



# Network Security

**Dr. Md. Imdadul Islam**

Professor, Department of Computer Science and  
Engineering Jahangirnagar University

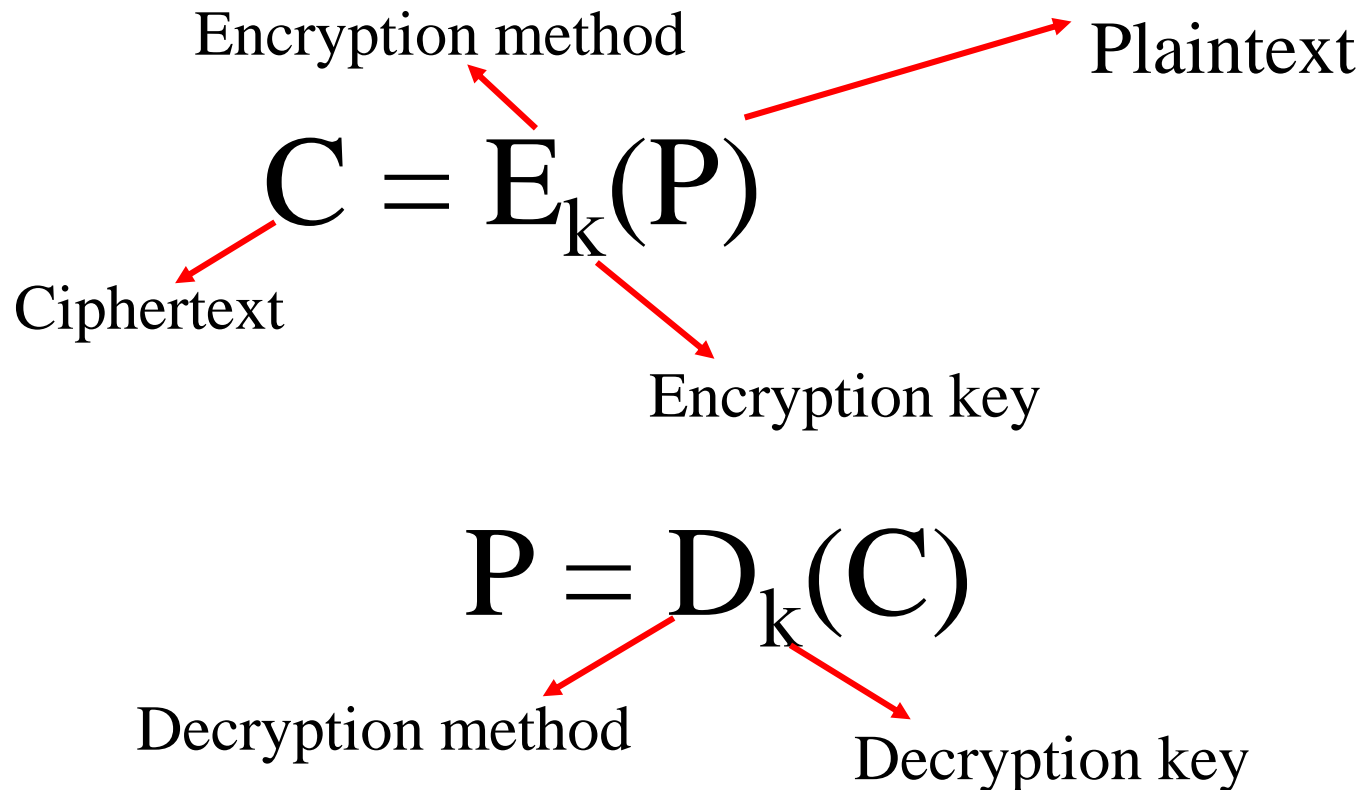
<https://www.juniv.edu/teachers/imdad>

- ✓ Computer networks are typically a shared resource used by many users having different interests.
- ✓ The Internet is shared by competing businesses, mutually antagonistic governments, and opportunistic criminals. Unless security measures are taken, a network conversation or a distributed application may be compromised by an adversary.
- ✓ The main tools for securing networked systems are **cryptography** and **firewalls**.

# An Introduction to Cryptography

- ✓ Cryptography comes from the Greek words for ‘**secret writing**.’ It has a long and colorful history going back thousands of years.
- ✓ Professionals make a distinction between **cipher** and **codes**. A cipher is a character-for-character or bit-for-bit transformation, without regard to the linguistic structure of the messages.
- ✓ In contrast, a code replace one word with another word or symbol (for example in Huffman code each symbol is represented by its own bit string of different length).
- ✓ Historically four groups of people have used and contributed to the art of cryptography: the military, the diplomatic corps, diarists and lovers.

The messages to be encrypted, known as *plaintext*, are transformed by a function that parameterized by a *key*. The output of the encryption process, known as the *ciphertext*, is then transmitted, often by messenger or radio. We assume that the enemy or intruder, hears and accurately copies down the complete ciphertext.

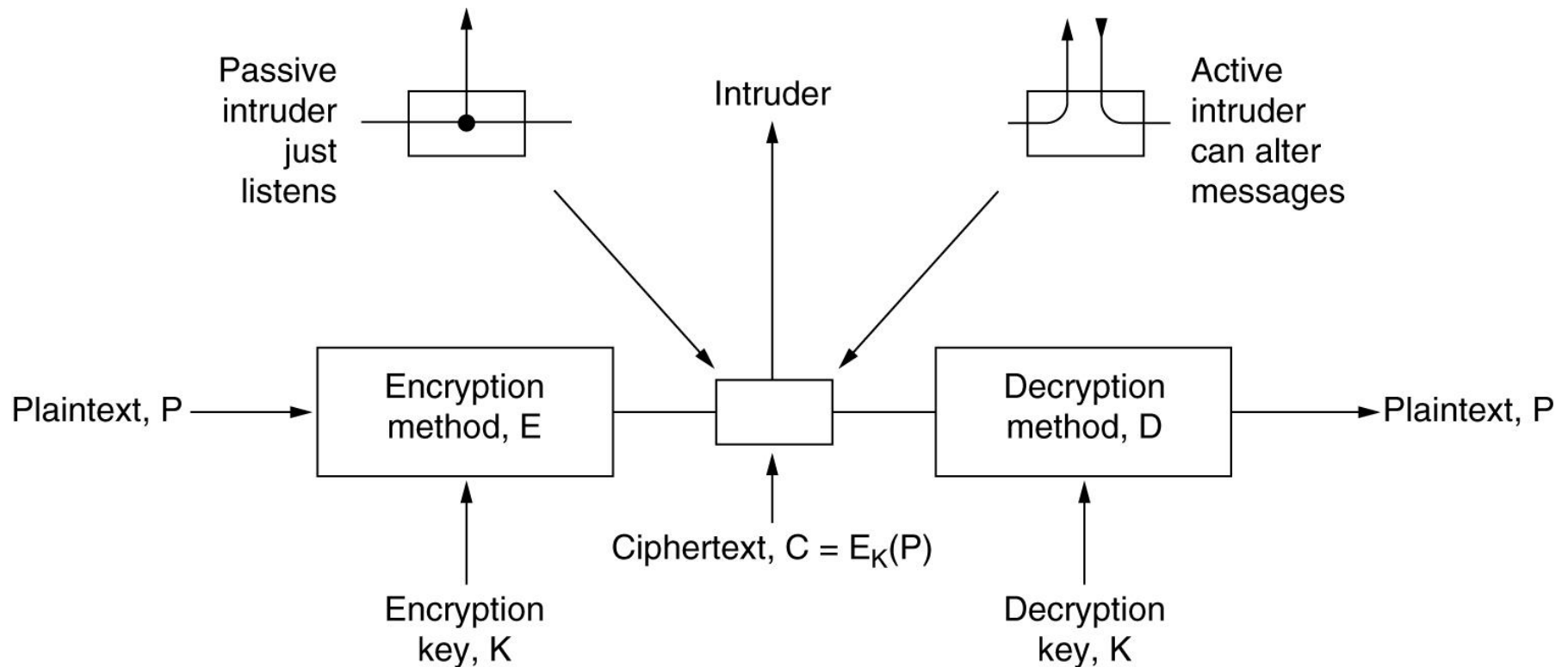


# Some Basic Terminology

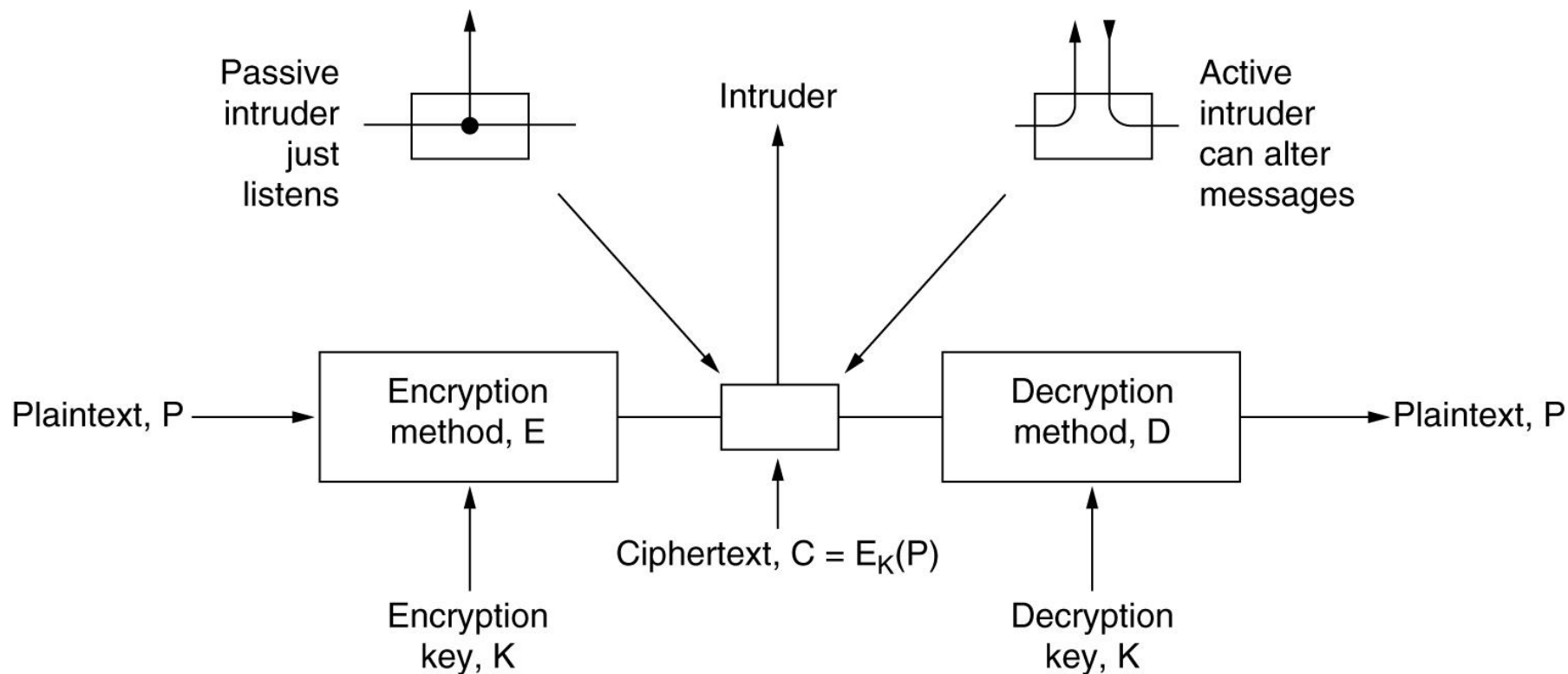
- **plaintext** - original message
- **ciphertext** - coded message
- **cipher** - algorithm for transforming plaintext to ciphertext
- **key** - parameter used in cipher known only to sender/receiver
- **encipher (encrypt)** - converting plaintext to ciphertext
- **decipher (decrypt)** - recovering ciphertext from plaintext
- **cryptography** - study of encryption principles/methods
- **cryptanalysis (codebreaking)** - study of principles/ methods of deciphering ciphertext *without* knowing key
- **cryptology** - field of both **cryptography** and **cryptanalysis**

**Cryptography** is the science and art of creating secret codes, **cryptanalysis** is the science and art of breaking those codes.

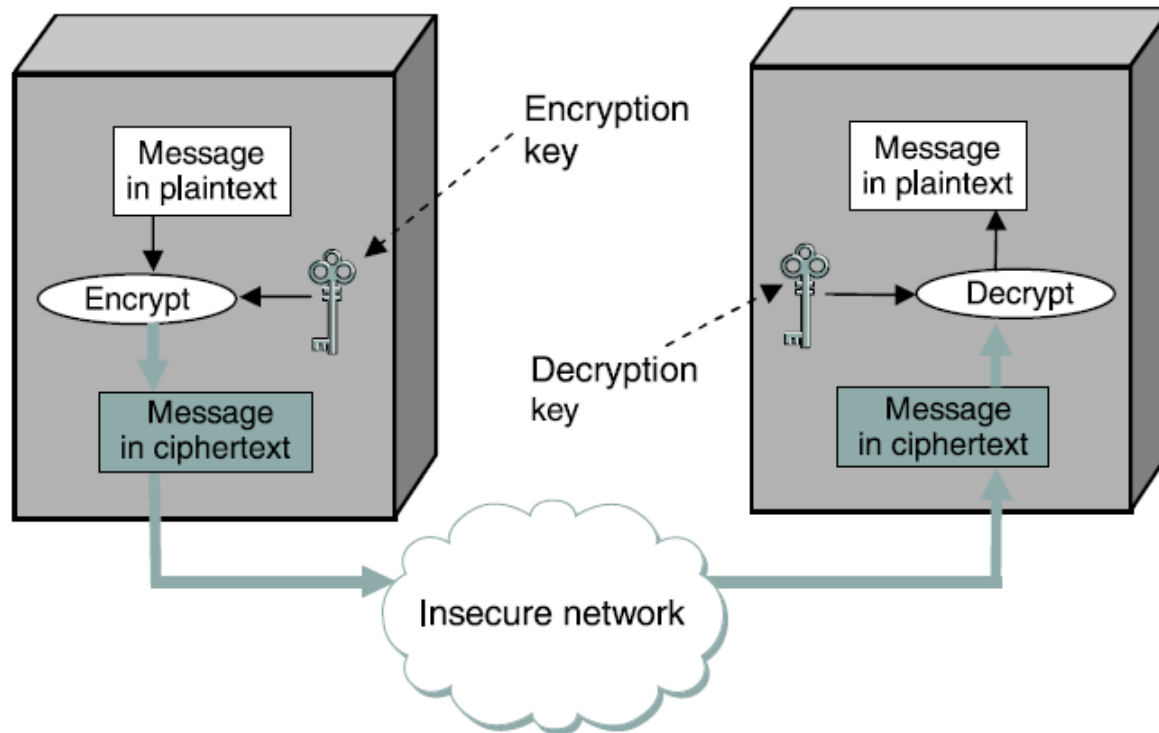
✓ Sometimes the intruder can only listen to the communication channel (*passive intruder*) but can also record messages and play them back later, inject his own messages or modify legitimate messages before they get to the receiver (*active intruder*).



✓ It will often be useful to have a notation for relating plaintext, ciphertext and keys. We will use  $C = E_K(P)$  to mean that the encryption of the plaintext  $P$  using key  $K$  gives the ciphertext  $C$ . Similarly  $P = D_K(C)$  represents the decryption of  $C$  to get the plaintext again. Therefore,  $D_K(E_K(P)) = P$ .



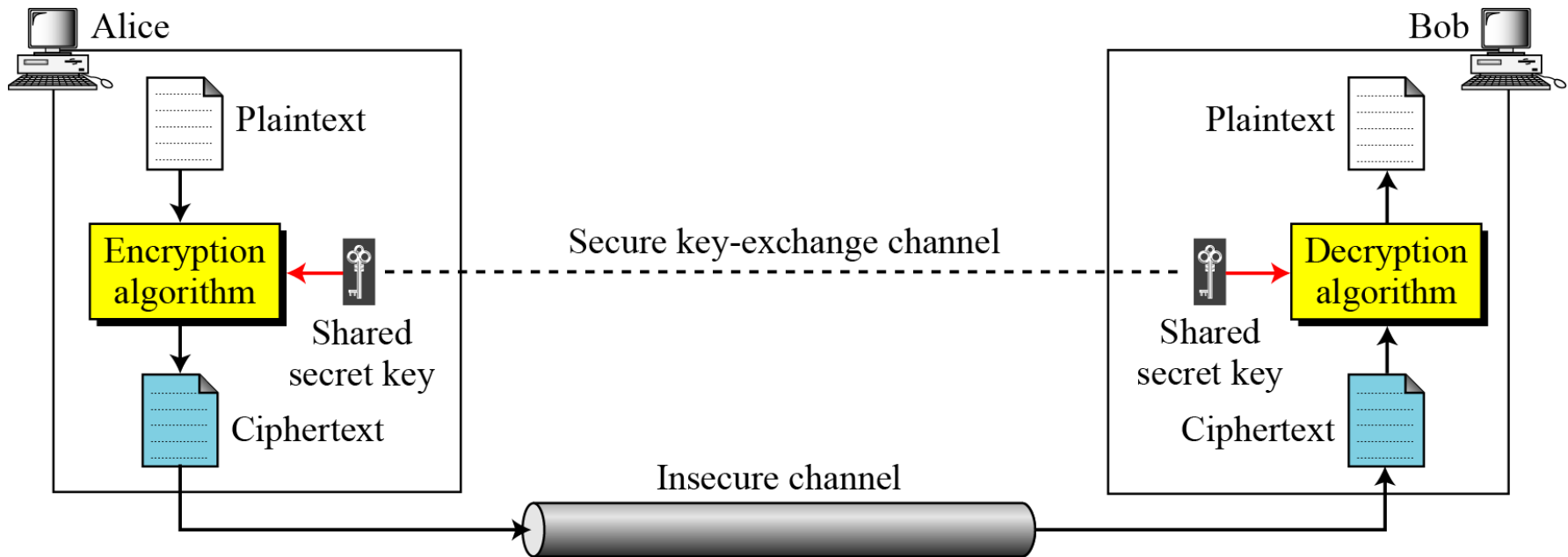
In a **symmetric-key cipher**, both participants (sender and receiver) in a communication share the same **key**. In other words, if a message is encrypted using a particular key, the same key is required for decrypting the message. In **asymmetric key cryptography**, where the encryption and decryption keys are different.



Symmetric-key encryption and decryption



# Basic of **symmetric-key cipher** from Alice to Bob

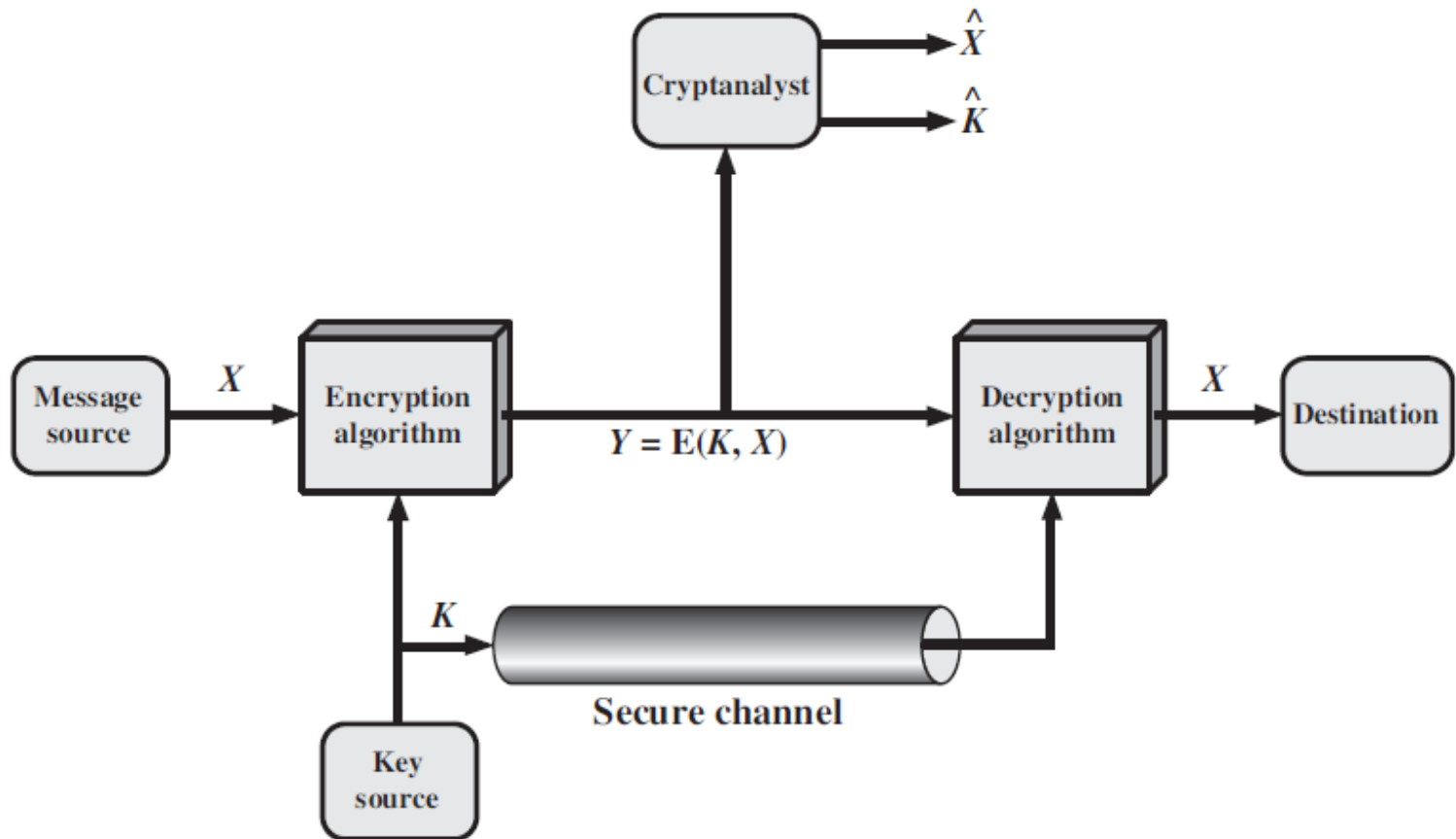


Here  $P$  is the plaintext,  $C$  is the ciphertext, and  $K$  is the key then,

$$\text{Alice: } C = E_k(P)$$

and

$$\text{Bob: } P_1 = D_k(C) = D_k(E_k(P)) = P$$



Model of Symmetric Cryptosystem

Encryption methods have historically been divided into two categories: *substitution ciphers* and *transposition ciphers*. Both of them are under symmetric key cyphering.

## Substitution Ciphers

- ✓ In substitution cipher each letter or group of letters is replaced by another letter or group of letters to disguise it.
- ✓ One of the oldest ciphers is the **Caesar** cipher, attributed to Julius Caesar. In this method *a* becomes *D*, *b* becomes *E*, *c* becomes *F*, ....., *z* becomes *C*. For example *attack* becomes *DWWDFN*.
- ✓ A slight generalization of the **Caesar** cipher allows the ciphertext alphabet to be shifted by *k* letters, instead of always 3. In this case *k* becomes a key to the general method of circularly shifted alphabets.

The next improvement is to have the symbols in the plaintext, say, the 26 letters for simplicity map onto some other letters. For example:

Plaintext: *a b c d e f g h i j k l m n o p q r s t u v w x y z*

Ciphertext: ***Q E R T Y U I O P A S D F G H J K L Z X C V B N M***

The general system of symbol-for-symbol substitution is called monoalphabetic substitution, with the key being 26-letter string corresponding to the full alphabet.

Now ***attack*** become ***QZZQEA***.

At first this might appeared to be safe because it requires  $26!$  Keys and take  $10^{10}$  years if each trial needs 1nsec.

Statistical distribution of letters will be changed.

In English, *e* is the most common letter, followed by *t*, *o*, *a*, *n*, *i* etc.

Digrams: *th*, *in*, *er*, *re* and *an*.

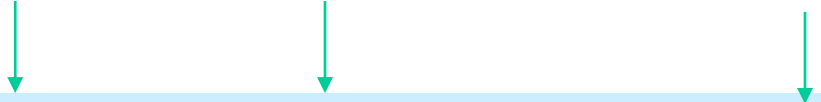
Trigrams: *the*, *ing*, *and* and *ion*.

From the most frequent letter,  $T = e$ .

*tXe* will be *the* i.e.  $X = h$

*thYt* will be *that* i.e.  $Y = a$

Example-1: Ciphertext from an **accounting firm** should contains *financial*



CTBMN BYCTC BTJDS QXBNS GSTJC BTSWX CTQTZ CQVUJ  
**QJSGS TJQZT** MNQJS VLNSX VSZJU JDSSTS JQUUS JUBXJ  
DSKSU JSNTK BGAQJ ZBGYQ TLCTZ BLCTZ BNYBN QJSW

Using our knowledge that *financial* has a repeated letter (i), with four other letters between their occurrences. We look for repeated letters in the ciphertext at this spacing. We find 12 hits at positions 6, 15, 27, 31, 42, 48, 56, 66, 70, 71, 76 and 82. However, only two of these, 31 and 42, have the next letter( corresponding to n in the plaintext) repeated in the popper place. Of these two only 31 also has the *a* correctly positioned, so we know that *financial* begins at position 30.

# Transposition Ciphers

Transpositional ciphers in contrast reorder the letters but do not disguise them. Therefore statistical distribution of letters will remain same. The cipher is keyed by a word or phrase not containing any repeated letters. In this example, MEGABUCK is the key. If the intruder assume that the plain text contains *milliondollar* then he can try to beak it

<u>M</u>	<u>E</u>	<u>G</u>	<u>A</u>	<u>B</u>	<u>U</u>	<u>C</u>	<u>K</u>
<u>7</u>	<u>4</u>	<u>5</u>	<u>1</u>	<u>2</u>	<u>8</u>	<u>3</u>	<u>6</u>
p	l	e	a	s	e	t	r
a	n	s	f	e	r	o	n
e	m	i	l	l	i	o	n
d	o	l	l	a	r	s	t
o	m	y	s	w	i	s	s
b	a	n	k	a	c	c	o
u	n	t	s	i	x	t	w
o	t	w	o	a	b	c	d

Plaintext

pleasetransferonemilliondollarsto  
myswissbankaccountsixtwotwo

Ciphertext

AFLLSKSOSELAWAIATOOSSCTCLNMOMANT  
ESILYNTWRNNTSOWDPAEDOBUEIRICXB

Fig.2 A transposition cipher.

<u>M</u>	<u>E</u>	<u>G</u>	<u>A</u>	<u>B</u>	<u>U</u>	<u>C</u>	<u>K</u>
<u>7</u>	<u>4</u>	<u>5</u>	<u>1</u>	<u>2</u>	<u>8</u>	<u>3</u>	<u>6</u>
p	l	e	a	s	e	t	r
a	n	s	f	e	r	o	n
e	m	i	l	l	i	o	n
d	o	l	l	a	r	s	t
o	m	y	s	w	i	s	s
b	a	n	k	a	c	c	o
u	n	t	s	i	x	t	w
o	t	w	o	a	b	c	d

Plaintext

pleasetransferonemilliondollarsto  
myswissbankaccountsixtwotwo

Ciphertext

AFLLSKSOSELAWAIATOOSCTCLNMOMANT  
ESILYNTWRNNTSOWDPAEDOBUEOERIRICXB

- ✓ The cryptanalyst must first aware he is dealing with transposition cipher (from the statistical distribution of letters).
- ✓ For example, suppose that our cryptanalyst suspects that the plaintext phrase *milliondollars* occurs somewhere in the message. Observe that digrams *MO*, *IL*, *LL*, *LA*, *IR* and *OS* occur in the ciphertext as a result of this phrase wrapping around.



✓The ciphertext letter *O* follows the ciphertext letter *M* (they are vertically adjacent in column 4) because they are separated in the probable phrase by a distance equal to the key length. If a key length seven had been used, the digrams: *MD*, *IO*, *LL*, *LL*, *IA* , *OR* and *NS* would have occurred instead.

✓By hunting for various possibilities, the cryptanalyst can easily find the key length.

✓The remaining step is to order the columns. If the key length is  $k$  then  $k(k-1)$  column pair can be examined to see if its digram frequencies match those for English plaintext.

# Asymmetric Key Cryptography

**Asymmetric cryptography**, also known as **public-key cryptography**. Asymmetric key cryptography uses two separate keys: one private and one public.

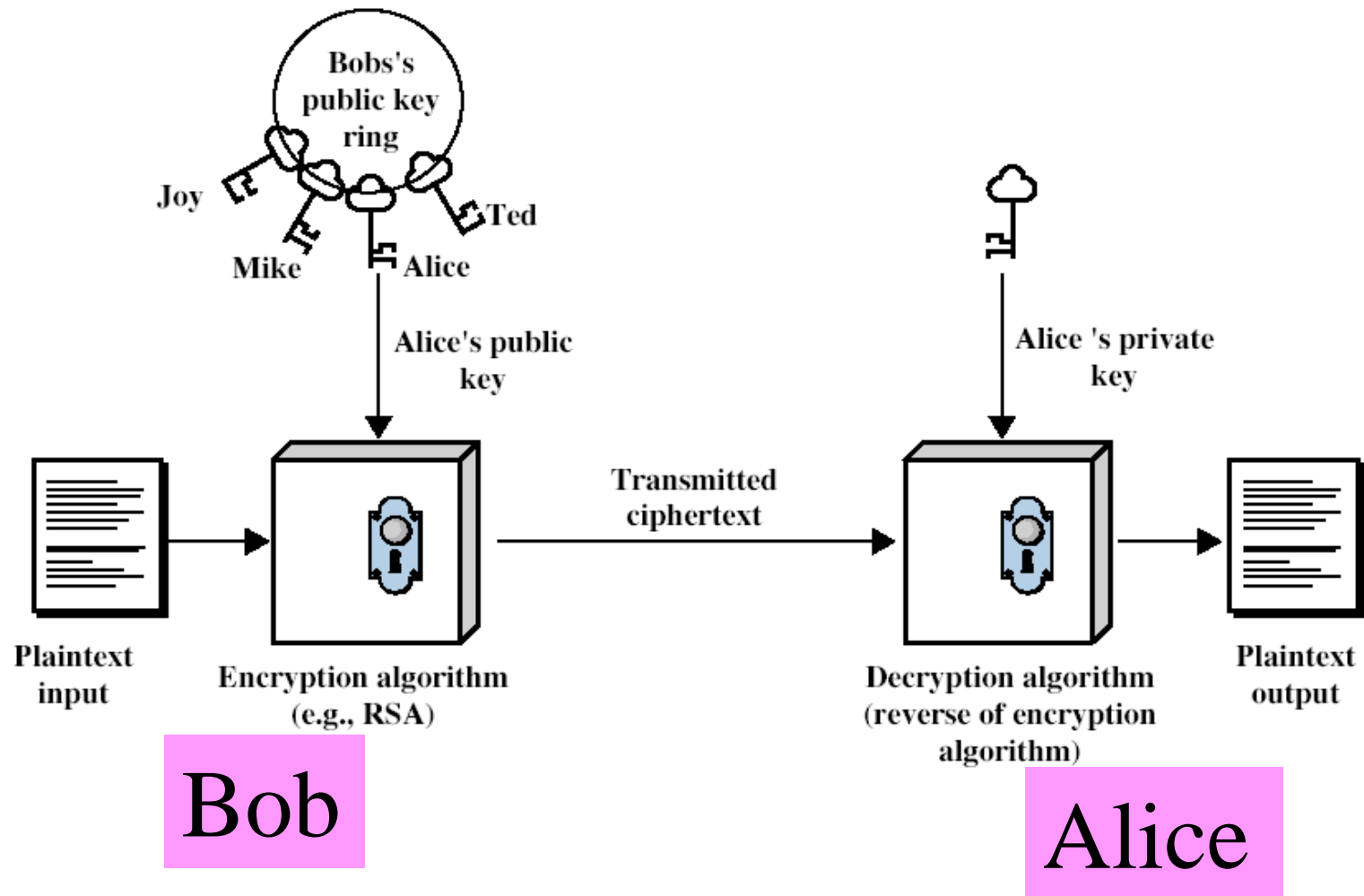
$$C = f(K_{public}, P) \quad P = g(K_{private}, C)$$

- ✓ There is a very important fact that is sometimes misunderstood: The advent of asymmetric-key cryptography does not eliminate the need for symmetric-key cryptography.

## Public-Key Algorithms:

- ✓ In 1976, two researchers at Stanford University, Diffie and Hellman(1976) proposed a radically new kind of cryptosystem, one in which the encryption and decryption keys were different, and the decryption key could not feasibly be derived from the encryption key.
- ✓ In their proposal, the (keyed) encryption algorithm,  $E$ , and the (keyed) decryption algorithm,  $D$ , had to meet three requirements:
  1.  $D(E(P)) = P$
  2. It is exceedingly difficult to deduce  $D$  from  $E$ .
  3.  $E$  can not be broken by a chosen plaintext attack.

Public-key cryptography requires each user to have two key: a *public key*, used by the entire world for encrypting messages to be sent to that user and a *private key*, which the user needs for decrypting message.



# RSA

One good method was discovered by a group at M.I.T. , 1978 known as RSA( Rivest, Shamir, Adleman). The RSA method is based on some principles from number theory.

1. Select two large primes:  $p = 17$ ,  $q = 11$  (typically 1024 bits)
2. Compute  $n = pq = 17 \times 11 = 187$  and  $z = (p-1)(q-1) = 16 \times 10 = 160$
3. Select  $d$ :  $\gcd(d, 160) = 1$  and  $d < 160$ . Actually  $d$  is a number relatively prime to  $z$ . (for example  $d = 7$ )
4. choose  $e$  such that  $de = 1 \pmod{z}$  (We will divide  $de = 7e$  by  $z = 160$  and the remainder will be 1 for example you can get  $e = 23$ )

Here  $e=23$  and  $d=7$  are the encryption and decryption key.

Here  $e=23$  and  $d=7$  are the encryption and decryption key.

Let  $P$  is the numerical value of message.

Encoded message,  $C = P^e(\text{mod } n)$

Decoded message,  $P = C^d(\text{mod } n)$

To encrypt a message,  $P$ , compute  $C = P^e \pmod n$ . To decrypt  $C$ , compute  $P = C^d \pmod n$ .

Let us consider another example:

$$p = 3, q = 11$$

$$n = pq = 33, z = (p-1)(q-1) = 20$$

$d = 7$ , since 20 and 7 does not have common factor

Now  $7e = 1 \pmod{20}$  which provides  $e = 3$

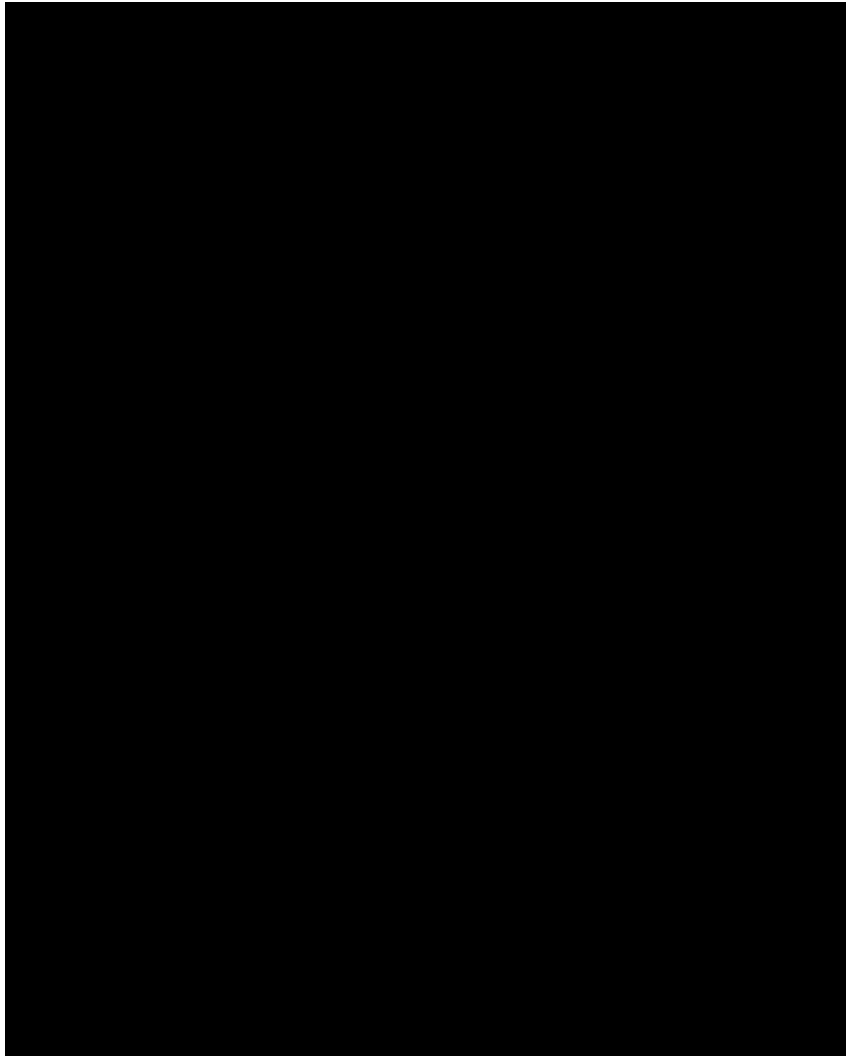
$C = P^3 \pmod{33}$  which provides encoded value  $C$

Let  $P$  is the numerical value of message.  
 Encoded message,  $C = P^e \pmod n$   
 Decoded message,  $P = C^d \pmod n$

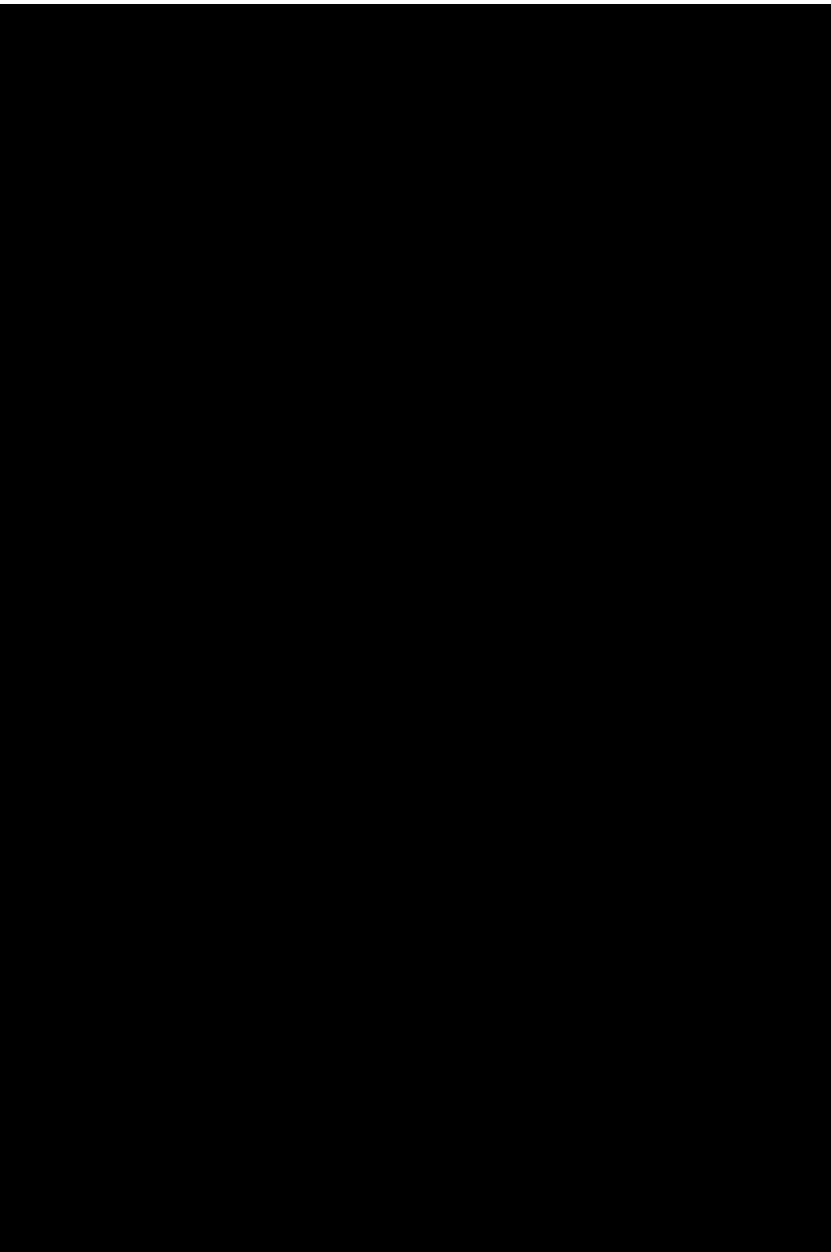
Plaintext (P)		Ciphertext (C)			After decryption	
Symbolic	Numeric	$P^3$	$P^3 \pmod{33}$	$C^7$	$C^7 \pmod{33}$	Symbolic
S	19	6859	28	13492928512	19	S
U	21	9261	21	1801088541	21	U
Z	26	17576	20	1280000000	26	Z
A	01	1	1	1	01	A
N	14	2744	5	78125	14	N
N	14	2744	5	78125	14	N
E	05	125	26	8031810176	05	E
Sender's computation				Receiver's computation		

Fig. An example of the RSA algorithm.

# RSA In Image Encryption and Decryption







Original Image



Encrypted Quiscent Image



Encrypted Remainder Image



Decrypted Image

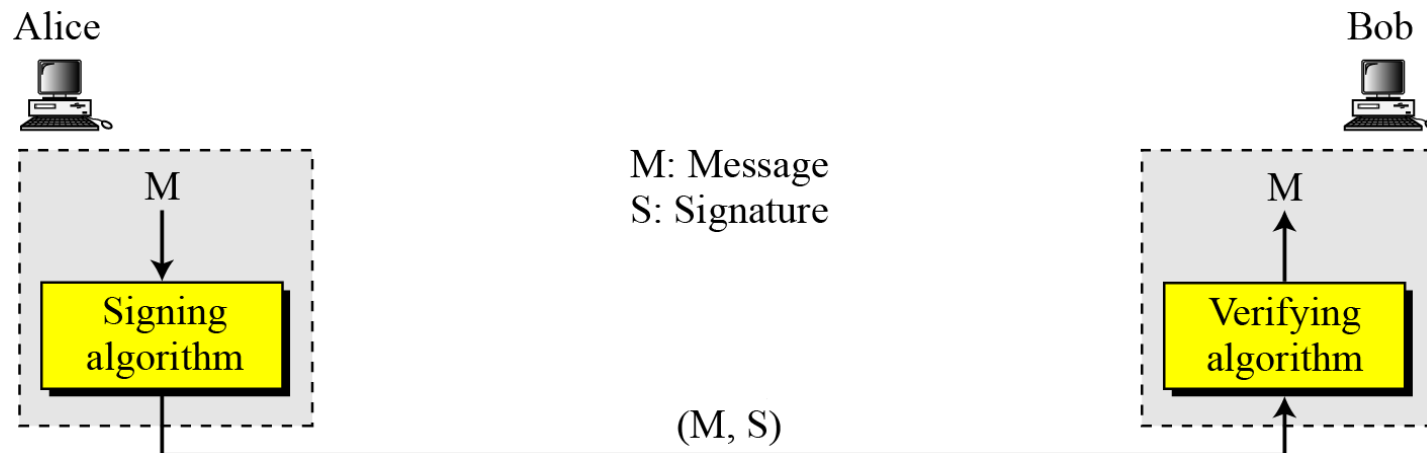


# Digital Signatures

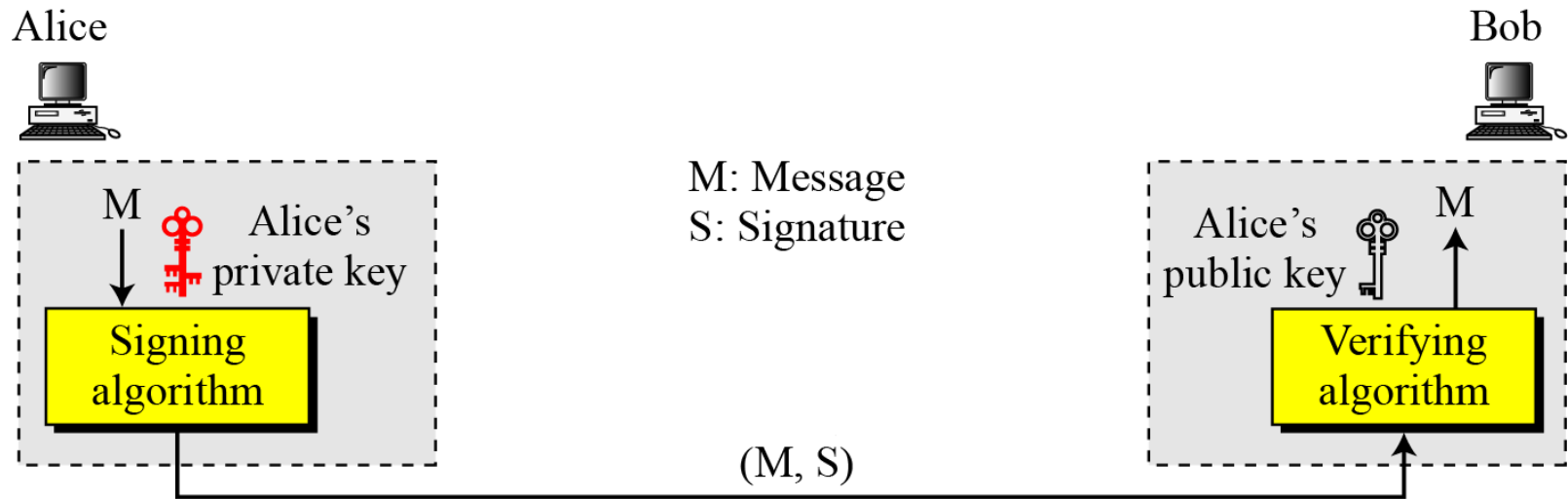
- ✓ The authority of many legal, financial and other documents is determined by the presence or absence of an authorized handwritten signature.
- ✓ **Inclusion:** A conventional signature is **included** in the document; it is part of the document. But when we sign a document digitally, we send the signature as a separate document.
- ✓ For a conventional signature, when the recipient receives a document, she compares the signature on the document with the signature on file i.e. on other paper.
- ✓ **Verification:** For a digital signature, the recipient receives the message and the signature. The recipient needs to apply a **verification** technique to the combination of the message and the signature to verify the authenticity.
- ✓ **Relationship:** For a conventional signature, there is normally a one-to-many **relationship** between a signature and documents. For a digital signature, there is a one-to-one relationship between a signature and a message.

- ✓ **Duplicity:** In conventional signature, a copy of the signed document can be distinguished from the original one on file. In digital signature, there is no such distinction unless there is a factor of time on the document.

Figure below shows the digital **signature process**. The sender uses a signing algorithm to sign the message. The message and the signature are sent to the receiver. The receiver receives the message and the signature and applies the verifying algorithm to the combination. If the result is true, the message is accepted; otherwise, it is rejected.



A digital signature needs a public-key system. The signer signs with her private key; the verifier verifies with the signer's public key.



Adding key to the digital signature process

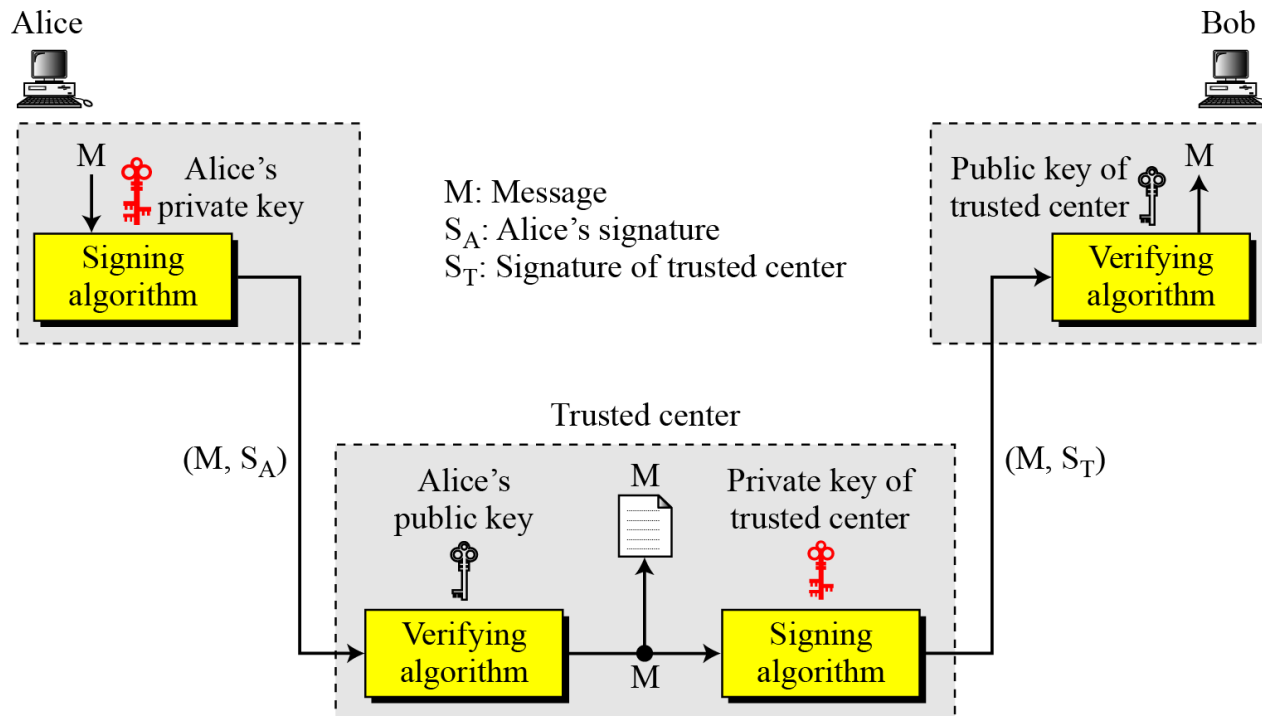
Basically, what is needed is a *system* by which one party can send a signed message to another party in such a way that the following conditions holds:

- ✓ The receiver can verify the claimed identity of the sender.
- ✓ The sender cannot later repudiate (deny) the contents of the message.
- ✓ The receiver cannot possibly have concocted the message himself.

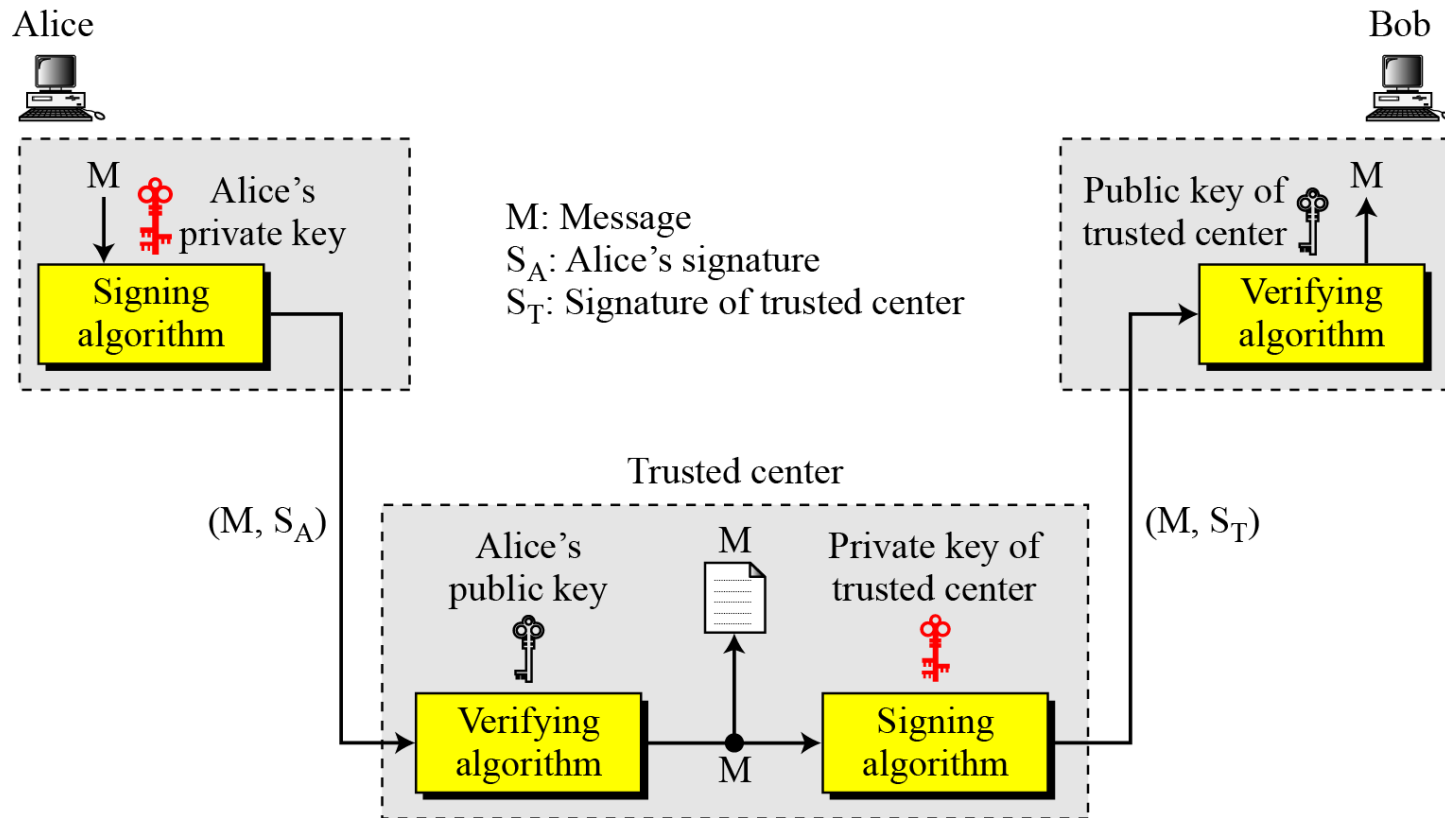
Two of the most important service of digital signature are **Non-repudiation** and **Confidentiality**.

# Non-repudiation:

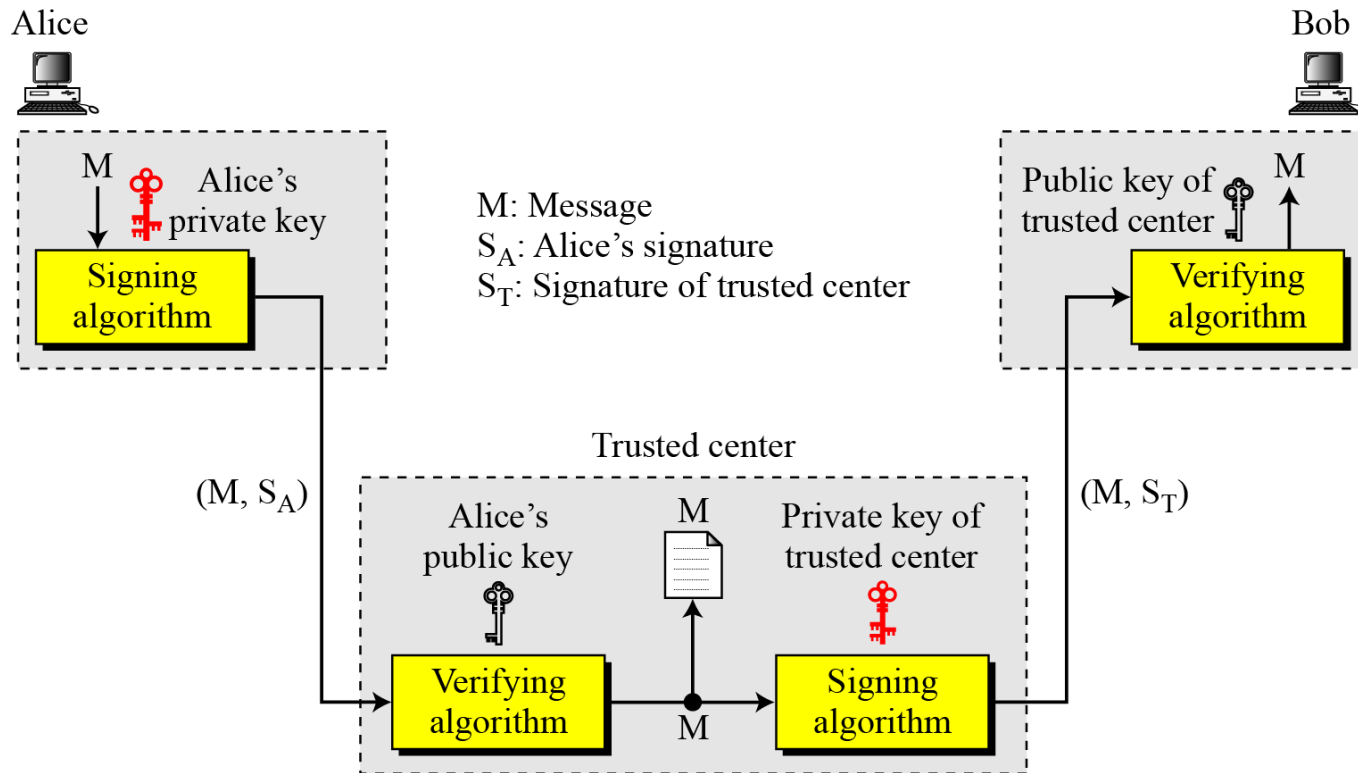
- ✓ If Alice signs a message and then denies it, can Bob later prove that Alice actually signed it?
- ✓ One solution is trusted third party say big brother (BB). People can create an established trusted third party among themselves.



- ✓ Alice creates a signature  $S_A$  from her message  $M$  and sends the message, her identity, Bob's identity and the signature to the center/BB. The center first checks that Alice's public key is valid then saves a copy of the message with the sender identity, recipient identity and a timestamp in its archive.



- ✓ The center uses its private key to create another signature  $S_T$ , Alice's identity and Bob's identity to Bob. Bob verifies the message using the public key of the trusted center.



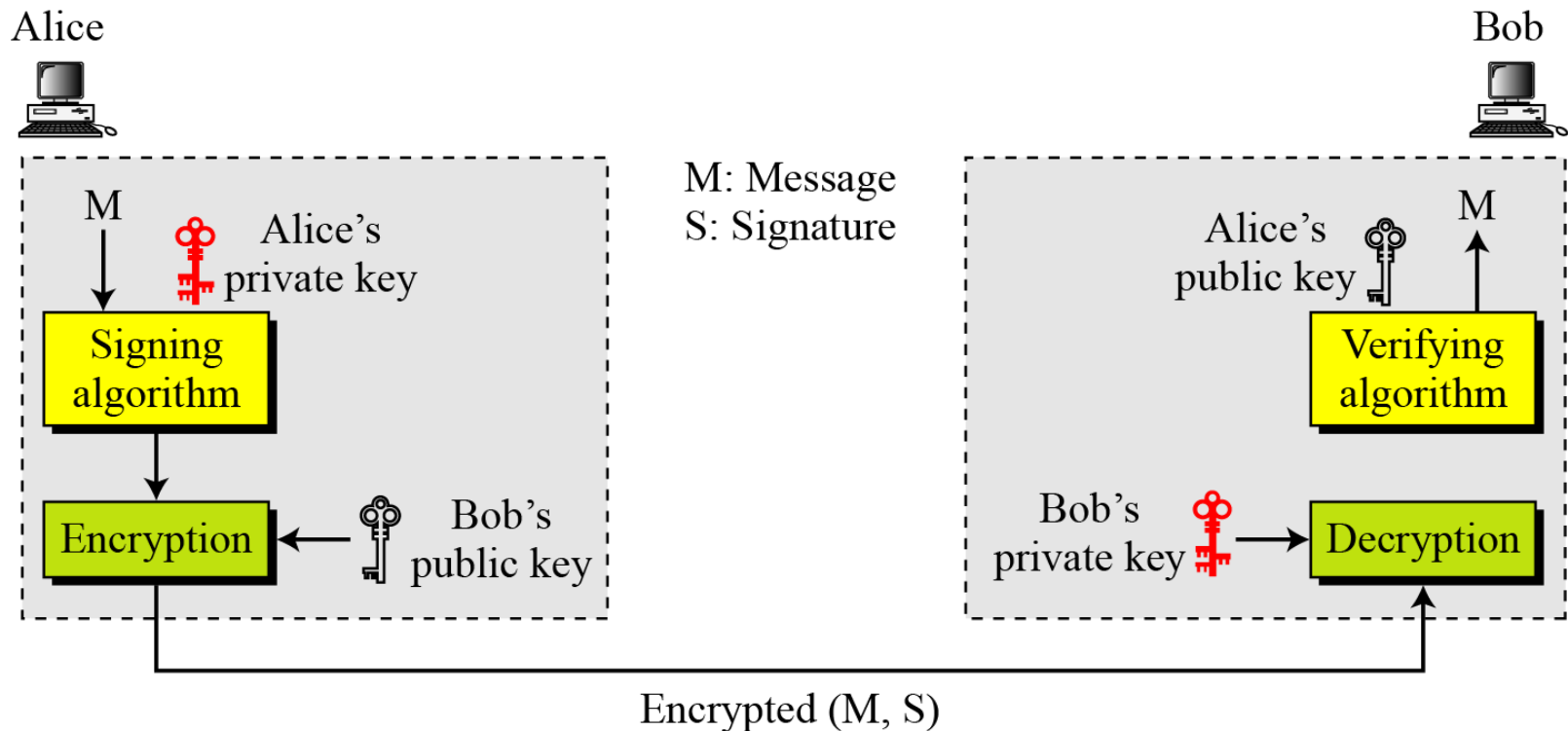
In future if Alice denies that she sent the message, the center will show the a copy of the saved message. Now Alice will lose the dispute.



# Confidentiality:

- ✓ For digital signature everyone has to agree to trust Big Brother, since Big Brother gets to read all signed messages. The most logical candidates for running the Big Brother server are the government, the banks, the accountants and the lawyers.
- ✓ Unfortunately none of these organizations inspire total confidence in all citizens. Hence it would be nice if signing documents did not require a trusted authority.

A digital signature does not provide privacy. If there is a need for privacy, another layer of encryption/decryption must be applied. In this case the message and the signature must be encrypted using either a secret-key or public-key cryptosystem.



Adding confidentiality to a digital signature scheme

## Problems:

1. Bob can prove that a message was sent by Alice only as long as  $K_A$  remains secret. If Alice discloses her secret key, the argument no longer holds, because anyone could have sent the message including Bob himself.
2. An other problem with signature scheme is what happens if Alice decides to change her key. Doing so is clearly legal, and it is probably a good idea to do so periodically. If a court case later arises, as described above, the judge will apply the current key  $K_A$  to *encrypted message* and discover that it does not produce *plain text*  $P$ . Bob will look pretty stupid at this point.

# Message Digests

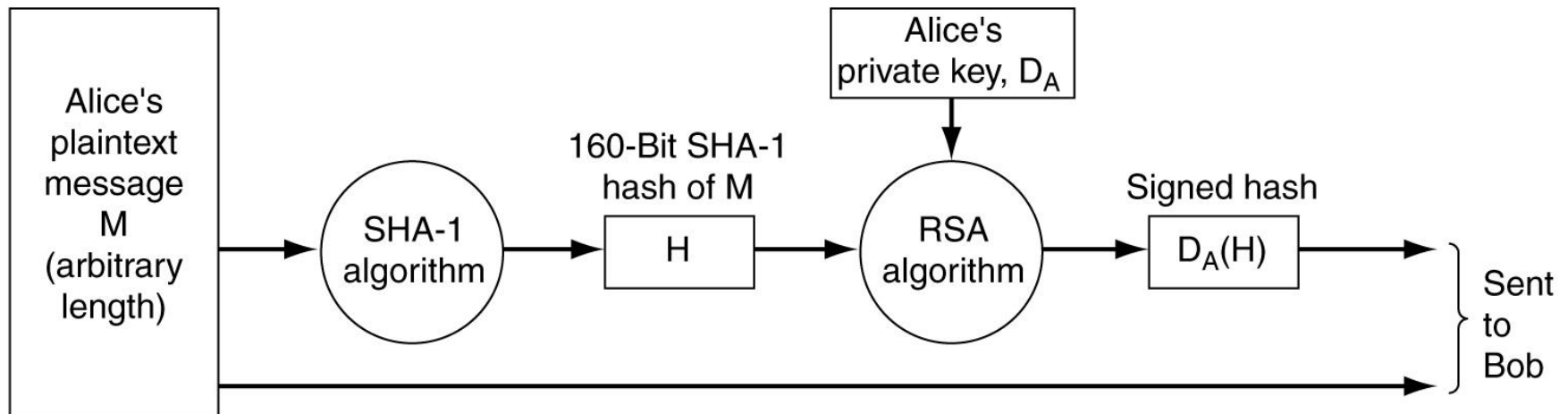
This scheme is based on the idea of a one way hash function that takes an arbitrary long piece of plaintext and from it computes a fixed length bit string. This hash function,  $MD$  often called a message digest, has four important properties:

1. Given  $P$ , it is easy to compute  $MD(P)$ .
2. Given  $MD(P)$ , it is effectively impossible to find  $P$ .
3. Given  $P$  no one can find  $P'$  such that  $MD(P') = MD(P)$ .
4. A change to the input of even 1 bit produces a very different output.

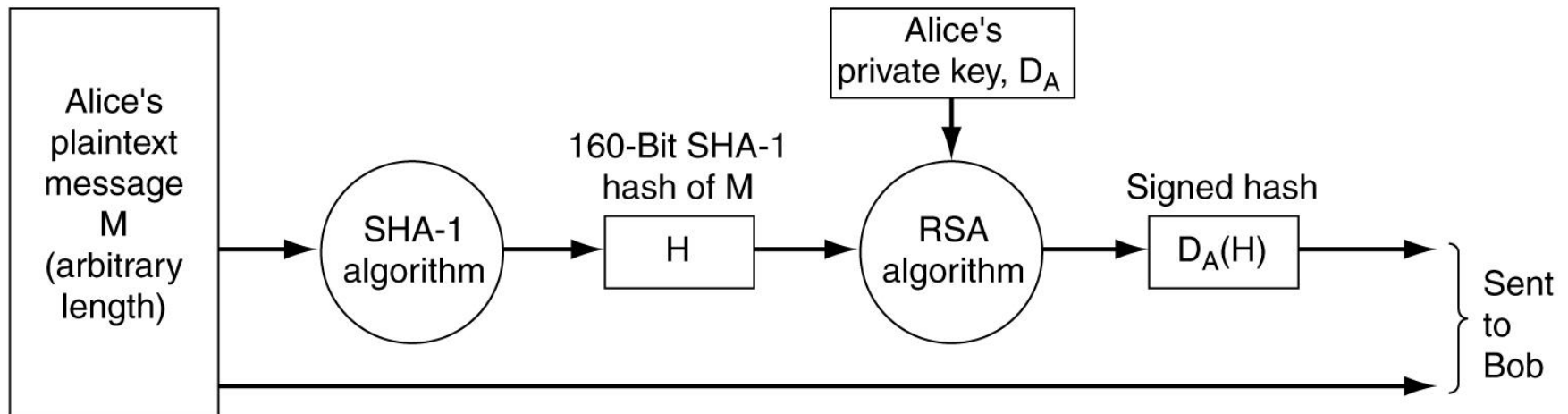
Computing a message digest from a piece of plaintext is much faster than encrypting that plaintext with a public-key algorithm, so message digests can be used to speed up digital signature algorithms.

# SHA-1

The other major message digest function is SHA-1 (Secure Hash Algorithm-1). A typical way for Alice to send a nonsecret but signed message to Bob is illustrated in fig. below. Here her plaintext message is fed into the SHA-1 algorithm to get a 160-bit SHA-1 hash. Alice then signs the hash with her RSA private key and sends both the plaintext message and the signed hash to Bob.



After receiving the message, Bob computes the SHA-1 hash himself and also applies, Alice's public key to the signed hash to get the original hash,  $H$ . If the two agree, the message is considered valid. Since there is no way for intruder to modify the message (plaintext) while it is in transit and produce a new one that hashes to  $H$ , Bob can easily detect any change.



If a dispute arises, Bob can show both  $P$  and SHA-1 hash. However since it is effectively impossible for Bob to find any other message that gives this hash, the judge will easily be convinced that Bob is telling the truth. Using message digest in this way saves both encryption time and message transport cost.

# IP Security (IPsec)

✓ One of the weaknesses of the original Internet Protocol is that it lacks any sort of general purpose mechanism for ensuring the authenticity and privacy of data as it is passed over the internetwork. Since IP datagrams must usually be routed between two devices over unknown networks, any information in them is subject to being intercepted and even possibly changed. With the increased use of the Internet for critical applications, security enhancements were needed for IP. To this end, a set of protocols called *IP Security* or *IPSec* was developed.

✓ **Internet Protocol Security (IPsec)** is a technology protocol suite for securing Internet Protocol (IP) communications by authenticating and encrypting each IP packet of a communication session.

✓ It can be used in protecting data flows between a pair of hosts (*host-to-host*), between a pair of security gateways (*network-to-network*), or between a security gateway and a host (*network-to-host*)



# IP Security (IPsec)

IPsec provides three degrees of freedom.

- ✓ First, it is highly **modular**, allowing users to select from a variety of *cryptographic algorithms* and *specialized security protocols*.
- ✓ Second, IPsec allows users to select from a large menu of security properties, including *access control*, *integrity*, *authentication*, *originality*, and *confidentiality*.
- ✓ Third, IPsec can be used to protect narrow streams (e.g., packets belonging to a particular TCP connection being sent between a pair of hosts) or wide streams (e.g., all packets flowing between a pair of gateways).

# Basic Concepts:

**Authentication** Verifies that each datagram was originated by the claimed sender that the sender is not impersonated by third party. (data origin authentication).

**Integrity** Verifies that the contents of a datagram were not changed in transit, either deliberately or due to random errors.

**Confidentiality** With certain security mechanism (so-called encryption/decryption), data is protected during transmission from third party 's knowing the content. (Conceals the content of a message, typically by using encryption.)

**Security Association (SA)** An agreement between two communication parties on knowing and using certain combination of security mechanisms for data transmission between them. It's based on destination address and a certain index, called Security Parameters Index (SPI).

# Transport and Tunnel Modes

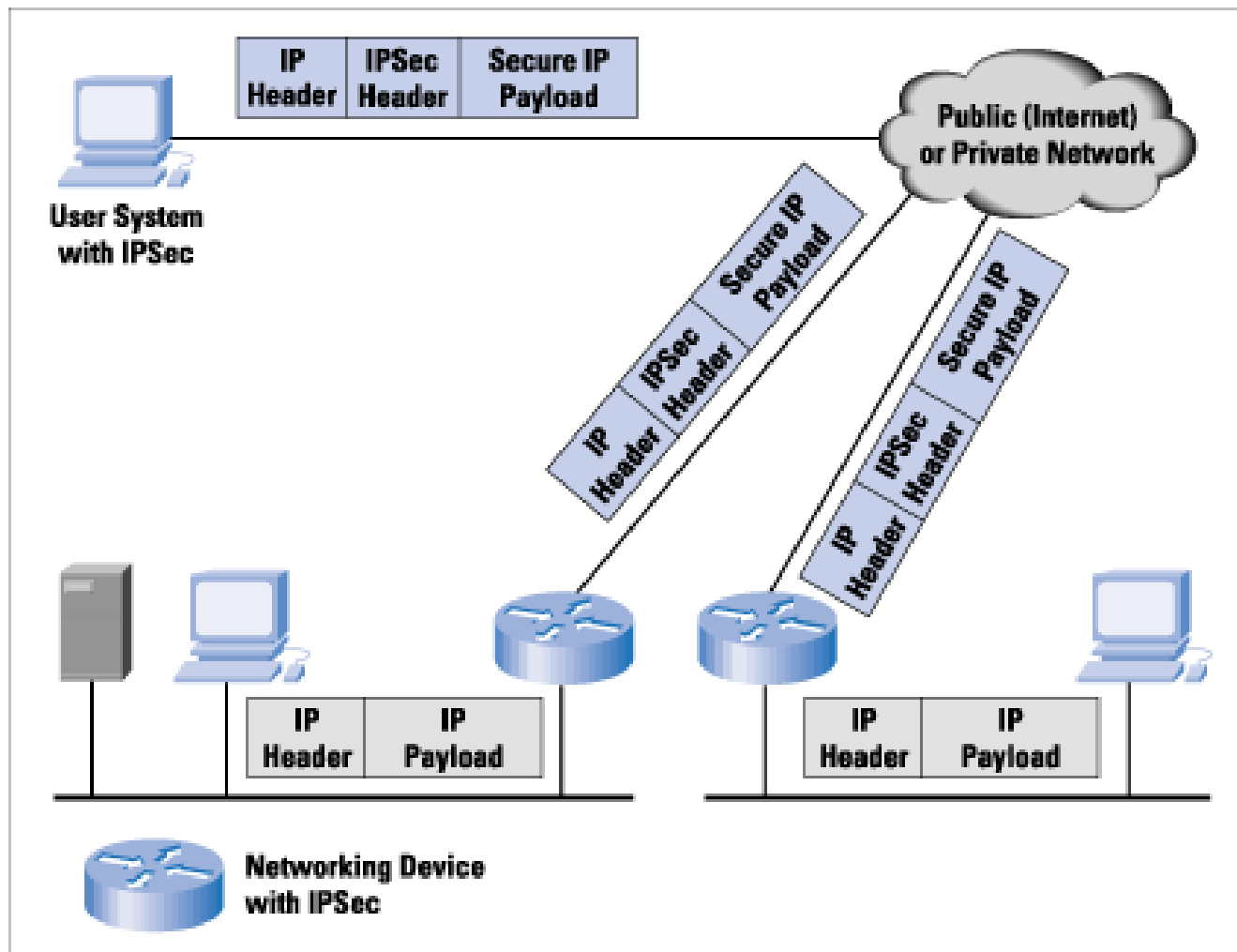
IPsec can be used in either of two modes. In *transport mode*, the IPsec header is inserted just after the IP header. In *tunnel mode*, the entire IP packet, header and all, is encapsulated in the body of a new IP packet with completely new IP header.

✓ **Transport mode is used to encrypt & optionally authenticate IP data**

- \* data protected but header left in clear
- \* can do traffic analysis but is efficient
- \* good for ESP host to host traffic

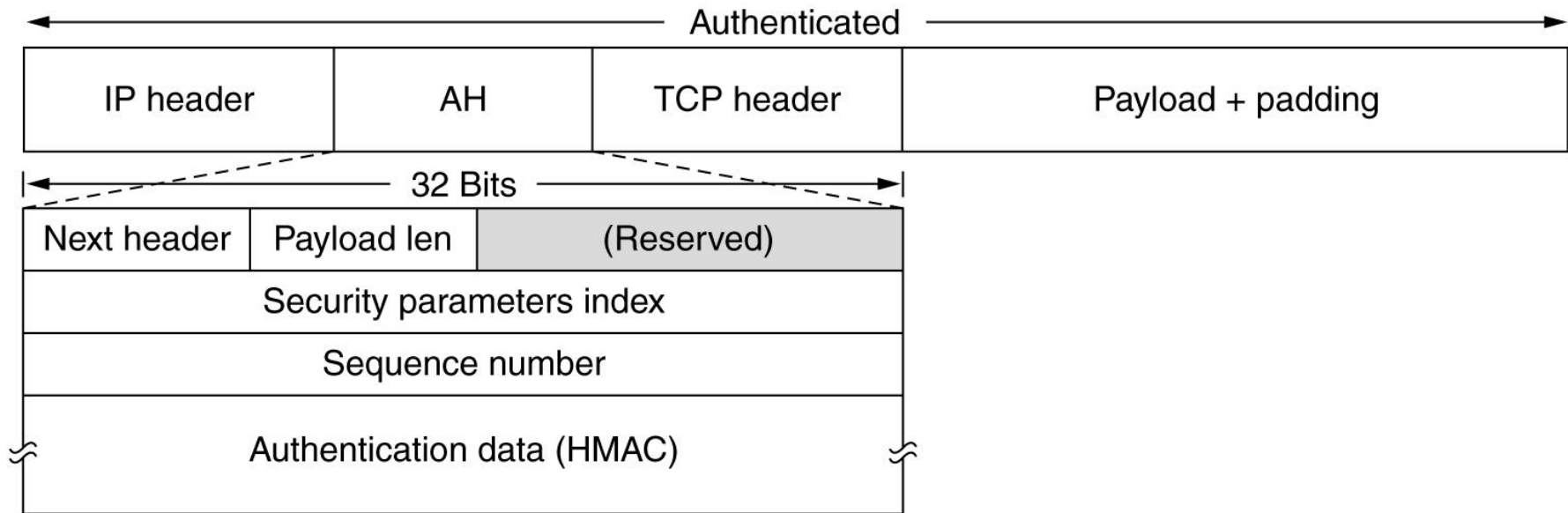
✓ **Tunnel mode encrypts entire IP packet**

- \* add new header for next hop
- \* good for VPNs, gateway to gateway security



An IP Security Scenario

The IPsec authentication header in *transport mode* for IPv4.



***Next Header:*** The Next Header is an 8-bit field that identifies the type of the next payload after the Authentication Header. The value of this field is chosen from the set of IP Protocol Numbers defined on the web page of Internet Assigned Numbers Authority (IANA). For example, a value of 4 indicates IPv4, a value of 41 indicates IPv6, and a value of 6 indicates TCP.

***Payload Length:*** This 8-bit field specifies the length of AH in 32-bit words (4-byte units), minus "2". For IPv6, the total length of the header must be a multiple of 8-octet units. (Note that although IPv6 characterizes AH as an extension header, its length is measured in 32-bit words, not the 64-bit words used by other IPv6 extension headers.) .

***Reserved:*** This 16-bit field is reserved for future use. It MUST be set to "zero" by the sender, and it SHOULD be ignored by the recipient.

**Security Parameters Index:** The *Security Parameters Index* is the connection identifier tag. It is inserted by the sender to indicate a particular record in the receiver's database. The record contains the shared key used on this connection and other information about the connection.

### **Sequence Number**

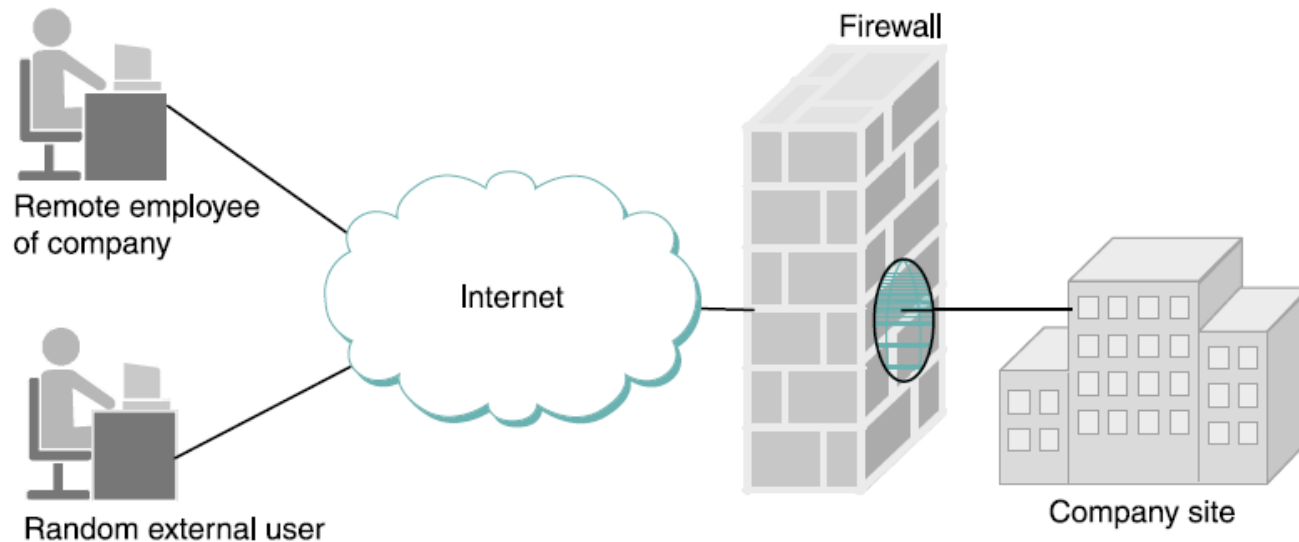
This unsigned 32-bit field contains a counter value that increases by one for each packet sent, i.e., a per-SA packet sequence number.

***Authentication Data:*** It is a variable-length field contains the payload's digital signature. Normally public-key-cryptography is not used here because packets must be processed extremely rapidly and all known public key algorithms are too slow. One simple way is to compute the hash over the packet plus the shared key. The shared key is not transmitted, of course. A scheme like this is called an HMAC (Hashed Message Authentication Code)



# Firewall

A firewall is a combination of hardware and software that isolates an organization's internal network from the Internet at large, allowing some packets to pass and blocking others.



A firewall filters packets flowing between a site and the rest of the Internet

- ✓ The firewall acts as a **packet filter**. It inspects each and every incoming and outgoing packet. Packets meeting some criterion described in rules formulated by the network administrator are forwarded normally. Those that fail the test are dropped.

Filtering decisions are typically based on:

- ❖ IP source or destination address
- ❖ Protocol type in IP datagram field: TCP, UDP, ICMP, OSPF, and so on
- ❖ TCP or UDP source and destination port
- ❖ TCP flag bits: SYN, ACK, and so on
- ❖ ICMP message type
- ❖ Different rules for datagrams leaving and entering the network
- ❖ Different rules for the different router interfaces

A network administrator configures the firewall based on the policy of the organization like table below.

Policy	Firewall Setting
No outside Web access.	Drop all outgoing packets to any IP address, port 80
No incoming TCP connections, except those for organization's public Web server only.	Drop all incoming TCP SYN packets to any IP except 130.207.244.203, port 80
Prevent Web-radios from eating up the available bandwidth.	Drop all incoming UDP packets — except DNS packets.
Prevent your network from being used for a smurf DoS attack.	Drop all ICMP ping packets going to a "broadcast" address (eg 130.207.255.255).
Prevent your network from being tracerouted	Drop all outgoing ICMP TTL expired traffic

- ✓ A filtering policy can be based on a combination of addresses and port numbers. For example, a filtering router could forward all Telnet datagrams (those with a port number of 23) except those going to and coming from a list of specific IP addresses.
- ✓ For example, a firewall might be configured to filter out (not forward) all packets that match the following description:  
    <192.12.13.14, 1234, 128.7.6.5, 80>
- ✓ This pattern says to discard all packets from port 1234 on host 192.12.13.14 addressed to port 80 on host 128.7.6.5. (Port 80 is the well-known TCP port for HTTP.)
- ✓ Of course it's often not practical to name every source host whose packets you want to filter, so the patterns can include wildcards. For example, <\*, \*, 128.7.6.5, 80>, says to filter out all packets addressed to port 80 on 128.7.6.5, regardless of what source host or port sent the packet.

# Stateful Packet Filters

- ✓ **Stateful filters** keeps tracking of all ongoing TCP connections in a connection table. This is possible because the firewall can observe the beginning of a new connection by observing a three-way handshake (SYN, SYNACK, and ACK); and it can observe the end of a connection when it sees a FIN packet for the connection.

- ✓ An example connection table for a firewall is shown in Table below. This connection table indicates that there are currently three ongoing TCP connections, all of which have been initiated from within the organization.

source address	dest address	source port	dest port
222.22.1.7	37.96.87.123	12699	80
222.22.93.2	199.1.205.23	37654	80
222.22.65.143	203.77.240.43	48712	80

- ✓ Additionally, the stateful filter includes a new column, “check connection,” in its access control list, based on rules of packet blocking.

- ✓ For example, an **Stateful filters** allow an external Web server to send packets to an internal host, but only if the internal host first establishes a connection with the external Web server. Such a rule is not possible with stateless designs that must either pass or drop all packets from the external Web server.

# Application Gateway

- ✓ Rather than just looking at raw packets, the **application gateway** operates at the application level.
  - ✓ A mail gateway for example, can be set up to examine each message going in or coming out. For each one the gateway decides whether to transmit or discard the message based on header fields, message size, or even the content (i.e. at a military installation, the presence of words like 'nuclear' or 'bomb' might cause some special action to be taken).
- 
- ✓ Let's design a firewall that allows only a restricted set of internal users to Telnet outside and prevents all external clients from Telneting inside. Such a policy can be accomplished by implementing a combination of a packet filter (in a router) and a Telnet application gateway.



- ✓ Even if the firewall is perfectly configured, plenty of security problems still exists. For example, if a firewall is configured to allow in packets from only specific networks ( i.e. company's other plants), an intruder outside the firewall can put in false source address to bypass this check.
- ✓ If an insider wants to ship out secret documents , he can encrypt them or even convert the message in jpeg image which bypasses any word filter.