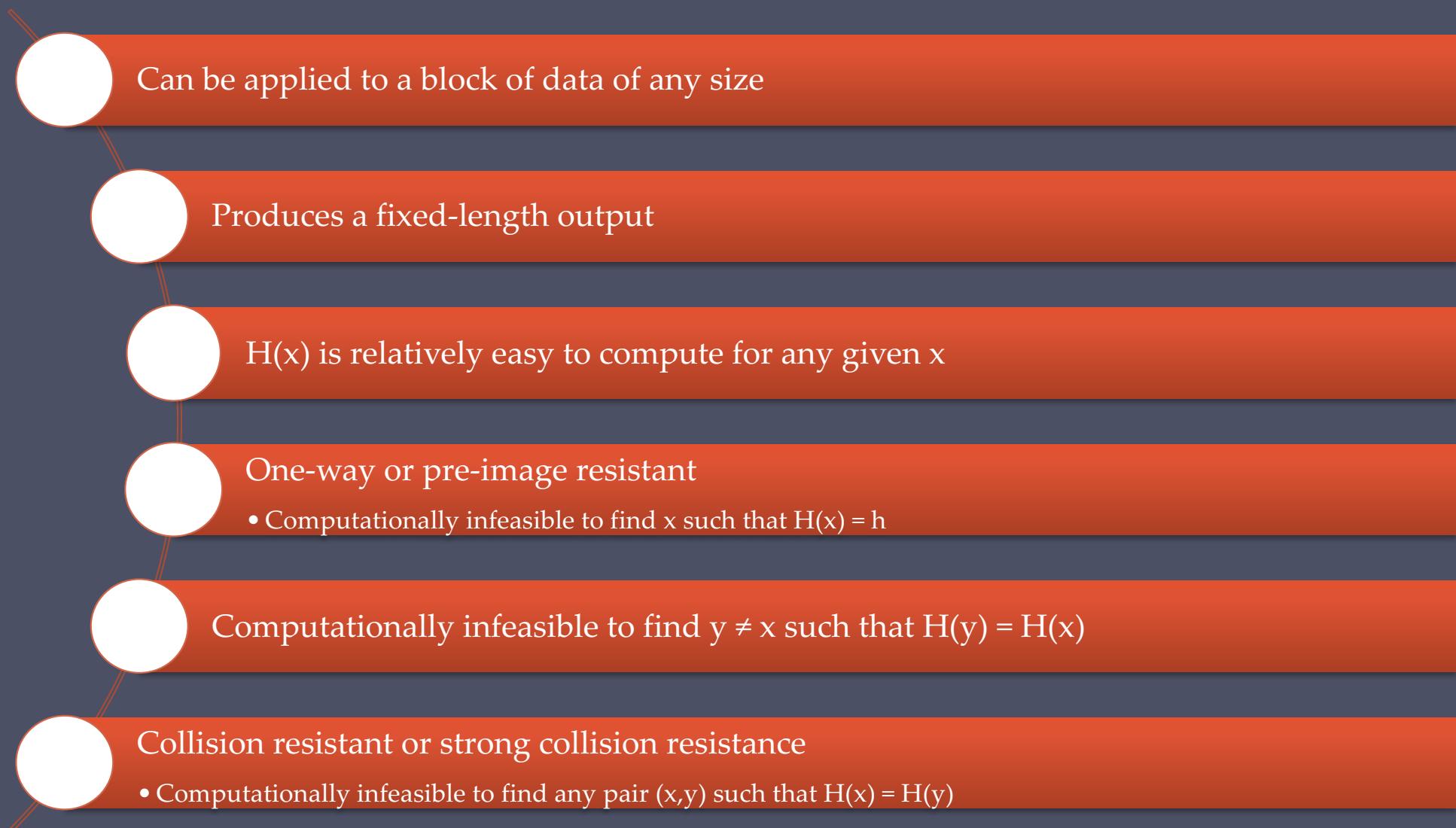


Figure 2.5 Message Authentication Using a One-Way Hash Function.

Hash Function Requirements



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

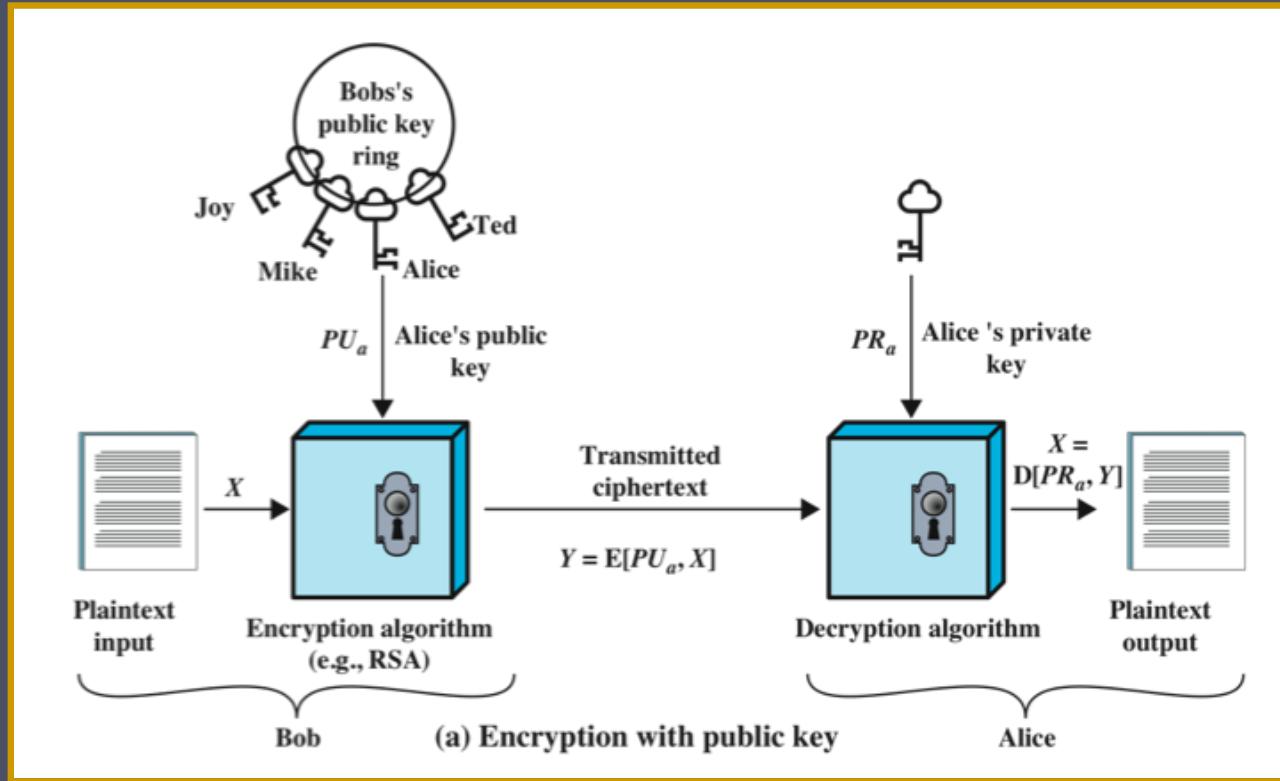
Based on mathematical functions

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

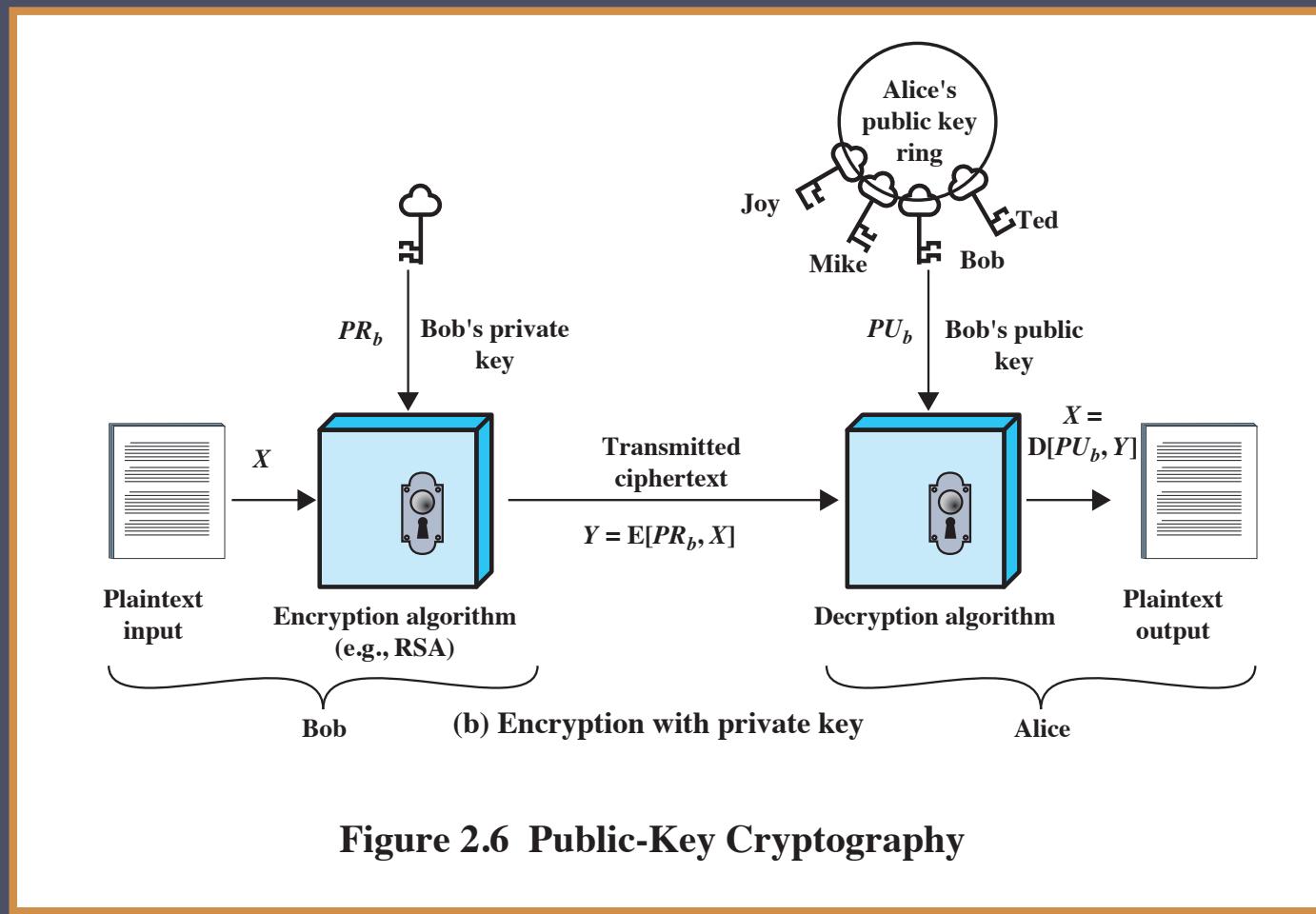


Figure 2.6 Public-Key Cryptography

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

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

Useful if either key can be used for each role

Computationally infeasible for opponent to otherwise recover original message

Computationally infeasible for opponent to determine private key from public key



Computationally easy to create key pairs

Computationally easy for sender knowing public key to encrypt messages

Computationally easy for receiver knowing private key to decrypt ciphertext

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

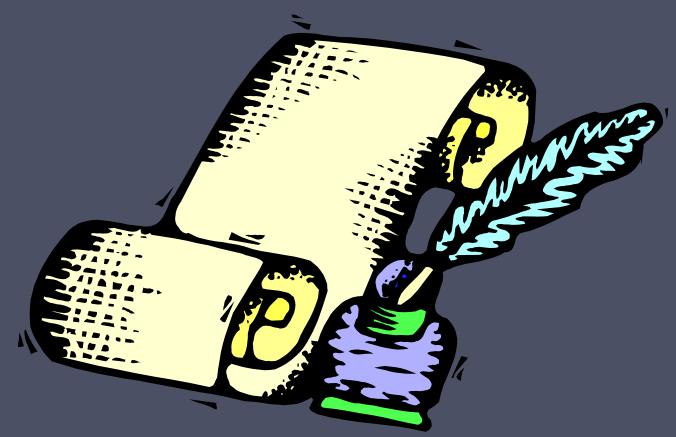
Cannot be used for encryption or key exchange

**Elliptic curve
cryptography
(ECC)**

Security like RSA, but with much smaller keys

Digital Signatures

- Used for authenticating both source and data integrity
- Created by encrypting hash code with private key
- Does not provide confidentiality
 - Even in the case of complete encryption
 - Message is safe from alteration but not eavesdropping



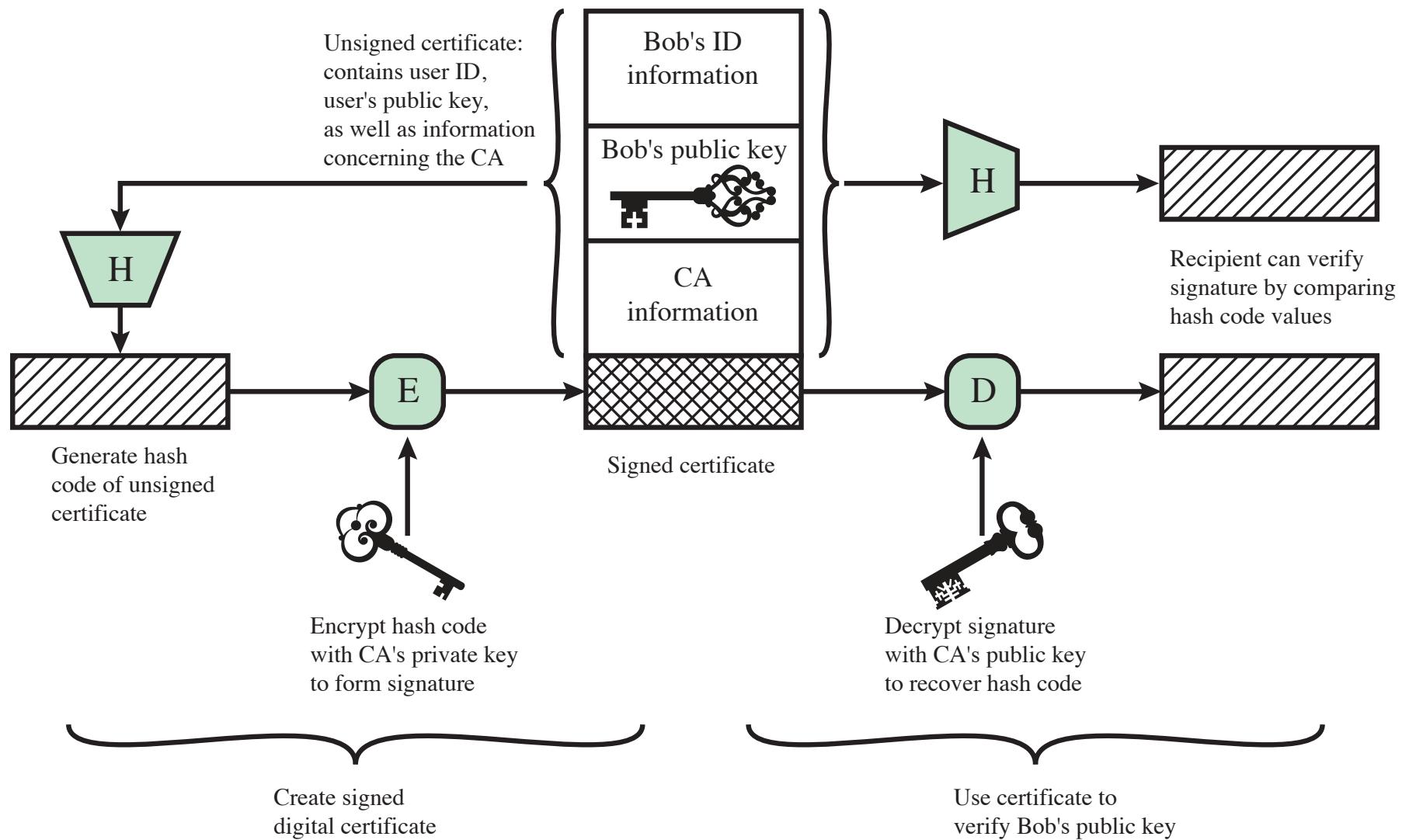


Figure 2.7 Public-Key Certificate Use

Digital Envelopes

- Protects a message without needing to first arrange for sender and receiver to have the same secret key
- Equates to the same thing as a sealed envelope containing an unsigned letter

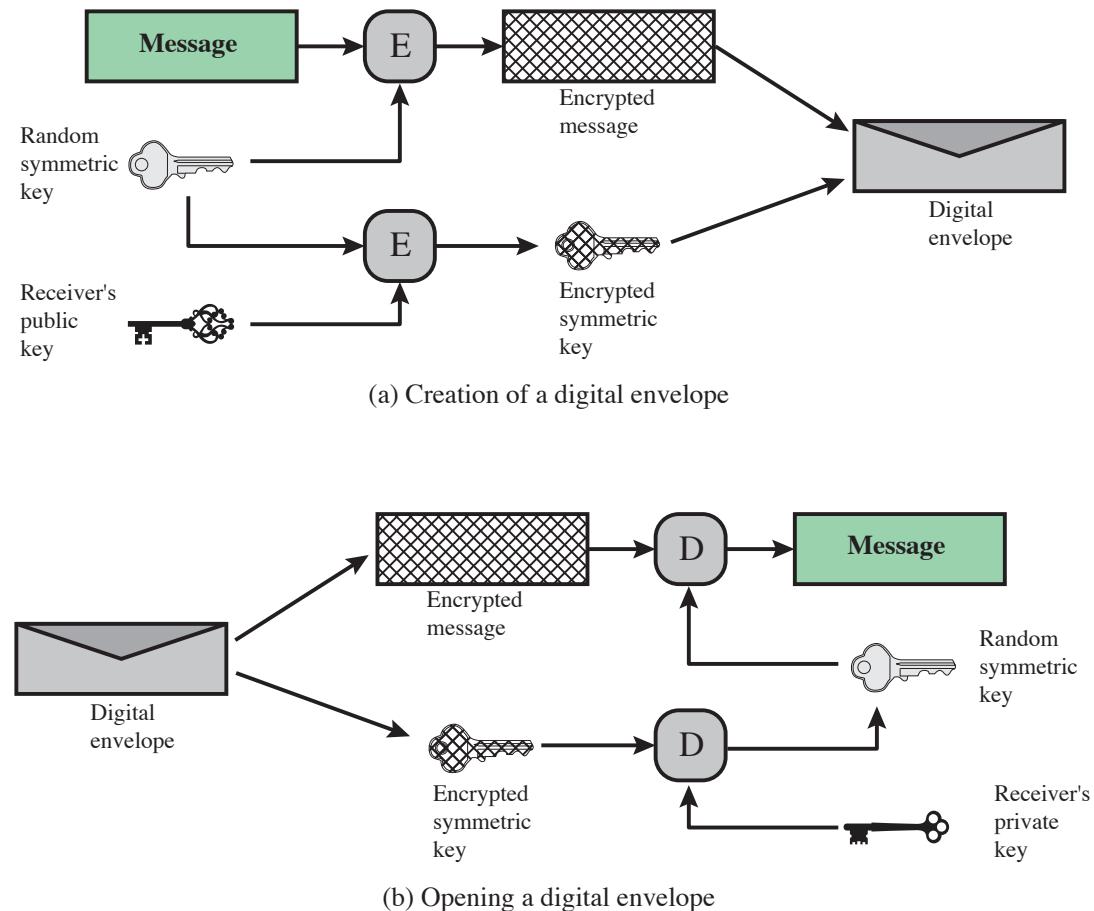


Figure 2.8 Digital Envelopes

Random Numbers



**Uses include
generation of:**

- Keys for public-key algorithms
- Stream key for symmetric stream cipher
- Symmetric key for use as a temporary session key or in creating a digital envelope
- Handshaking to prevent replay attacks
- Session key

Random Number Requirements

Randomness

- Criteria:
 - Uniform distribution
 - Frequency of occurrence of each of the numbers should be approximately the same
 - Independence
 - No one value in the sequence can be inferred from the others

Unpredictability

- Each number is statistically independent of other numbers in the sequence
- Opponent should not be able to predict future elements of the sequence on the basis of earlier elements

Random versus Pseudorandom

Cryptographic applications typically make use of algorithmic techniques for random number generation

- Algorithms are deterministic and therefore produce sequences of numbers that are not statistically random

Pseudorandom numbers are:

- Sequences produced that satisfy statistical randomness tests
- Likely to be predictable

True random number generator (TRNG):

- Uses a nondeterministic source to produce randomness
- Most operate by measuring unpredictable natural processes
 - e.g. radiation, gas discharge, leaky capacitors
- Increasingly provided on modern processors

Practical Application: Encryption of Stored Data

Common to encrypt transmitted data



Much less common for stored data

There is often little protection beyond domain authentication and operating system access controls

Data are archived for indefinite periods

Even though erased, until disk sectors are reused data are recoverable

Approaches to encrypt stored data:

Use a commercially available encryption package

Back-end appliance

Library based tape encryption

Background laptop/PC data encryption

Summary

- Confidentiality with symmetric encryption
 - Symmetric encryption
 - Symmetric block encryption algorithms
 - Stream ciphers
 - Message authentication and hash functions
 - Authentication using symmetric encryption
 - Message authentication without message encryption
 - Secure hash functions
 - Other applications of hash functions
 - Random and pseudorandom numbers
 - The use of random numbers
 - Random versus pseudorandom
 - Public-key encryption
 - Structure
 - Applications for public-key cryptosystems
 - Requirements for public-key cryptography
 - Asymmetric encryption algorithms
 - Digital signatures and key management
 - Digital signature
 - Public-key certificates
 - Symmetric key exchange using public-key encryption
 - Digital envelopes
- 