Cryptography and Cryptograhic Algorithms [LH 4]

Unit 2

Outlines

- Cryptography
- Data Encryption standard
- Symmetric Key Crytography (Block and Stream Ciphers)
- Asymmetric Key Cryptography
- Public Key Cryptography (RSA)
- Message Digest 5
- Hash Function
- Message Authentication Code (MAC)

Cryptography

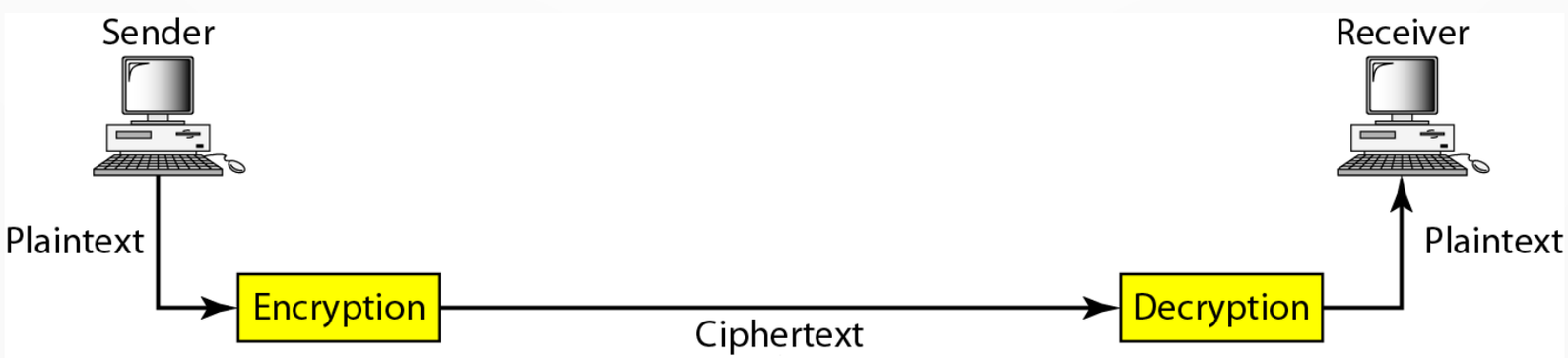
- Cryptography, a word with Greek origins, means "secret writing." and is the art and science of concealing meaning.
- The Concise Oxford English Dictionary (9th ed.) defines cryptography as "*the art of writing or solving codes*." This is historically accurate, but does not capture the current breadth of the field or its modern scientific foundations. The definition focuses solely on the codes that have been used for centuries to enable secret communication.
- But cryptography nowadays encompasses much more than this: it deals with mechanisms for ensuring integrity, techniques for exchanging secret keys, protocols for authenticating users, electronic voting, cryptocurrency, and more.
- Modern cryptography involves *the study of mathematical techniques for securing digital information, systems, and distributed computations against adversarial attacks.*
- It involves three distinct mechanisms: symmetric-key encipherment, asymmetric-key encipherment, and hashing.

Cryptography

- It is the techniques of converting ordinary plain text into unintelligible text and vice-versa.
- It is the practice and study of techniques for secure communication in the presence of third parties.
- Techniques used for deciphering message without any knowledge of the enciphering details fall into the area of **cryptanalysis**. Cryptanalysis is what the layperson calls "*breaking the code*".
- The area of cryptography and cryptanalysis together are called **cryptology**.

Cryptography

- Before message is sent by the sender to the network , the message the user entered (plain-text) will be encrypted (converting plain-text to cipher-text) and after receiving the cipher-text will be decrypted (converting cipher-text to plain-text) and used by receiver.
- Encryption and decryption algorithms are referred as ***ciphers***.



Cryptography basic terminology

- **Plain text**: Original message fed to encryption algorithm; readable.
- **Encryption algorithm**: changes plain text to coded ciphertext by various substitution and transformation methods. The process of converting plaintext to ciphertext is called *enciphering* or *encryption*.
- **key**: input to the encryption algorithm. Known to sender and receiver only.
- **Ciphertext**: coded/scrambled output message by the algorithm. Different secret key applied on plain text produces different ciphertext.
- **Decryption Algorithm**: the encryption algorithm that run in reverse i.e. takes ciphertext and key to produce the original transmitted plain text. The process is known as *deciphering* or *decryption*.

Encryption and Decryption

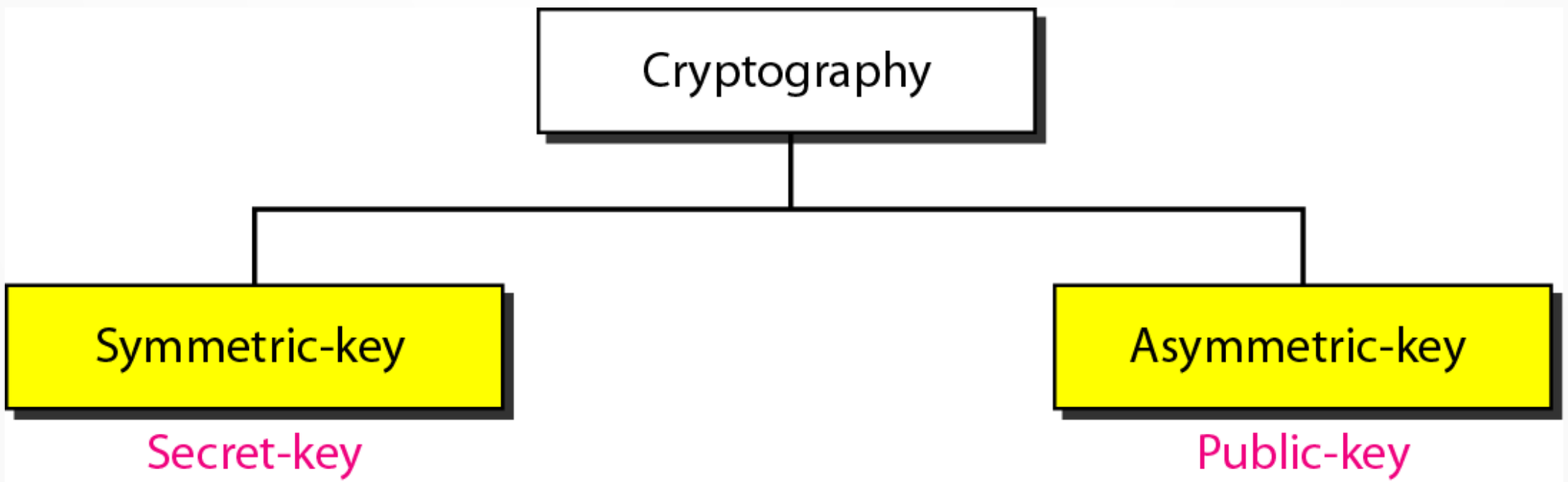
Encryption

- Encryption is the process of encoding a message or information in such a way that only authorized parties can access it and those who are not authorized Cannot.
- Encryption does not itself prevent interference, but denies the intelligible content to a would-be interceptor.

Decryption

- Decryption is the process of taking encoded or encrypted text or other data
- and converting it back into text that you or the computer can read and understand (original form).
- It is generally a reverse process of encryption. It decodes the encrypted information so that an authorized user can only decrypt the data
- because decryption requires a secret key or password

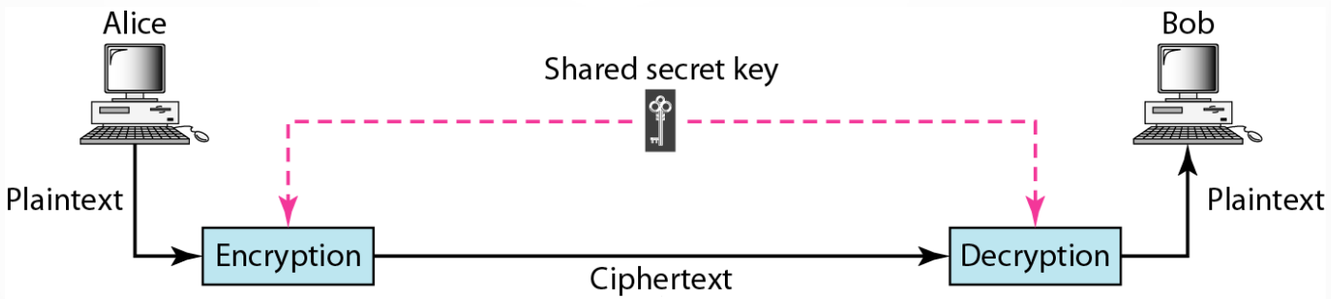
Cryptography Category

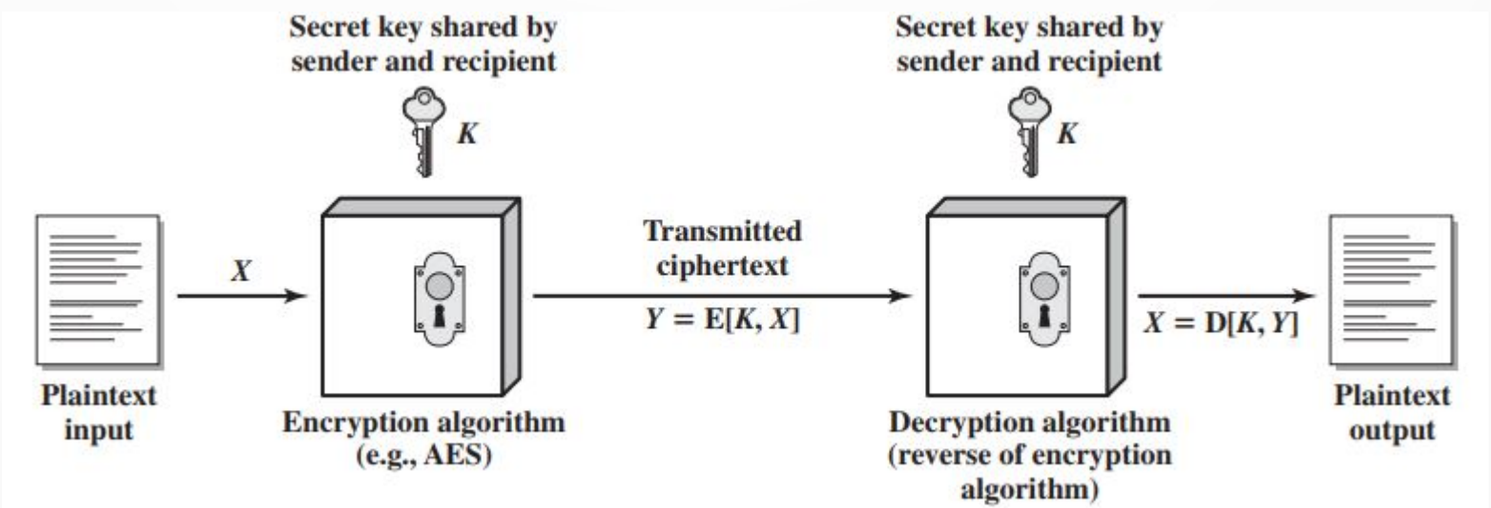


Symmetric Key

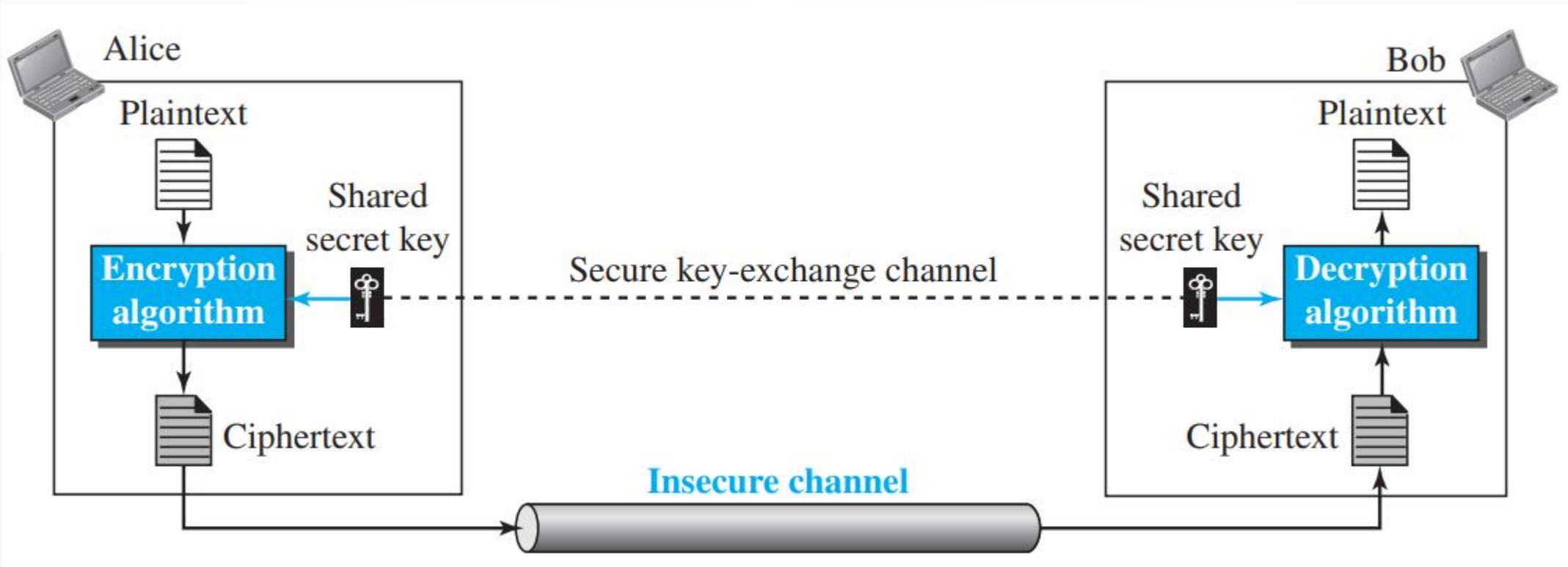
- Symmetric encryption (also called *classical cryptosystems*), also referred to as *conventional encryption*, *secret-key*, or single-key encryption, was the only type of encryption in use prior to the development of public-key encryption in the late 1970s.
- Same key is shared by sender (for encryption) and receiver (for decryption).
- Ciphertext is generated either by *substitution* or *transformation* method by encryption algorithm.
- A symmetric encryption has five ingredients:
  - Plaintext
  - Encryption algorithm
  - Secret key
  - Ciphertext
  - Decryption algorithm

Symmetric Key





Symmetric Key



Symmetric Key

There are two requirements for secure use of symmetric encryption:

1. We need a strong algorithm.
   - At minimum, an operator who knows the algorithm and has access to one or more ciphertext would be unable to decipher the ciphertext.
2. Sender and receiver must have obtained copies of the secret key in a secure fashion and must keep the key secure.

Symmetric Cryptosystem/Classical Cryptosystem

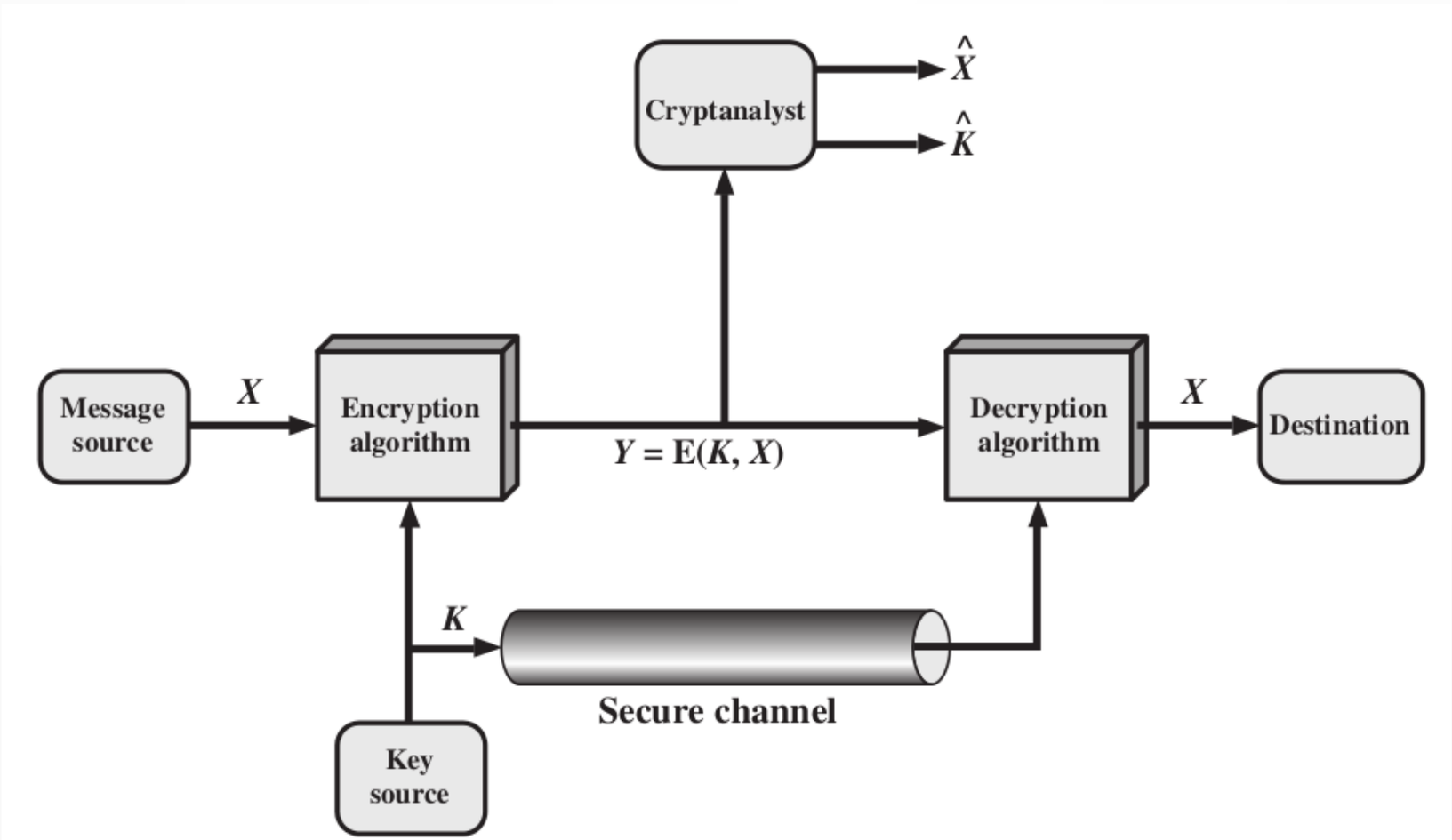


Fig: Model of Symmetric Cryptosystem

Classical ciphers

- There are two basic types of classical ciphers:
  i. transposition ciphers and
  ii. substitution ciphers.

Transposition Ciphers

- A transposition cipher rearranges the characters in the plaintext to form the cipher text. These letters are not changed.
- For example: "***HELLO WORLD***" could be written as

  HLOOL

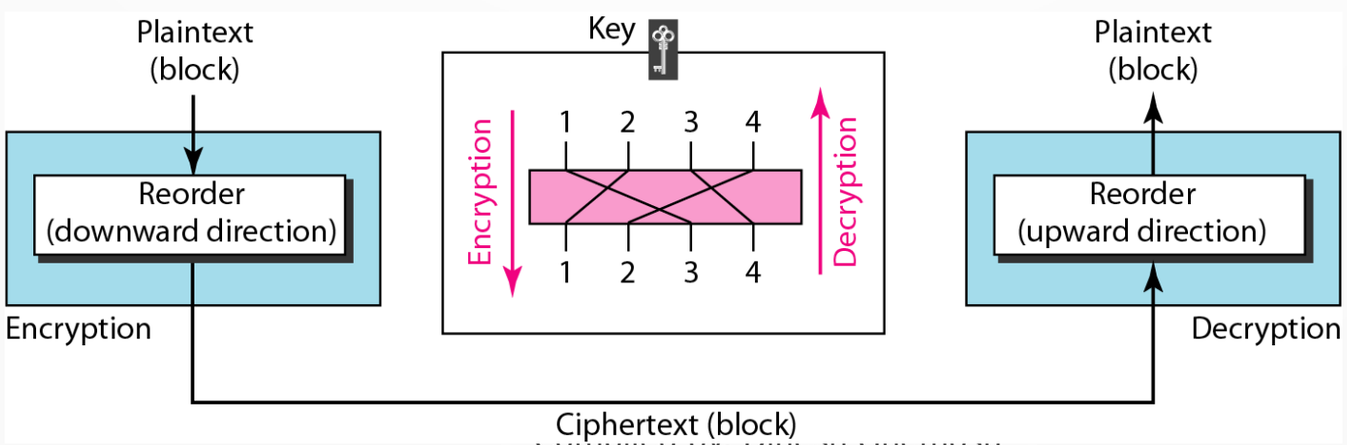
  ELWRD

  resulting in the ciphtertext "*HLOOLELWRD*"

  A transposition cipher is a permutation cipher. It rearranges the given information without modifying it.

Transformation Cipher

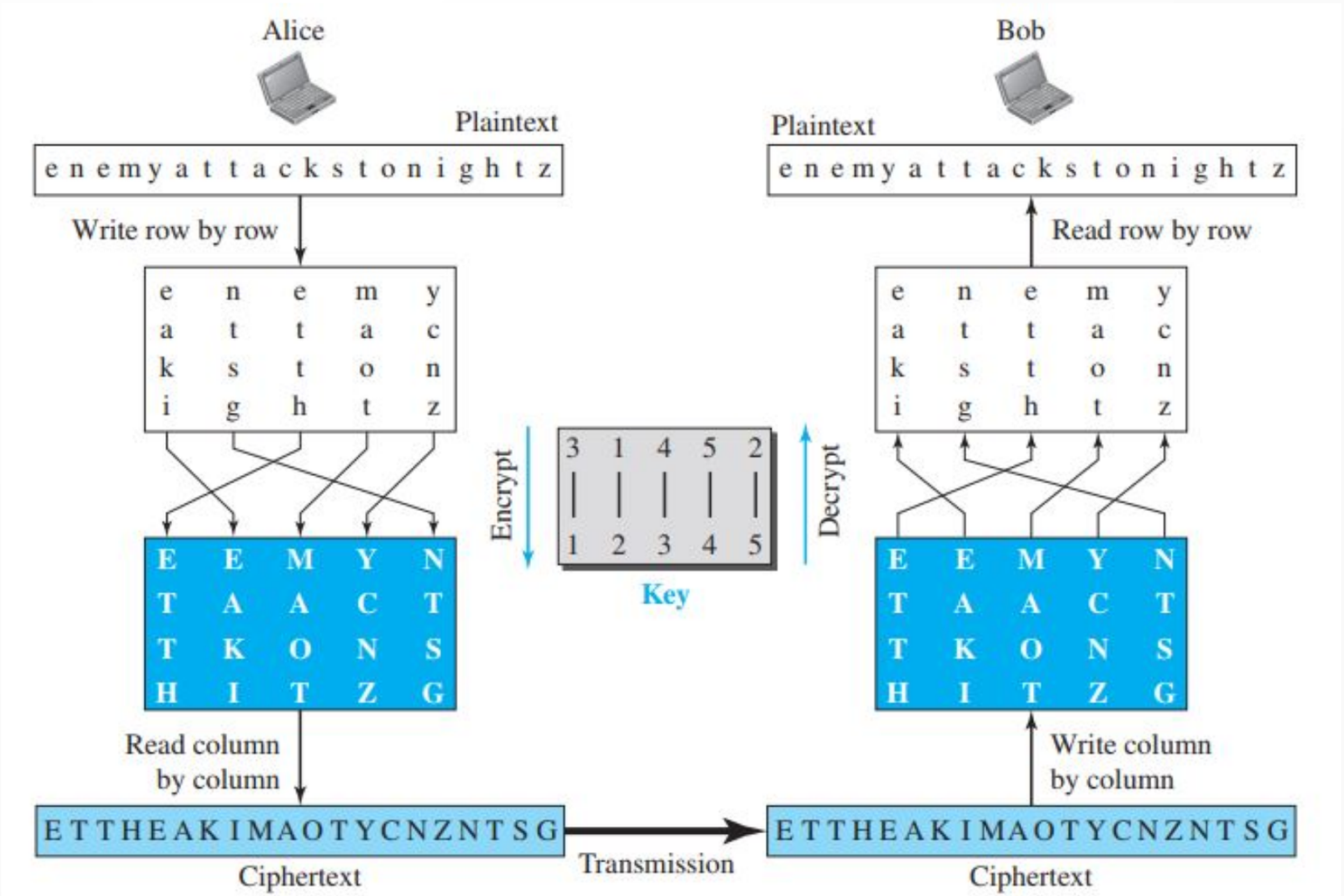
- A transposition cipher does not substitute one symbol for another; instead it changes the location of the symbols.
- A symbol in the first position of the plaintext may appear in the tenth position of the ciphertext.
- A symbol in the eighth position in the plaintext may appear in the first position of the ciphertext.
- In other words, a transposition cipher reorders (transposes) the symbols.



Transposition Cipher

For example:

Suppose Alice wants to secretly send the message “Enemy attacks tonight” to Bob. The encryption and decryption is shown in Figure. Note that we added an extra character (z) to the end of the message to make the number of characters a multiple of 5.



Substitution Cipher

- A substitution cipher changes characters in the plaintext to produce the ciphertext.
- For example: Caesar cipher, Vigenere Cipher, One-time pad are example of substitution cipher.

Caesar Cipher

- Substitution cipher
- Shift cipher
- Earliest known and simplest substitution scheme developed by Julis Ceasar.
- Replaces each letter of the alphabet with the letter standing three places further down the alphabet.

  plain: meet me after the toga party

  cipher: PHHW PH DIWHU WKH WRJD SDUWB
- A Caesar cipher is susceptible to a statistical ciphertext-only attack

Data Encryption Standard

- The Data Encryption Standard (DES) works by using the same key to encrypt and decrypt a message, so both the sender and the receiver must know and use the same private key.
- DES is an outdated symmetric-key method of data encryption.
- DES has been upgraded by the more secure Advanced Encryption Standard (AES) algorithm.

Data Encryption Standard

- Originally designed by researchers at IBM in the early 1970s, DES was adopted by the U.S. government as an official Federal Information Processing Standard (FIPS) in 1977 for the encryption of commercial and sensitive yet unclassified government computer data.
- It was the first encryption algorithm approved by the U.S. government for public disclosure.

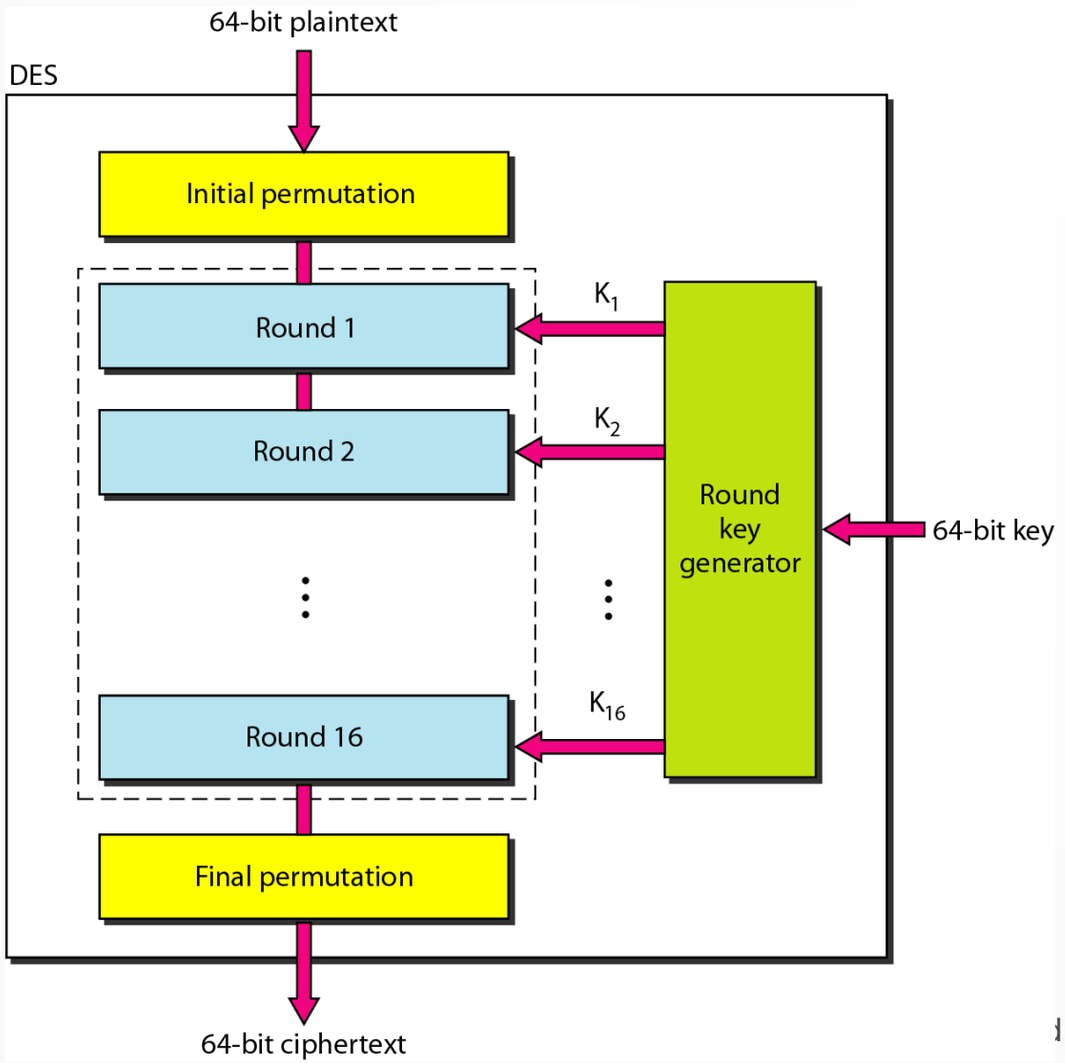
Data Encryption Standard

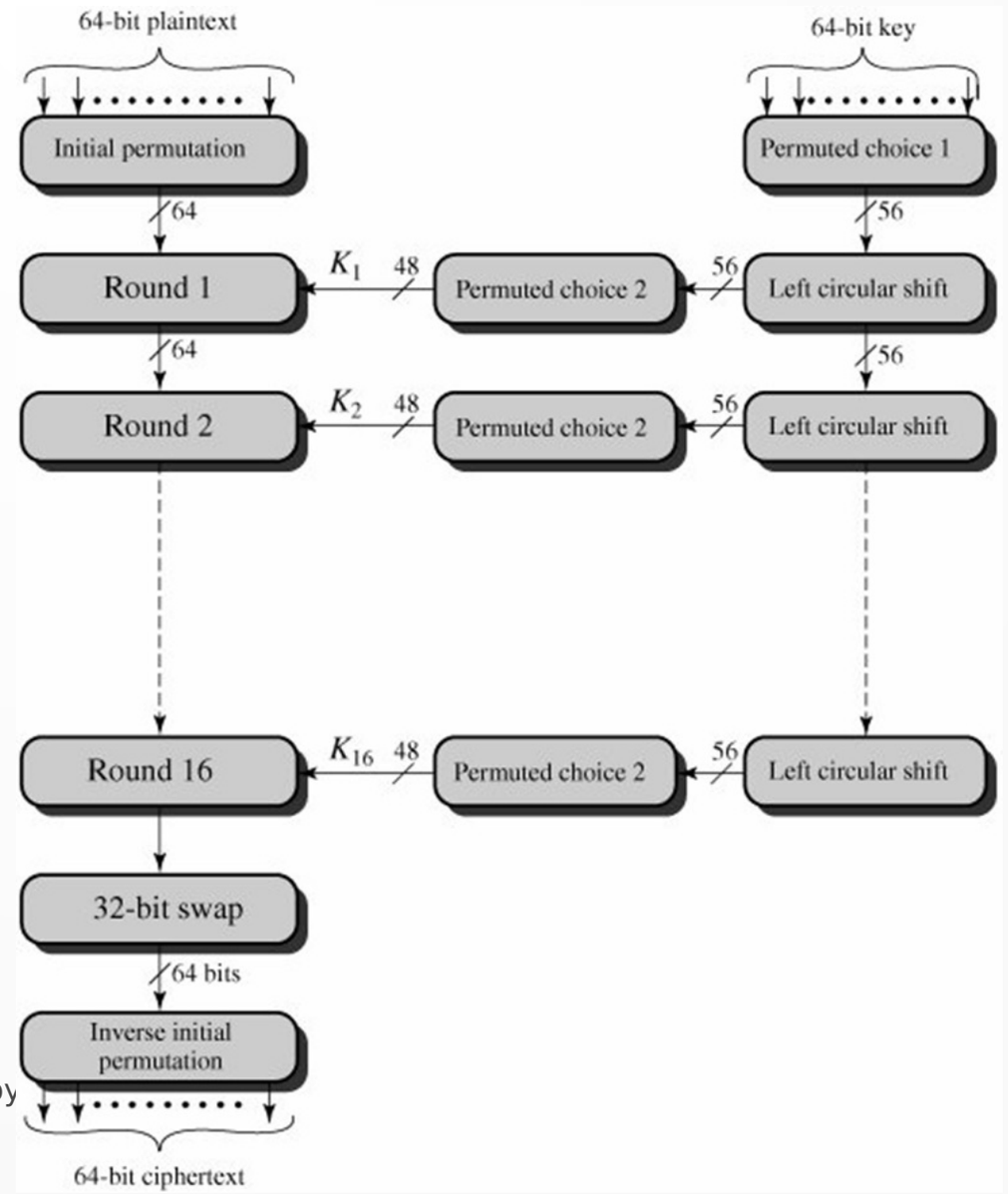
- The Data Encryption Standard is a block cipher, meaning a cryptographic key and algorithm are applied to a block of data simultaneously rather than one bit at a time.
- To encrypt a plaintext message, DES groups it into 64-bit blocks. Each block is enciphered using the secret key into a 64-bit ciphertext by means transposition and substitution.

Data Encryption Standard

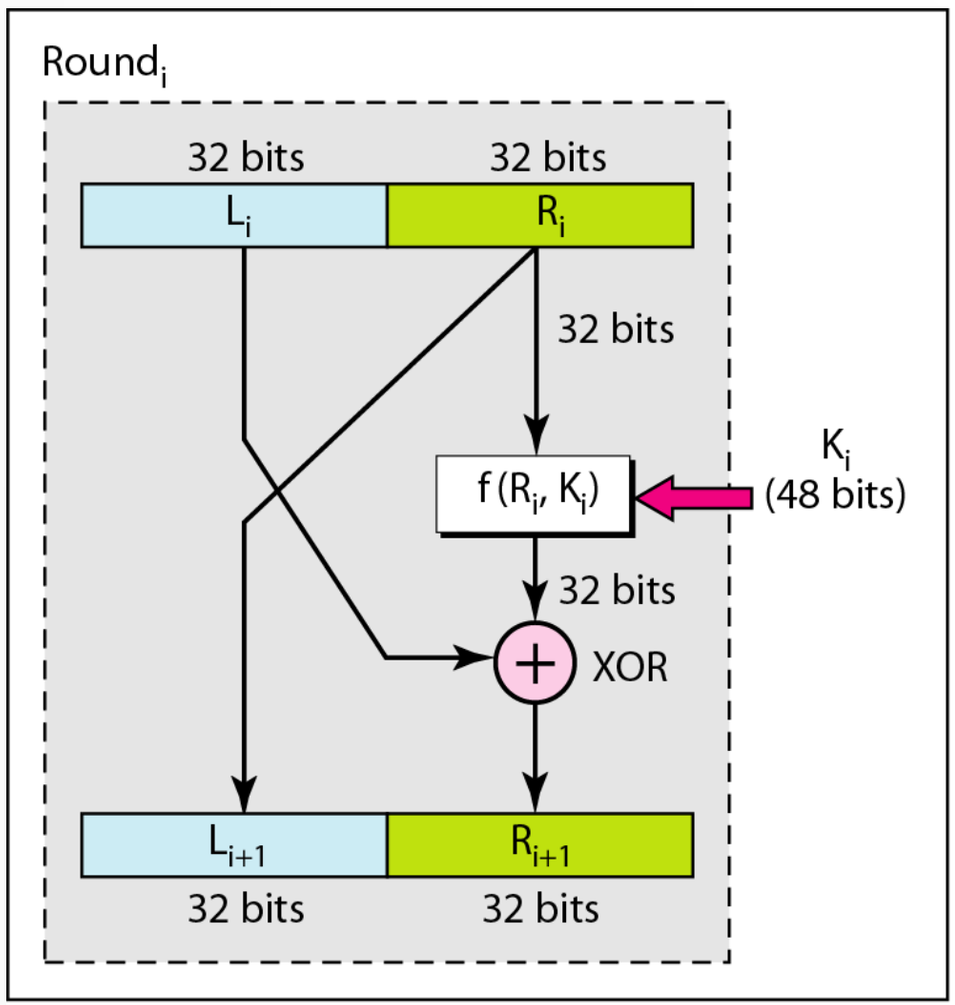
- The process involves 16 rounds and encrypting blocks individually or making each cipher block is dependent on all the previous blocks.
- DES has an effective key length of 56 bits, since 8 of the 64 bits of the key are not used by the encryption algorithm (function as check bits only).
- The check bits or parity bits are used to check if the key was indeed correctly retrieved.

Data Encryption Standard

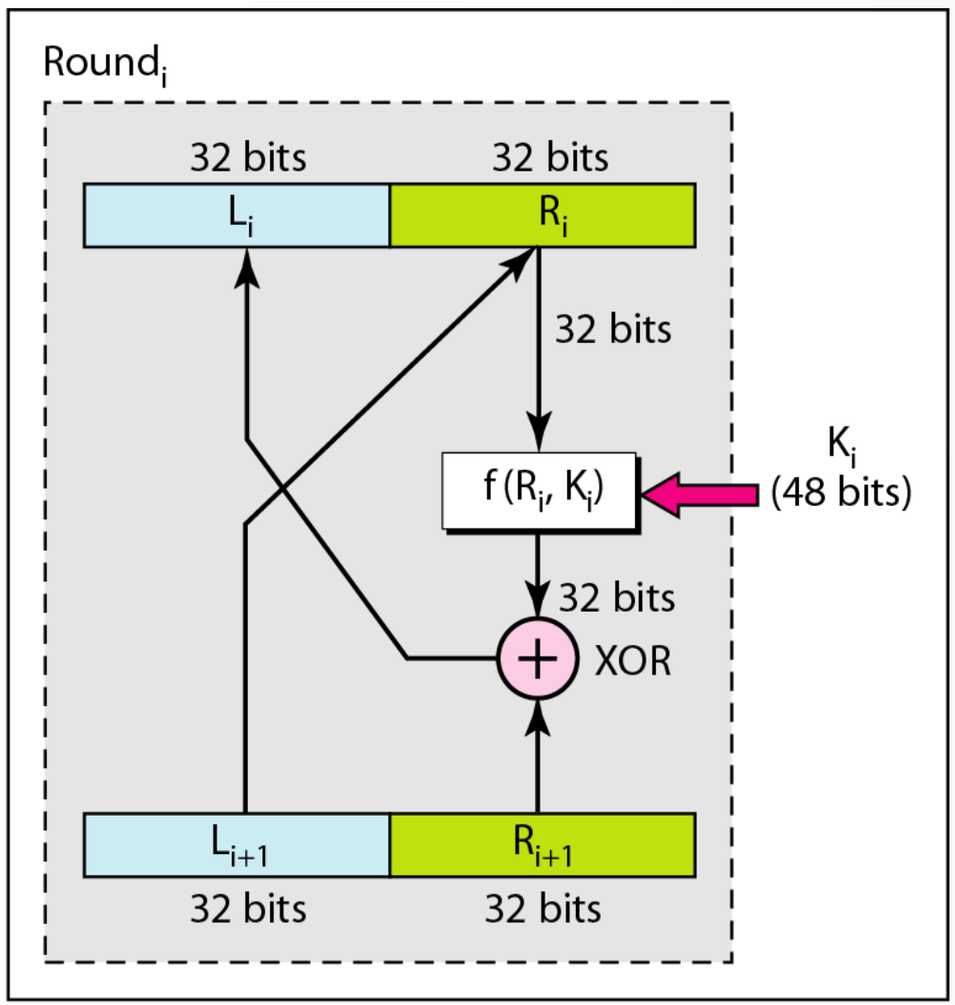




Data Encryption Standard



a. Encryption round



b. Decryption round

Data Encryption Standard

- The rounds are executed sequentially, the input of one round being the output of the previous round.
- The right half of the input, and the round key, are run through a function f that produces 32 bits of output; that output is then XOR'ed into the left half, and the resulting left and right halves are swapped.

- The keys for each round is separate which is just the result of left circular shift operation of the original key.
- The round key generator is the component which is responsible to generate 16 sub keys for 16 rounds.
- The round operation is nothing but the XOR operation between the plain text and the key.
- The final key to the cipher text is the resulting key at the end of 16 rounds.

Data Encryption Standard

- Decryption is simply the inverse of encryption, following the same steps but reversing the order in which the keys are applied.
- It would take maximum of 2^{56} or 72,057,594,037,927,936 attempts to find the correct key.
- For any cipher, the most basic method of attack is brute force, which involves trying each key until you find the right one.

Data Encryption Standard

- Even though few messages encrypted using DES encryption are likely to be subjected to this kind of code-breaking effort, many security experts felt the 56-bit key length was inadequate even before DES was adopted as a standard.
- Thus, DES is upgraded to more secure Advanced Encryption Standard (AES).

Stream Cipher

- A stream cipher is a symmetric key cipher where plaintext digits are combined with a keystream.
- A keystream is a stream of random characters that are combined with a plaintext message to produce an encrypted message.
- In a stream cipher, each plain digit is encrypted one at a time with the corresponding digit of the keystream, to give a digit of the ciphertext stream.
- https://www.okta.com/identity-101/stream-cipher/

Block Cipher

- A block cipher is an encryption method that applies a deterministic algorithm along with a symmetric key to encrypt a block of text, rather than encrypting one bit at a time as in stream ciphers.
- For example, a common block cipher, AES, encrypts 128 bit blocks with a key of predetermined length: 128, 192, or 256 bits.

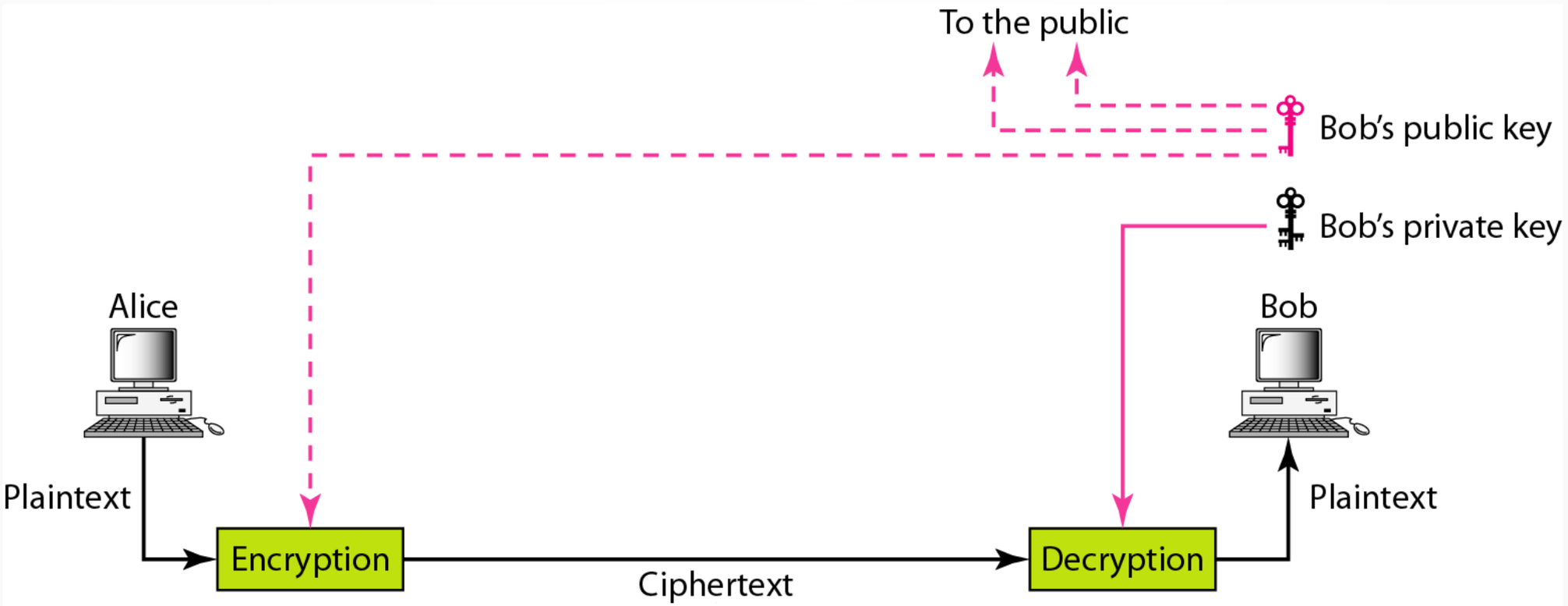
Asymmetric Key Cryptography

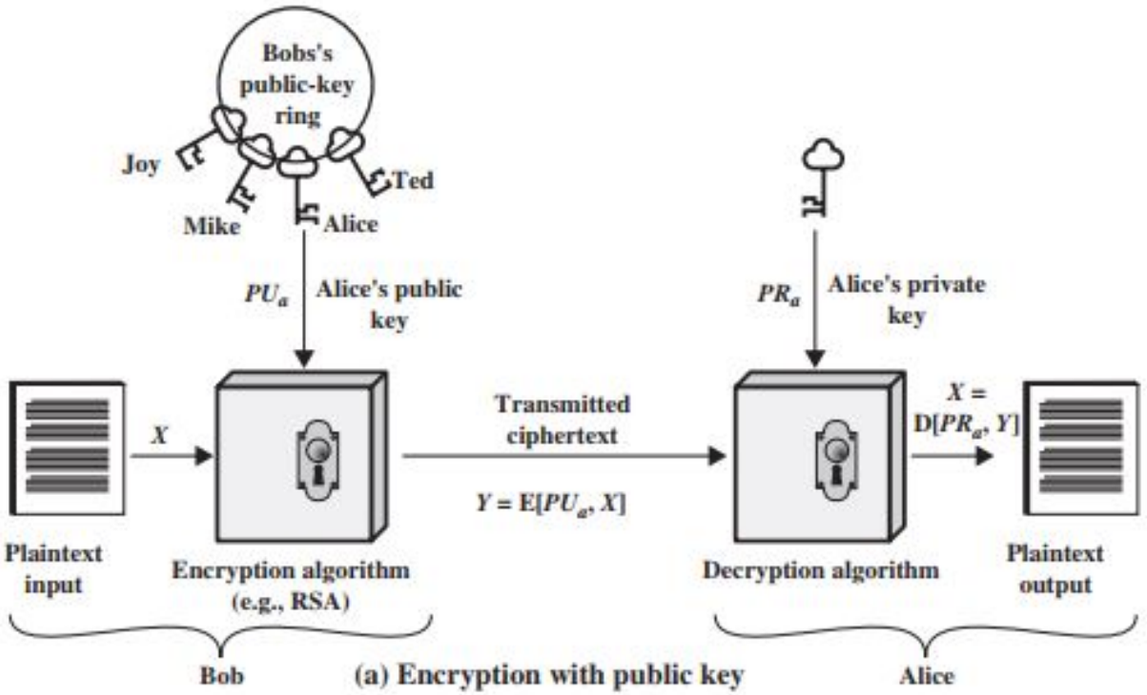
- Also known as ***Public Key Cryptography***.
- Used two keys: *public-key* and *private key*.

  Public key:
  - Shared with the public who wants to communicate with the receiver.
  - Used for enciphering the sender's plaintext into cipher text.

  Private key
  - Is kept secret by the receiver.
  - Used to deciphering ciphertext into plaintext by the receiver.
- Proposed by Diffie and Hellman in 1976
- **RSA** uses public-key cryptosystem.

Asymmetric Key Cryptography





(a) Encryption with public key

The essential steps:

I. Each user generates a pair of keys to be used for the encryption and decryption of messages.

II. Each user places one of the two keys in a public register or other accessible file. This is the public key. The companion key is kept private. As Figure suggests, each user maintains a collection of public keys obtained from others.

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

IV. 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.

Asymmetric Key Cryptography

- Because one key is public, and its complementary key must remain secret, a public key cryptosystem must meet the following three conditions:
  1. It must be computationally easy to encipher and decipher a message given the appropriate key.
  2. It must be computationally infeasible to derive the private key from the public key.
  3. It must be computationally infeasible the private key from a chosen plaintext attack.

RSA Algorithm

- RSA is one of the first public key cryptosystem and is widely used for secure data transmission.
- Developed in 1977 by Ron **R**ivest, Adi **S**hamir, and Len **A**dleman at MIT and first published in 1978.
- Most widely used asymmetric key encryption.
- Used in security protocol such as IPSEC, SSH, TLS etc.
- The **RSA** scheme is a cipher in which the plaintext and ciphertext are integers between 0 and *n* - 1 for some *n*. A typical size for *n* is 1024 bits, or 309 decimal digits. That is, *n* is less than 2^{1024} .
- RSA uses two exponents, *e* and *d*, where *e* is public and *d* is private.
- Suppose *P* is the plaintext and *C* is the ciphertext. Alice uses $C = P^e \bmod n$ to create ciphertext C from plaintext P; Bob uses $P = C^d \bmod n$ to retrieve the plaintext sent by Alice. The modulus *n*, a very large number, is created during the key generation process.

RSA Algorithm

- A user of RSA creates and then published a public key based on two large prime numbers, along with an auxiliary value. The prime numbers must be kept secret.
- The RSA algorithm involves four steps:
  - Key generation
  - Key distribution
  - Encryption
  - Decryption

RSA : Key Generation

The keys for RSA algorithm are generated in the following way:

- Generate two large random prime numbers, *p* and *q*. (p and q ≥ 2^{512})
- Compute *n = p x q, n is called modulus*
- *Compute the totient* $\Phi(n) = (p-1)(q-1)$.
- Choose *e* such that 1<e< $\Phi(n)$, *where e and* $\Phi(n)$ *do share factors other than one i.e. gcd(e,* $\Phi(n)$*)=1.*
- Computer *d* such that *d x e = 1 mod* $\Phi(n)$ *(i.e. de{mod* $\Phi(n)$*}=1*
- Publish *e* and *n* as the public key (Sender's public key) for encryption.
- Keep *d* and *n* as private key (Receiver's private key) for decryption

RSA: Key Generation

- The totient $\Phi(n)$ of positive integer n is the numbers less than n with no factors in common with n (i.e. they share no factors except 1).
- For example:
  - Let n = 10. The numbers that are less than 10 and are relatively prime to (have no factors in common with) n are 1,3,7, and 9. Hence, $\Phi(n)$= 4.
  - Similarly, if n =21, the numbers that are relatively prime to n are 1,2,4,5,8,10, 11, 12, 13,16,19, and 20. So $\Phi(n)$=12

RSA: Key Distribution

- Suppose that Bob wants to send information to Alice. IF they decide to use RSA, Bob must know Alice's public key to encrypt the message and Alice must use her private key to decrypt the message.
- To enable Bob to send his encrypted messages, Alice transmit her public key (e ,n) to Bob via a reliable,but not necessarily secret, route. Alice' private key (d, n) is never distributed.

RSA: Encryption and Decryption

- Given public key (e ,n) and private key (d, n) are computed
- To encrypt bit pattern, *P*, compute cipher text C as:
  $C = P^e \bmod n$
- To decrypt bit pattern, *C*, compute
  $P = C^d \bmod n$

RSA: Example

- Pick two prime numbers: p=3, q = 5.
- n=p*q = 3*5=15
- $\Phi(n)$=(p-1)(q-1) = (3-1)(5-1) = 2*4 = 8.
- Choose e satisfying 1<e< $\Phi(n)$.

  Let us choose e=3,which do not share any common factors with 8 rather than 1.

- Compute d satisfying de{mod $\Phi(n)$ }=1

  So d*3{mod 8} = 1

  Let us choose d =11 which satisfy the relation

RSA : Example

- So public key (e,n) is (3,15) which is released publicly and the persons that want to send the message use this key to encrypt the message and send it to the receiver.
- Private key(d,n) is (11,15) which is kept secret by the receiver.

RSA : Example

- Let us consider the message be 2.
- So, at encryption process, the sender uses the public key to encrypt the message. Resulting cipher text will be:

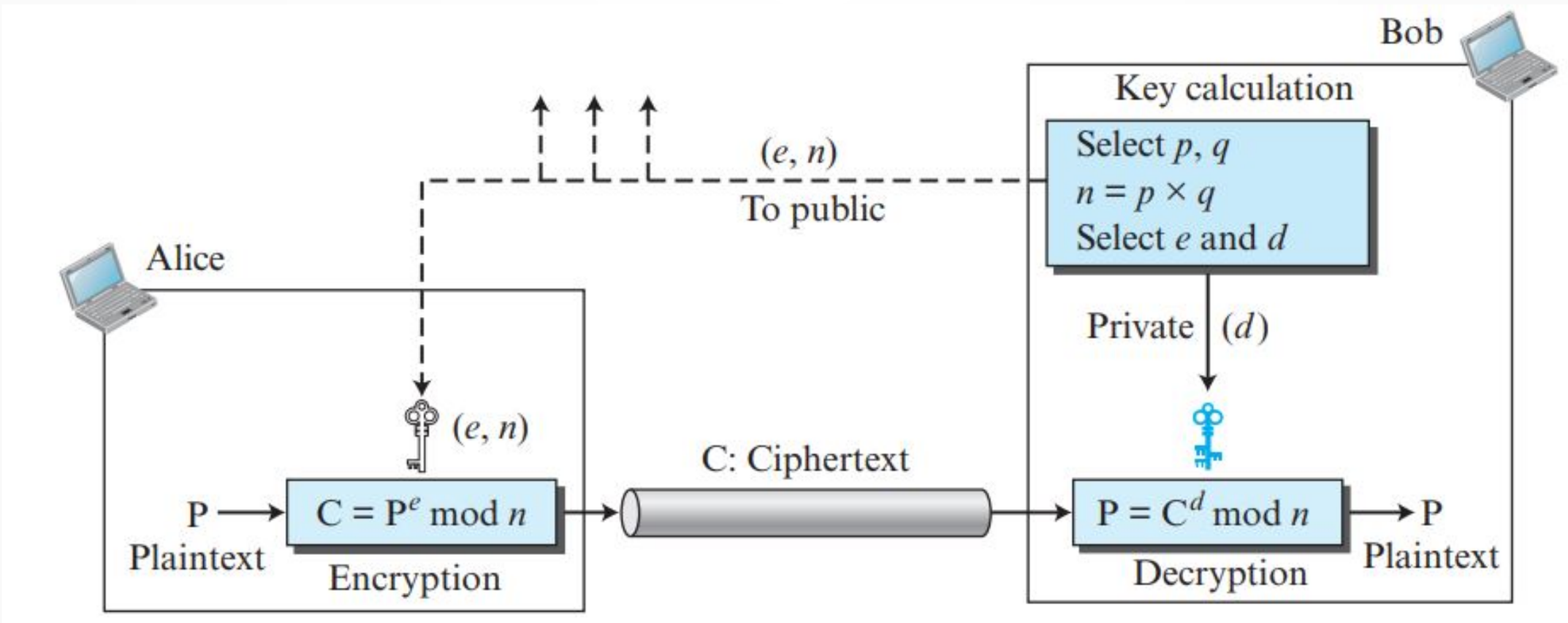
  C = m^e(mod n) = 2^3(mod 15) = 8.

- At decryption process, the private key is used to decrypt the cipher text. Plain text is obtained as:

  P = c^d(mod n) = 8^11(mod 15) = 2.

- Hence the original message 2 is obtained at receiver end after decryption.

RSA Algorithm



Hashing

- **Hashing** is the transformation of a string of characters into a usually shorter fixed-length value or key, called *hash*, that represent the original string.
- A hash is generated by a *hash function*. A hash function is a mathematical function that converts a input value into another compressed output value (often called as “message digest” , or simply a “hash”) The input to the hash function is of arbitrary length but output is always of fixed length.
- A hash function provides encryption using an algorithm and no key. They are called “*one-way hash functions*” because there is no way to reverse the encryption.

Hashing

- The figure depicts the general operation of a cryptographic hash function.
- Typically, the input is padded out to an integer multiple of some fixed length (e.g., 1024 bits), and the padding includes the value of the length of the original message in bits.
- The length field is a security measure to increase the difficulty for an attacker to produce an alternative message with the same hash value.

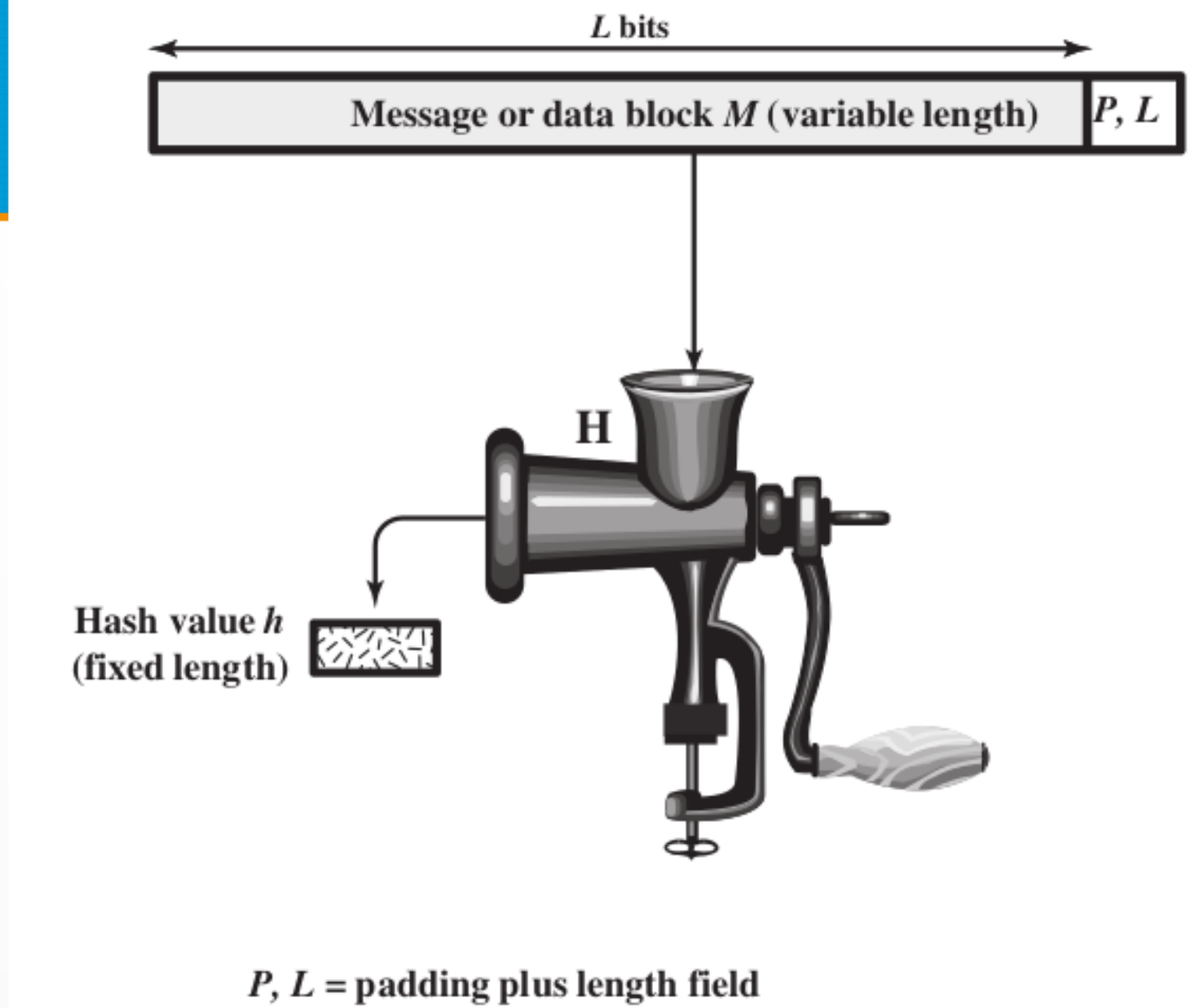


Figure: Cryptographic Hash Function; h=H(M)

Input-output property of Hash function

- A “good” hash function produces ***evenly distributed and apparently random outputs*** as a result of applying the function ***to a large set of inputs***.
- A *change to any bit or bits* in the message *M* most probably results in a ***change to the hash code***.

Properties of Hash functions

1. Each hash value or output must be unique.
2. Hashing speed is also a factor. A hash function should be reasonably quick to produce a hash value.
3. A hash function needs to be secure. Even a slight change to the input file should produce a vastly different hash value.
4. It is immutable in the sense that the same input must produce the exact same hash. (h(x) = z)
5. It is irreversible, i.e., it's not possible to find original message from its hash value. (*preimage resistance*).
6. Given x_1, and thus h(x1),it is computationally infeasible to find any x_2 such that $h(x_1) = h(x_2)$. (*second preimage resistance*)
7. It is infeasible to find two different messages ($x_1 \neq x_2$) with the same hash value ($h(x_1) = h(x_2)$) (collision resistance)

Uses of Hashing

Data Integrity

- When a user sends a secure message, a hash of the intended message is generated and encrypted, and is sent along with the message.
- When the message is received, the receiver decrypts the hash as well as the another hash from the message.
- If the two hashes are identical when compared, then a secure transmission has occurred. This hashing process ensures that the message is not altered by an unauthorized end users.

Uses of Hashing

- Cryptographic hash functions have many information-security applications, notably in digital signatures, message authentication codes (MACs), and other forms of authentication.
- They can also be used as ordinary hash functions, to index hash tables for fingerprinting, to detect duplicate data or uniquely identify files, and as checksums to detect accidental data corruption.

Hashing Algorithm

- SHA-1
- SHA-2
- RIPE-MD 160
- Whirlpool
- MD5

Hashing

- https://sectigostore.com/blog/hashing-vs-encryption-the-big-players-of-the-cyber-security-world/
- https://www.tutorialspoint.com/cryptography/cryptography_hash_functions.htm
- https://cybersecurityglossary.com/hashing/
-

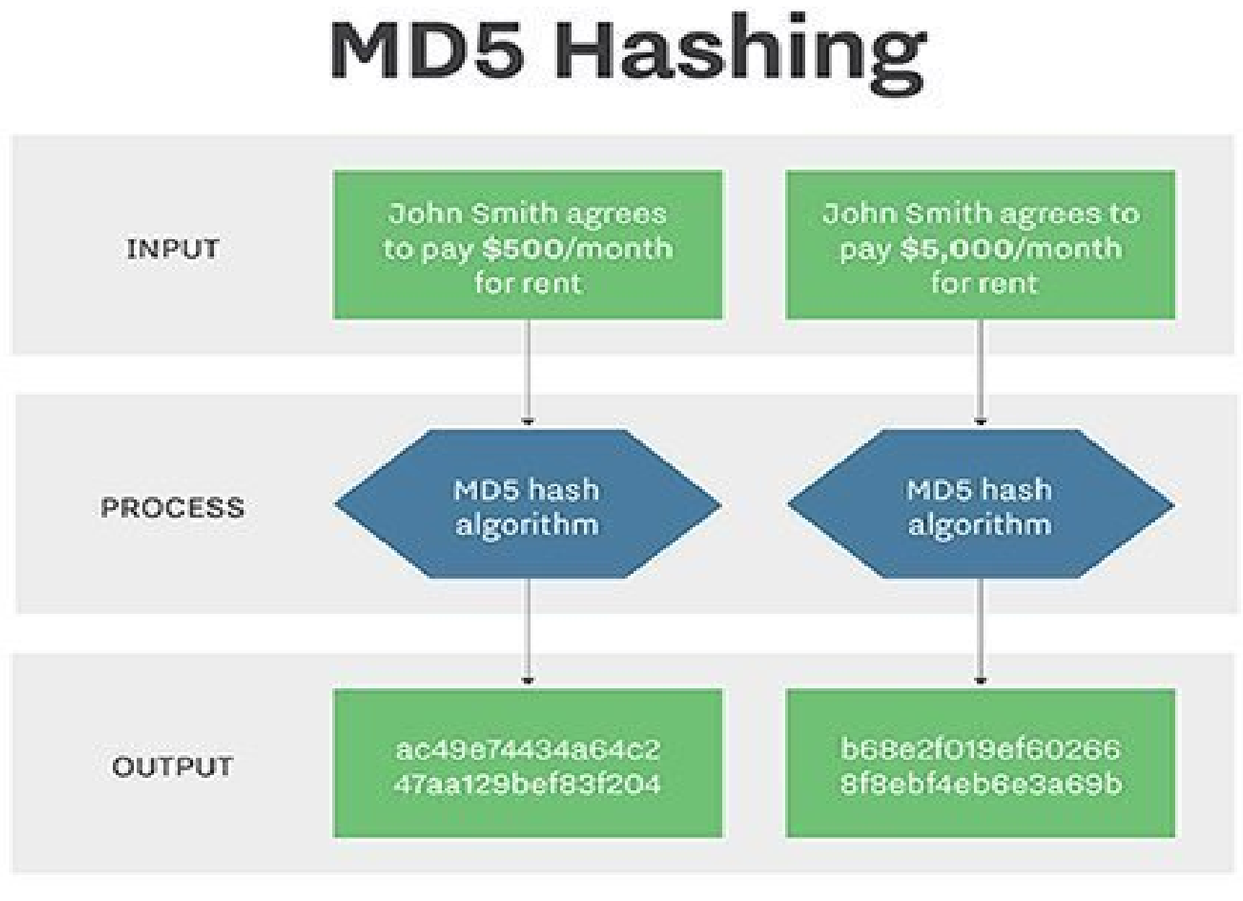
Message Digest 5 (MD5)

- MD5 (Message-Digest algorithm 5) is a widely used cryptographic hash function that results in a ***128-bit hash value*** from any length of input string.
- The 128-bit (16-byte) MD5 hashes (also termed message digests) typically are represented as 32-digit hexadecimal numbers (for example, ec55d3e698d289f2afd663725127bace)
- The MD5 hash function was originally designed for use as a secure cryptographic hash algorithm for authenticating digital signatures and data integrity.
-

Message Digest 5 (MD5)

- MD5 has been deprecated for uses other than a non-cryptographic checksum to verify data integrity and detect unintentional data corruption.
- MD5 is now no longer used as cryptographic checksum because researcher have demonstrated techniques capable of easily generating MD5 collision on commercial off-the shelf computers.
- MD-5 typically is not recommended for wireless LAN implementations because it may expose the user's password, and because several collision-based weaknesses have been demonstrated.

MD5 Hashing Example



Message Authentication Code (MAC)

Message Authentication Requirements

- In the context of communication across a network, the following attacks can be identified.

1. Disclosure
2. Traffic Analysis
3. Masquerade
4. Content modification
5. Sequence modification
6. Timing modification
7. Source repudiation
8. Destination repudiation

3-6: Message Authentication
7: digital signature
8: digital signature and some protocols

Message Authentication

- Message authentication is a mechanism or service used to verify the integrity of a message. Message authentication assures that data received are exactly as sent (i.e., there is no modification, insertion, deletion, or replay).
- Also, message authentication assures the purported (alleged) identity of the sender is valid. That is, it confirm that that the message came from the intended sender (authenticity) and has not been changed.
- When a hash function is used to provide message authentication, the hash function value is often referred to as a **message digest**.

Message Authentication Code (MAC)

- A message authentication code (MAC), sometimes known as a tag, is a short piece of information used for authenticating a message.
- In other words, to confirm that the message came from the stated sender (its authenticity) and has not been changed.
- The MAC value protects a message's data integrity, as well as its authenticity, by allowing verifiers (who also possess the secret key) to detect any changes to the message content.

Message Authentication Code (MAC)

- Informally, a message authentication code system consists of three algorithms:
  - A *key generation algorithm* selects a key from the key space uniformly at random.
  - A *signing algorithm* efficiently returns a tag given the key and the message.
  - A *verifying algorithm* efficiently verifies the authenticity of the message given the key and the tag. That is, return accepted when the message and tag are not tampered with or forged, and otherwise return rejected.

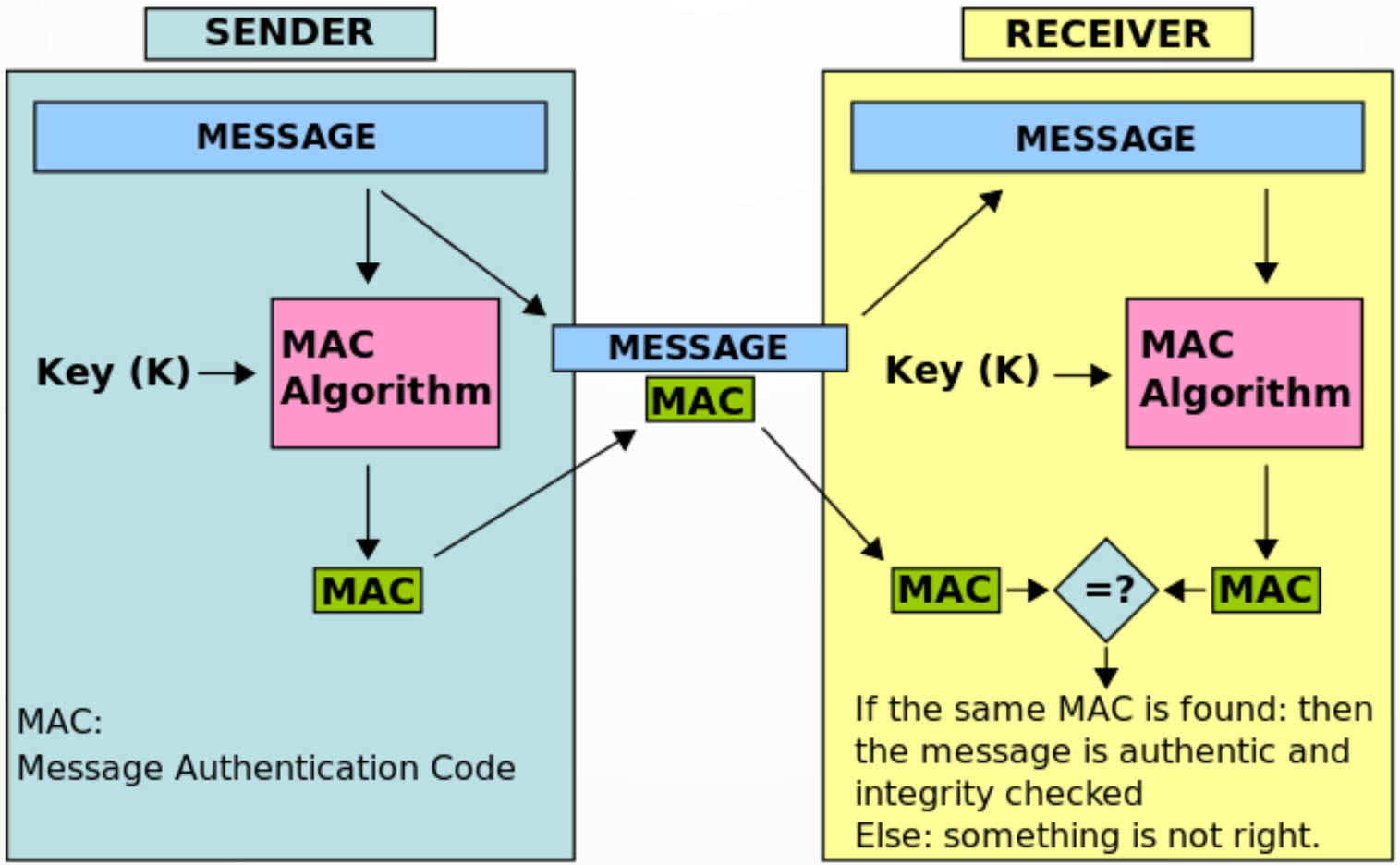
Message Authentication Code (MAC)

- MAC algorithm is a symmetric key cryptographic technique to provide message authentication. For establishing MAC process, the sender and receiver share a symmetric key K.

  Essentially, a MAC is an encrypted checksum generated on the underlying message that is sent along with a message to ensure message authentication.

- The process of using MAC for authentication is depicted in the following illustration

Message Authentication Code (MAC)



Message Authentication Code (MAC)

- The sender uses some publicly known MAC algorithm, inputs the message and the secret key K and produces a MAC value.
- Similar to hash, MAC function also compresses an arbitrary long input into a fixed length output. The major difference between hash and MAC is that MAC uses secret key during the compression.
- The sender forwards the message along with the MAC. Here, we assume that the message is sent in the clear, as we are concerned of providing message origin authentication, not confidentiality. If confidentiality is required then the message needs encryption.

  On receipt of the message and the MAC, the receiver feeds the received message and the shared secret key K into the MAC algorithm and re-computes the MAC value.
- The receiver now checks equality of freshly computed MAC with the MAC received from the sender. If they match, then the receiver accepts the message and assures himself that the message has been sent by the intended sender.
- If the computed MAC does not match the MAC sent by the sender, the receiver cannot determine whether it is the message that has been altered or it is the origin that has been falsified. As a bottom-line, a receiver safely assumes that the message is not the genuine.

Limitation of MAC

- The shared key should be shared through secret connection establishment.
- Since MACs are based on symmetric principles, it does not provide non-repudiation.