2. Cryptographic Tools

Prof. Seunghyun Park (sp@hansung.ac.kr)

Division of Computer Engineering

한성대학교 HANSUNG UNIVERSITY

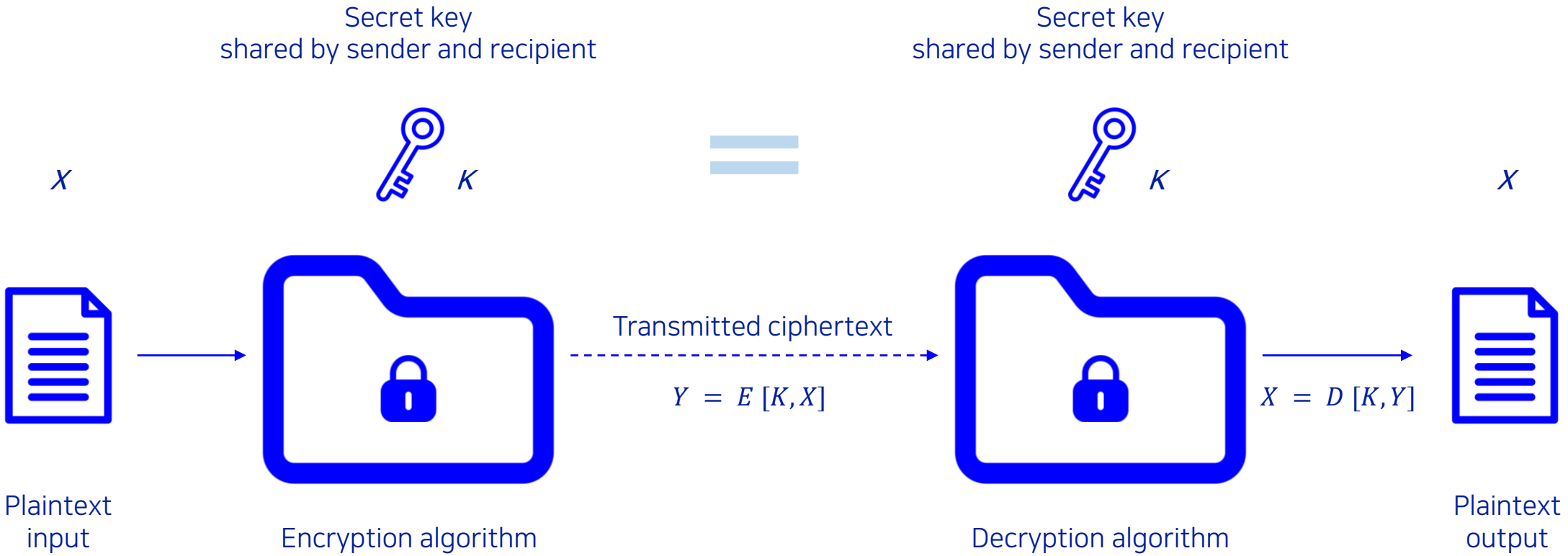
Objectives

- We are able to …
  - explain the basic operation of symmetric block encryption algorithms.
  - compare and contrast block encryption and stream encryption.
  - discuss the use of secure hash functions for message authentication.
  - list other applications of secure hash functions.
  - explain the basic operation of asymmetric block encryption algorithms.
  - present an overview of the digital signature mechanism and the concept of digital envelopes.
  - explain the significance of random and pseudorandom numbers in cryptography.

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

Symmetric encryption model



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

Comparison of 3 popular symmetric encryption algorithms

	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

Average time required for exhaustive key search

Key size (bits)	Cipher	# 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

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

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)

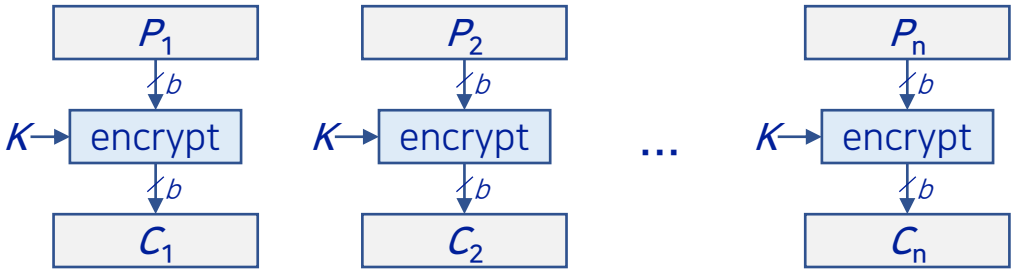
- Needed a replacement for 3DES
  - 3DES was not reasonable for long term use
- NIST called for proposals for a new AES in 1997
  - Should have a security strength equal to or better than 3DES
  - Significantly improved efficiency
  - Symmetric block cipher
  - 128-bit data and 128/192/256 bit keys
-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

Practical security issues

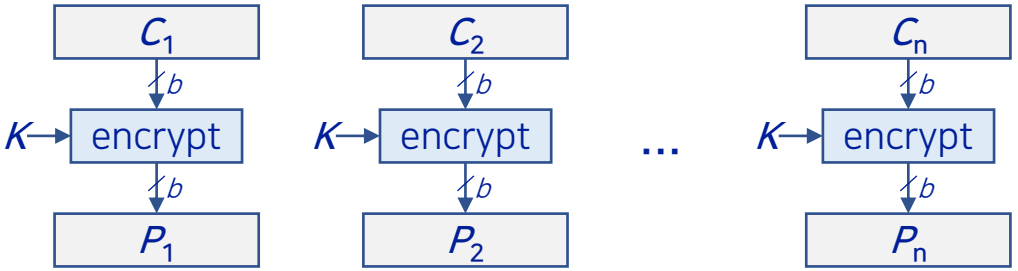
- Typically symmetric encryption is applied to a unit of data larger than a single 64-bit or 128-bit block
- Electronic codebook (ECB) mode is the simplest approach to multiple-block encryption
  - Each block of plaintext is encrypted using the same key
  - Cryptanalysts may be able to exploit regularities 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

Types of symmetric encryption

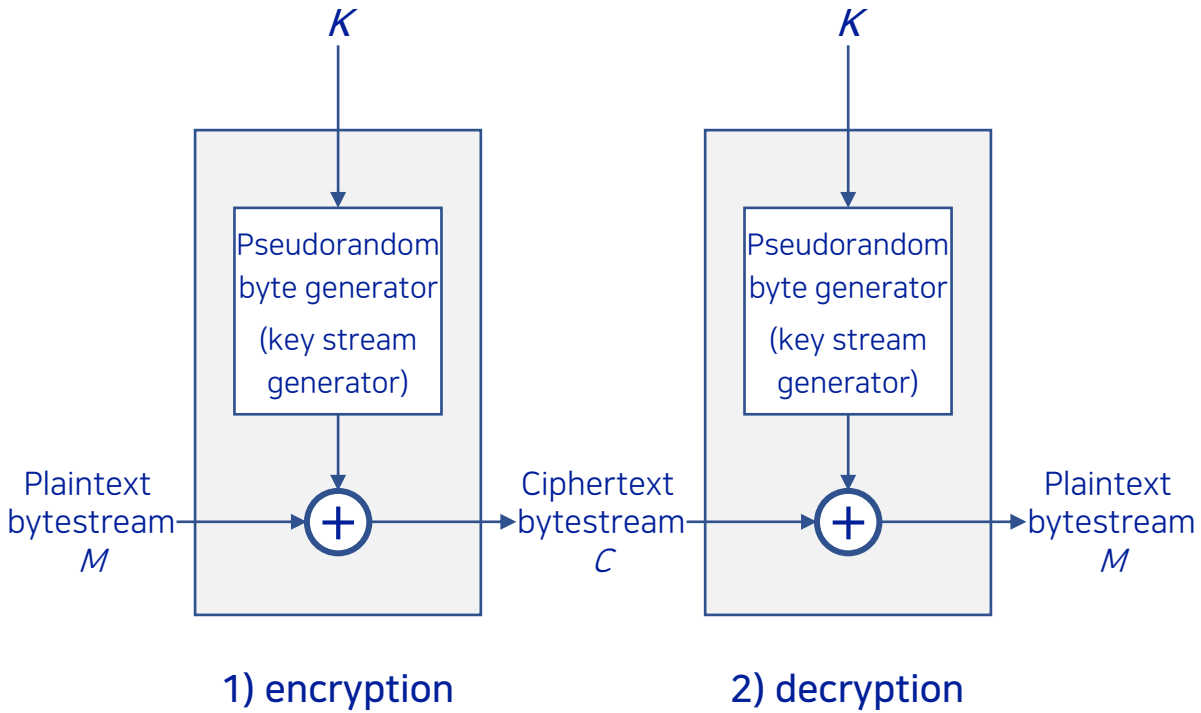
1) encryption



2) decryption



(a) Block cipher encryption (electronic codebook mode)



(b) Stream encryption

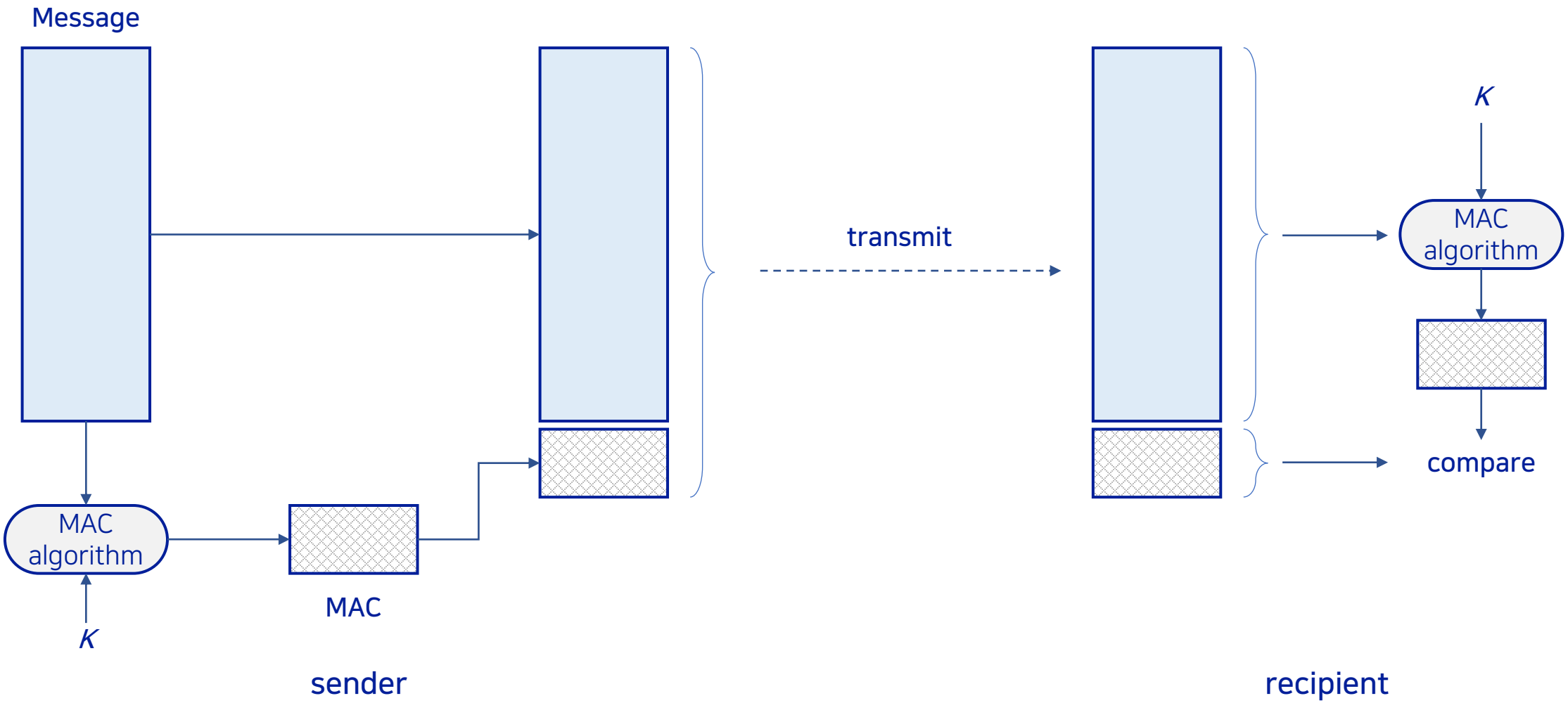
Block & stream ciphers

- 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

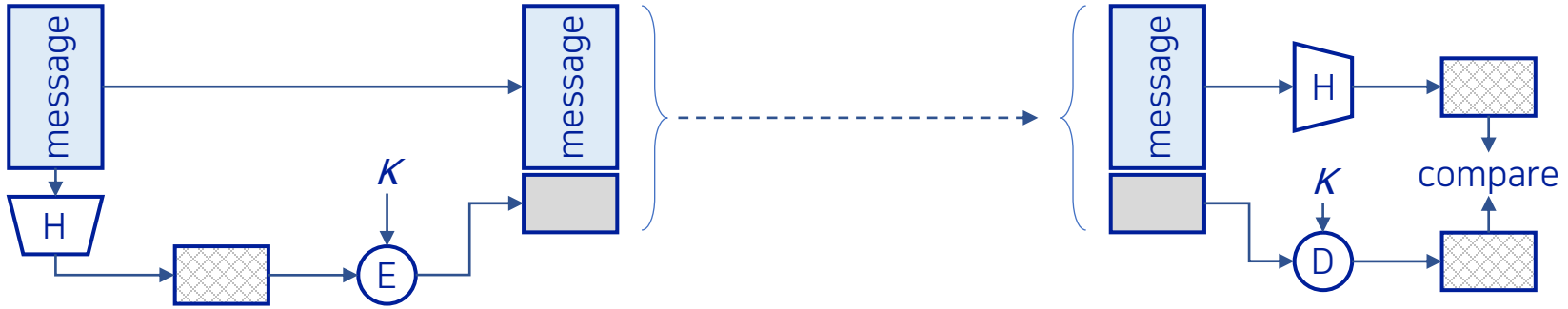
- Protects against active attacks
- Verifies received message is authentic
  - contents have not been altered
  - from authentic source
  - timely and in correct sequence
- Can use conventional encryption
  - Only sender and receiver share a key

Message authentication using a message authentication code

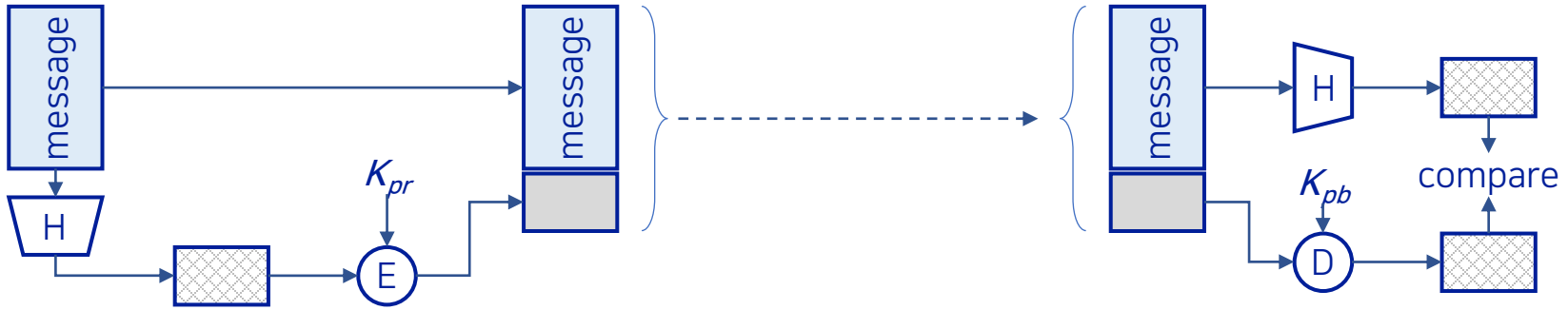


Message authentication using a one-way hash function

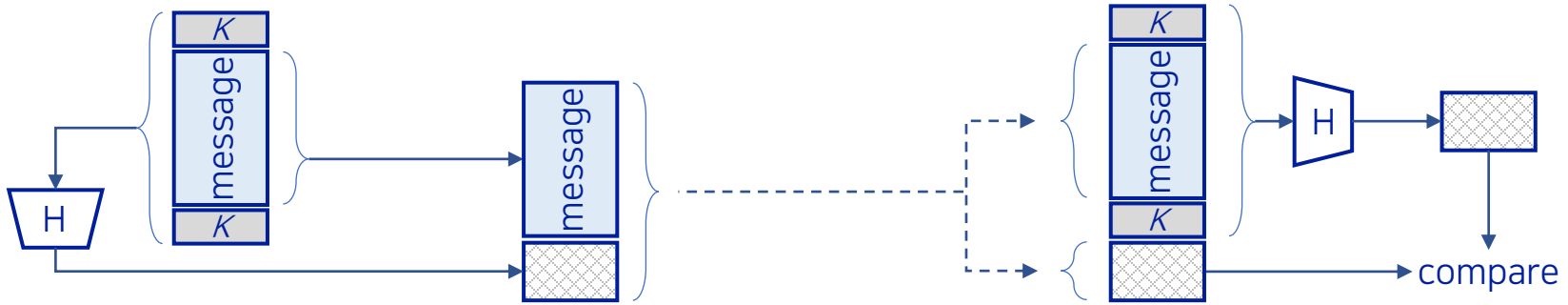
(a) Using symmetric encryption



(b) using public-key encryption

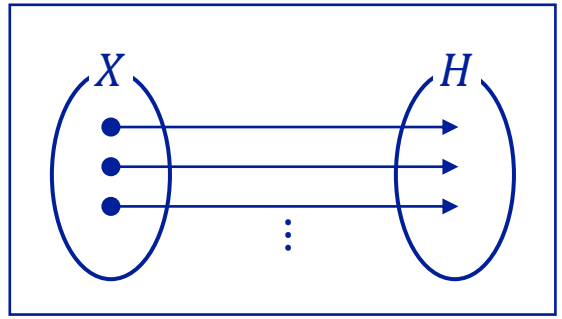


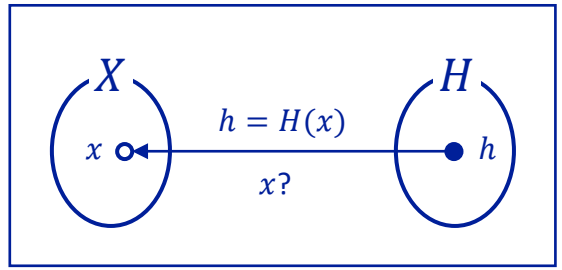
(c) using secret value

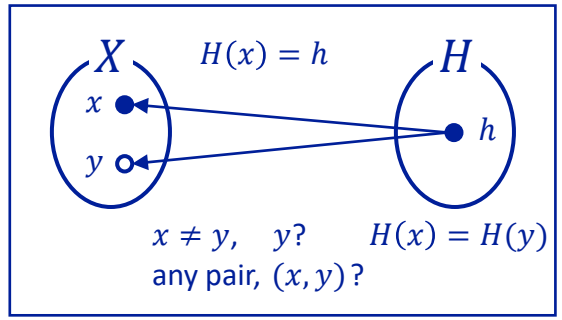


Hash function requirements

- Hash function requirements
  - 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 $H(x) = H(y)$







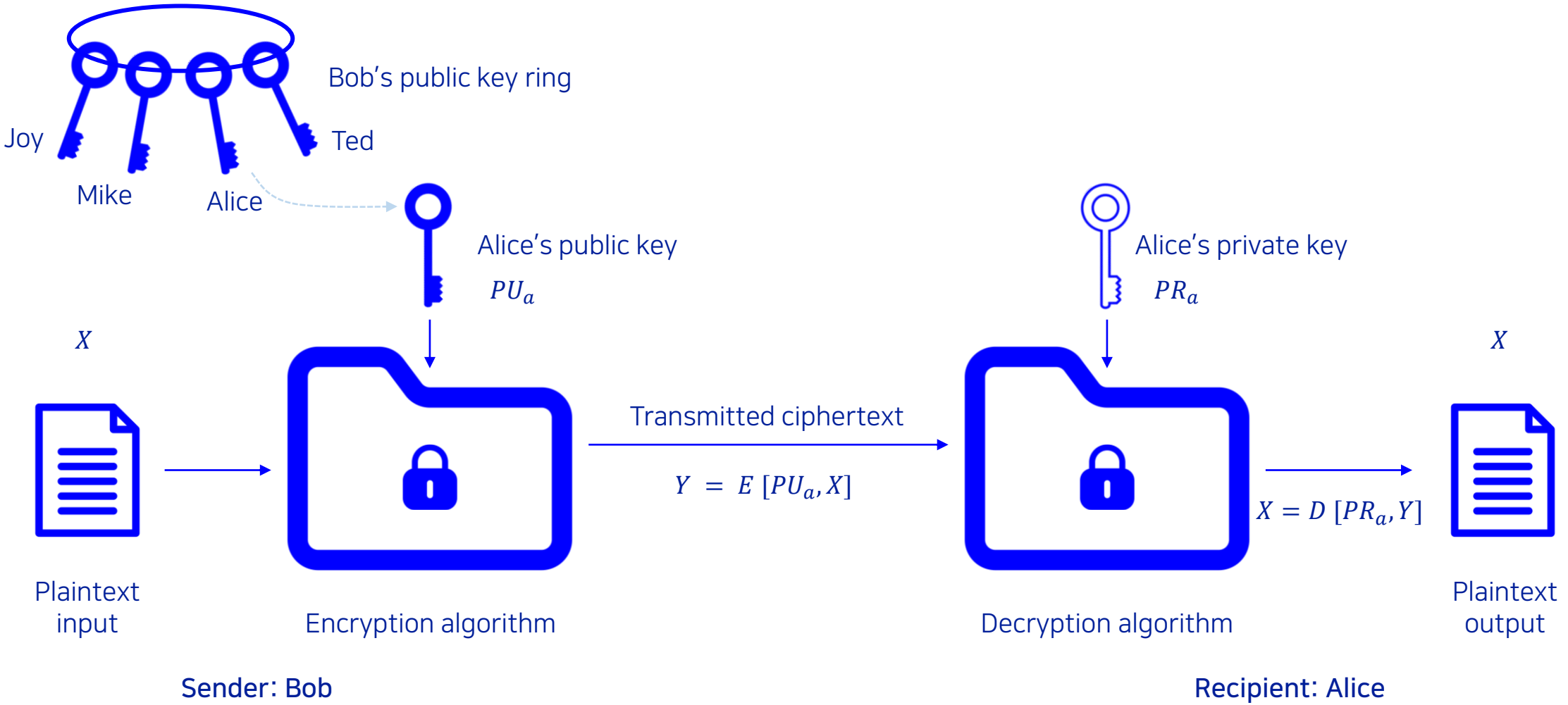
Security of hash functions

- Attacking 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
- Secure Hash Algorithm (SHA)
  - Most widely used hash algorithm
  - SHA-1: produces a hash value of 160 bits (FIPS 180, in 1993)
  - SHA-2 (SHA-256, SHA-384, SHA-512): hash value of 256, 384, and 512 bits (FIPS 180-2, in 2002)
  - SHA-3 (in 2015)

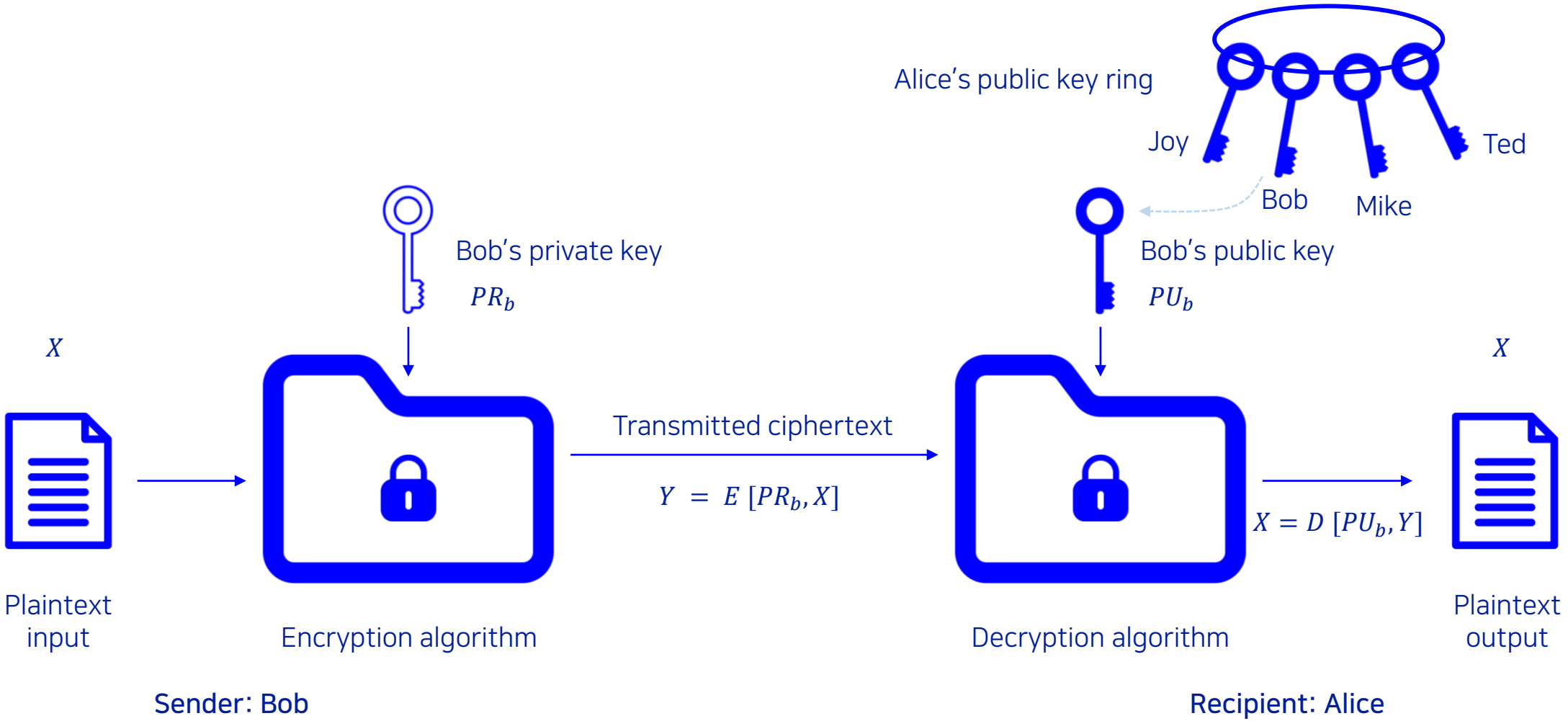
Public-key encryption structure

- 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

Encryption with public key



Encryption with private key



Requirements for public-key cryptosystems

- Computationally easy to create key pairs
- Computationally easy for sender knowing public key to encrypt messages
  - ciphertext $C = E(PU_b, M)$
- Computationally easy for receiver knowing private key to decrypt ciphertext
  - original message $M = D(PR_b, C) = D[PR_b, E(PU_b, M)]$
- Computationally infeasible for opponent to determine private key from public key
- Computationally infeasible for opponent to otherwise recover original message
- Useful if either key can be used for each role
  - $M = D[PU_b, E(PR_b, M)] = D[PR_b, E(PU_b, M)]$

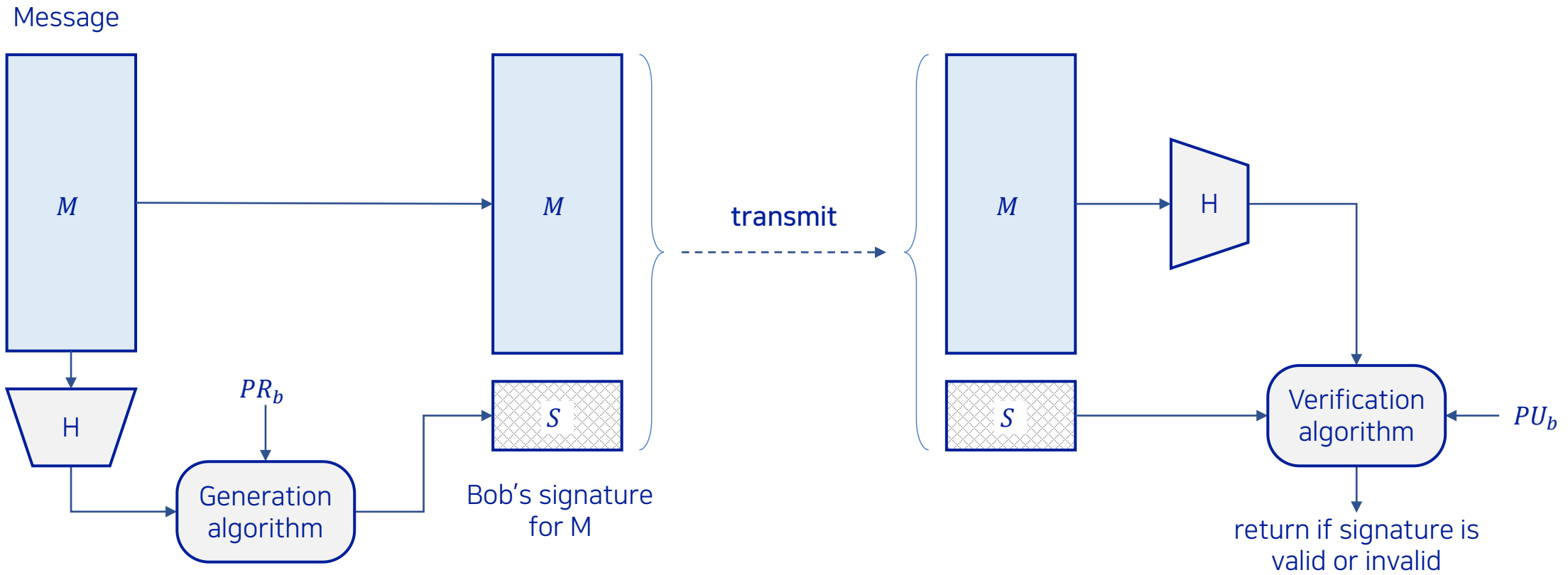
Asymmetric encryption algorithms

- 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

- NIST FIPS PUB 186-4 defines a digital signature as:
  - "The result of a cryptographic transformation of data that, when properly implemented, provides a mechanism for verifying origin authentication, data integrity and signatory non-repudiation."
- Thus, a digital signature is a data-dependent bit pattern, generated by an agent as a function of a file, message, or other form of data block
- FIPS 186-4 specifies the use of one of three digital signature algorithms:
  - Digital Signature Algorithm (DSA)
  - RSA Digital Signature Algorithm
  - Elliptic Curve Digital Signature Algorithm (ECDSA)

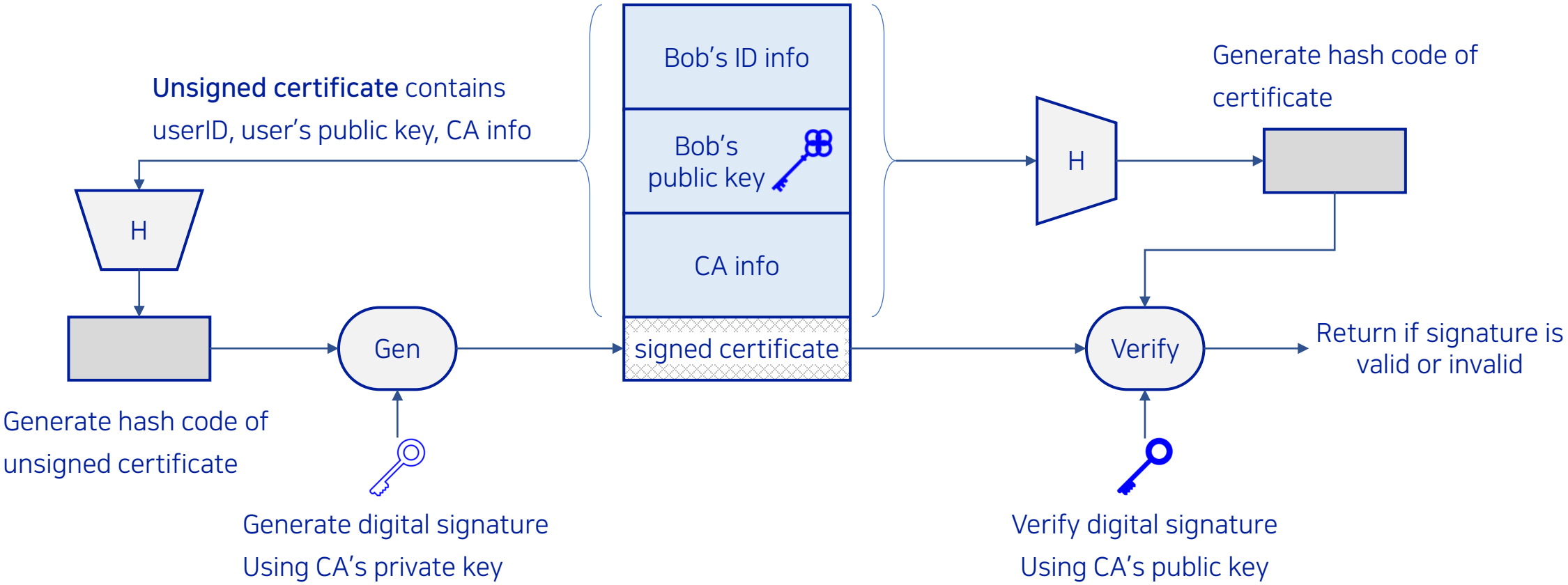
Digital signature process



Bob signs a message

Alice verifies the signature

Public-key certificate use

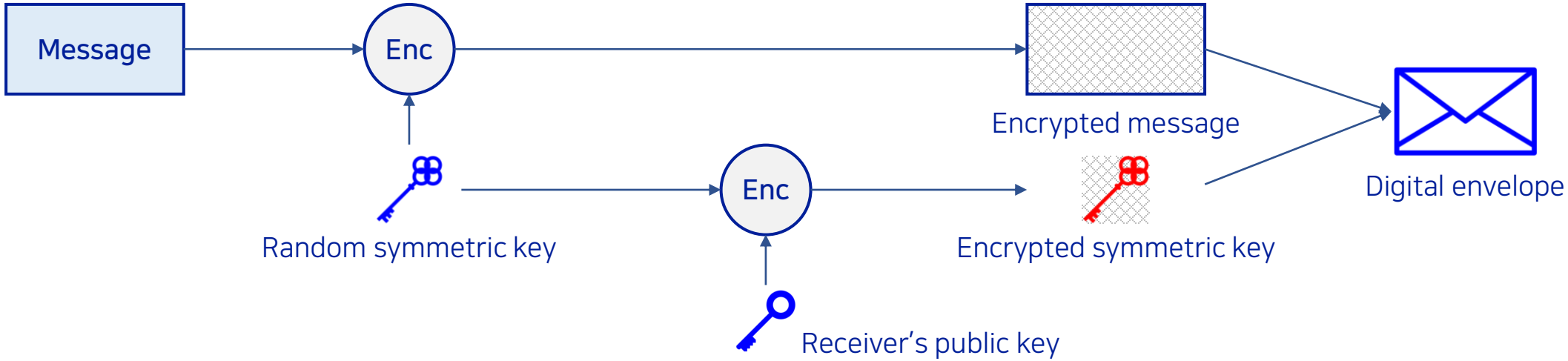


Create signed digital certificate

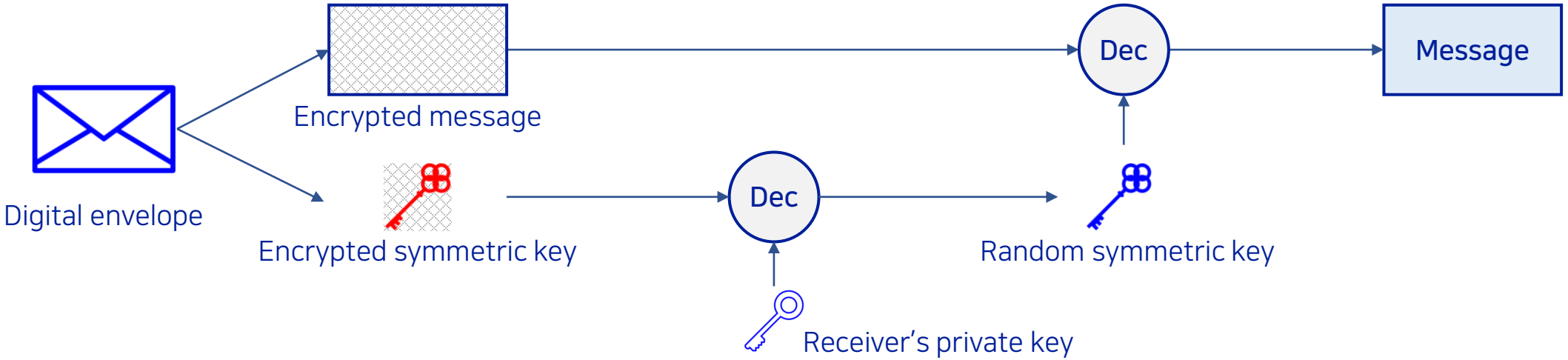
Use certificate to verify Bob's public key

Digital envelops

(a) Creation of a digital envelope



(b) Opening a digital envelope



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
  - 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 vs. 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

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
- 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
- Random and pseudorandom numbers