UNIT-I

- Two developments lead to '**computer security**' and '**network security**'.
- **computer security** deals with collection of tools designed to protect data.
- **Network security** measures are needed to protect data during transmission.

Three aspects of IS are:

- **Security Attack:** Any action that comprises the security of information.
- **Security Mechanism:** A mechanism that is designed to detect, prevent, or recover from a security.

- **Security Service:** It is a processing or communication service that enhances the security of the data processing systems and information transfer.

❖ Security Attacks

- Security attacks can be classified in terms of **Passive attacks** and **Active attacks.**

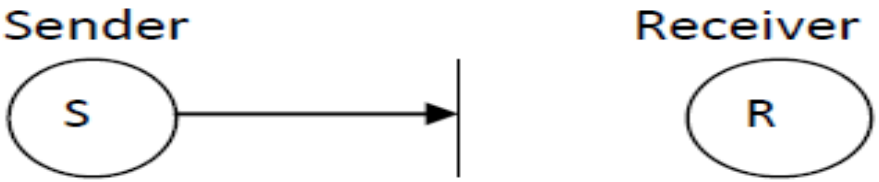
- **Different kinds of attacks are:**
- **Interruption**

An asset of the system is destroyed or becomes unavailable or unusable. It is an attack on availability.

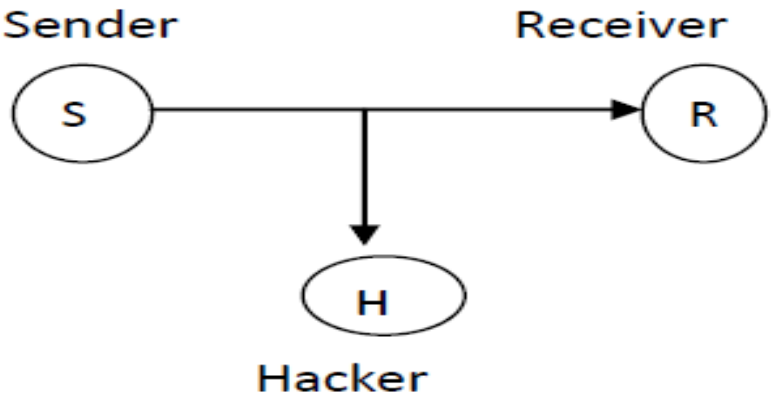
Examples:

- Destruction of some hardware
- jamming wireless signals
- Disabling file management systems

Interruption



Interception



- **Interception**
- An unauthorized party gains access to an asset. Attack on confidentiality.

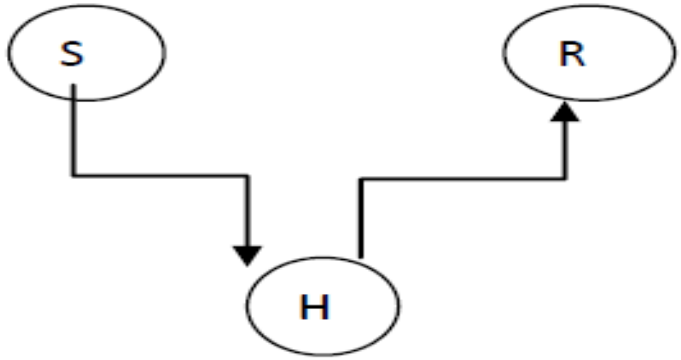
Examples:

- Wire tapping to capture data in a network.
- Illicitly copying data or programs
- Eavesdropping
- **Modification:**
- When an unauthorized party gains access and tampers an asset. Attack is on Integrity.

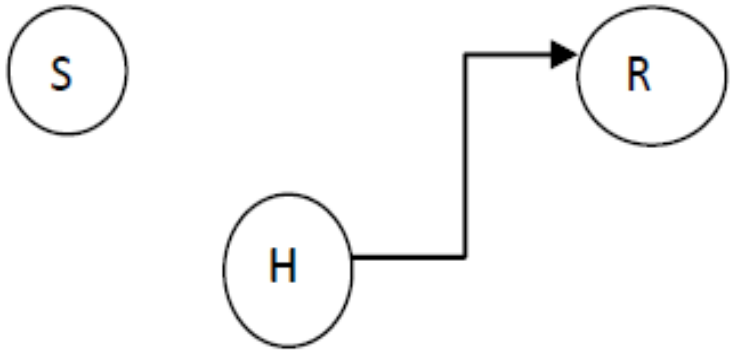
Examples:

- Changing data file
- Altering a program and the contents of a message

Modification:



Fabrication



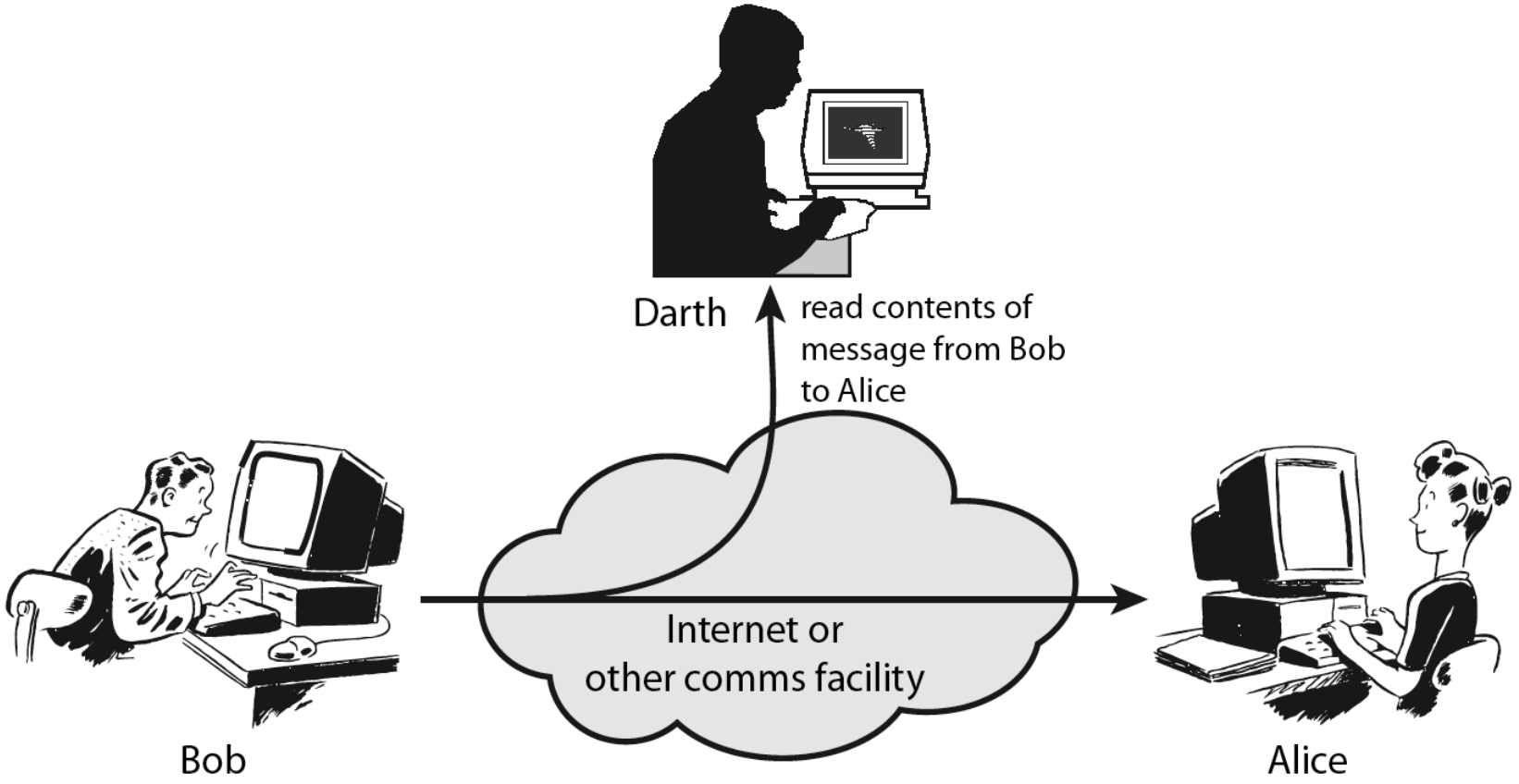
- **Fabrication**
- An unauthorized party inserts a counterfeit object into the system. Attack on Authenticity. Also called **impersonation.**

Examples:

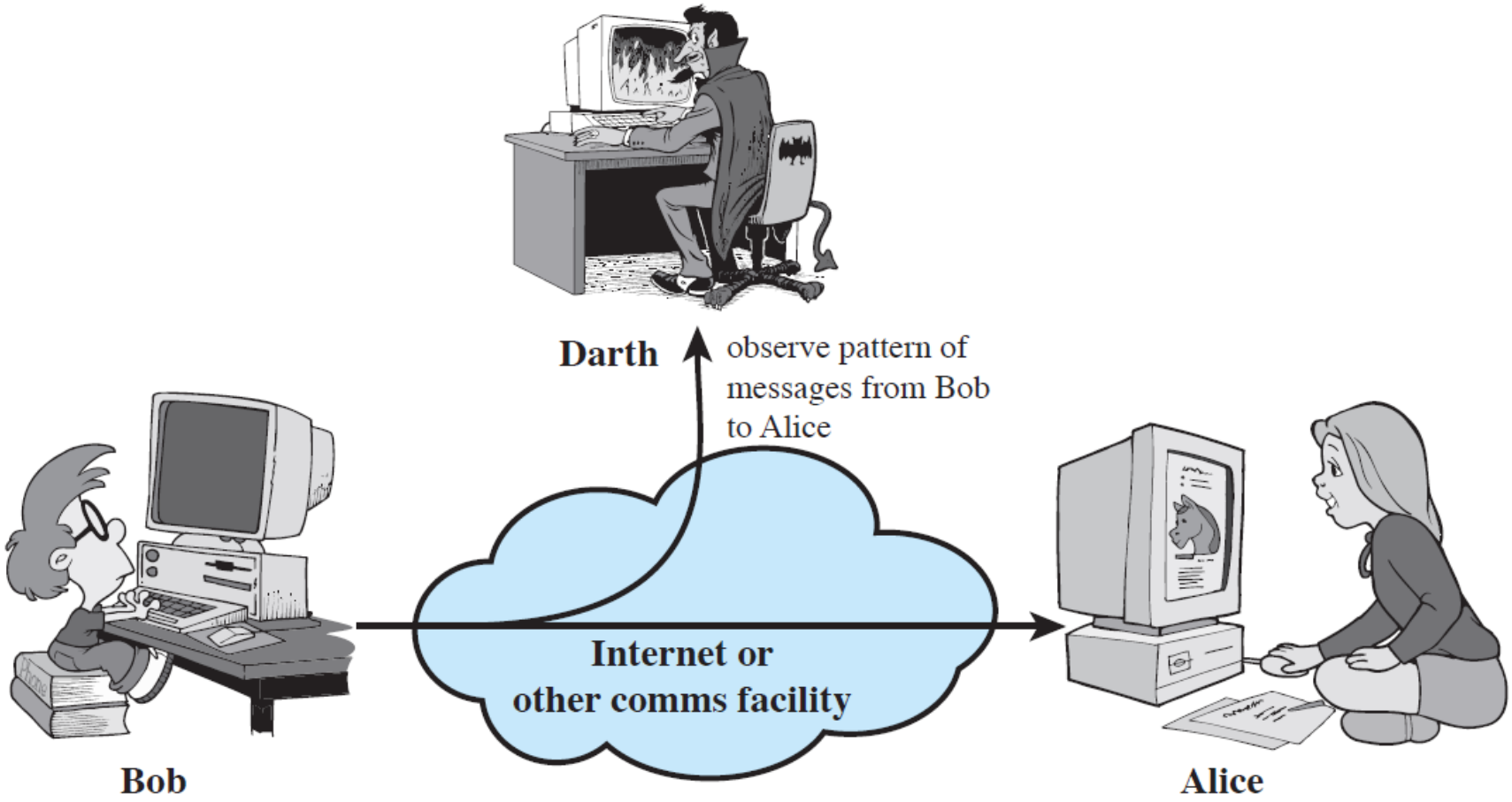
- Hackers gaining access to a personal email and sending message
- Insertion of records in data files
- insertion of spurious messages in a network

- **Passive Attacks:**
- A Passive attack attempts to learn or make use of information from the system, but does not affect system resources.
- **Two types:**
- **Release of message content**
- It may be desirable to prevent the opponent from learning the contents (i.e sensitive or confidential info) of the transmission.

Passive Attacks (1)
Release of Message Contents



Passive Attacks (2)
Traffic Analysis

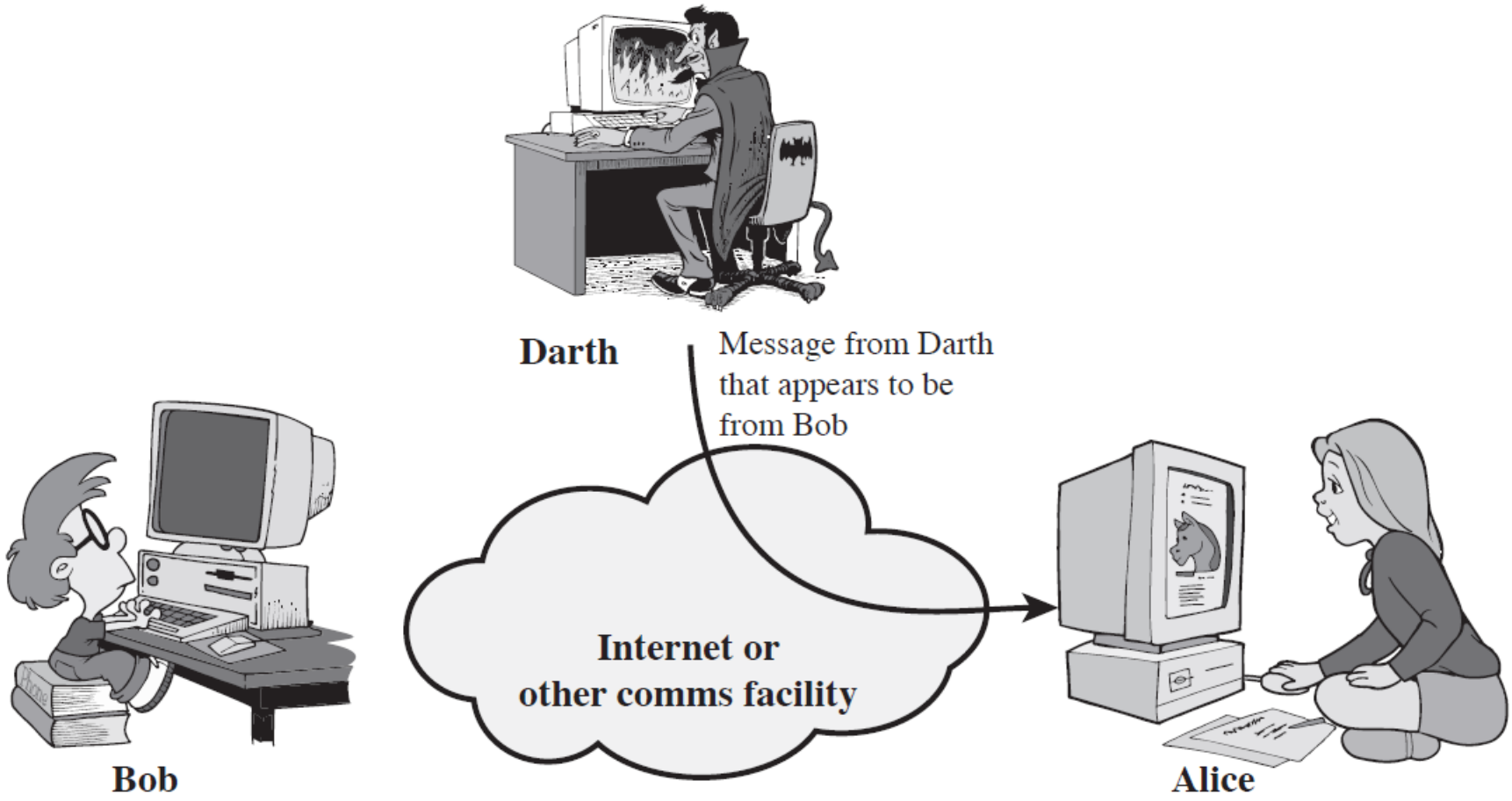


- **Traffic analysis:**
- A more subtle technique where the opponent could **determine** the location and identity of communicating hosts and could observe the frequency & length of encrypted messages being exchanged there by guessing the nature of communication taking place.

- **Passive attacks:**
- Passive attacks are very difficult to detect because they do not involve any alternation of the data.
- As the communications take place in a very normal fashion, neither the sender nor receiver is aware that *a* third party has read the messages or observed the traffic pattern.
- So, the emphasis in dealing with passive attacks is on prevention rather than detection.

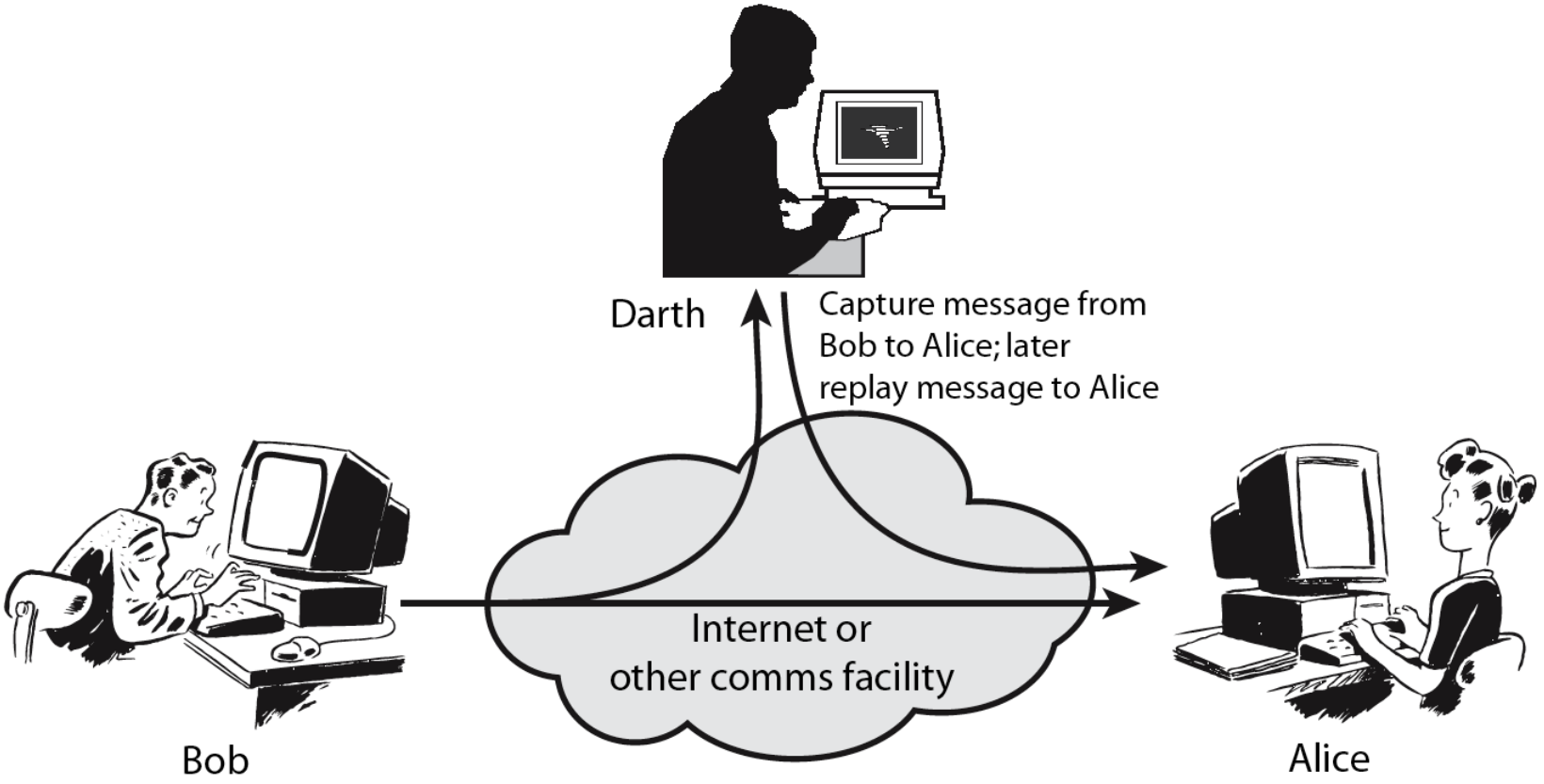
- **ActiveAttacks:** Activeattacks involve some modification of the data stream or creation of a false stream.
- An active attack attempts to alter system resources or affect their operation.
- ***Four types:***
- **Masquerade:** Here, an entity pretends to be some other entity. It usually includes one of the other forms of active attack.

Active Attacks (1)
Masquerade

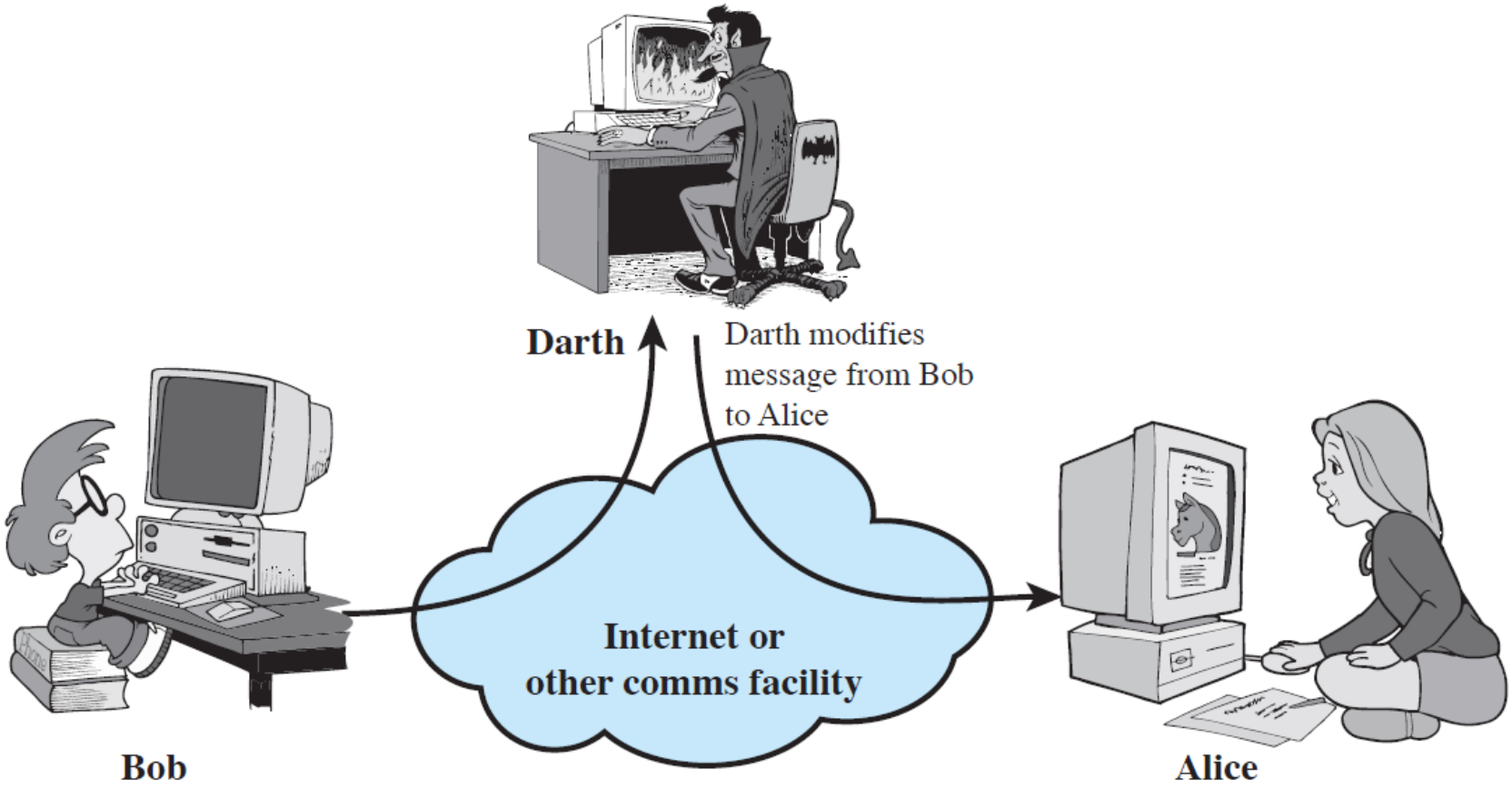


- **Replay:** It involves the passive capture of a data unit and its subsequent retransmission to produce an unauthorized effect.
- **Modification of messages:** It means that some portion of a legitimate message is altered, or that messages are delayed to produce an unauthorized effect.
- Ex: “John’s acc no is 2346” is modified as “John’s acc no is 7892”

Active Attacks (2)
Replay

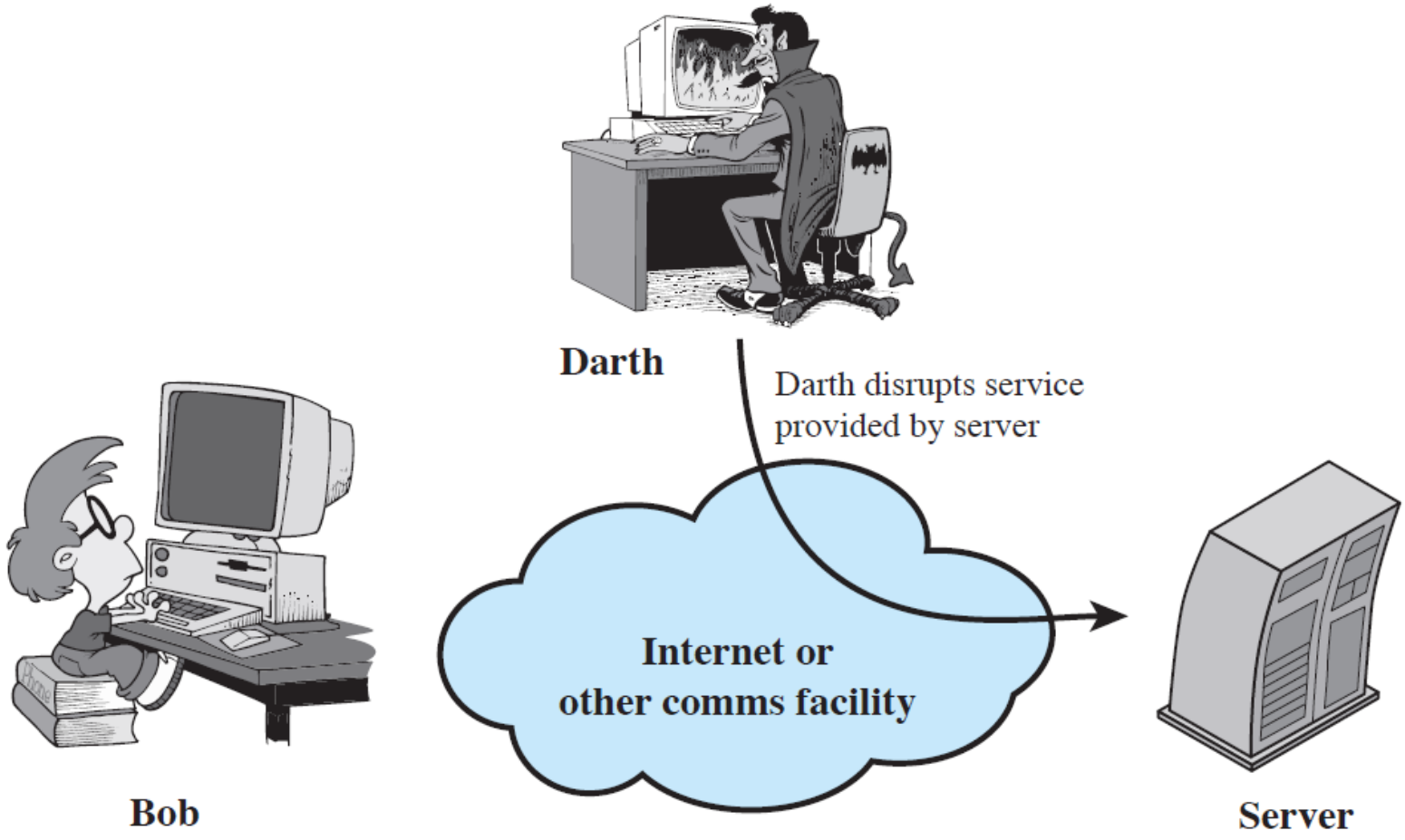


Active Attacks (3)
Modification of Messages



- **Denial of service:** This attack prevents or inhibits the normal use or management of communication facilities.
- Ex:
- a: Disruption of entire network by disabling it
- b: Suppression of all messages to a particular destination by a third party.

Active Attacks (4)
Denial of Service



Active attacks:

- It is quite difficult to **prevent** active attacks absolutely, because of the wide variety of potential physical, software and network vulnerabilities.

- Instead, the goal is to **detect** active attacks and to **recover** from any disruption or delays caused by them.

➢ Security Services:

- ➢ It is a processing or communication service that is provided by a system to give a specific kind of protection to system resources.

- ➢ Security services implement security policies and are implemented by security mechanisms.

Authentication

- Concerned with assuring that a communication is authentic In the case of a single message, assures the recipient that the message is from the source that it claims to be from.
- In the case of ongoing interaction, assures the two entities are authentic and that the connection is not interfered with in such a way that a Third party can masquerade as one of the two legitimate parties

- **PEER ENTITY AUTHENTICATION**

  Verifies the identities of the peer entities involved in communication. Provides confidence against a **masquerade or a replay attack**

- **DATA ORIGIN AUTHENTICATION**

Assumes the authenticity of source of data unit, but does not provide protection against duplication or modification of data units.

Access Control

- The ability to limit and control the access to host systems and applications via communications links
- To achieve this, each entity trying to gain access must first be indentified, or authenticated, so that access rights can be tailored to the individual

Data Confidentiality

- The protection of transmitted data from passive attacks Broadest service protects all user data transmitted between two users over a period of time
- Narrower forms of service include the protection of a single message or even specific fields within a message The protection of traffic flow from analysis
- This requires that an attacker not be able to observe the source and destination, frequency, length, or other characteristics of the traffic on a communications facility

Data Integrity

- Can apply to a stream of messages, a single message, or selected fields within a message
- Connection-oriented integrity service deals with a stream of messages and assures that messages are received as sent with no duplication, insertion, modification, reordering, or replays
- A connectionless integrity service deals with individual messages without regard to any larger context and generally provides protection against message modification only

Nonrepudiation

- Prevents either sender or receiver from denying a transmitted message
- When a message is sent, the receiver can prove that the alleged sender in fact sent the message
- When a message is received, the sender can prove that the alleged receiver in fact received the message

Availability service

- Availability

  The property of a system or a system resource being accessible and usable upon demand by an authorized system entity, according to performance specifications for the system

- Availability service

  One that protects a system to ensure its availability

  Addresses the security concerns raised by denial-of-service attacks

  Depends on proper management and control of system resources

Security Mechanisms:

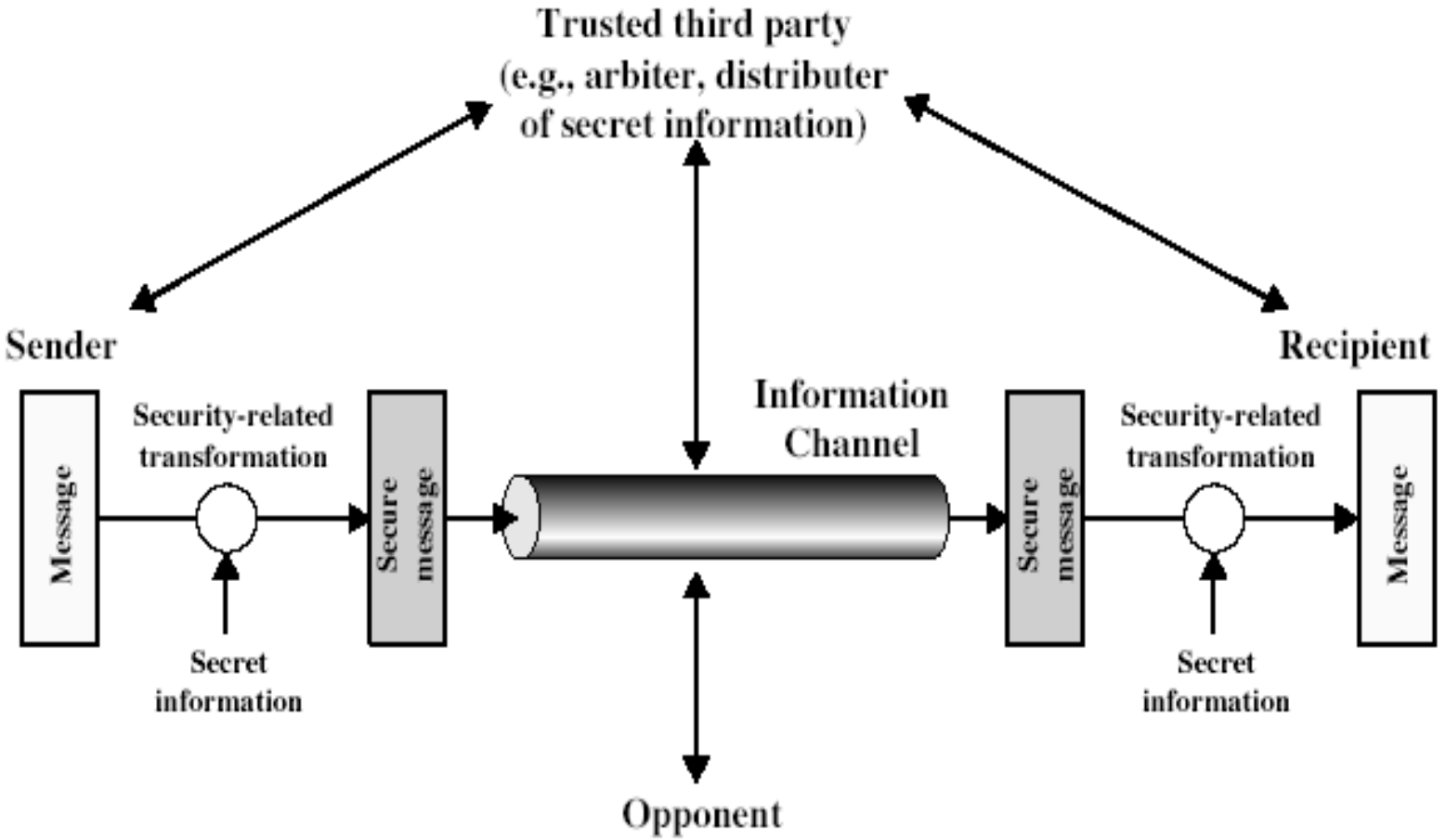
- **Specific Security Mechanisms:**
- **Encipherment:** It refers to the process of applying mathematical algorithms for converting data into a form that is not intelligible. This depends on algorithm used and encryption keys.
- **Digital Signature:** The appended data or a cryptographic transformation applied to any data unit allowing to prove the source and integrity of the data unit and protect against forgery.
- **Access Control:** A variety of techniques used for enforcing access permissions to the system resources.
- **Data Integrity:** A variety of mechanisms used to assure the integrity of a data unit or stream of data units.

- **Authentication Exchange:** A mechanism intended to ensure the identity of an entity by means of information exchange.
- **Traffic Padding:** The insertion of bits into gaps in a data stream to frustrate traffic analysis attempts.
- **Routing Control:** Enables selection of particular physically secure routes for certain data and allows routing changes once a breach of security is suspected.
- **Notarization:** The use of a trusted third party to assure certain properties of a data exchange

pervasive security mechanisms:

- **Trusted Functionality:** That which is perceived to be correct with respect to some criteria
- **Security Level:** The marking bound to a resource (which may be a data unit) that names or designates the security attributes of that resource
- **Event Detection:** It is the process of detecting all the events related to network security
- **Security Audit Trail**: Data collected and potentially used to facilitate a security audit, which is an independent review and examination of system records and activities
- **Security Recovery:** It deals with requests from mechanisms, such as event handling and management functions, and takes recovery actions

A Model Of Inter Network Security



- The general model shows that there are **four** basic tasks in designing a particular security service:

1. **Design an algorithm** for performing the security-related transformation. The algorithm should be such that an opponent cannot defeat its purpose
2. **Generate the secret information** to be used with the algorithm
3. **Develop methods** for the distribution and sharing of the secret information
4. **Specify a protocol** to be used by the two principals that makes use of the security algorithm and the secret information to achieve a particular security service

Data is transmitted over network between two communicating parties, who must cooperate for the exchange to take place.

A logical information channel is established by defining a route through the internet from source to destination by use of communication protocols by the two parties

Two components are present in almost all the security providing techniques.

A **security-related transformation** on the information to be sent making it unreadable by the opponent, and the addition of a code based on the contents of the message, used to verify the identity of sender.

Some **secret information** shared by the two principals and, it is hoped, unknown to the opponent. An example is an encryption key used in conjunction with the transformation to scramble the message before transmission and unscramble it on reception

A **trusted third party** may be needed to achieve secure transmission. It is responsible for distributing the secret information to the two parties, while keeping it away from any opponent

Network Access Security Model

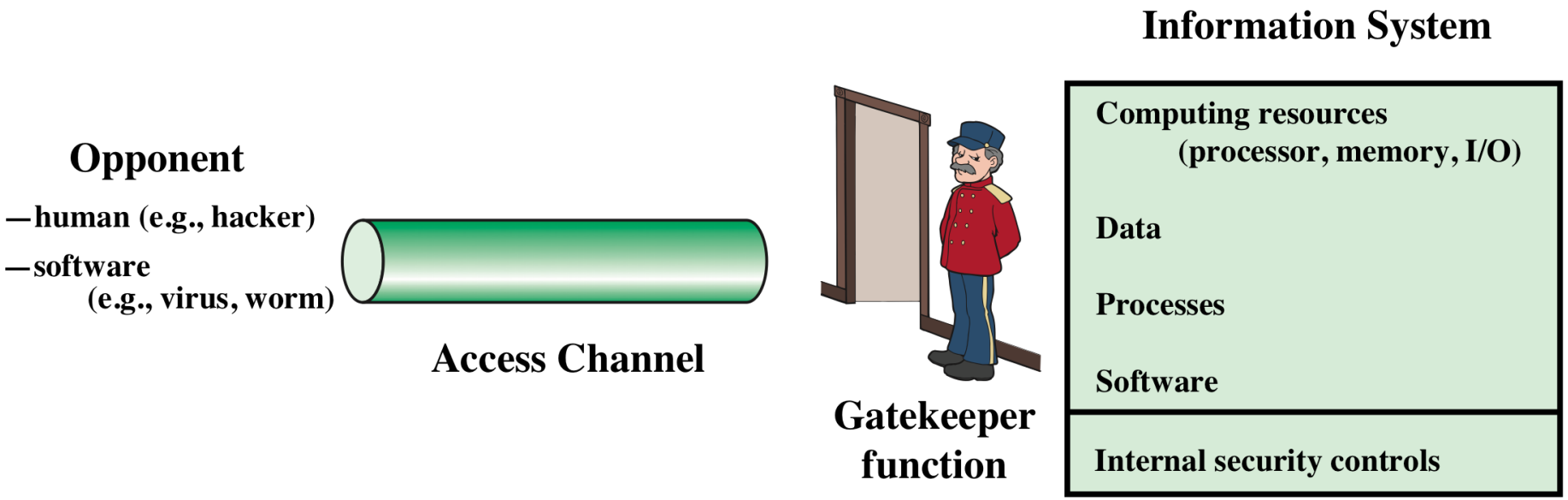


Figure 1.3 Network Access Security Model

Another threat is placement of some logic in computer system affecting various applications and utility programs. This inserted code presents two kinds of threats

Information access threats intercept or modify data on behalf of users who should not have access to that data

Service threats exploit service flaws in computers to inhibit use by legitimate users

The security mechanisms needed to cope with unwanted access fall into two broad categories

Placing a gatekeeper function, which includes a password-based login methods that provide access to only authorized users and screening logic to detect and reject worms, viruses etc

An internal control, monitoring the internal system activities analyzes the stored information and detects the presence of unauthorized users or intruders

Internet standards and RFCs

- The Internet society
  - Internet Architecture Board (IAB)
  - Internet Engineering Task Force (IETF)
  - Internet Engineering Steering Group (IESG)

Most of the protocols related to TCP/IP protocol suite are already standardized or under the process of standardization

An organization known as internet society is responsible for development and publication of these standards

An internet society refers to the organization responsible for monitoring and coordinating internet design, engineering and management

Three organizations under the internet society are responsible for actual work of standards development & publication

INTERNET ARICHITECTURE BOARD (IAB): Responsible for defining the overall architecture of the internet, providing guidance and broad direction to IETF

INETRNET ENGINEERING TASK FORCE (IETF): The protocol engineering and development arm of the internet

INTERNET ENGINEERING STEERING GROUP (IESG): Responsible for technical management of IETF activities and the internet standards process

The RFC publication process is shown below, in which a specification passes through a sequence of steps called standards track, in order to qualify as a standard.

It involves excessive scrutinizing and testing. The actual process starts after the approval of internet draft documentation as an RFC by IESG.

For a specification to act as a draft standard it must pass through at least two non- dependent interoperable implementations for achieving proper operational experience

once, necessary implementations and operational experience is achieved, it can be regarded as internet standard

Internet Standard Categories All the internet standards fall into two categories

TECHINICAL SPECIFICATION (TS): TS defines a protocol, service, procedure, convention or format. Most internet standards are TS‘s.

APPLICABILITY STATEMENT (AS): AS specifies how, and under what circumstances, one or more TS may be applied to support a particular internet capability.

Session Hijacking:

- **Session Hijacking** is security threat to which most systems are prone to.
- **Session hijack** is a process whereby the attacker inserts themselves into an existing communication session between two computers.
- The **three** main protocols that manage the data flow on which **session hijacking** occurs are **TCP, UDP, and HTTP**.

- **The goal of the TCP session hijacker** is to **create a state** where the client and server are unable to exchange data, so that he **can forge** acceptable packets for both ends, which **mimic** the real packets.
- **why the client and server will drop packets sent between them:**

➢ **because**

➢ the server's sequence number no longer matches the client's ACK number and likewise,

➢ the client's sequence number no longer matches the server's ACK number.

- **To hijack the session in the TCP network the hijacker should employ following techniques:**

- **IP Spoofing**: IP spoofing is “a technique used to gain unauthorized access to computers.
- **Blind Hijacking:** If source routing is disabled, the session hijacker can also employ blind hijacking where he injects his malicious data into intercepted communications in the TCP session. It is called “blind” because the hijacker can send the data or commands, but cannot see the response.
- **Man in the Middle attack (packet sniffing):** This technique involves using a packet sniffer that intercepts the communication between the client and server.

ARP Attacks

An ARP table controls the **Media Access Control (MAC)-address-to-IP-address mapping** on each machine

ARP is designed to be a dynamic protocol, so as new machines are added to a network or existing machines get new MAC addresses

Address Resolution Protocol (ARP) spoofing, also known as **ARP poisoning** or **ARP Poison Routing** (APR), is a technique used to attack an Ethernet wired or wireless network

ARP Spoofing allows an attacker to sniff **data frames on a local area network (LAN**), modify the traffic, or stop the traffic altogether

The principle of ARP spoofing is to send fake, or "spoofed", ARP messages to an Ethernet LAN. Generally, the aim is to associate the attacker's MAC address with the IP address of another node (such as the default gateway).

Any traffic meant for that IP address would be mistakenly sent to the attacker instead. The attacker could then choose to forward the traffic to the actual default gateway **(passive sniffing**) or modify the data before forwarding it **(man-in-the-middle attack)**

The attacker could also launch a **denial-of-service** attack against a victim by associating a non existent MAC address to the IP address of the victim's default gateway.

Some detection techniques are **ARP Watch (Free UNIX Program), Reverse ARP (RARP- used to detect MAC cloning) and Promiscuous Mode Sniffing**

The best defense against ARP attacks are having a **static ARP, DHCP Snooping (access control based on IP, MAC, and port) and detection**

Some detection techniques are **ARP Watch (Free UNIX Program), Reverse ARP (RARP- used to detect MAC cloning) and Promiscuous Mode Sniffing**

Any traffic meant for that IP address would be mistakenly sent to the attacker instead. The attacker could then choose to forward the traffic to the actual default gateway **(passive sniffing**) or modify the data before forwarding it **(man-in-the-middle attack)**

The attacker could also launch a **denial-of-service** attack against a victim by associating a non existent MAC address to the IP address of the victim's default gateway.

The best defense against ARP attacks are having a **static ARP, DHCP Snooping (access control based on IP, MAC, and port) and detection.**

ROUTE TABLE MODIFICATION

An attacker would be able to put himself in such a position to block packets by modifying routing tables, so that packets flow through a system he has **control of (Layer 3 redirection), by changing bridge tables by playing games with spanning-tree frames (Layer 2 redirection**)

By rerouting **physical cables** so that the frames must flow through the attacker's system **(Layer 1 redirection)**

Most of the time, an attacker will try to change route tables remotely

A more locally workable attack might be to spoof **Internet Control Message Protocol** (ICMP) and redirect packets

Many **OS's accept ICMP redirects in their default configuration**. Unless, the connection is to be broken entirely (or proxy it in some way), the packets have to be forwarded back to the real router, so they can reach their ultimate destination

If the attacker has managed to change route tables to get packets to flow through his system, some of the intermediate routers will be aware of the route change, either because of route tables changing or possibly because of an Address Resolution Protocol (ARP) table change

- **Format string vulnerability** attacks fall into **three categories**:

➢ **denial of service**, **reading** and **writing**.

➢ **denial of service attacks** are characterized by utilizing multiple instances of the **%s** format specifier to read data off of the stack until the program attempts to read data from an illegal address, which will cause the program to crash.

- **reading attacks** typically utilize the **%x** format specifier to print **sections of memory** that we **do not** normally have access to. This is a **serious problem** and can lead to disclosure of **sensitive information**.
- **writing attacks** utilize the **%d, %u or %x** format specifiers **to overwrite** the Instruction Pointer and force execution of user-supplied shell code. This is exploited using single write method or multiple writes method.

Buffer Overflow

- **Buffer Overflow:** A buffer overflow occurs when a **program or process tries to store more data in a buffer** than it was intended to hold.

- **It happens** when the attacker intentionally enters more data than a program was written to handle.
- **This allows** an attacker to overwrite data that **controls** the program and can take over control of the program to execute the **attacker's code** instead of programmer's code.

Buffer Injection Techniques

For creating an exploit, it is important to determine a way of getting a **large buffer into the over flowable buffer.** A simple process of filling a buffer over the network

Injection vector: It refers to the **customized operational code needed to monitor and control an instruction pointer on the remote system**. It depends on host and targeted machine and is used to execute the payload

Payload: Something like a virus that can run at anytime, anywhere irrespective of its injection into a remote machine

Determining the location of payload

Files on disk, which are then **loaded into memory**

Environment variables controlled by a local user

Environment variables passed within a web request

User-controlled fields within a network protocol

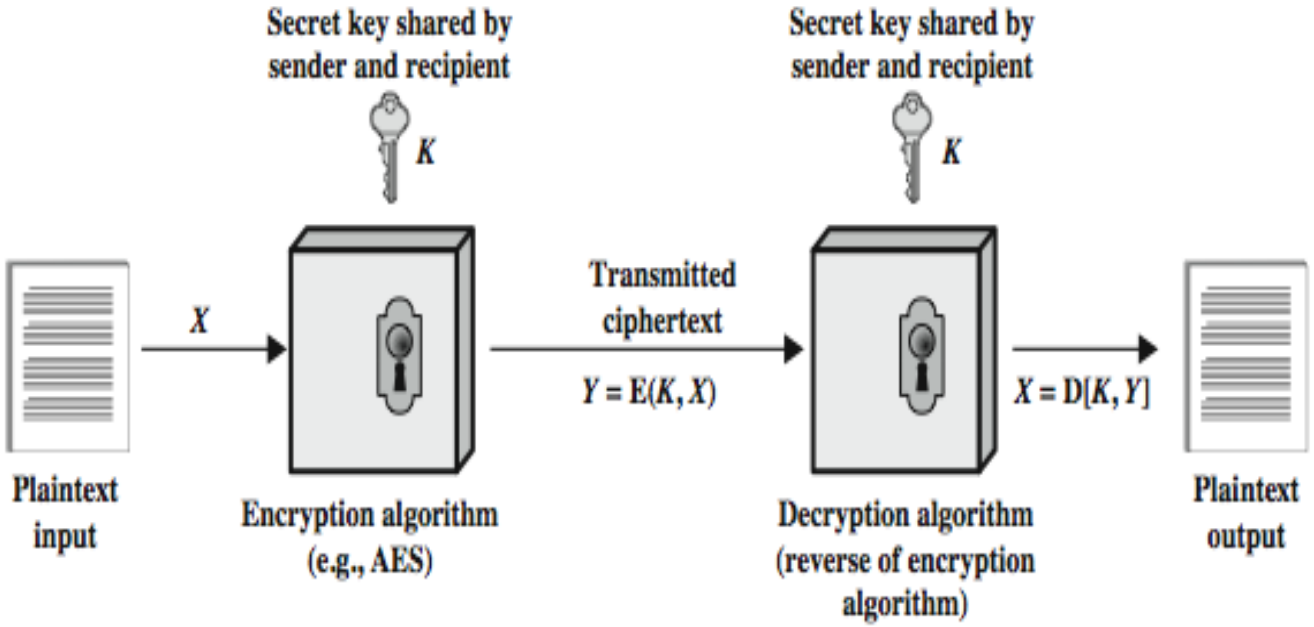
Symmetric Encryption

- or conventional / private-key / single-key
- sender and recipient share a common key
- all classical encryption algorithms are private-key
- was only type prior to invention of public-key in 1970’s
- and by far most widely used

Basic Terminology

- **plaintext -** original message
- **ciphertext -** coded message
- **cipher -** algorithm for transforming plaintext to ciphertext
- **key -** info 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
- **cryptology -** field of both cryptography and cryptanalysis

Symmetric Cipher Model



Requirements(Symmetric Cipher Model)

- two requirements for secure use of conventional encryption:
  - a strong encryption algorithm
  - a secret key known only to sender / receiver
- mathematically have:

  $Y = E(K, X)$(x is plain text, k is key, y is cipher text, E is encryption)

  $X = D(K, Y)$(D is decryption)
- assume encryption algorithm is known
- implies a secure channel to distribute key

cryptography

They are characterized along three independent dimensions:

1) The type of operations used for transforming plain text to cipher text:

encryption algorithms are based on 2 general principles

a)SUBSTITUTION: each element in plain text is mapped into another element

In this example each letter replaces with the letter standing three places further down the alphabet

example plain text A B C

cipher text D E F

b)TRANSPOSITION: elements in plain text are rearranged

example A C encrypted message : acbd

B D

cryptography

2) The number of keys used:

- If same key is used the system is referred to as symmetric, single key or conventional encryption.
- If different keys are used the system is referred as asymmetric or public key encryption

cryptography

- 3) The way in which the plaintext is processed:

A block cipher processes the input one block of elements at a time, producing an output block for each input block

A stream cipher processes the input elements continuously producing output one element at a time

cryptanalysis

- Two general approaches to attack a conventional encryption scheme:
- Cryptanalysis:

  This type of attacks relay on the nature of algorithm plus perhaps some knowledge of the general characteristics of the plain text.
- Brute force attack:

  The attacker tries every possible key on a piece of cipher text until an intelligible translation into plain text is obtained.

  on average , half of all possible keys must be tried to achieve success

Cryptanalysis

- An encryption scheme: computationally secure if
  - **The cost of breaking the cipher exceeds the value of information**
  - **The time required to break the cipher exceeds the lifetime of information**

An encryption scheme: unconditionally secure if

- The cipher text generated by the scheme does not contain enough information to determine uniquely the corresponding plain text, no matter how much cipher text is available

Location of Encryption Device

- **Link encryption:**
  - A lot of encryption devices
  - High level of security
  - Decrypt each packet at every switch
- **End-to-end encryption**
  - The source encrypt and the receiver decrypts
  - Payload encrypted
  - Header in the clear
- **High Security:** Both link and end-to-end encryption are needed

Location of Encryption Device

- **Link encryption**
- In link encryption, each vulnerable communications link is equipped on both ends with an encryption device.
- This recourse requires a lot of encryption devices in a large network.
- One of its disadvantages is that message must be decrypted each time it enters a switch because the switch must read the address in packet header to route the frame
- Thus message is vulnerable at each switch

Link encryption

- All potential links in a path from source to destination must use link encryption

- Each pair of nodes that share a link should share a unique key, with a different key used on each link. Thus, many keys must be provided.

END TO END ENCRYPTION

- With end –to- end encryption, the encryption process is carried out at the two end systems
- The source host encrypts the data. The data in encrypted form are then transmitted unaltered across the network to the destination terminal
- The destination shares a key with the source and so is able to decrypt the data
- Thus , with end-to-end encryption the user data are secure. but traffic pattern is not because packet headers are transmitted in the clear

END TO END ENCRYPTION

- If two end systems share an encryption key , then a recipient is assured that any message that it receives comes from the alleged sender
- When both forms of encryption are employed, the host encrypts the user data portion of a packet using an end-to-end encryption key
- The entire packet is then encrypted using a link encryption key
- As the packet traverses the network ,each switch decrypts the packet, using a link encryption key to read the header, and then encrypts the entire packet again for sending it out on the next link.

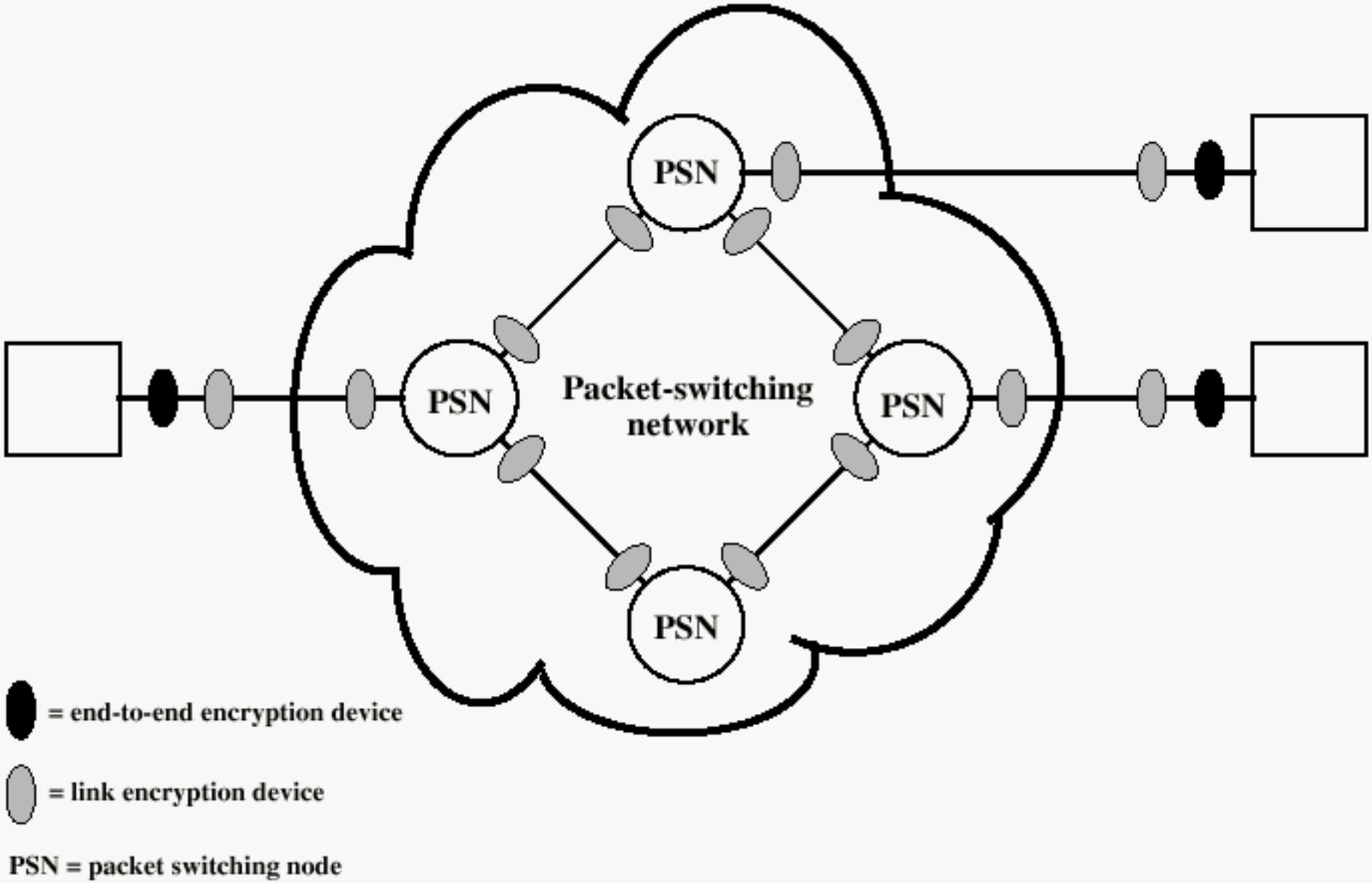


Figure 2.9 Encryption Across a Packet-Switching Network

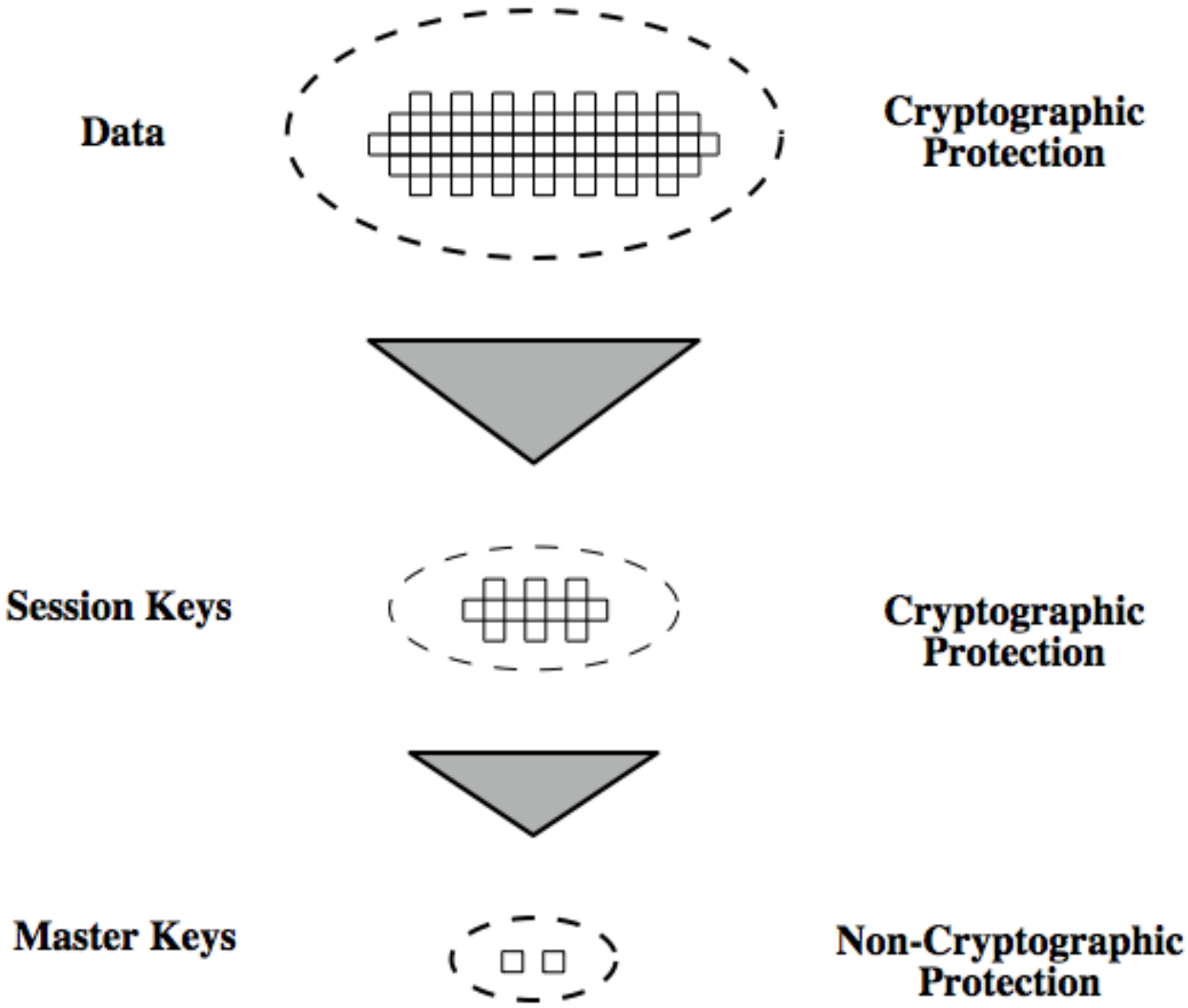
Key Distribution

- symmetric schemes require both parties to share a common secret key
- issue is how to securely distribute this key
- often secure system failure due to a break in the key distribution scheme

Key Distribution

- Given parties A and B have various **key distribution** alternatives:
  1. A can select key and physically deliver to B
  2. third party can select & deliver key to A & B
  3. if A & B have communicated previously can use previous key to encrypt a new key
  4. if A & B have secure communications with a third party C, C can relay key between A & B

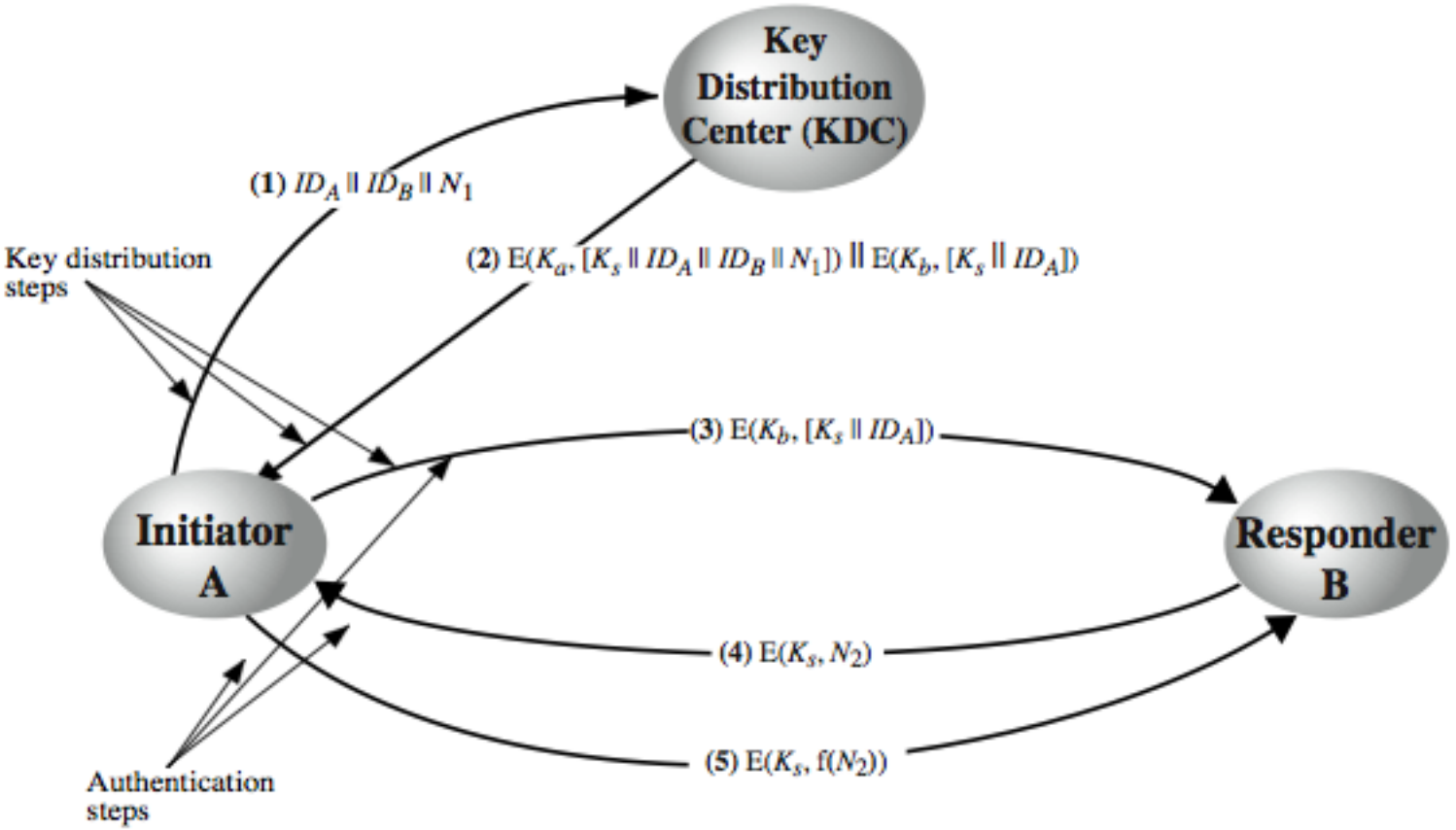
Key Hierarchy



Key Hierarchy

- Typically have a hierarchy of keys
- session key
  - temporary key
  - used for encryption of data between users
  - for one logical session then discarded
- master key
  - used to encrypt session keys
  - shared by user & key distribution center

Key Distribution Scenario



KDC

- Key distribution center:
- Allows systems to communicate with each other
- Permission is granted for two systems to establish a connection
- KDC provides a one-time session key for that connection

SSM(SECURITY SERVICE MODULE)

- It consist of functionality at one protocol layer
- Performs end-to-end encryption
- Obtains session keys on behalf of users

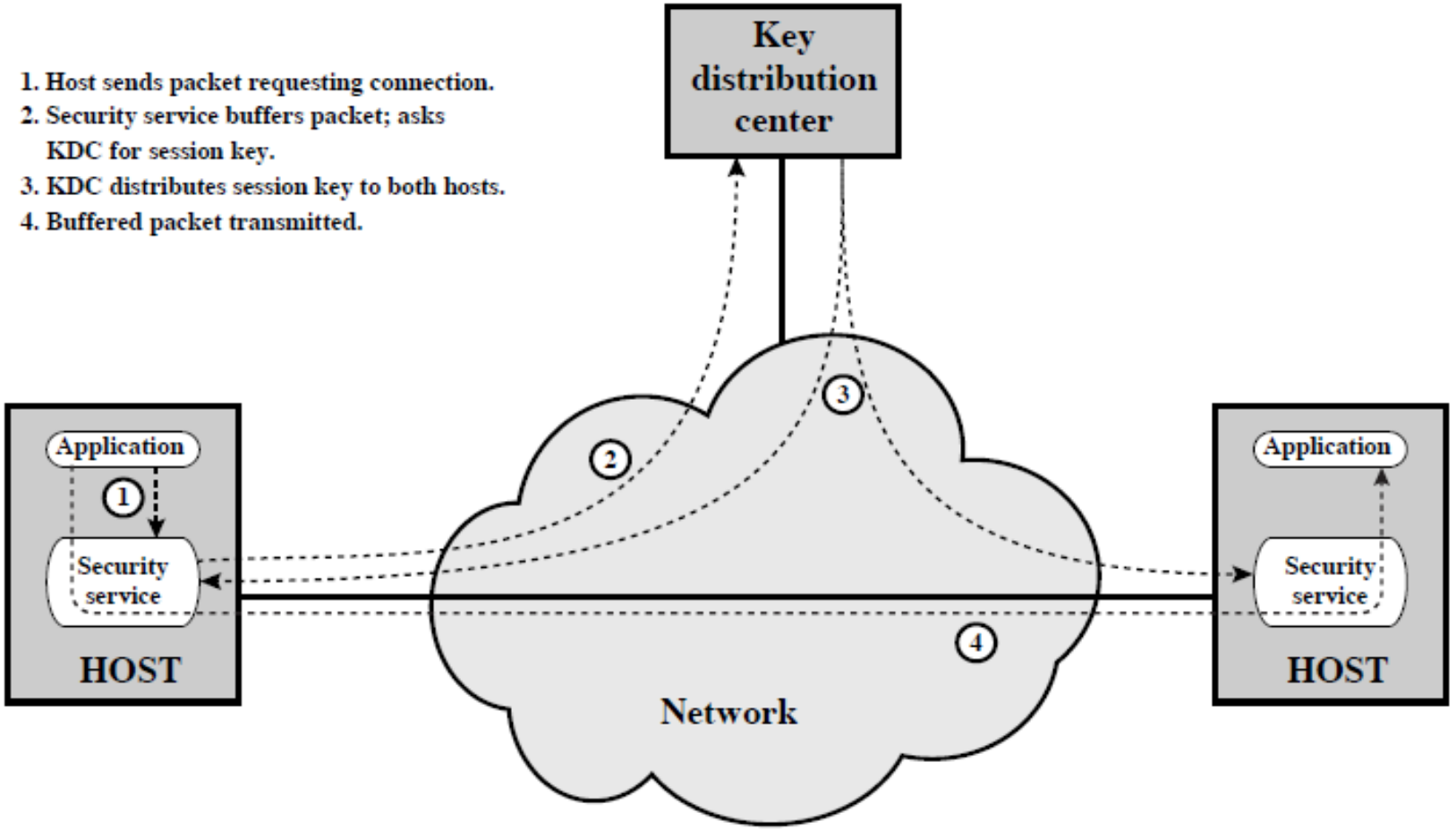


Figure 7.10 Automatic Key Distribution for Connection-Oriented Protocol

Establishing a connection

- Host will set up a connection to another host. it transmits a connection –request packet(step 1)
- SSM saves the packet & applies to KDC for permission to establish connection(step 2)
- Communication between SSM and KDC is encrypted using a master key
- KDC generates a session key and deliver it two SSM'S using a unique permanent key(master key) for each SSM(step 3)
- User data between 2 end systems are encrypted by their respective SSM'S using one time session key

Authentication

- Requirements - must be able to verify that:
  1. Message came from apparent source or author,
  2. Contents have not been altered,
  3. Sometimes, it was sent at a certain time or sequence.

- Protection against active attack (falsification of data and transactions)

Approaches to Message Authentication

- Authentication Using Conventional Encryption
  - Only the sender and receiver should share a key
  - If the message includes and error-detection code and a sequence number receiver is assured that no alterations have been made and sequencing is proper
- Message Authentication without Message Encryption
  - An authentication tag is generated and appended to each message
- Message Authentication Code
- This technique involves the use of a secret key to generate a small block of data. Known as MAC, that is appended to the message

Message Authentication Code

- This technique assumes that two communications parties , A and B, share a common secret key K.
- When A has a message to send to B , it calculates the message authentication code as a function of the message and the key
- MAC = F(K,M)
- The message plus code are transmitted to the intended recipient
- The recipient performs the same calculation on the received message, using the same secret key ,to generate a new MAC
- The received code is compared to the calculated code

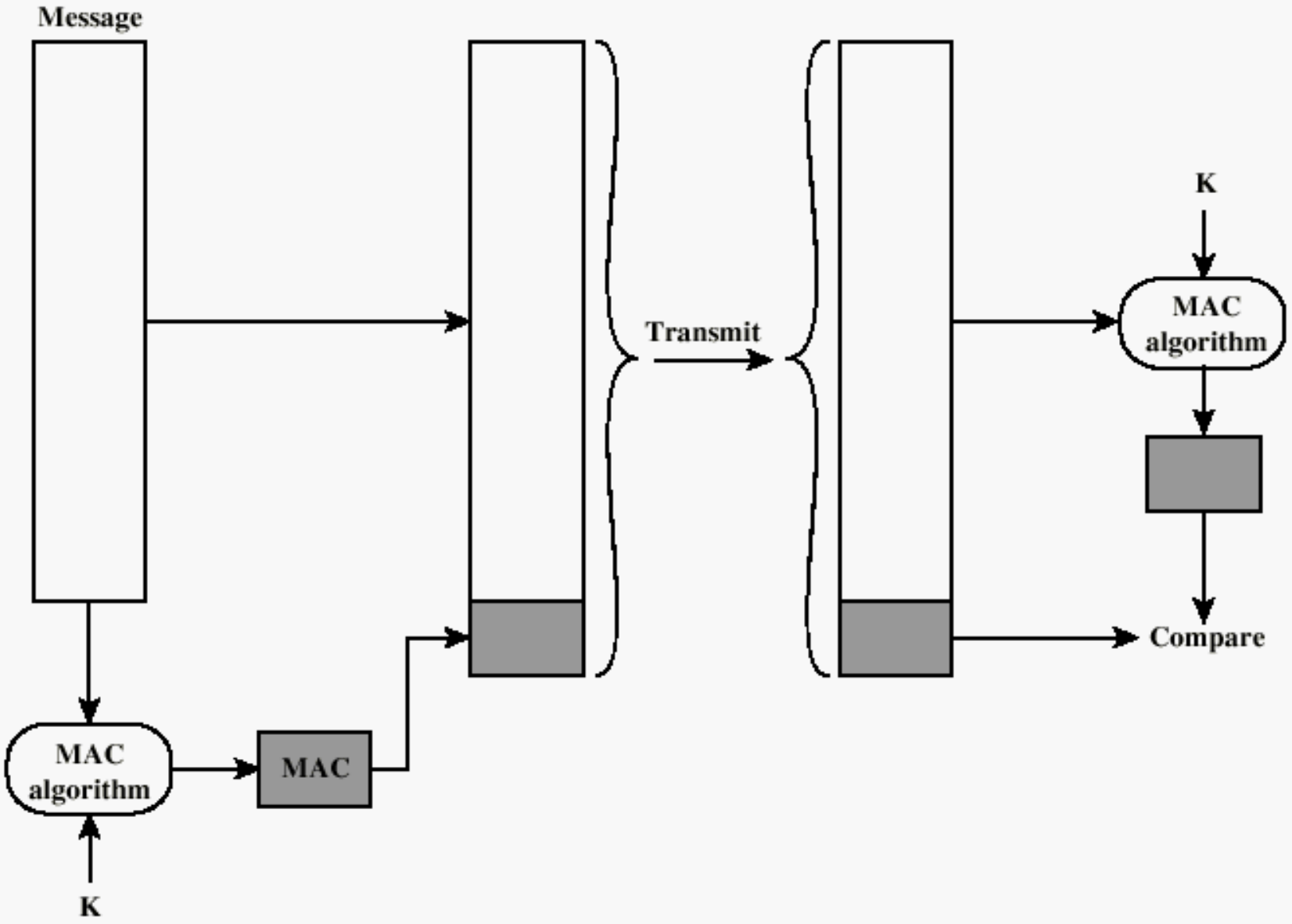
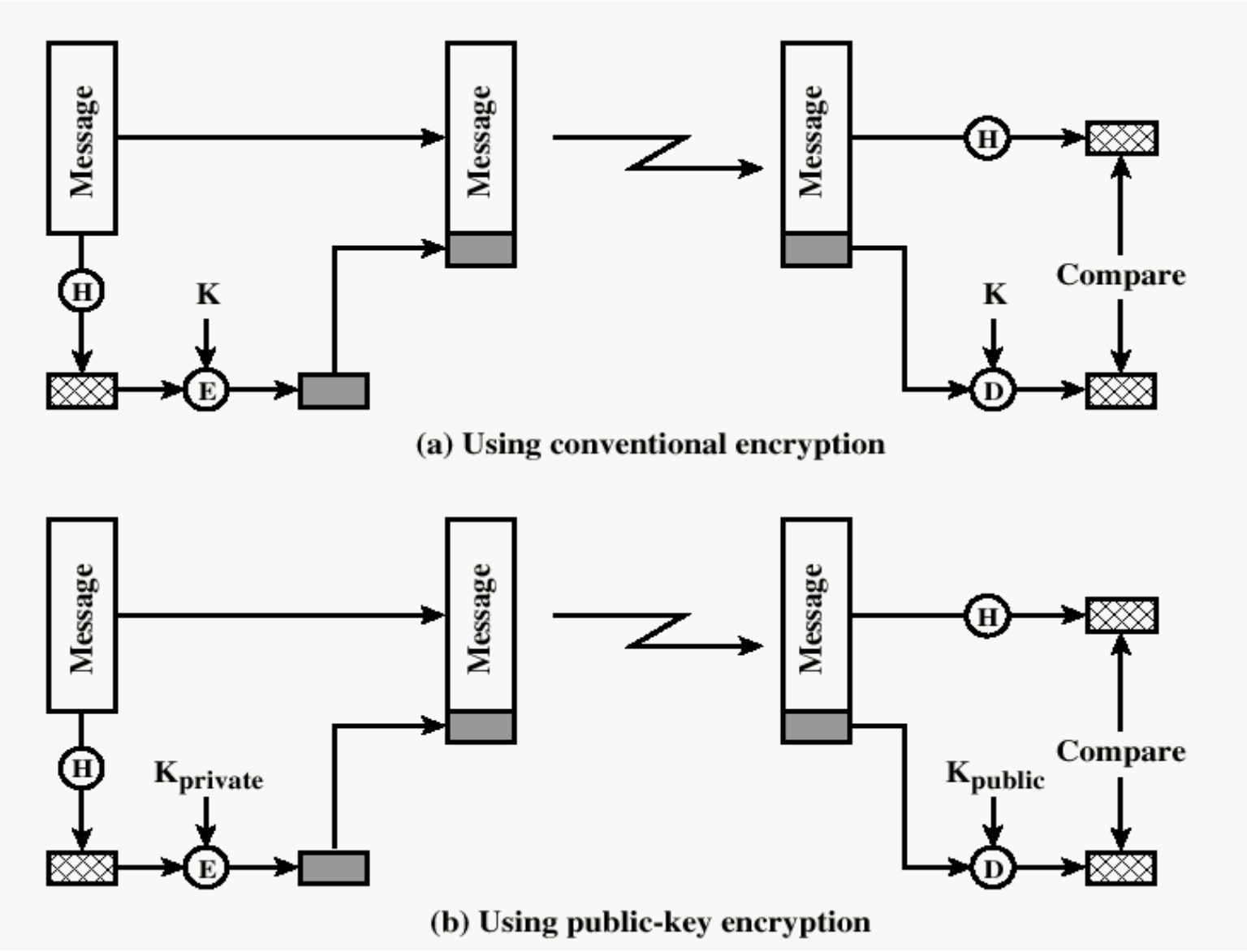


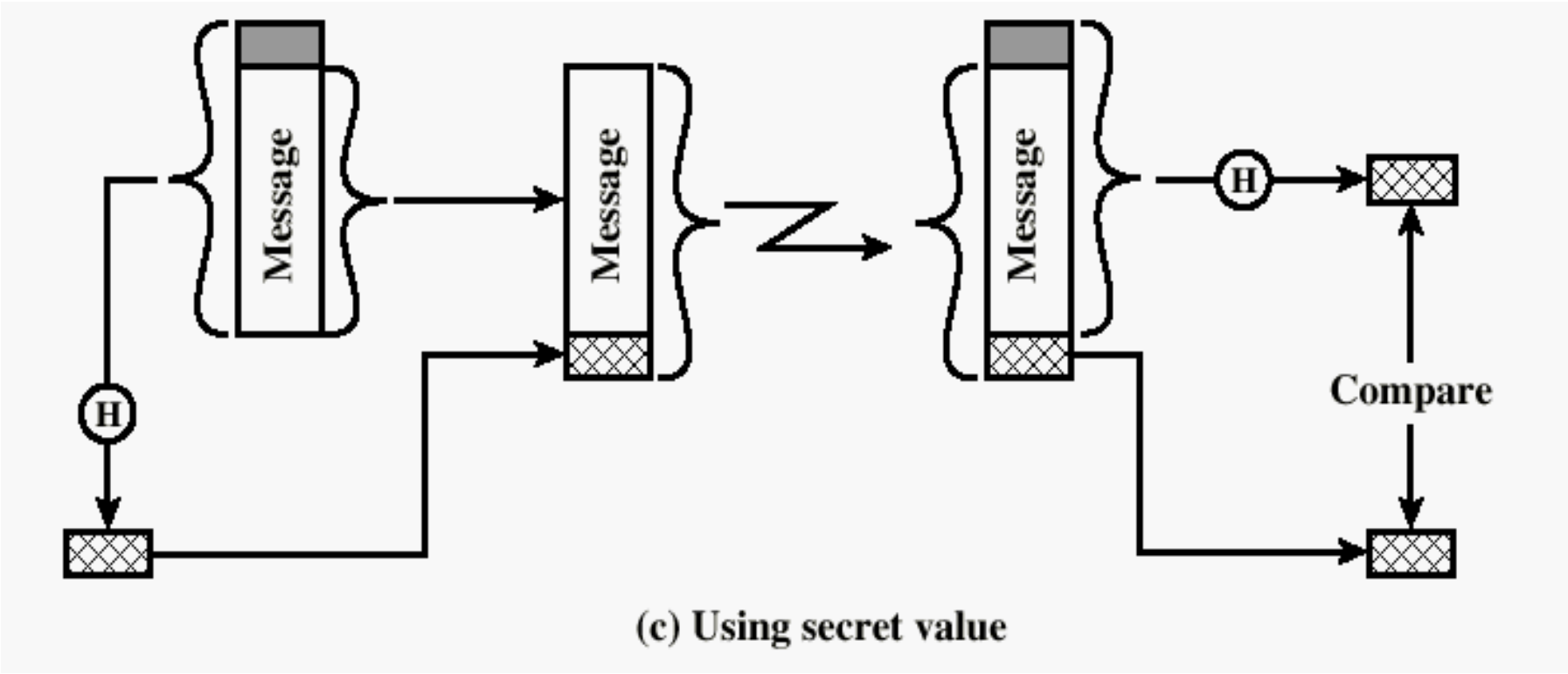
Figure 3.1 Message Authentication Using a Message Authentication Code (MAC)

One-way HASH function



One-way HASH function

- Secret value is added before the hash and removed before transmission.



(c) Using secret value

One-way Hash Functions

- Accepts a variable-size message *M* as input and produces a fixed-size message digest H(*M*) as output
- Does not take a secret key as input
- To authenticate a message, the message digest is sent with the message in such a way that the message digest is authentic

One-way Hash Functions

- The message digest can be encrypted using conventional encryption (part a); if it is assumed that only the sender and receiver share the encryption key, then authenticity is assured.
- The message digest can be encrypted using public-key encryption (part b)The public-key approach has two advantages: (1) It provides a digital signature as well as message authentication. (2) It does not require the distribution of keys to communicating parties.

One-way Hash Functions

- Figure c shows a technique that uses a hash function but no encryption for message authentication.
- Because the secret value itself is not sent, it is not possible for an attacker to modify an intercepted message.
- As long as the secret value remains secret, it is also not possible for an attacker to generate a false message.

Secure HASH Functions

- Purpose of the HASH function is to produce a ”fingerprint”.
- Properties of a HASH function H :
  1. H can be applied to a block of data at any size
  2. H produces a fixed length output
  3. H(x) is easy to compute for any given x.
  4. For any given value h, it is computationally infeasible to find x such that H(x) = h
  5. For any given block x, it is computationally infeasible to find $y \neq x$ with H(y) = H(x).
  6. It is computationally infeasible to find any pair (x, y) such that H(x) = H(y)

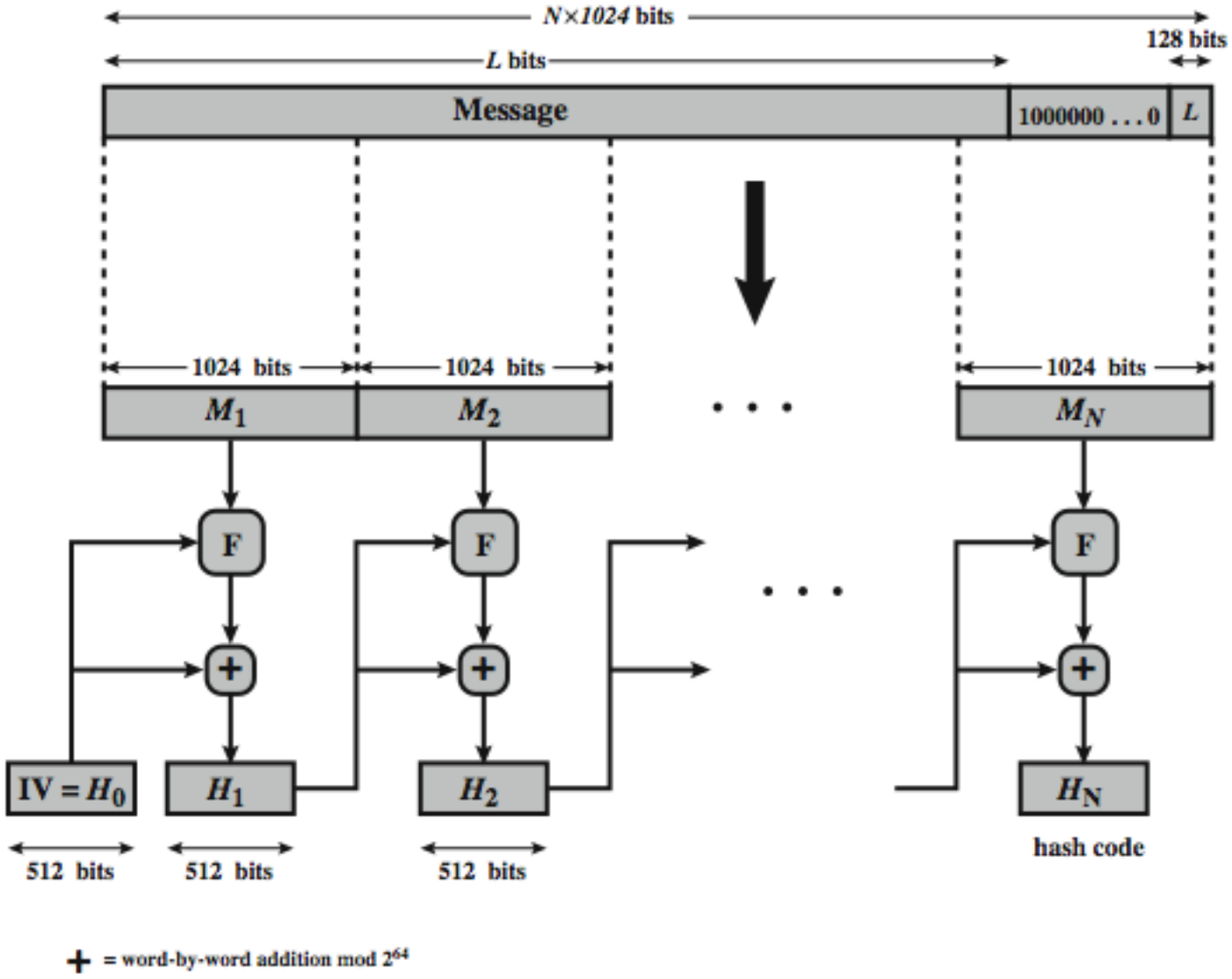
Simple Hash Function Using Bitwise XOR

	bit 1	bit 2	• • •	bit n
block 1	b_{11}	b_{21}		b_{n1}
block 2	b_{12}	b_{22}		b_{n2}
	•	•	•	•
	•	•	•	•
	•	•	•	•
block m	b_{1m}	b_{2m}		b_{nm}
hash code	C_1	C_2		C_n

Simple hash function

- The input is viewed as a sequence of n-bit blocks.
- The input is processed one block at a time in a iterative fashion to produce an n-bit hash function
- One of the simplest hash function is the bit – by – bit exclusive –OR(XOR) of every block
- bit-by-bit exclusive-OR (XOR) of every block
  - $C_i = b_{i1}\ xor\ b_{i2}\ xor\ \ldots\ xor\ b_{im}$
  - a longitudinal redundancy check
  - reasonably effective as data integrity check

SHA-512 Overview



Comparison of SHA Parameters

	SHA-1	SHA-224	SHA-256	SHA-384	SHA-512
Message Digest Size	160	224	256	384	512
Message Size	$< 2^{64}$	$< 2^{64}$	$< 2^{64}$	$< 2^{128}$	$< 2^{128}$
Block Size	512	512	512	1024	1024
Word Size	32	32	32	64	64
Number of Steps	80	64	64	80	80

Note: All sizes are measured in bits.

Now examine the structure of SHA-512, noting that the other versions are quite similar.

. The processing consists of the following steps:

• Step 1: Append padding bits, consists of a single 1-bit followed by the necessary number of 0-bits, so that its length is congruent to 896 modulo 1024

• Step 2: Append length as an (big-endian) unsigned 128-bit integer

• Step 3: Initialize hash buffer to a set of 64-bit integer constants

• Step 4: Process the message in 1024-bit (128-word) blocks, which forms the heart of the algorithm. Each round takes as input the 512-bit buffer value H_i, and

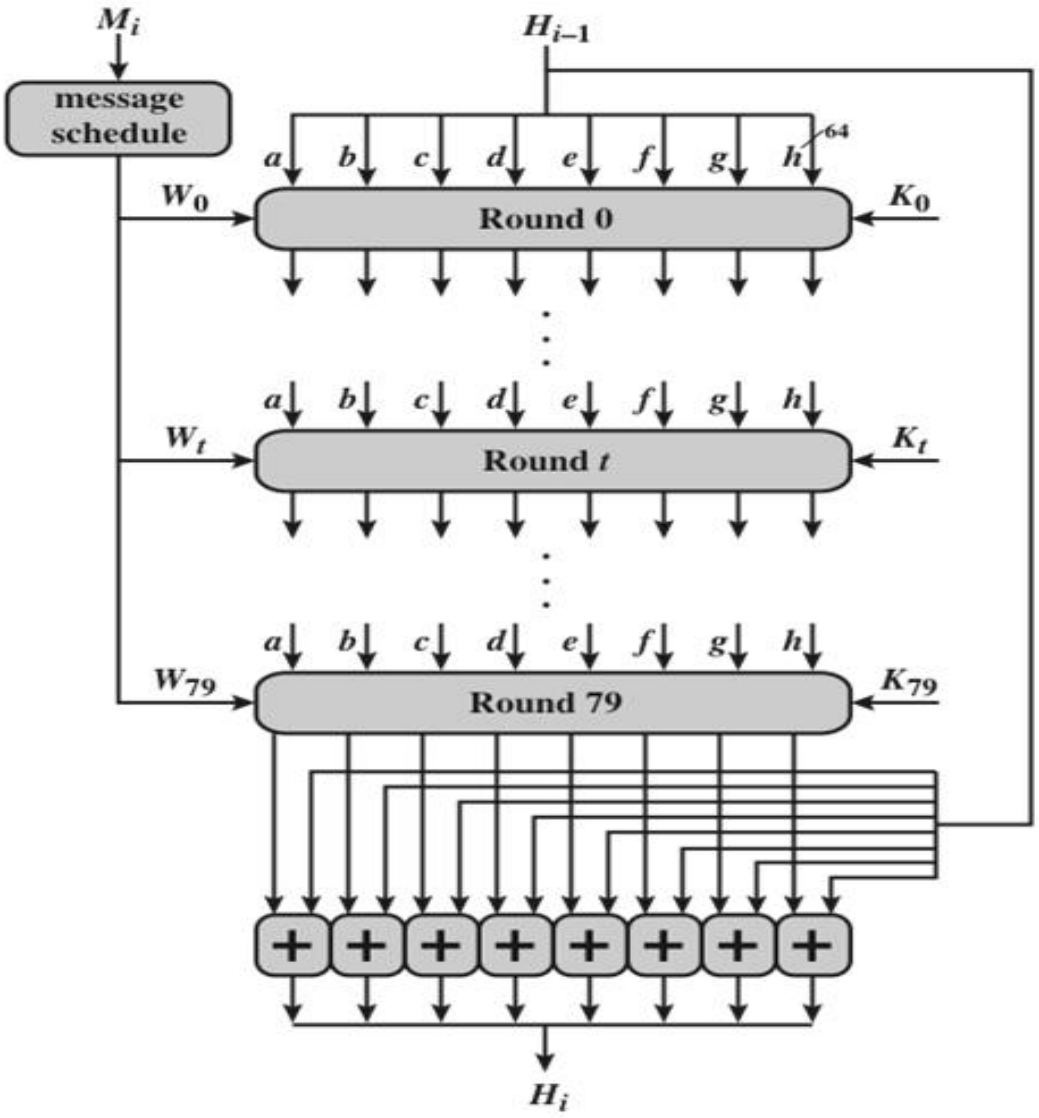


Figure 3.5 SHA-512 Processing of a Single 1024-Bit Block

Feistel Cipher Structure

- Virtually all conventional block encryption algorithms, including DES have a structure first described by Horst Feistel of IBM in 1973
- The realisation of a Fesitel Network depends on the choice of the following parameters and design features

Feistel Cipher Structure

- **Block size:** larger block sizes mean greater security
- **Key Size:** larger key size means greater security
- **Number of rounds:** multiple rounds offer increasing security
- **Subkey generation algorithm:** greater complexity will lead to greater difficulty of cryptanalysis.
- **Fast software encryption/decryption:** the speed of execution of the algorithm becomes a concern

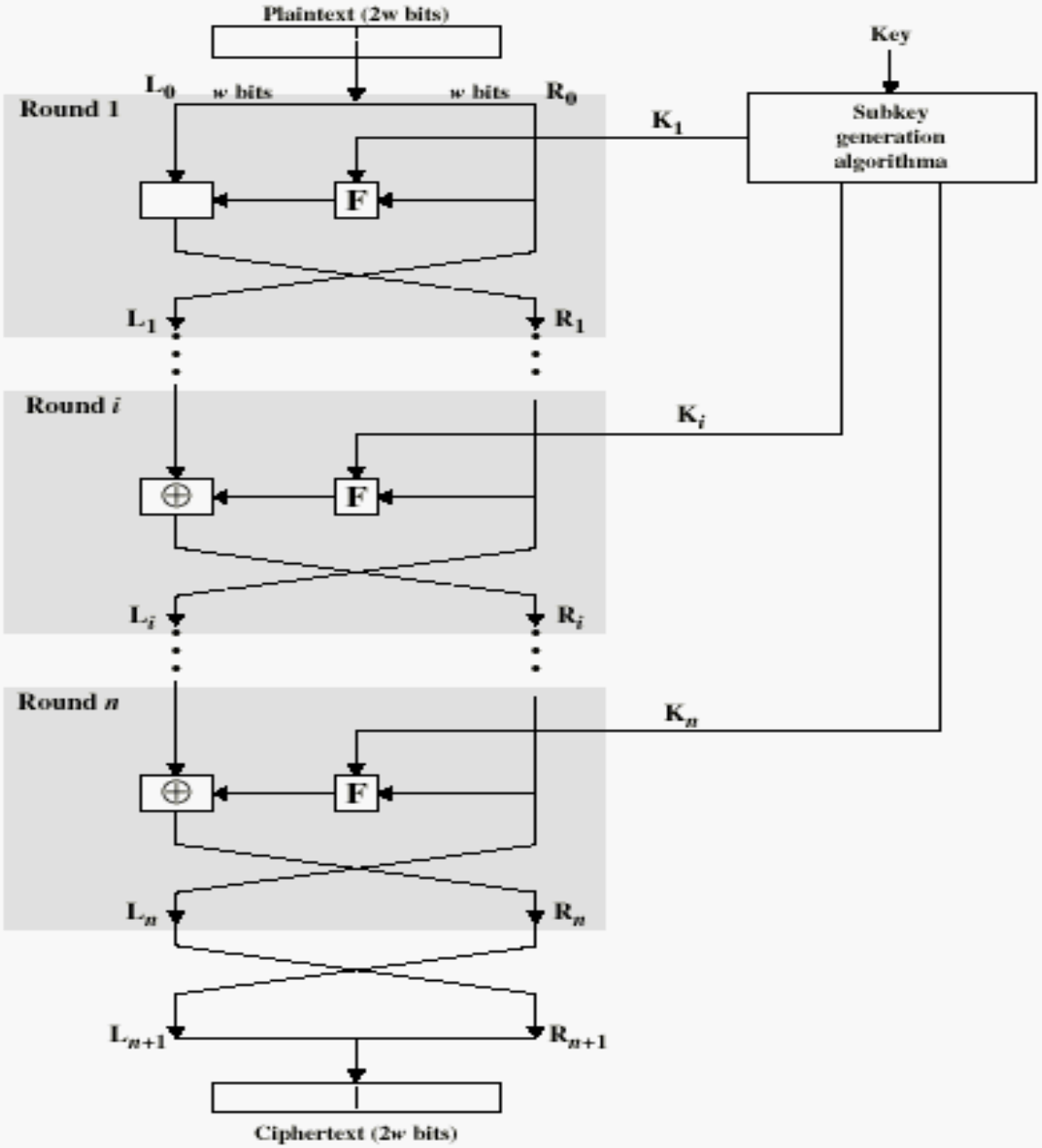


Figure 2.2 Classical Feistel Network

Feistel Cipher Structure

- The inputs to the encryption algorithm are a plaintext block of length 2w bits and a key K.
- The plaintext block is divided into two halves, L_0 and R_0.
- The two halves of the data pass through n rounds of processing and then combine to produce the cipher text block.
- Each round i has as inputs L_{i-1} and R_{i-1},

Feistel Cipher Structure

- In general, the a sub key K_i are different from K and from each other and are generated from the key by a sub key generation algorithm
- A substitution is performed on the left half of the data
- This is done by applying a round function F to the right half of the data and then taking the X-OR of the output of that function and the left half of the data
- Following this substitution, a permutation is performed that consists of the interchange of the two halves of the data

Conventional Encryption Algorithms

- Data Encryption Standard (DES)
  - The most widely used encryption scheme
  - The algorithm is reffered to the Data Encryption Algorithm (DEA)
  - DES is a block cipher
  - The plaintext is processed in 64-bit blocks
  - The key is 56-bits in length

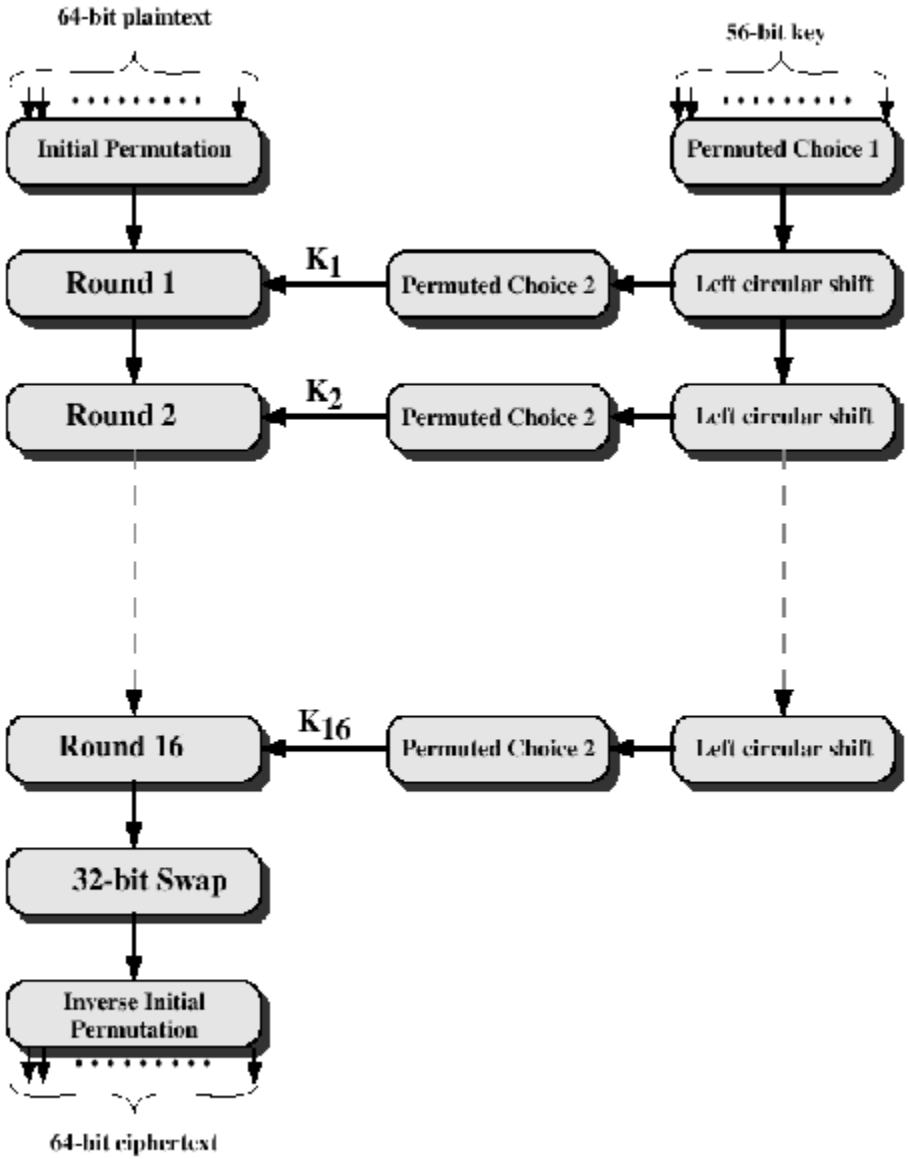


Figure 2.3 General Depiction of DES Encryption Algorithm

UNIT-II

Data Encryption Standard (DES)

- There are 2 inputs to the encryption function: the plain text to be encrypted and the key
- The plain text must be 64 bits in length and the key is 56 bits in length
- First the 64-bit plain text passes through an initial permutation(IP) that rearranges the bits to produce the permuted input
- Next phase consisting of 16 rounds same function, which involves both permutation and substitution functions
- The left and right halves of the output are swapped to produce the pre output
- Finally the output is passed through inverse of initial permutation, to produce the 64-bit cipher text

Data Encryption Standard (DES)

- The right hand portion in which the 56-bit key is used
- Initially ,the key is passed through a permutation function.
- Then , for each of the 16 rounds , a sub key is produced by the combination of a left circular shift and permutation
- The permutation function is the same for each round, but a different

Sub key is produced because of the repeated shifts of the key bits

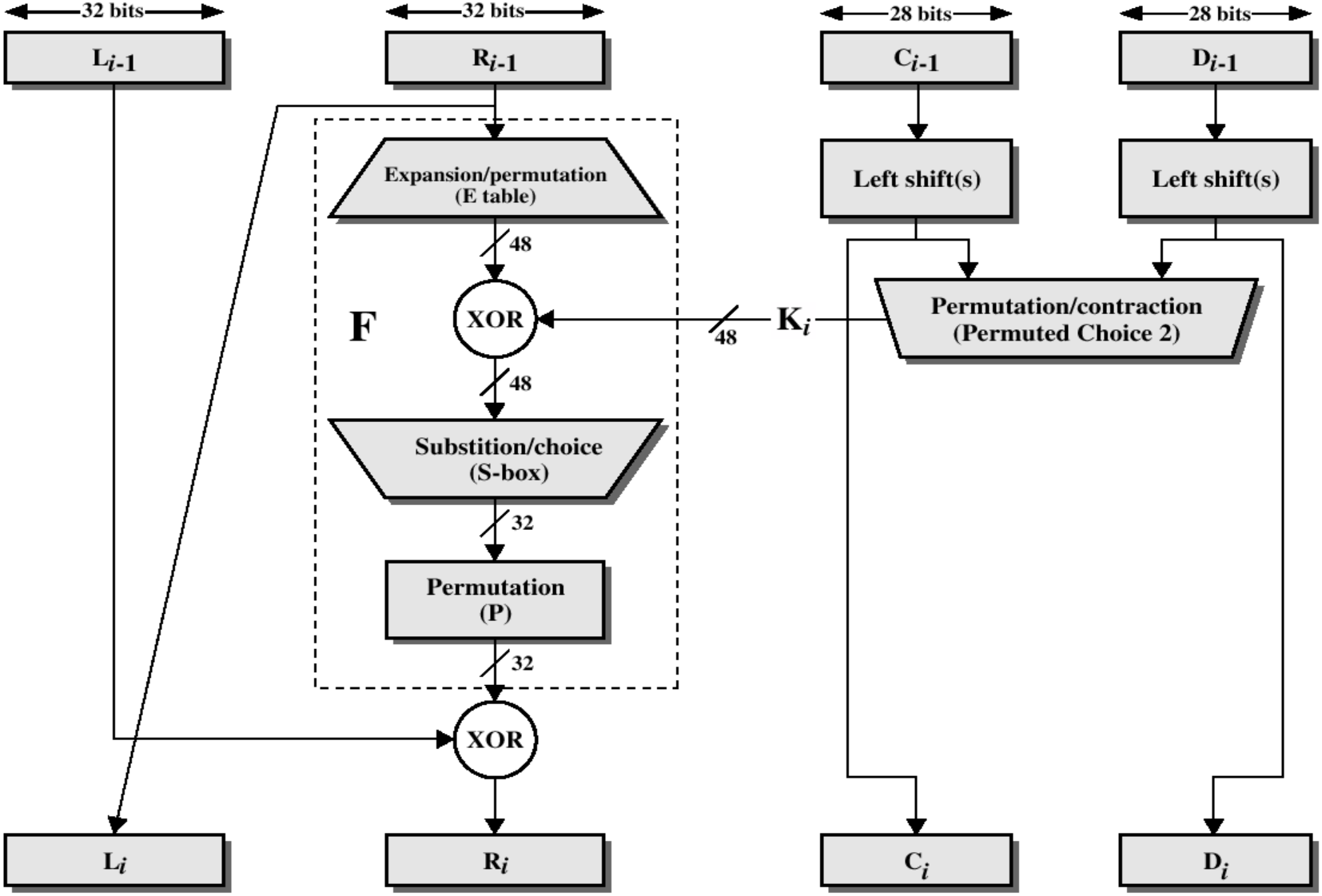


Figure 2.4 Single Round of DES Algorithm

Details of single round

- The left and right halves of each 64 -bit intermediate value are treated as separate 32-bit quantities L(left) and R(right)
- **The overall processing at each iteration:**
  - Li = Ri-1
  - Ri = Li-1 F(Ri-1, Ki)
- The round key ki is 48 bits. The R input is first expanded to 48bits by using permutation plus an expansion that involves duplication of 16 of R bits
- The resulting 48 bits are XORed with Ki
- This 48-bit result passes through a substitution function that produces a 32-bit output

Details of single round

- The 56-bit key is treated as two 28 bit quantities
- Labled C0 and D0. at each round Ci-1 and Di-1 are subjected to a circular left shift or rotation of 1 0r 2 bits
- These shifted values serves as input to the next round
- They also serve as input to permuted choice two which produces a 48 bit output that serves a input to the function F(Ri-1,ki)

STRENGTH OF DES

- Concern about the strength of DES falls into two categories i.e. strength of algorithm itself and use of 56- bit key.
- Cryptanalysis is possible by exploiting the characteristics of the DES algorithm
- DES was proved insecure in JULY 1998, when EFF(Electronic Frontier Foundation) had broken a DES encryption using a DES CRACKER
- There is more to a key search attack than simply running through all possible keys
- If the message is a numerical file and if it is compressed , the problem becomes even more difficult to automate

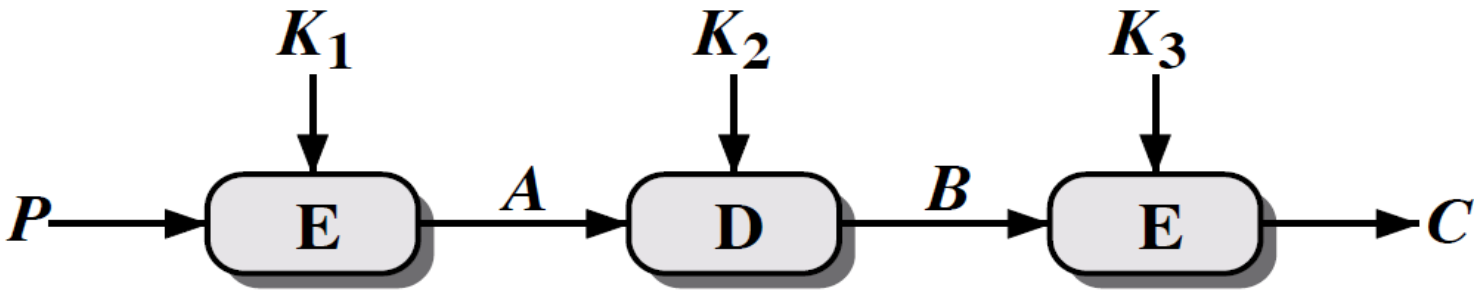
STRENGTH OF DES

- Thus , to supplement the brute – force approach, some degree of knowledge about the expected plain text is needed.
- The only form of attack that could be made on an encryption algorithm is brute force , then way to counter such attacks is : USE LONG KEYS
- If a key of size 128 bits is used, it takes approximately so many years to break the code making the algorithm unbreakable by using EFF cracker

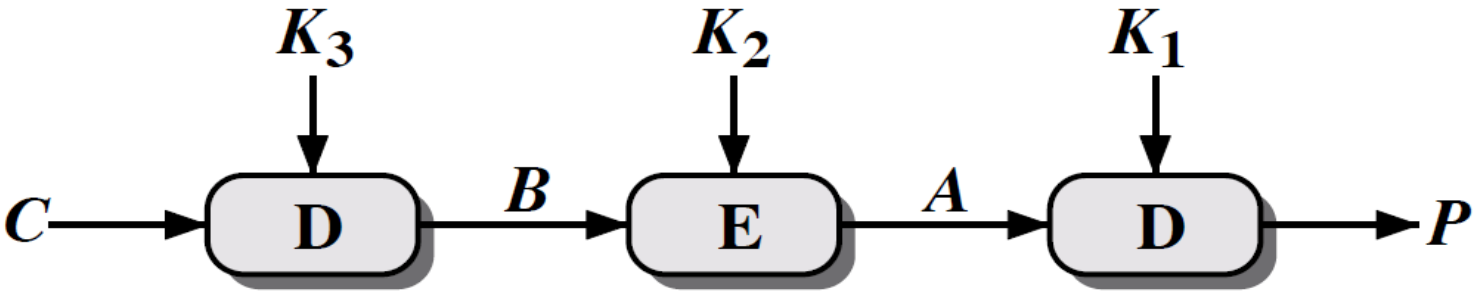
3DES guidelines

- FIPS 46-3 includes the following guidelines for 3DES:
  - 3DES is the FIPS-approved symmetric encryption algorithm of choice
  - The original DES, which uses a single 56-bit key, is permitted under the standard for legacy systems only; new procurements should support 3DES
  - Government organizations with legacy DES systems are encouraged to transition to 3DES
  - It is anticipated that 3DES and the Advanced Encryption Standard (AES) will coexist as FIPS-approved algorithms, allowing for a gradual transition to AES

Triple DES



(a) Encryption



(b) Decryption

$C = E_{K3}[D_{K2}[E_{K1}[P]]]$

$P = D_{K1}[E_{K2}[D_{K3}[C]]]$

Advanced encryption standard (AES)

- AES uses a block length of 128 bits and a key length can be 128,192 or 256 bits
- The input to the encryption and decryption algorithms is a single 128-bit block
- This block is depicted as a square matrix of bytes
- This block is copied into the STATE array which is modified at each stage of encryption or decryption
- Similarly 128 – bit key is depicted as a square matrix of bytes
- This key is then expanded into an array of key schedule words
- Each word is 4 bytes and total key is 44 words for 128-bit key

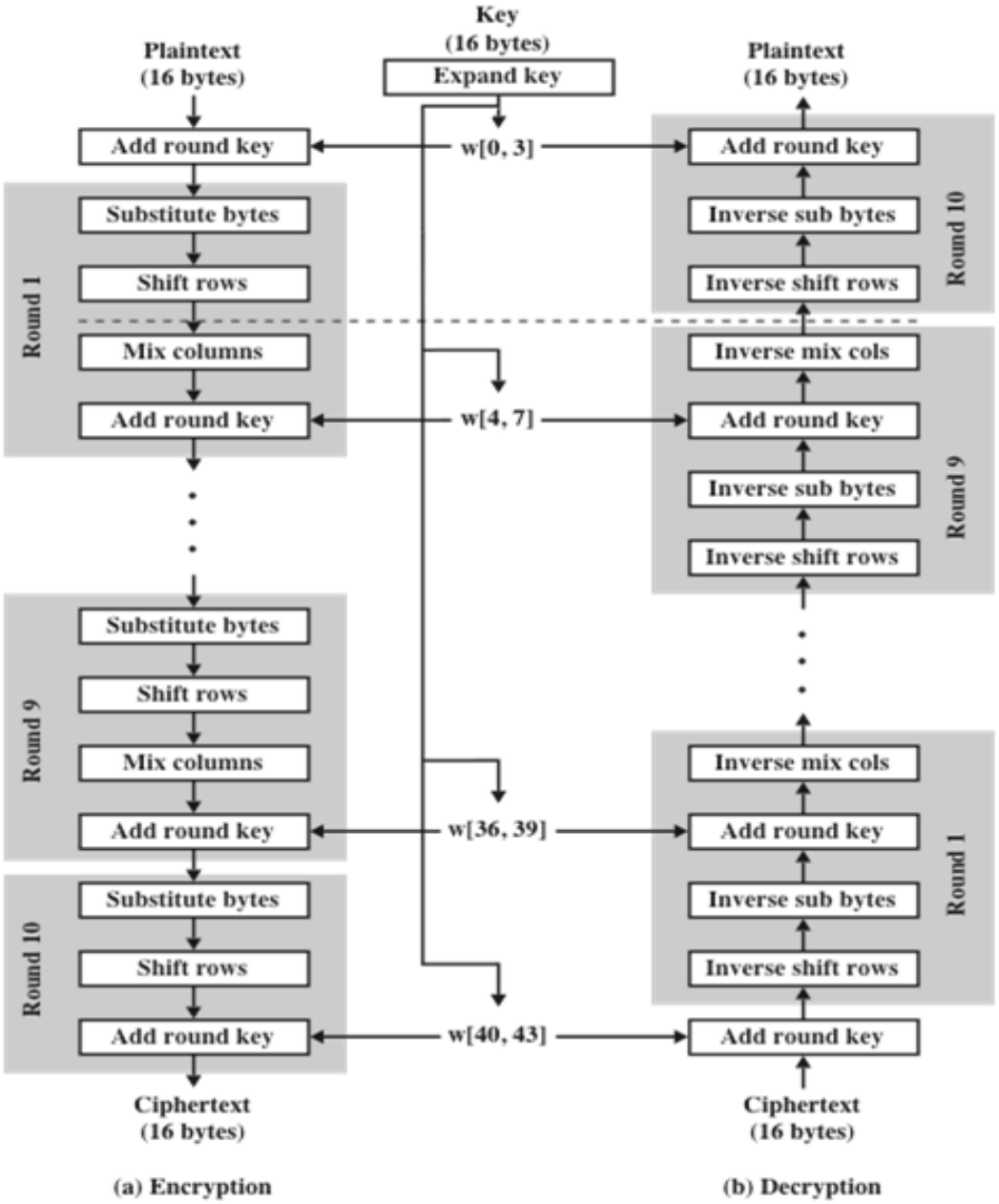


Figure 2.4 AES Encryption and Decryption

Advanced encryption standard (AES)

- Four different stages are used, one of permutation and three of substitution

- Substitute bytes: Uses a table, referred to as an S-box, to perform a byte-by-

byte substitution of the block.

- Shift rows: A simple permutation that is performed row by row.

- Mix columns: A substitution that alters each byte in a column as a function

of all of the bytes in the column.

- Add round key: A simple bitwise XOR of the current block with a portion

of the expanded key.

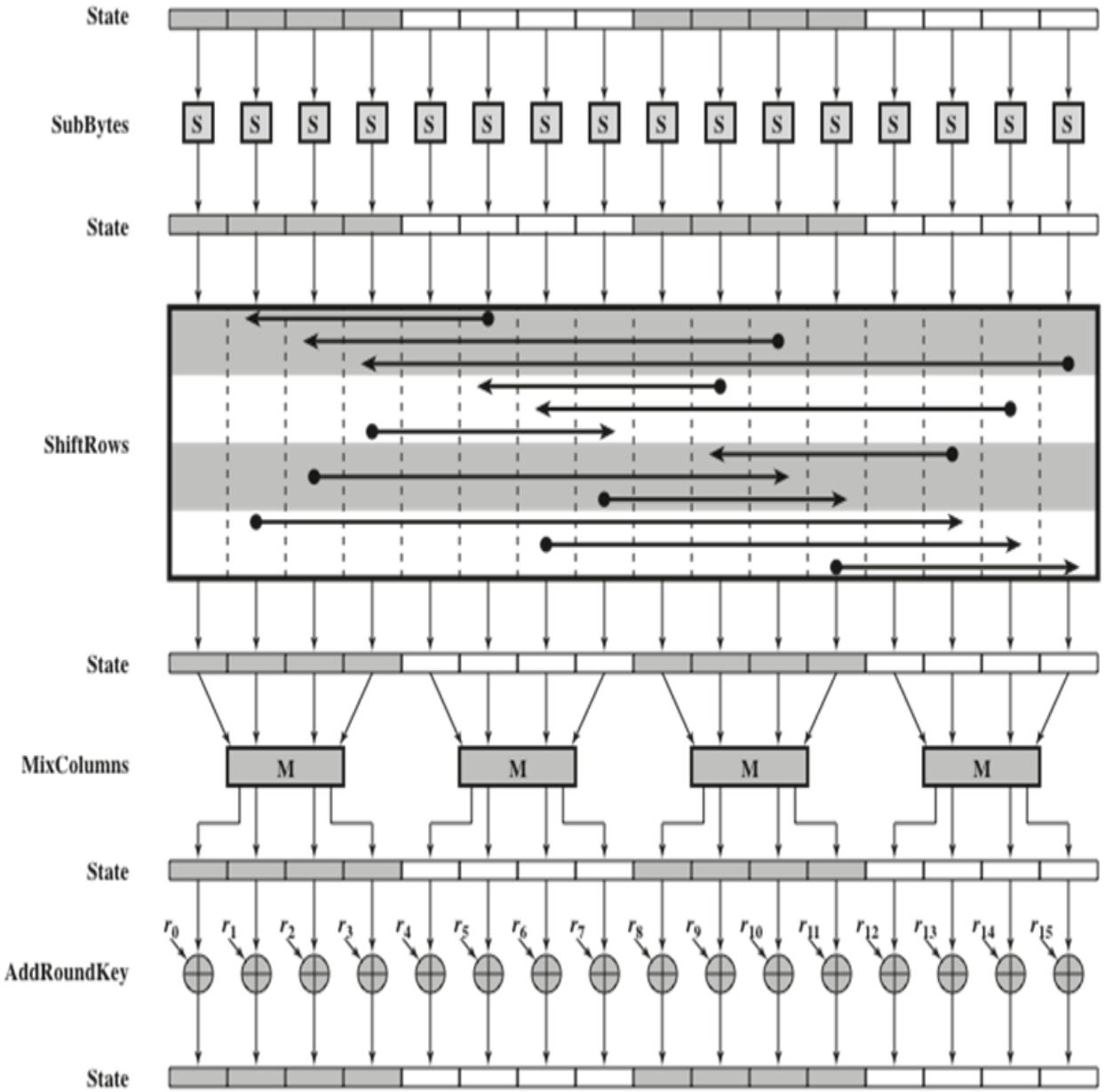


Figure 2.5 AES Encryption Round

UNIT-3

- Public Key Cryptography:

➢ It is *asymmetric, involving the use of two separate keys, in contrast to symmetric encryption, which uses only one key*

➢ Public key schemes are neither more nor less secure than private key (security depends on the key size for both).

➢ The concept of public-key cryptography evolved from an attempt to attack two of the most difficult problems associated with symmetric encryption:

1.) ***key distribution*** ***– how to have secure communications in general without having to trust a KDC with your key***
2.) ***digital signatures*** ***– how to verify a message comes intact from the claimed sender***

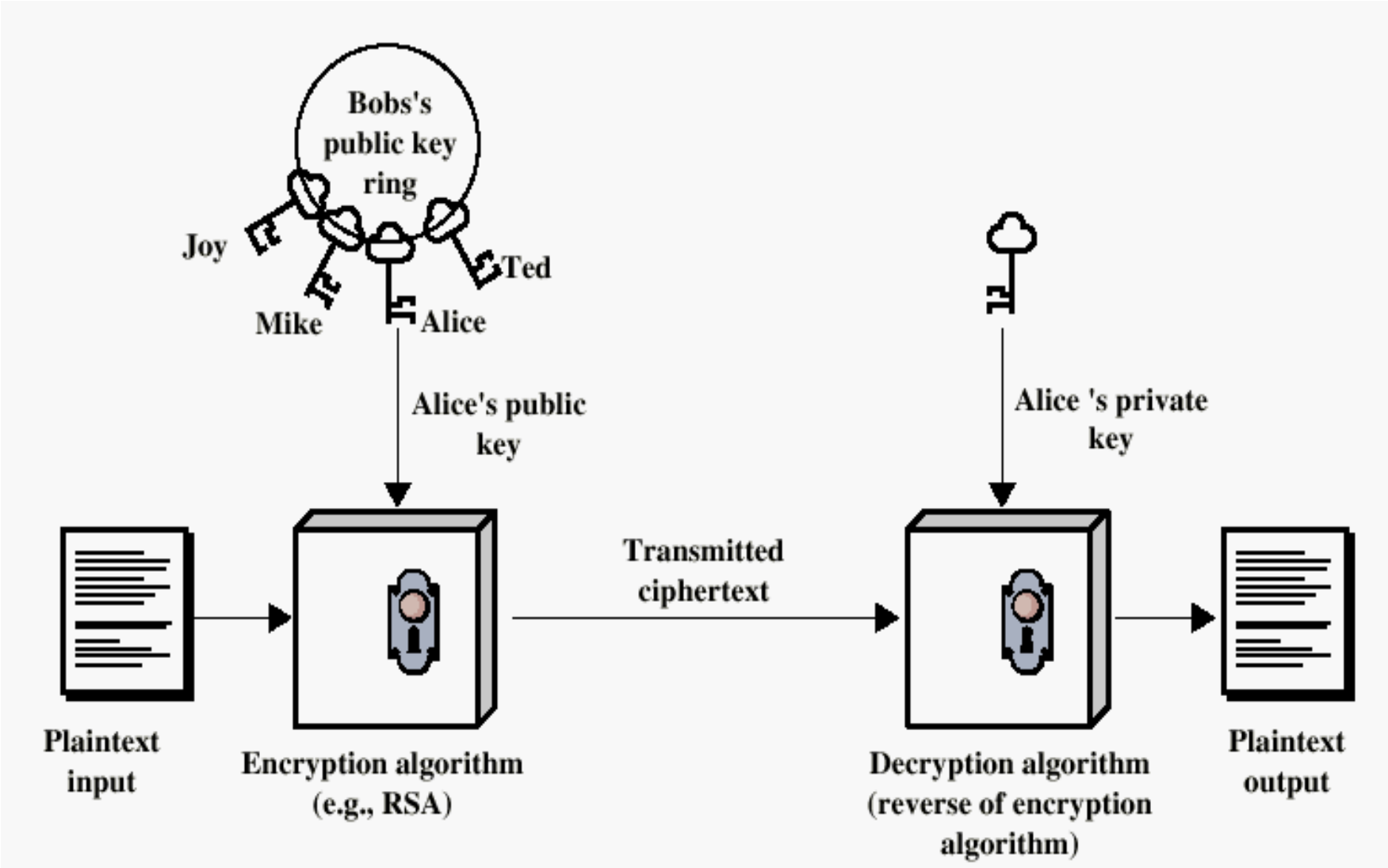
Public Key Cryptography

- Public-key/two-key/asymmetric cryptography involves the use of two keys:
- a public-key, which may be known by anybody, and can be used to encrypt messages, and verify signatures
- a private-key, known only to the recipient, used to decrypt messages, and sign (create) signatures
- is asymmetric because those who encrypt messages or verify signatures cannot decrypt messages or create signatures

Public-Key Cryptography Principles

- The use of two keys has consequences in: key distribution, confidentiality and authentication.
- The scheme has six ingredients
  - Plaintext- readable message
  - Encryption algorithm- transformations on plaintext
  - Public and private key- 1 for encryption and 1 for decryption
  - Ciphertext- scrambled message
  - Decryption algorithm – cipher text and key

Encryption using Public-Key system



Encryption using Public-Key system

- The essential steps involved in a public-key encryption

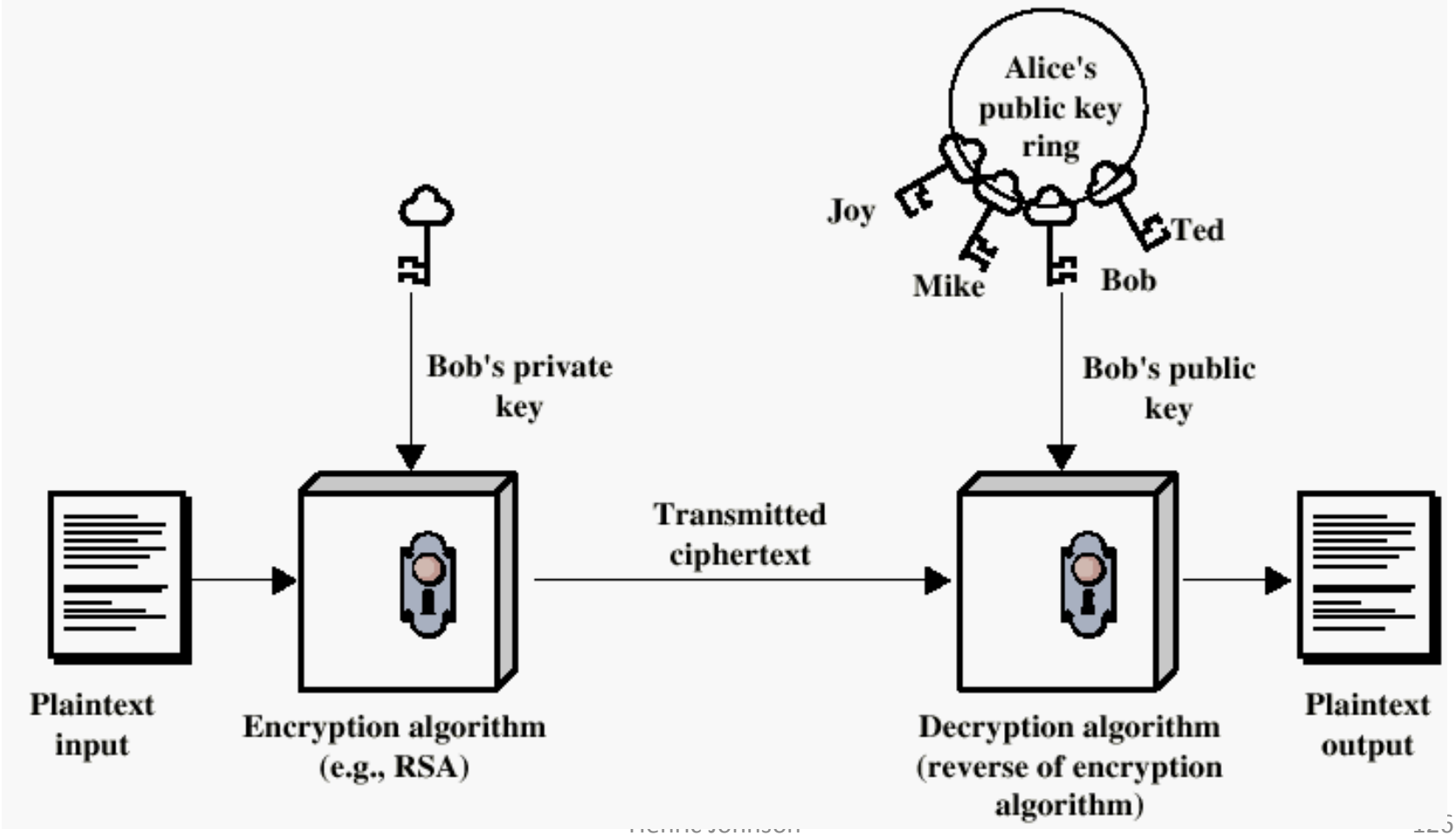
1.) Each user generates a pair of keys to be used for encryption and decryption

2)Each user places one of the two keys in a public register and the other key is kept private.

3) If B wants to send a confidential message to A, B encrypts the message using A's public key.

4) When A receives the message, she decrypts it using her private key.

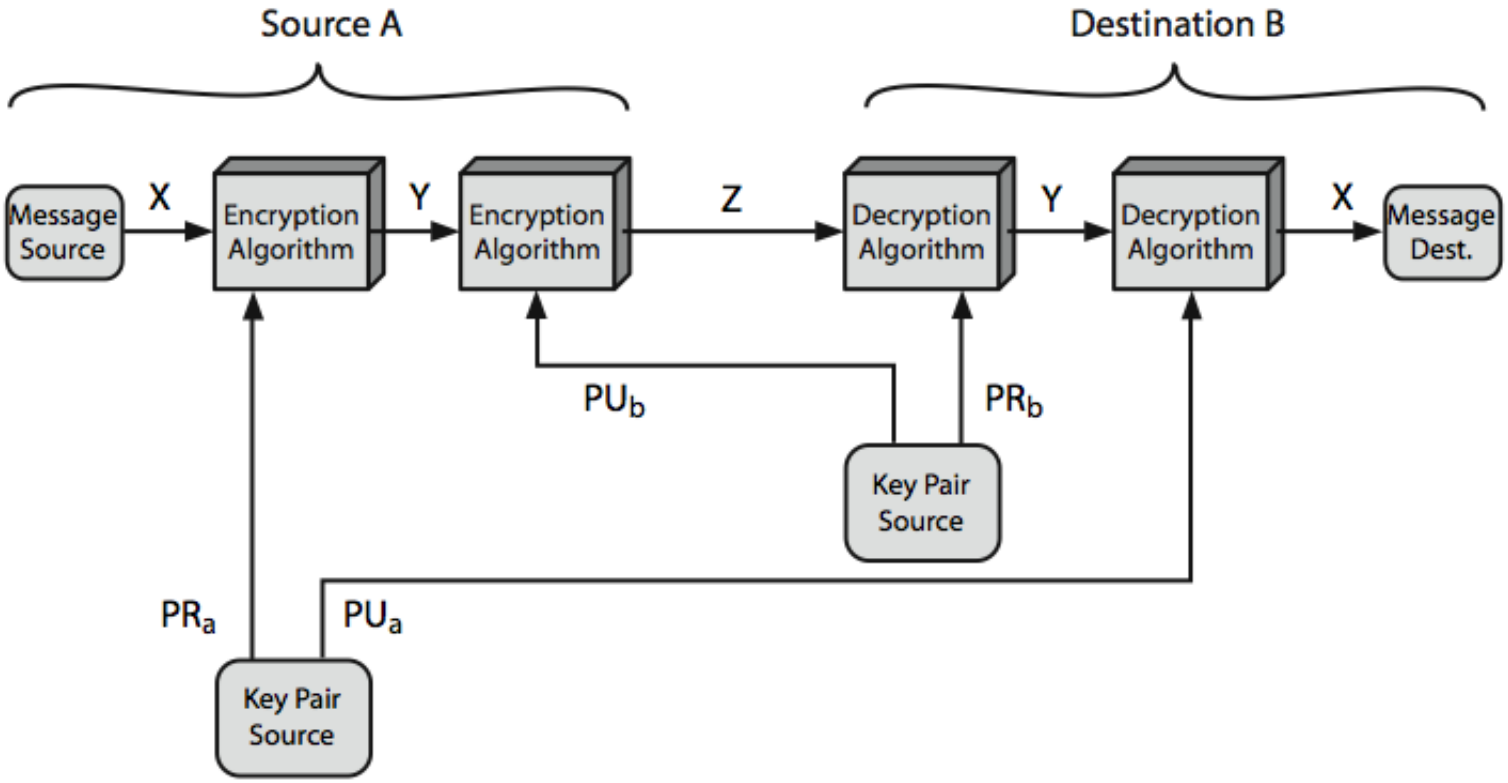
Authentication using Public-Key System



Authentication using Public-Key System

- B will encrypt the message using his private key: Y=E(KRb,X).
- Receiver decrypts using B's public key KRB.
- This shows the authenticity of the sender because (supposedly) he is the only one who knows the private key.
- The entire encrypted message serves as a digital signature

Public-Key Cryptosystems



Applications of public- key crypto systems

- Can classify the use of public key cryptosystems into three categories

- Encryption/decryption: sender encrypts message with recipient's public key

- Digital signature: sender "signs" a message with its private key

- Key exchange : Two sides cooperate to exchange a session key

Security of Public Key Schemes

- Like private key schemes brute force **exhaustive search** attack is always theoretically possible
- But keys used are too large (>512bits)
- The key sizes that have been proposed do make brute-force attack impractical but result in en/dec speeds that are too slow.
- Public-key encryption is currently confined to key management and signature applications

Direct Digital Signature

- Direct Digital Signatures involve the direct application of public-key algorithms involving only the communicating parties(source, destination)

- It is important to perform the signature function first and then an outer confidentiality function, since in case of dispute, some third party must view the message and its signature

- But these approaches are dependent on the security of the sender’s private-key. Will have problems if it is lost/stolen and signatures forged

- Need time-stamps and timely key revocation.

Arbitrated Digital Signatures

- The problems associated with direct digital signatures can be addressed by using an arbiter, in a variety of possible arrangements
- The arbiter plays a sensitive and crucial role in this sort of scheme, and all parties must have a great deal of trust that the arbitration mechanism is working properly
- These schemes can be implemented with either private or public-key algorithms, and the arbiter may or may not see the actual message contents

Using Public Key Encryption

(1) $X \rightarrow A: ID_X \| E(PR_x, [ID_X \| E(PU_y, E(PR_x, M))])$

(2) $A \rightarrow Y: E(PR_a, [ID_X \| E(PU_y, E(PR_x, M)) \| T])$

(c) Public-Key Encryption, Arbiter Does Not See Message

Notation:

X = sender
Y = recipient
A = Arbiter

M = message
T = timestamp

Public Key Encryption

- X double encrypts a message M first with X's private key,PRx, and then with Y's public key, PUy. This is a signed, secret version of the message
- This signed message, together with X's identifier , is encrypted again with PRx and, together with IDx, is sent to A.
- The inner, double encrypted message is secure from the arbiter (and everyone else except Y)
- A can decrypt the outer encryption to assure that the message must have come from X (because only X has PRx)
- Then A transmits a message to Y, encrypted with PRa. The message includes IDx, the double encrypted message, and a timestamp.
- Arbiter does not see message

Key Management

- public-key encryption helps address key distribution problems
- Have two aspects of this:
  - distribution of public keys
  - use of public-key encryption to distribute secret keys

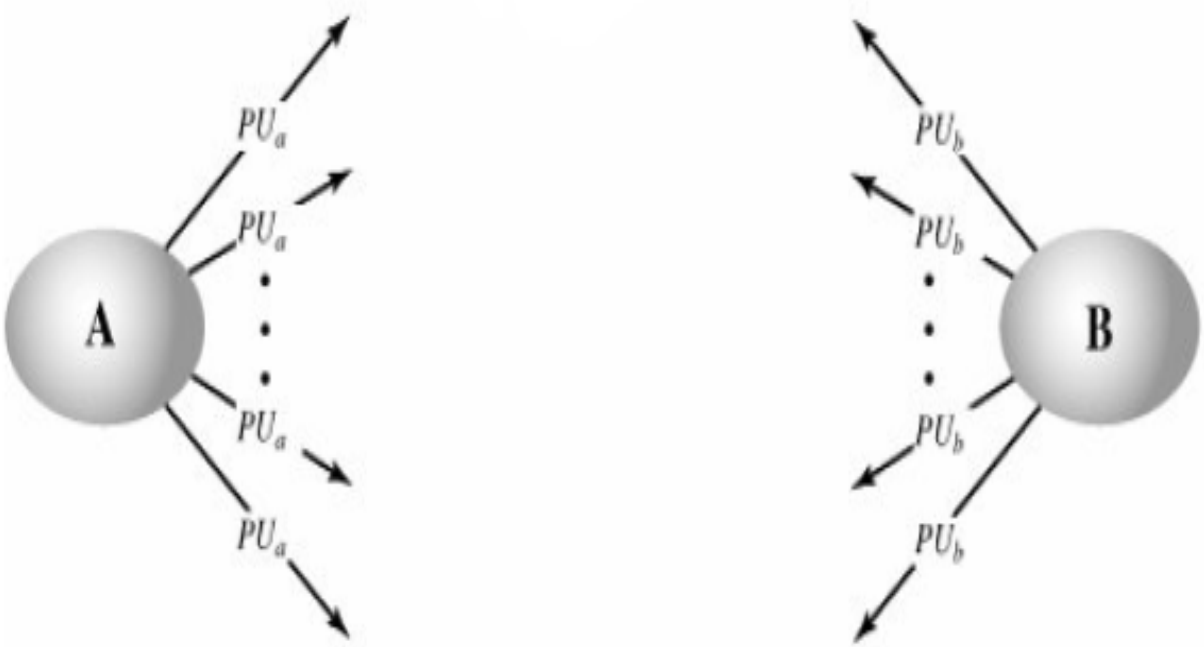
Distribution of Public Keys

- can be considered as using one of:
  - public announcement
  - publicly available directory
  - public-key certificates

Public Announcement

- users distribute public keys to recipients or broadcast to community at large
  - eg. append PGP keys to email messages or post to news groups or email list
- major weakness is forgery
  - anyone can create a key claiming to be someone else and broadcast it
  - until forgery is discovered can masquerade as claimed user

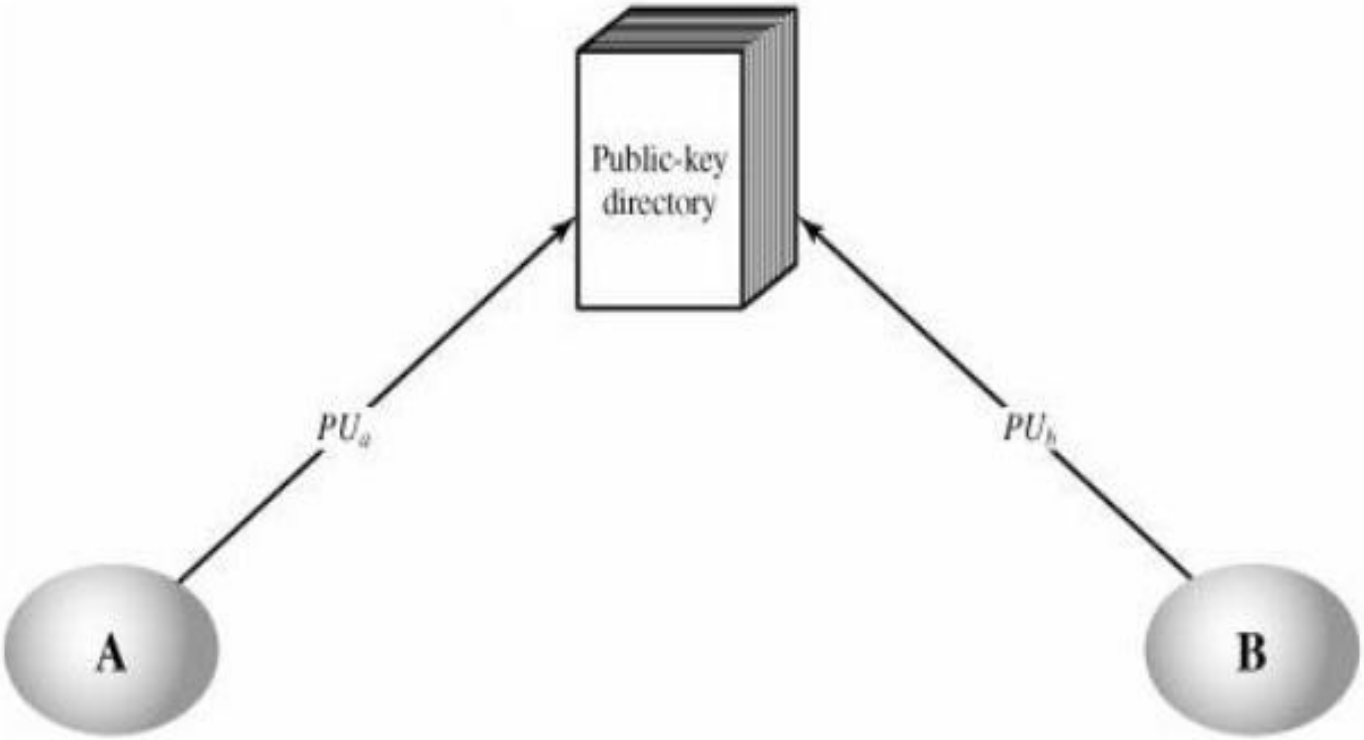
Uncontrolled Public-Key Distribution



Publicly Available Directory

- can obtain greater security by registering keys with a public directory
- directory must be trusted with properties:
  - contains {name,public-key} entries
  - participants register securely with directory
  - participants can replace key at any time
  - directory is periodically published
  - directory can be accessed electronically
- still vulnerable to tampering or forgery

Public-Key Publication



Public-Key Certificates

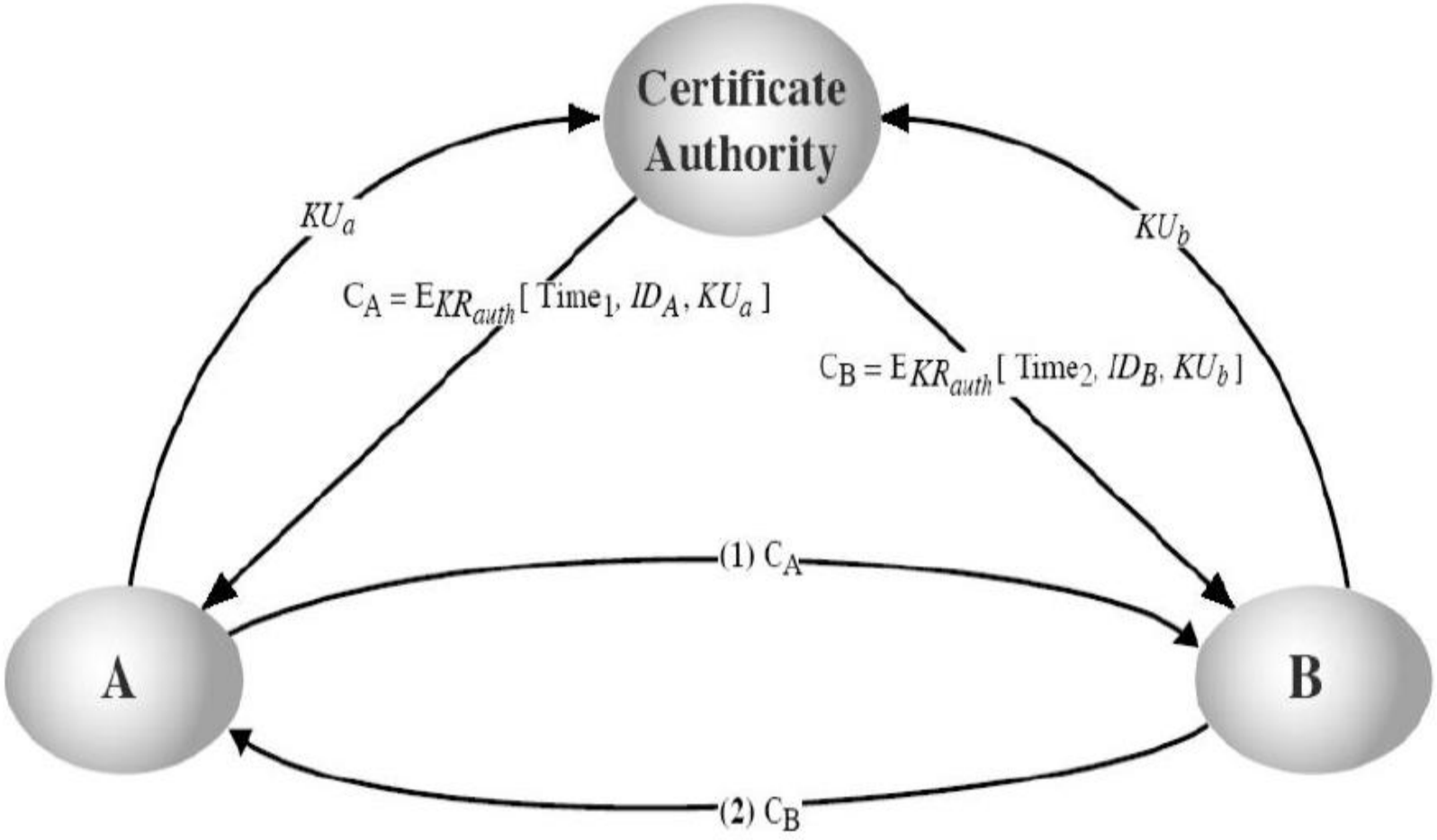
- A certificate binds an identity to public key, with all contents signed by a trusted Public-Key or Certificate Authority (CA).

- A user can present his or her public key to the authority in a secure manner, and obtain a certificate. The user can then publish the certificate.

- A participant can also convey its key information to another by transmitting its certificate. Other participants can verify that the certificate was created by the authority.

Public-Key Certificates

- This certificate issuing scheme does have the following requirements:
- 1. Any participant can read a certificate to determine the name and public key of the certificate's owner.
- 2. Any participant can verify that the certificate originated from the certificate authority and is not counterfeit.
- 3. Only the certificate authority can create and update certificates.
- 4. Any participant can verify the currency of the certificate.


Certificate Authority
KU_a
KU_b
$C_A = E_{KR_{auth}}[\text{Time}_1, ID_A, KU_a]$
$C_B = E_{KR_{auth}}[\text{Time}_2, ID_B, KU_b]$
(1) C_A
A
B
(2) C_B

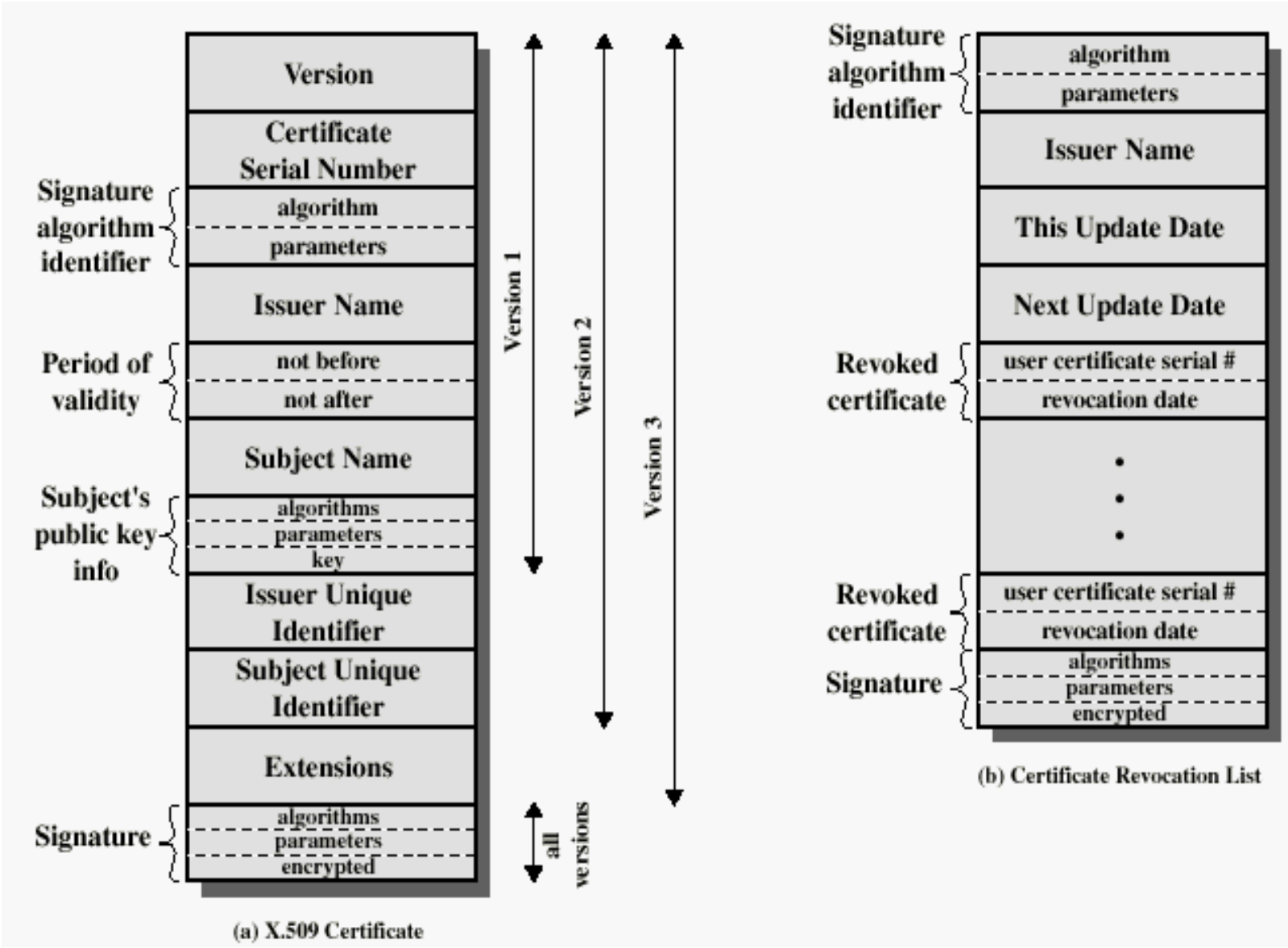
X.509 Authentication Service

- Distributed set of servers that maintains a database about users.
- Each certificate contains the public key of a user and is signed with the private key of a CA.
- Is used in S/MIME, IP Security, SSL/TLS and SET.
- RSA is recommended to use.

X.509 Authentication Service

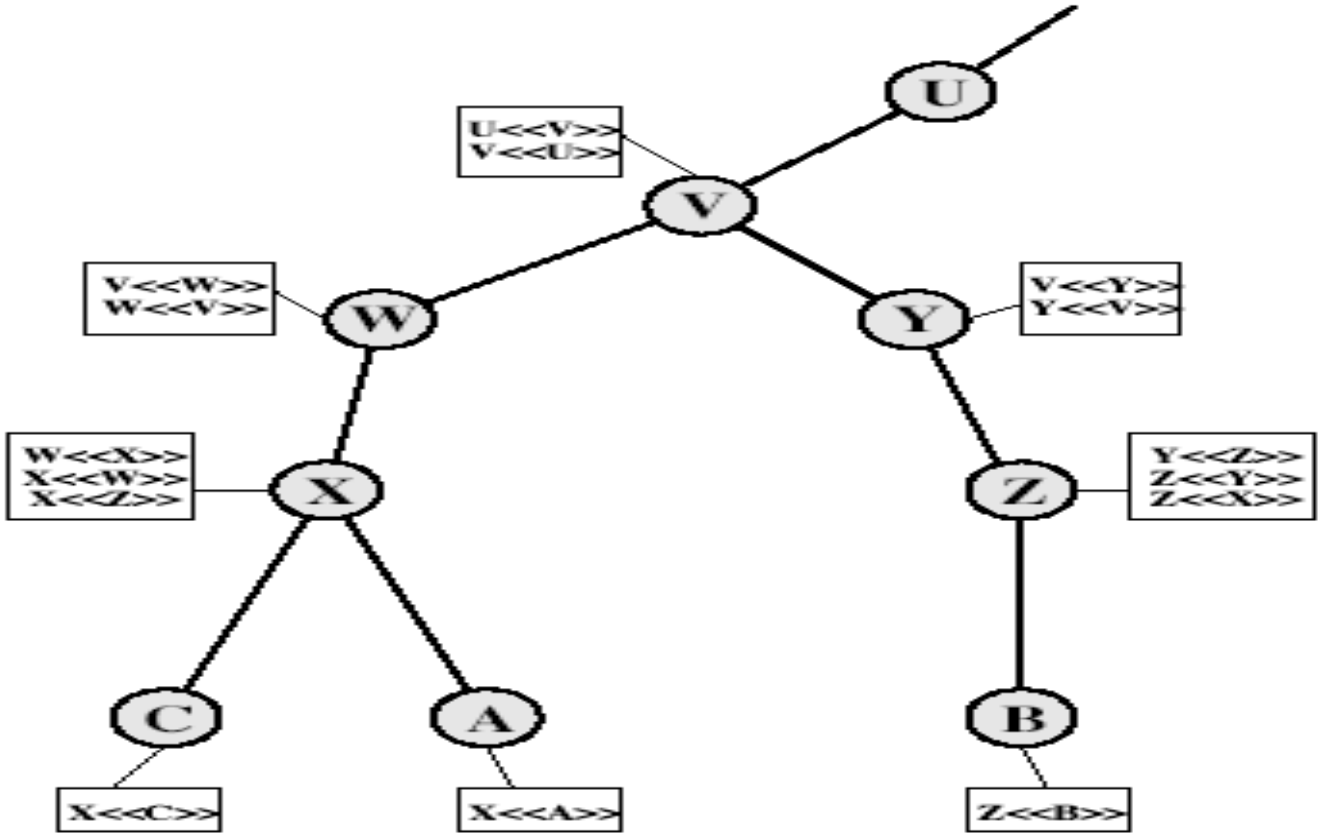
- The heart of the X.509 scheme is the public-key certificate associated with each user
- These user certificates are assumed to be created by some trusted certification authority (CA) and placed in the directory by the CA or by the user
- The directory server itself is not responsible for the creation of public keys or for the certification function; it merely provides an easily accessible location for users to obtain certificates.

X.509 Formats



X.509 Certificates

- issued by a Certification Authority (CA), containing:
  - version (1, 2, or 3)
  - serial number (unique within CA) identifying certificate
  - signature algorithm identifier
  - issuer X.500 name (CA)
  - period of validity (from - to dates)
  - subject X.500 name (name of owner)
  - subject public-key info (algorithm, parameters, key)
  - issuer unique identifier (v2+)
  - subject unique identifier (v2+)
  - extension fields (v3)
  - signature (of hash of all fields in certificate)



Track chains of certificates:

A acquires B certificate using chain: X<<W>>W<<V>>V<<Y>>Y<<Z>>Z<<B>>

B acquires A certificate using chain: Z<<Y>>Y<<V>>V<<W>>W<<X>>X<<A>>

KERBEROS

- Users wish to access services on servers.
- Three threats exist:
  - User pretends to be another user.
  - User alters the network address of a workstation.
  - User eavesdrops on exchanges and uses a replay attack.

Kerberos Requirements

- first published report identified its requirements as:
  - security
  - reliability
  - transparency
  - scalability

KERBEROS

- Provides a centralized authentication server to authenticate users to servers and servers to users.
- Relies on conventional encryption, making no use of public-key encryption
- Version 4 makes use of DES

Kerberos 4 Overview

- a basic third-party authentication scheme
- have an *Authentication Server (AS)*
  - users initially negotiate with AS to identify themselves
  - AS provides a non-corruptible *authentication credential (ticket granting ticket TGT)*
- have a *Ticket Granting server (TGS)*
  - users subsequently request access to other services from TGS on basis of users TGT

Kerberos Version 4

- Terms:
  - C = Client
  - AS = authentication server
  - V = server
  - IDc = identifier of user on C
  - IDv = identifier of V
  - P_c = password of user on C
  - ADc = network address of C
  - Kv = secret encryption key shared by AS and V
  - TS = timestamp
  - ‖ = concatenation

A Simple Authentication Dialogue

(1) C → AS: $IDc \| P_c \| IDv$

(2) AS → C: Ticket

(3) C → V: $IDc \| Ticket$

$Ticket = E_{Kv}[IDc \| AD_c \| IDv]$

TICKET GRANTING SERVER

- Introducing a ticket-granting server (TGS)
  - The user first requests a ticket-granting ticket ($Ticket_{tgs}$) from the AS;
  - The user then authenticates itself to TGS for a ticket ($Ticket_{v}$) for accessing new service;
  - The user finally authenticate itself to V for requesting a particular service.

Authentication Dialogue II

- Once per user logon session
- (1) C -> AS: $ID_C \| ID_{TGS}$
- (2) AS -> C: $E_{K(C)}[Ticket_{TGS}]$
- $Ticket_{TGS}$ is equal to
  - $E_{K(TGS)}[ID_C \| AD_C \| ID_{TGS} \| TS_1 \| Lifetime_1]$

Explaining the fields

- TGS = Ticket-granting server
- ID_{TGS} = Identifier of the TGS
- $Ticket_{TGS}$ = *Ticket-granting ticket or TGT*
- TS_1 = timestamp
- $Lifetime_1$ = lifetime for the TGT
- $K_{(C)}$ = key derived from user's password

Messages (3) and (4)

- Once per type of service
- (3) C -> TGS: $ID_C \| ID_V \| Ticket_{TGS}$
- (4) TGS -> C : $Ticket_V$
- $Ticket_V$ is equal to
  - $E_{K(V)} [ID_C \| AD_C \| ID_V \| TS_2 \| Lifetime_2]$

  K(V): key shared between V and TGS

  Is called the *service-granting ticket (SGT)*

Message 5

- Once per service session
- (5) C -> V: ID_C || $Ticket_V$

Table 4.1
Summary of Kerberos Version 4 Message Exchanges

(a) Authentication Service Exchange to obtain ticket-granting ticket

(1) C → AS $ID_c \| ID_{tgs} \| TS_1$

(2) AS → C $E(K_c, [K_{c,tgs} \| ID_{tgs} \| TS_2 \| Lifetime_2 \| Ticket_{tgs}])$

$$Ticket_{tgs} = E(K_{tgs}, [K_{c,tgs} \| ID_C \| AD_C \| ID_{tgs} \| TS_2 \| Lifetime_2])$$

(3) C → TGS $ID_v \| Ticket_{tgs} \| Authenticator_c$

(4) TGS → C $E(K_{c,tgs}, [K_{c,v} \| ID_v \| TS_4 \| Ticket_v])$

$$Ticket_{tgs} = E(K_{tgs}, [K_{c,tgs} \| ID_C \| AD_C \| ID_{tgs} \| TS_2 \| Lifetime_2])$$

$$Ticket_v = E(K_v, [K_{c,v} \| ID_C \| AD_C \| ID_v \| TS_4 \| Lifetime_4])$$

$$Authenticator_c = E(K_{c,tgs}, [ID_C \| AD_C \| TS_3])$$

(5) C → V $Ticket_v \| Authenticator_c$

(6) V → C $E(K_{c,v}, [TS_5 + 1])$ (for mutual authentication)

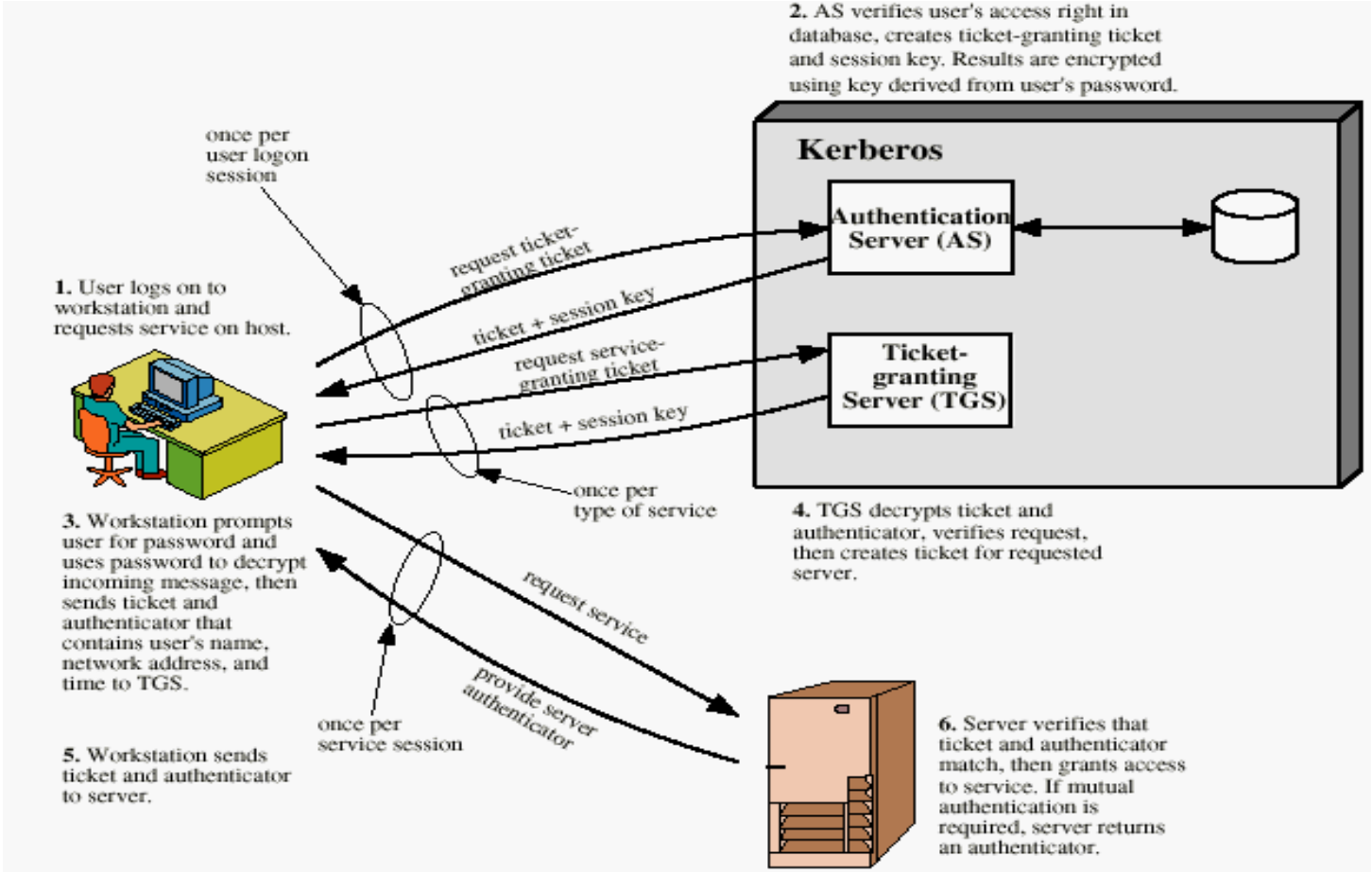
$$Ticket_v = E(K_v, [K_{c,v} \| ID_C \| AD_C \| ID_v \| TS_4 \| Lifetime_4])$$

$$Authenticator_c = E(K_{c,v}, [ID_C \| AD_C \| TS_5])$$

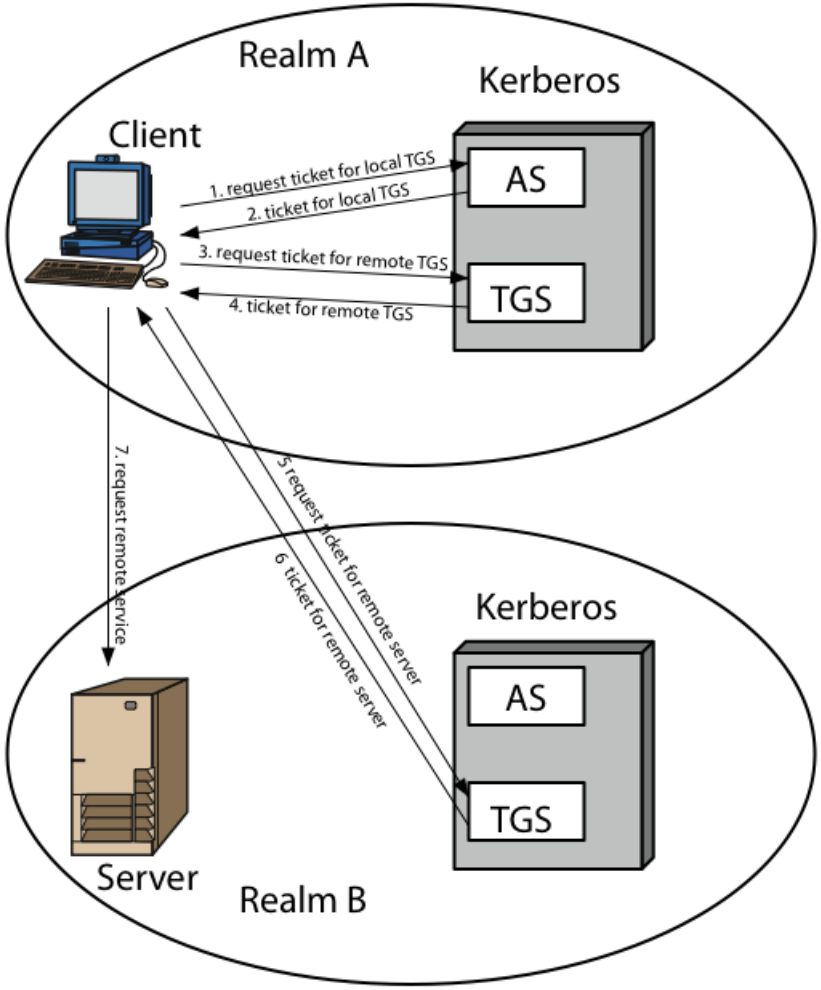
Kerberos Realms

- a Kerberos environment consists of:
  - a Kerberos server
  - a number of clients, all registered with server
  - application servers, sharing keys with server
- this is termed a realm
  - typically a single administrative domain
- if have multiple realms, their Kerberos servers must share keys and trust

Overview of Kerberos



Kerberos Realms



RSA

- by Rivest, Shamir & Adleman of MIT in 1977
- best known & widely used public-key scheme
- uses large integers (eg. 1024 bits)
- security due to cost of factoring large numbers

RSA Key Setup

- each user generates a public/private key pair by:
- selecting two large primes at random: p, q
- computing their system modulus n=p.q
  - note ø(n)=(p-1)(q-1)
- selecting at random the encryption key e
  - where $1<e<ø(n)$, $gcd(e,ø(n))=1$
- solve following equation to find decryption key d
  - $e.d=1 \bmod ø(n)$ and $0 \leq d \leq n$
- publish their public encryption key: PU={e,n}
- keep secret private decryption key: PR={d,n}

RSA En/decryption

- to encrypt a message M the sender:
  - obtains **public key** of recipient PU={e,n}
  - computes: $C = M^e \bmod n$, where $0 \leq M < n$
- to decrypt the ciphertext C the owner:
  - uses their private key PR={d,n}
  - computes: $M = C^d \bmod n$

RSA Example - Key Setup

1. Select primes: p=17 & q=11
2. Calculate $n = pq$ =17 x 11=187
3. Calculate $\phi(n)=(p-1)(q-1)$=16x10=160
4. Select e: gcd(e,160)=1; choose e=7
5. Determine d: de=1 mod 160 and $d < 160$ Value is d=23 since 23x7=161= 10x160+1
6. Publish public key PU={7,187}
7. Keep secret private key PR={23,187}

RSA Example - En/Decryption

- sample RSA encryption/decryption is:
- given message M = 88
- encryption:

  $C = 88^7 \bmod 187 = 11$
- decryption:

  $M = 11^{23} \bmod 187 = 88$

Diffie-Hellman Key Exchange

- The purpose of the algorithm is to enable two users to exchange a secret key securely.
- The algorithm itself is limited to the exchange of the keys
- The D-H algorithm depends for its effectiveness on the difficulty of computing discrete logarithms

Diffie-Hellman Key Exchange

- First, a primitive root of a prime number p, can be defined as one whose powers generate all the integers from 1 to p-1.
- If a is a primitive root of the prime number p, then the numbers, ***a mod p, a2 mod p,..., ap-1 mod p, are distinct and consist of the integers from 1 through p 1 in some permutation.***

Diffie-Hellman Key Exchange

- all users agree on global parameters:
  - large prime integer or polynomial q
  - α a primitive root mod q

we can find a unique exponent i such that .The exponent i is referred to as the discrete logarithm of b for the base a, mod p.

Diffie-Hellman Key Exchange

- shared session key for users A & B is K_{AB}:

  $K_{AB} = \alpha^{x_A . x_B} \bmod q$

  $= y_A^{x_B} \bmod q$ (which **B** can compute)

  $= y_B^{x_A} \bmod q$ (which **A** can compute)
- K_{AB} is used as session key in private-key encryption scheme between Alice and Bob
- if Alice and Bob subsequently communicate, they will have the **same** key as before, unless they choose new public-keys
- attacker needs an x, must solve discrete log

Diffie-Hellman Example

- users Alice & Bob who wish to swap keys:
- agree on prime q=353 and α=3
- select random secret keys:
  - A chooses x_A=97, B chooses x_B=233
- compute public keys:
  - $y_A = 3^{97} \bmod 353 = 40$ (Alice)
  - $y_B = 3^{233} \bmod 353 = 248$ (Bob)
- compute shared session key as:

$K_{AB} = y_B^{x_A} \bmod 353 = 248^{97} = 160$ (Alice)

$K_{AB} = y_A^{x_B} \bmod 353 = 40^{233} = 160$ (Bob)

UNIT-III

USM-CRYPTOGRAPHIC FUNCTIONS

- Remote users:
- Any principal at a remote SNMP engine for which communication is desired

- USM allows the use of one of two alternative authentication protocols :HMAC-MD5-96 and HMAC-SHA-96

- USM uses the CIPHER BLOCK CHAINING (CBC) mode of the Data encryption standard (DES) for standard

Intruders

- Three classes of intruders (hackers or crackers):
  - Masquerader
  - Misfeasor
  - Clandestine user

Intrusion Techniques

- System maintain a file that associates a password with each authorized user.
- Password file can be protected with:
  - One-way encryption
  - Access Control

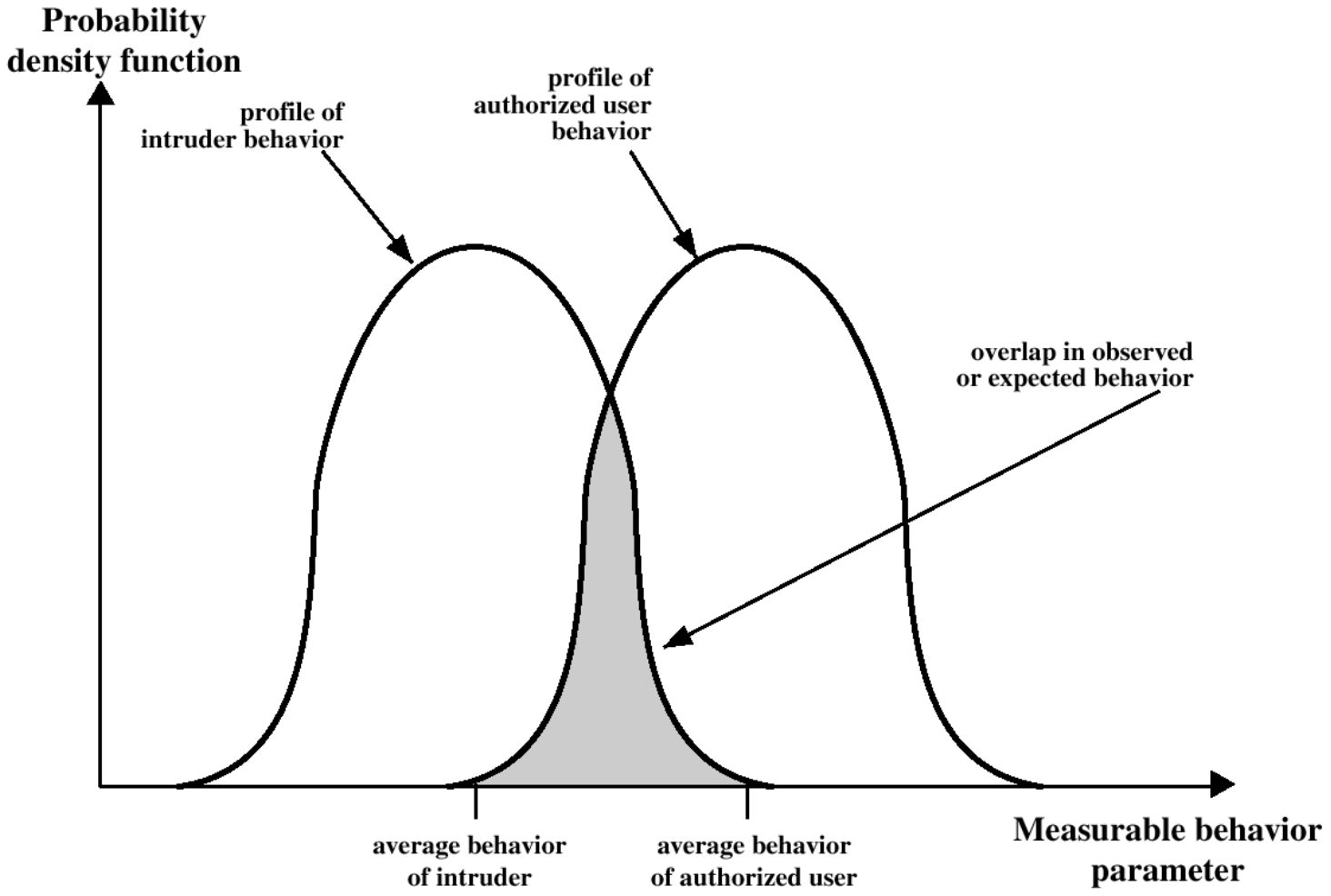
Password Selecting Strategies

- User ducation
- Computer-generated passwords
- Reactive password checking
- Proactive password checking

Intrusion Detection

- The intruder can be identified and ejected from the system.
- An effective intrusion detection can prevent intrusions.
- Intrusion detection enables the collection of information about intrusion techniques that can be used to strengthen the intrusion prevention facility.

Profiles of Behavior of Intruders and Authorized Users



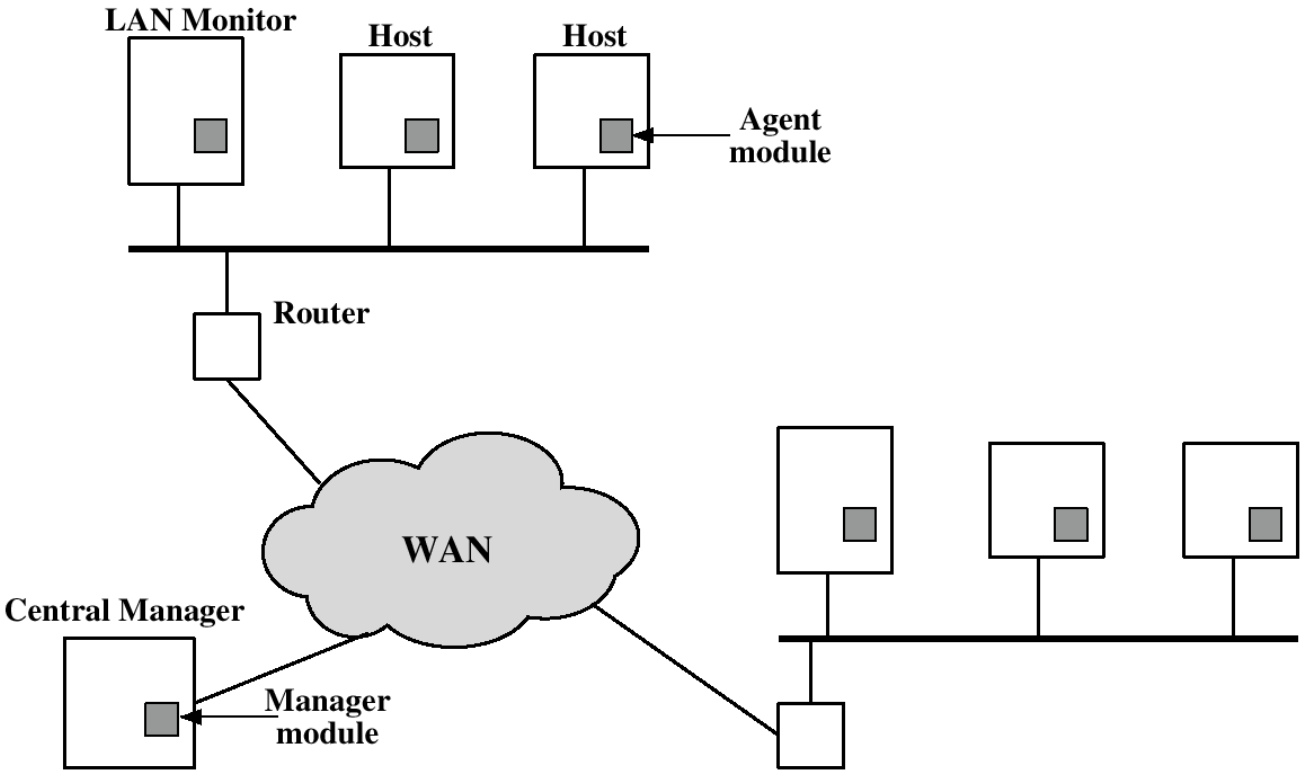
Intrusion Detection

- Statistical anomaly detection
  - Treshold detection
  - Profile based
- Rule based detection
  - Anomaly detection
  - Penetration identidication

Measures used for Intrusion Detection

- Login frequency by day and time.
- Frequency of login at different locations.
- Time since last login.
- Password failures at login.
- Execution frequency.
- Execution denials.
- Read, write, create, delete frequency.
- Failure count for read, write, create and delete.

Distributed Intrusion Detection

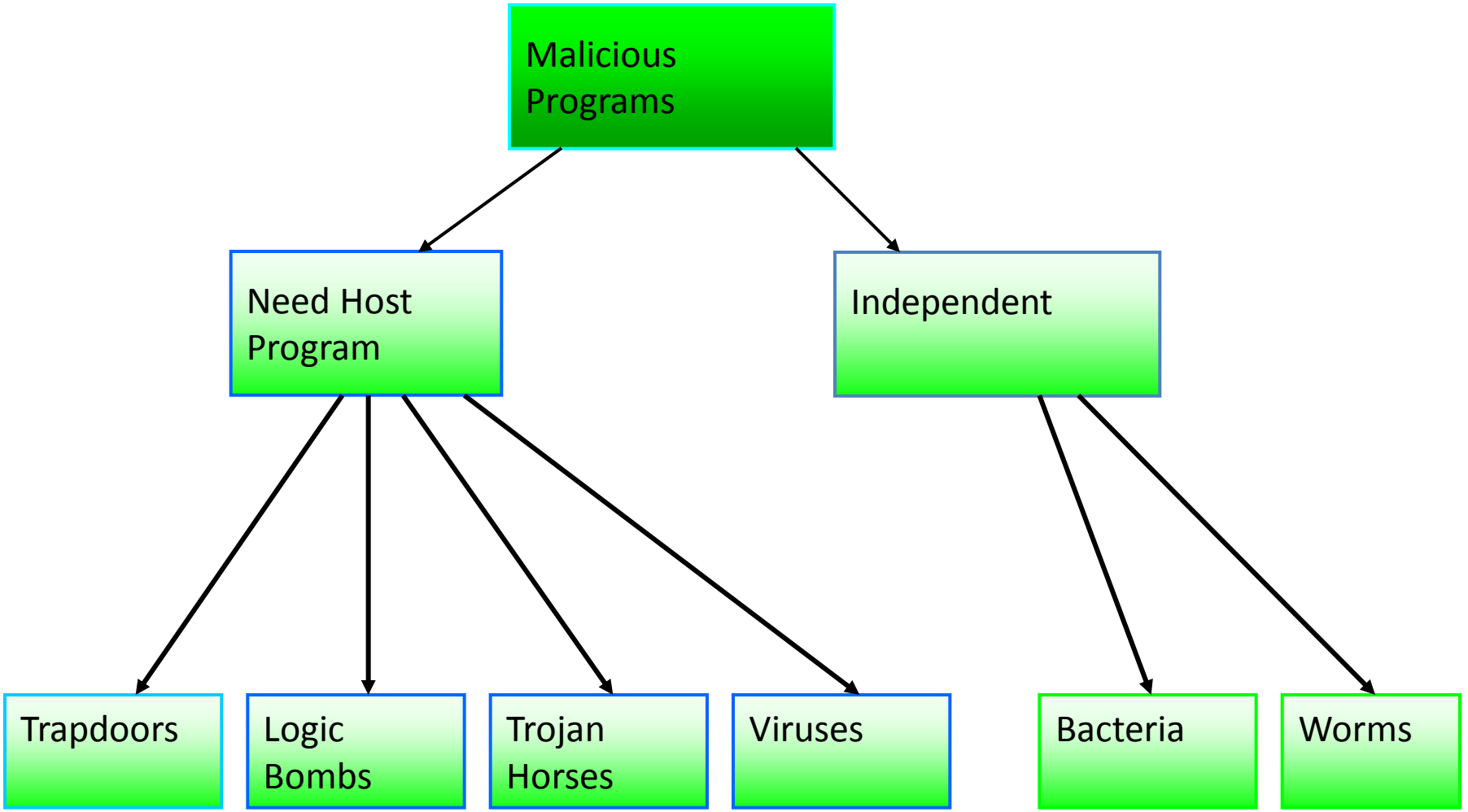


Viruses and ”Malicious Programs”

- Computer “Viruses” and related programs have the ability to replicate themselves on an ever increasing number of computers. They originally spread by people sharing floppy disks. Now they spread primarily over the Internet (a “Worm”).

- Other “Malicious Programs” may be installed by hand on a single machine. They may also be built into widely distributed commercial software packages. These are very hard to detect before the payload activates (Trojan Horses, Trap Doors, and Logic Bombs).

Taxanomy of Malicious Programs



Definitions

- Virus - code that copies itself into other programs.
- A "Bacteria" replicates until it fills all disk space, or CPU cycles.
- Payload - harmful things the malicious program does, after it has had time to spread.
- Worm - a program that replicates itself across the network (usually riding on email messages or attached documents (e.g., macro viruses).

Definitions

- Trojan Horse - instructions in an otherwise good program that cause bad things to happen (sending your data or password to an attacker over the net).
- Logic Bomb - malicious code that activates on an event (e.g., date).
- Trap Door (or Back Door) - undocumented entry point written into code for debugging that can allow unwanted users.

Virus Phases

- **Dormant phase -** the virus is idle
- **Propagation phase -** the virus places an identical copy of itself into other programs
- **Triggering phase –** the virus is activated to perform the function for which it was intended
- **Execution phase** – the function is performed

Types of Viruses

- **Parasitic Virus** - attaches itself to executable files as part of their code. Runs whenever the host program runs.
- **Memory-resident Virus** - Lodges in main memory as part of the residual operating system.
- **Boot Sector Virus** - infects the boot sector of a disk, and spreads when the operating system boots up (original DOS viruses).
- **Stealth Virus** - explicitly designed to hide from Virus Scanning programs.
- **Polymorphic Virus** - mutates with every new host to prevent signature detection.

Macro Viruses

- Microsoft Office applications allow “macros” to be part of the document. The macro could run whenever the document is opened, or when a certain command is selected (Save File).
- Platform independent.
- Infect documents, delete files, generate email and edit letters.

Firewall Design Principles

- Information systems undergo a steady evolution (from small LAN`s to Internet connectivity)
- Strong security features for all workstations and servers not established

Firewall Design Principles

- The firewall is inserted between the premises network and the Internet
- Aims:
  - Establish a controlled link
  - Protect the premises network from Internet-based attacks
  - Provide a single choke point

Firewall Characteristics

- Design goals:
  - All traffic from inside to outside must pass through the firewall (physically blocking all access to the local network except via the firewall)
  - Only authorized traffic (defined by the local security police) will be allowed to pass

Firewall Characteristics

- Design goals:
  - The firewall itself is immune to penetration (use of trusted system with a secure operating system)

Firewall Characteristics

- Four general techniques:
- Service control
  - Determines the types of Internet services that can be accessed, inbound or outbound
- Direction control
  - Determines the direction in which particular service requests are allowed to flow

Firewall Characteristics

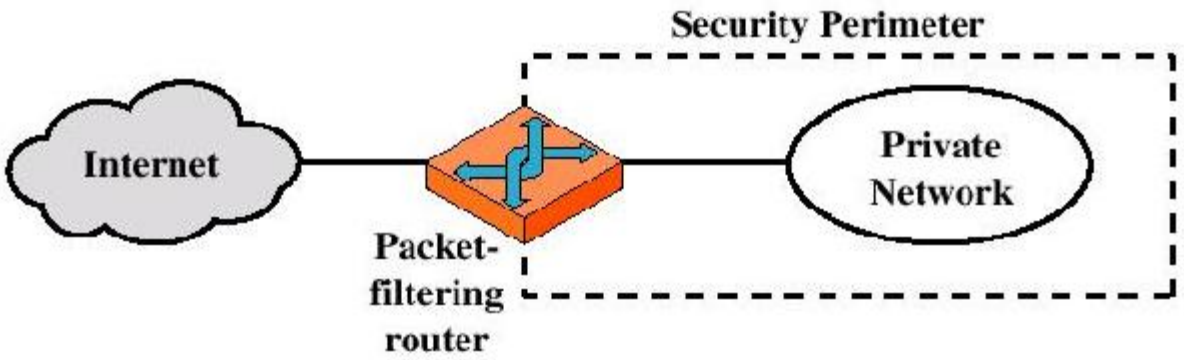
- User control
  - Controls access to a service according to which user is attempting to access it
- Behavior control
  - Controls how particular services are used (e.g. filter e-mail)

Types of Firewalls

- Three common types of Firewalls:
  - Packet-filtering routers
  - Application-level gateways
  - Circuit-level gateways
  - (Bastion host)

Types of Firewalls

- Packet-filtering Router



Types of Firewalls

- Packet-filtering Router
  - Applies a set of rules to each incoming IP packet and then forwards or discards the packet
  - Filter packets going in both directions
  - The packet filter is typically set up as a list of rules based on matches to fields in the IP or TCP header
  - Two default policies (discard or forward)

Types of Firewalls

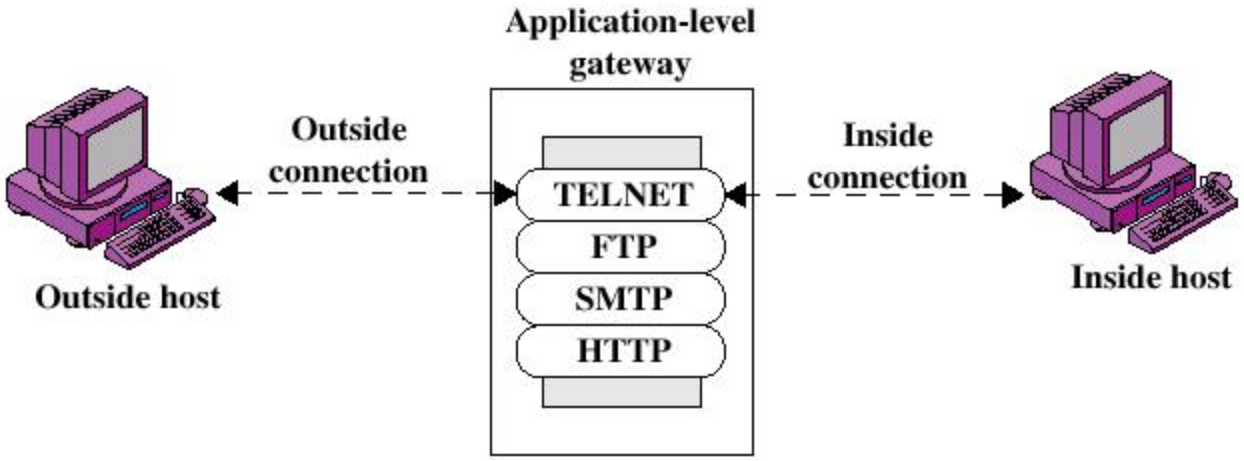
- Advantages:
  - Simplicity
  - Transparency to users
  - High speed
- Disadvantages:
  - Difficulty of setting up packet filter rules
  - Lack of Authentication

Types of Firewalls

- Possible attacks and appropriate countermeasures
  - IP address spoofing
  - Source routing attacks
  - Tiny fragment attacks

Types of Firewalls

- Application-level Gateway



Types of Firewalls

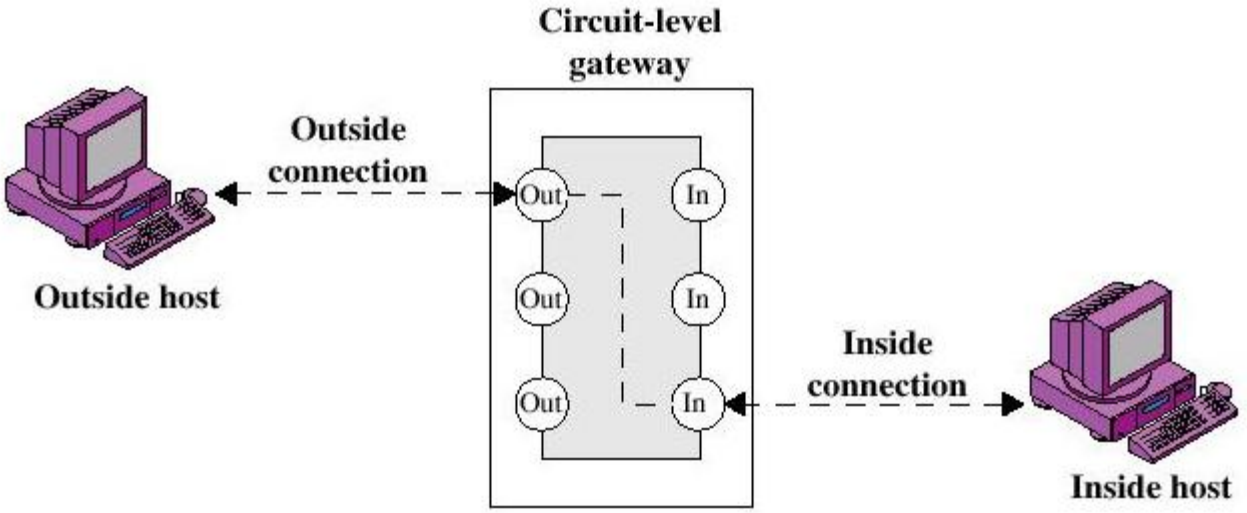
- Application-level Gateway
  - Also called proxy server
  - Acts as a relay of application-level traffic

Types of Firewalls

- Advantages:
  - Higher security than packet filters
  - Only need to scrutinize a few allowable applications
  - Easy to log and audit all incoming traffic
- Disadvantages:
  - Additional processing overhead on each connection (gateway as splice point)

Types of Firewalls

- Circuit-level Gateway



Types of Firewalls

- Circuit-level Gateway
  - Stand-alone system or
  - Specialized function performed by an Application-level Gateway
  - Sets up two TCP connections
  - The gateway typically relays TCP segments from one connection to the other without examining the contents

Types of Firewalls

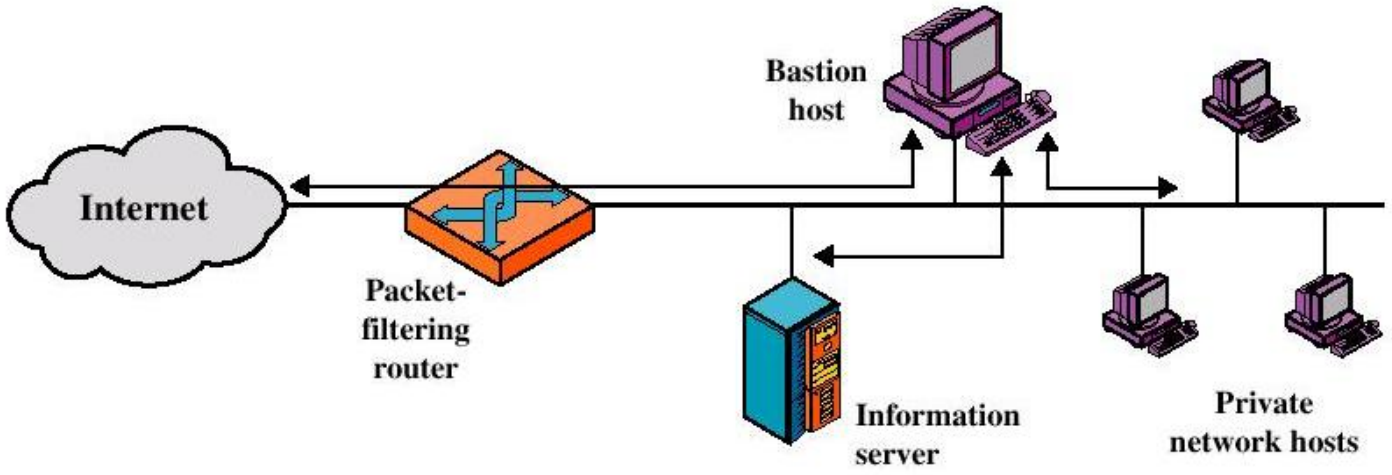
- Bastion Host
  - A system identified by the firewall administrator as a critical strong point in the network´s security
  - The bastion host serves as a platform for an application-level or circuit-level gateway

Firewall Configurations

- In addition to the use of simple configuration of a single system (single packet filtering router or single gateway), more complex configurations are possible
- Three common configurations

Firewall Configurations

- Screened host firewall system (single-homed bastion host)



Firewall Configurations

- Screened host firewall, single-homed bastion configuration
- Firewall consists of two systems:
  - A packet-filtering router
  - A bastion host

Firewall Configurations

- Configuration for the packet-filtering router:
  - Only packets from and to the bastion host are allowed to pass through the router
- The bastion host performs authentication and proxy functions

Firewall Configurations

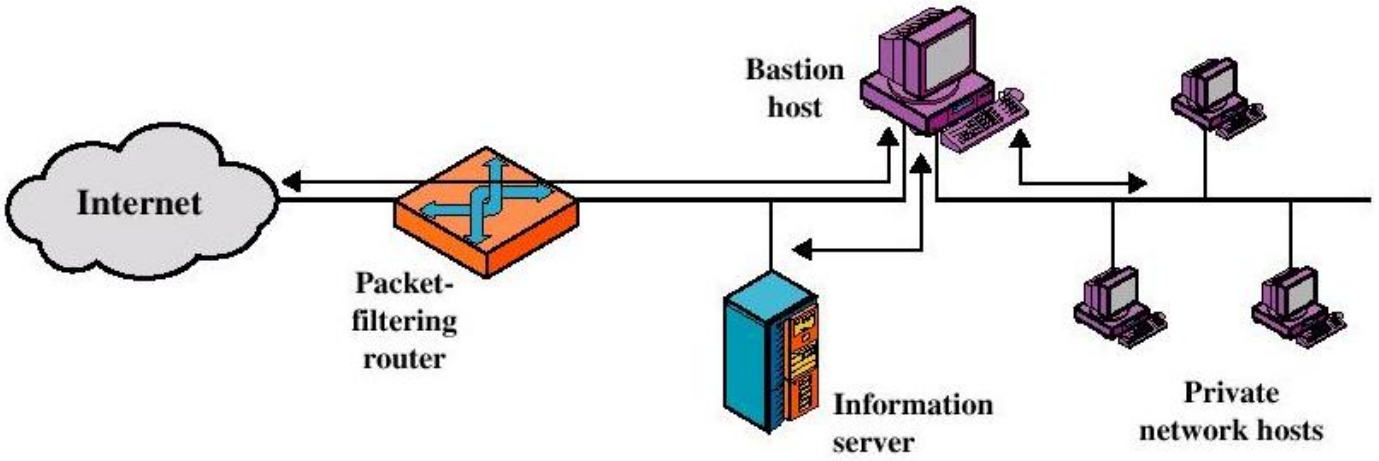
- Greater security than single configurations because of two reasons:
  - This configuration implements both packet-level and application-level filtering (allowing for flexibility in defining security policy)
  - An intruder must generally penetrate two separate systems

Firewall Configurations

- This configuration also affords flexibility in providing direct Internet access (public information server, e.g. Web server)

Firewall Configurations

- Screened host firewall system (dual-homed bastion host)

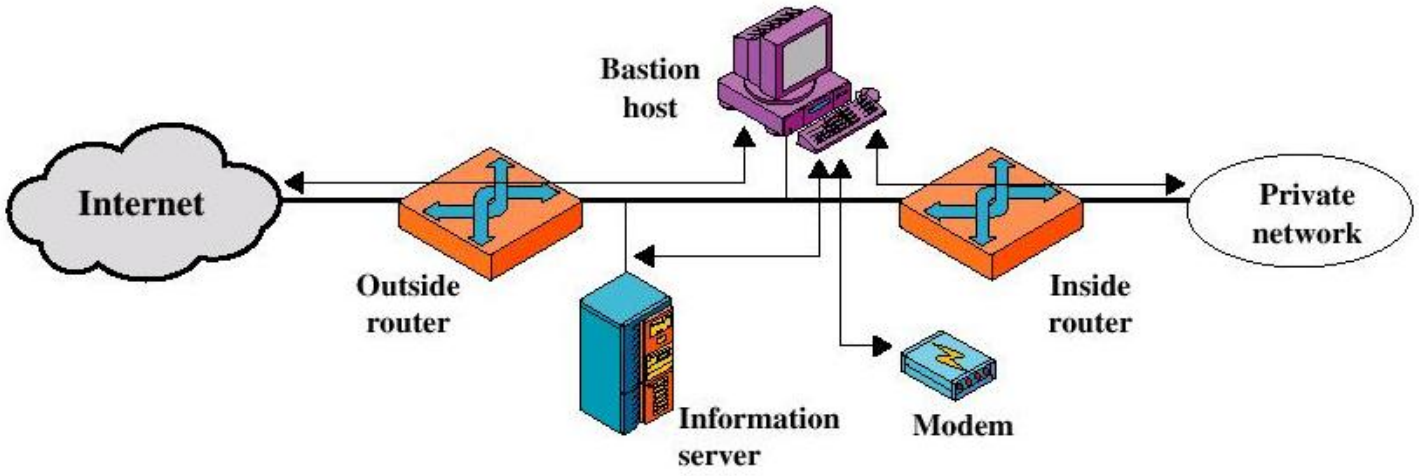


Firewall Configurations

- Screened host firewall, dual-homed bastion configuration
  - The packet-filtering router is not completely compromised
  - Traffic between the Internet and other hosts on the private network has to flow through the bastion host

Firewall Configurations

- Screened-subnet firewall system



Firewall Configurations

- Screened subnet firewall configuration
  - Most secure configuration of the three
  - Two packet-filtering routers are used
  - Creation of an isolated sub-network

Firewall Configurations

- Advantages:
  - Three levels of defense to thwart intruders
  - The outside router advertises only the existence of the screened subnet to the Internet (internal network is invisible to the Internet)

Trusted Systems

- One way to enhance the ability of a system to defend against intruders and malicious programs is to implement trusted system technology

Data Access Control

- Through the user access control procedure (log on), a user can be identified to the system
- Associated with each user, there can be a profile that specifies permissible operations and file accesses
- The operation system can enforce rules based on the user profile

Data Access Control

- General models of access control:
  - Access matrix
  - Access control list
  - Capability list

Data Access Control

- Access Matrix

	Program1	• • •	SegmentA	SegmentB
Process1	Read Execute		Read Write	
Process2				Read
•				
•				
•				

Data Access Control

- Access Matrix: Basic elements of the model
  - Subject: An entity capable of accessing objects, the concept of subject equates with that of process
  - Object: Anything to which access is controlled (e.g. files, programs)
  - Access right: The way in which an object is accessed by a subject (e.g. read, write, execute)

Data Access Control

- Access Control List: Decomposition of the matrix by columns

Access Control List for Program1: Process1 (Read, Execute)
Access Control List for SegmentA: Process1 (Read, Write)
Access Control List for SegmentB: Process2 (Read)

Data Access Control

- Access Control List
  - An access control list lists users and their permitted access right
  - The list may contain a default or public entry

Data Access Control

- Capability list: Decomposition of the matrix by rows

Capability List for Process1:
Program1 (Read, Execute)
SegmentA (Read, Write)
Capability List for Process2:
SegmentB (Read)

Data Access Control

- Capability list
  - A capability ticket specifies authorized objects and operations for a user
  - Each user have a number of tickets

The Concept of Trusted Systems

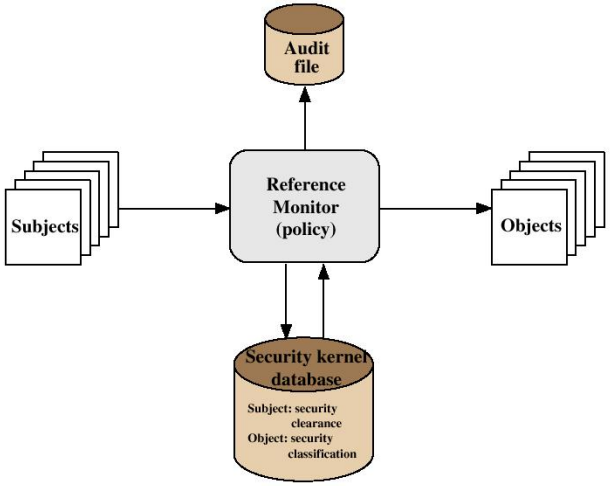
- Trusted Systems
  - Protection of data and resources on the basis of levels of security (e.g. military)
  - Users can be granted clearances to access certain categories of data

The Concept of Trusted Systems

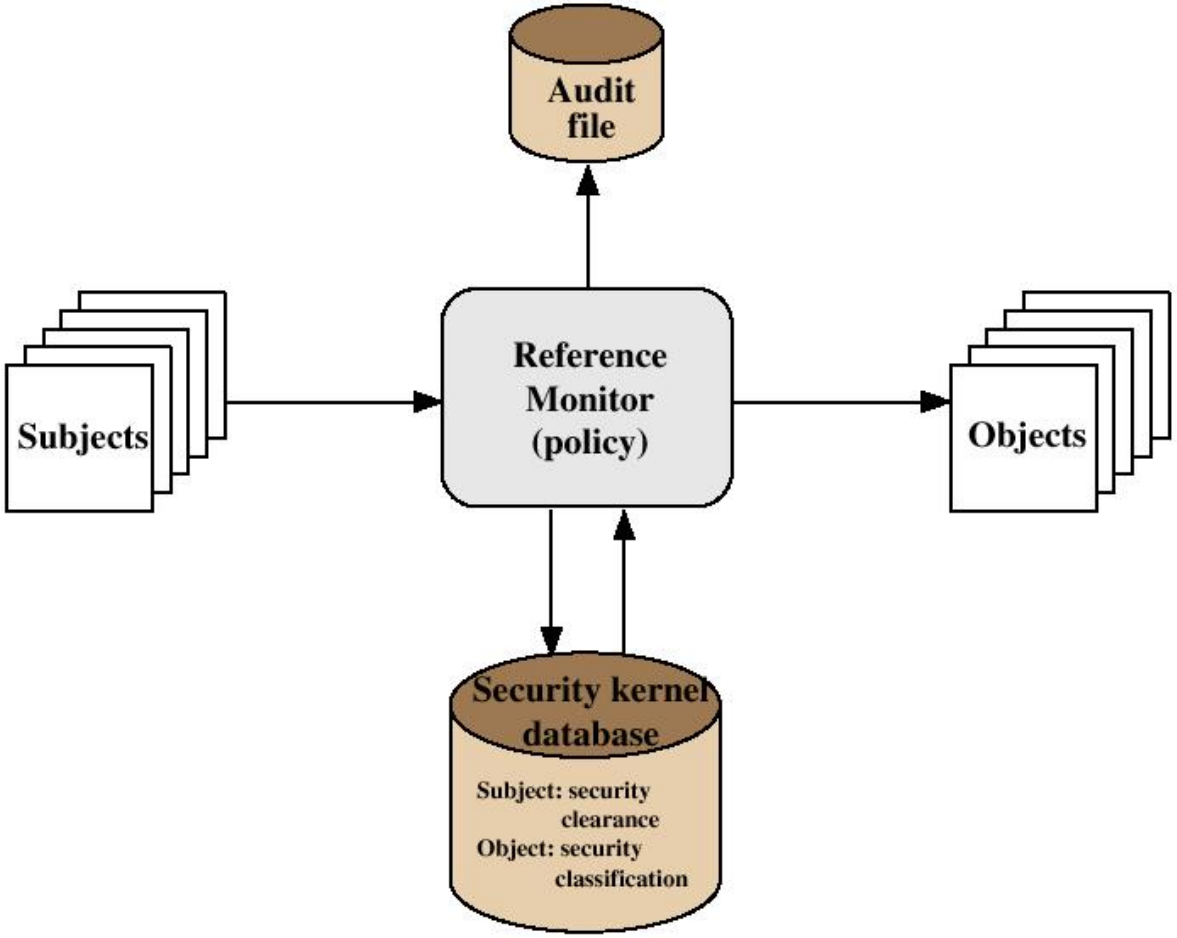
- Multilevel security
  - Definition of multiple categories or levels of data
- A multilevel secure system must enforce:
  - No read up: A subject can only read an object of less or equal security level (Simple Security Property)
  - No write down: A subject can only write into an object of greater or equal security level (*-Property)

The Concept of Trusted Systems

- Reference Monitor Concept: Multilevel security for a data processing system



The Concept of Trusted Systems



The Concept of Trusted Systems

- Reference Monitor
  - Controlling element in the hardware and operating system of a computer that regulates the access of subjects to objects on basis of security parameters
  - The monitor has access to a file (security kernel database)
  - The monitor enforces the security rules (no read up, no write down)

The Concept of Trusted Systems

- Properties of the Reference Monitor
  - Complete mediation: Security rules are enforced on every access
  - Isolation: The reference monitor and database are protected from unauthorized modification
  - Verifiability: The reference monitor's correctness must be provable (mathematically)

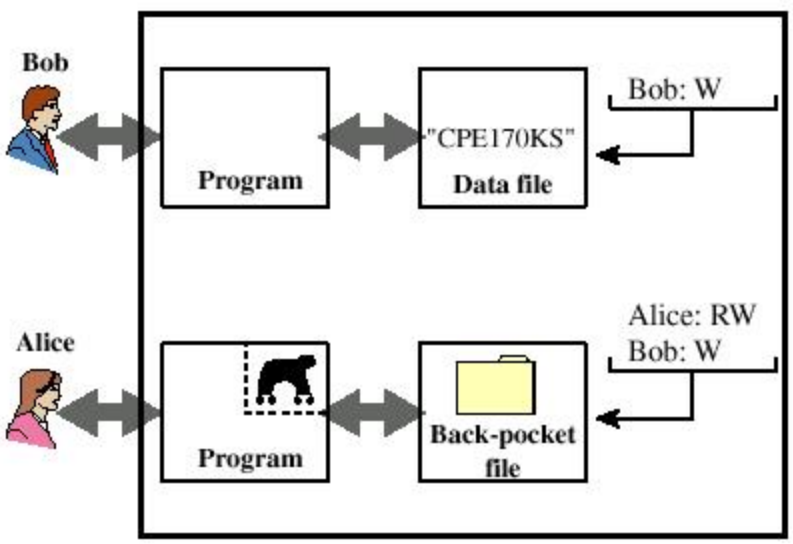
The Concept of Trusted Systems

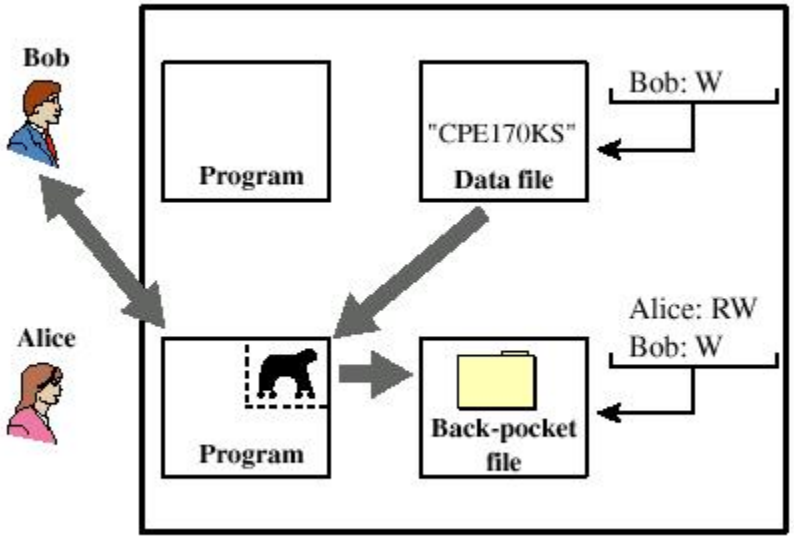
- A system that can provide such verifications (properties) is referred to as a trusted system

Trojan Horse Defense

- Secure, trusted operating systems are one way to secure against Trojan Horse attacks

Trojan Horse Defense





UNIT-IV

Pretty Good Privacy

- Philip R. Zimmerman is the creator of PGP.
- PGP provides a confidentiality and authentication service that can be used for electronic mail and file storage applications. : Pretty Good

IP Security

- Internet community considered some application specific security mechanisms
  - eg. S/MIME, PGP, Kerberos, SSL/HTTPS
- however there are security concerns that cut across protocol layers
- would like security implemented by the network for all applications

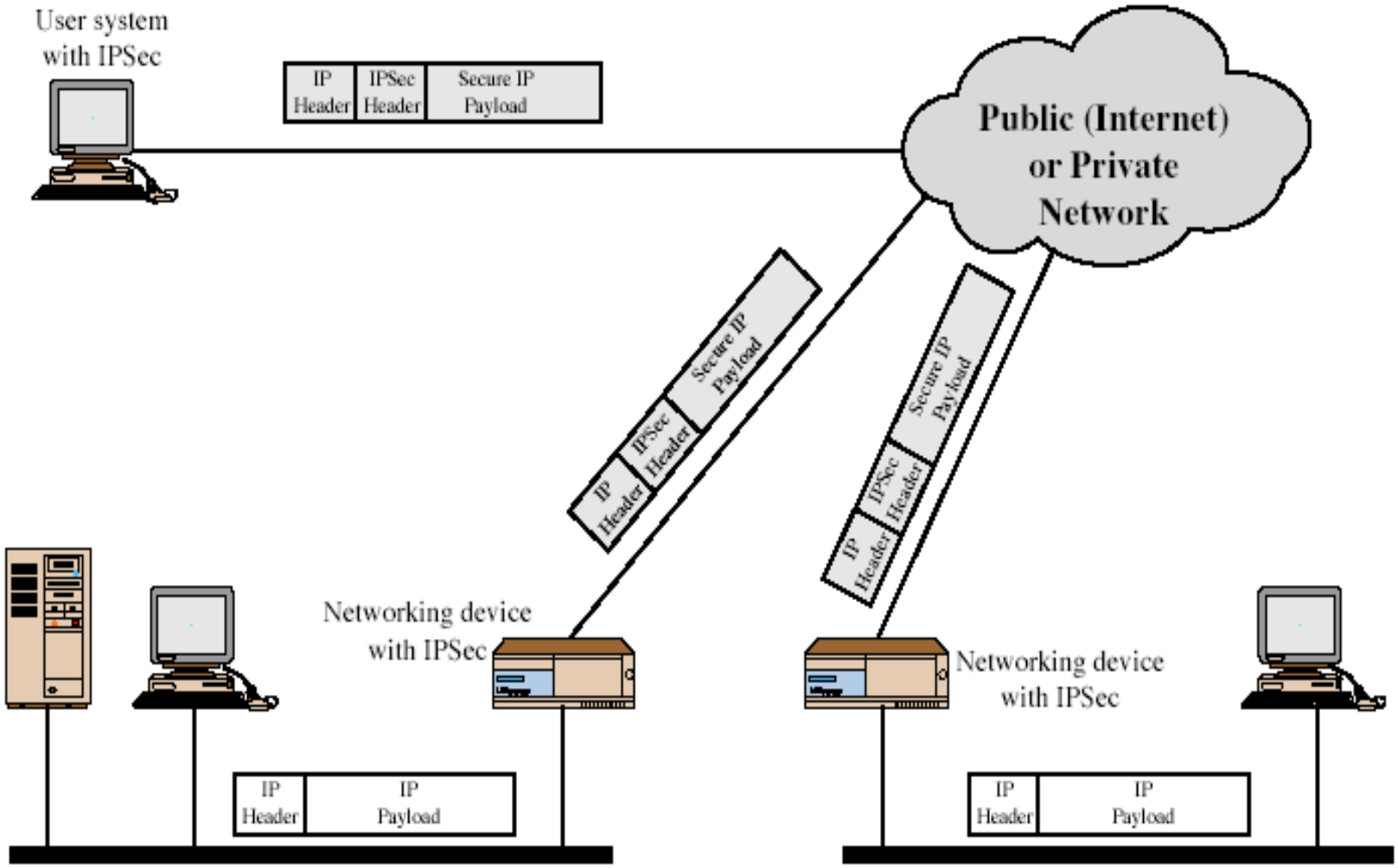
IPSec

- general IP Security mechanisms
- Encompasses 3 functional areas
  - authentication
  - confidentiality
  - key management
- applicable to use over LANs, across public & private WANs, & for the Internet

IP Security Overview

- Applications of IPSec
  - Secure branch office connectivity over the Internet
  - Secure remote access over the Internet
  - Establsihing extranet and intranet connectivity with partners
  - Enhancing electronic commerce security

IP Security Overview



Benefits of IPSec

- in a firewall/router provides strong security to all traffic crossing the perimeter
- is resistant to bypass
- is below transport layer, hence transparent to applications
- can be transparent to end users
- can provide security for individual users if desired

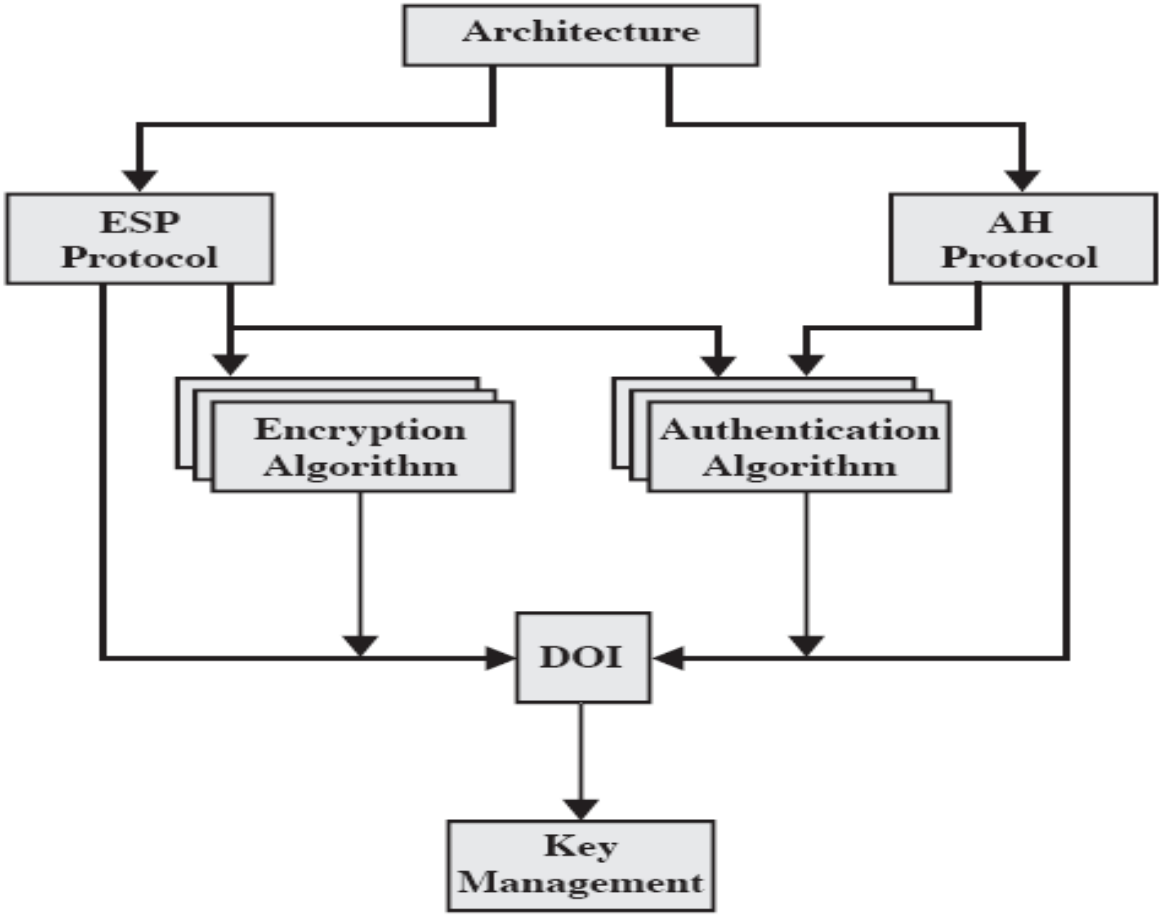
IP Security Architecture

- IPSec documents:
  - RFC 2401: An overview of security architecture
  - RFC 2402: Description of a packet authentication extension to IPv4 and IPv6
  - RFC 2406: Description of a packet encryption extension to IPv4 and IPv6
  - RFC 2408: Specification of key managament capabilities

IP Security Architecture

- specification is quite complex
- defined in numerous RFC's
  - incl. RFC 2401/2402/2406/2408
  - many others, grouped by category
- mandatory in IPv6, optional in IPv4
- have two security header extensions:
  - Authentication Header (AH)
  - Encapsulating Security Payload (ESP)

IP Security Architecture



IPSec Services

- Two protocols are used to provide security:
  - Authentication Header Protocol (AH)
  - Encapsulation Security Payload (ESP)
- Services provided are:
  - Access control
  - Connectionless integrity
  - Data origin authentication
  - Rejection of replayed packets
    - a form of partial sequence integrity
  - Confidentiality (encryption)
  - Limited traffic flow confidentiality

IPSec Services

	AH	ESP (encryption only)	ESP (encryption plus authentication)
Access control	✓	✓	✓
Connectionless integrity	✓		✓
Data origin authentication	✓		✓
Rejection of replayed packets	✓	✓	✓
Confidentiality		✓	✓
Limited traffic flow confidentiality		✓	✓

Security Associations

- a one-way relationship between sender & receiver that affords security services for traffic flow
- defined by 3 parameters:
  - Security Parameters Index (SPI)
    - a bit string
  - IP Destination Address
    - only unicast allowed
    - could be end user, firewall, router
  - Security Protocol Identifier
    - indicates if SA is AH or ES

  The security association is uniquely identified by the destination address in the IP V4 OR IP V6 header

SA Parameters

- In each IPSec implementation, there is a nominal Security Association Database that defines the parameters associated with each SA.
- A security association is normally defined by the following parameters:
- **Sequence Number Counter**
- **Sequence Counter Overflow**
- **Anti-Replay Window**
- **AH Information**
- **ESP Information**
- **Lifetime of This Security Association**
- **IPSec Protocol Mode**
- **Path MTU**

Transport and Tunnel Modes

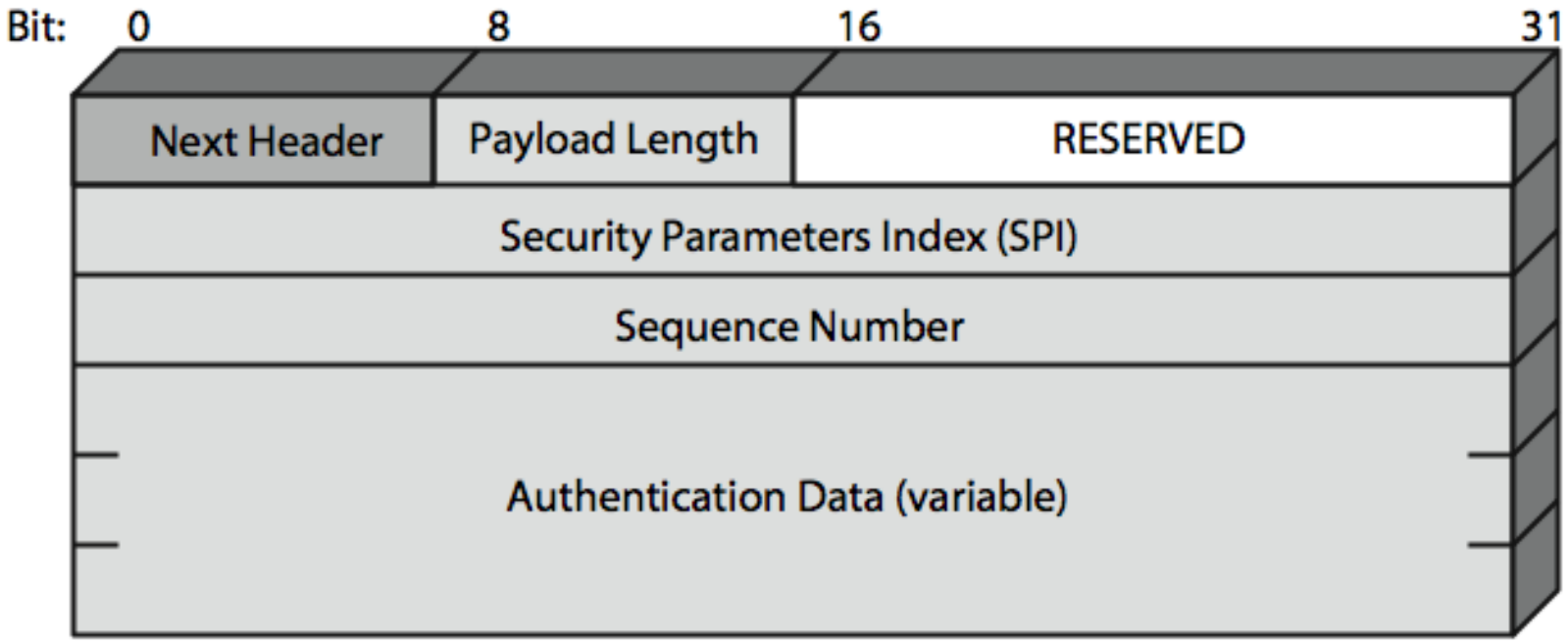
Both AH and ESP support two modes of use: transport and tunnel mode

	Transport Mode SA	Tunnel Mode SA
AH	**Authenticates** IP payload and selected portions of IP header and IPv6 extension headers	**Authenticates** entire inner IP packet plus selected portions of outer IP header
ESP	**Encrypts** IP payload and any IPv6 extesion header	**Encrypts** inner IP packet
ESP with authentication	**Encrypts** IP payload and any IPv6 extesion header. **Authenticates** IP payload but no IP header	**Encrypts** inner IP packet. **Authenticates** inner IP packet.

Authentication Header (AH)

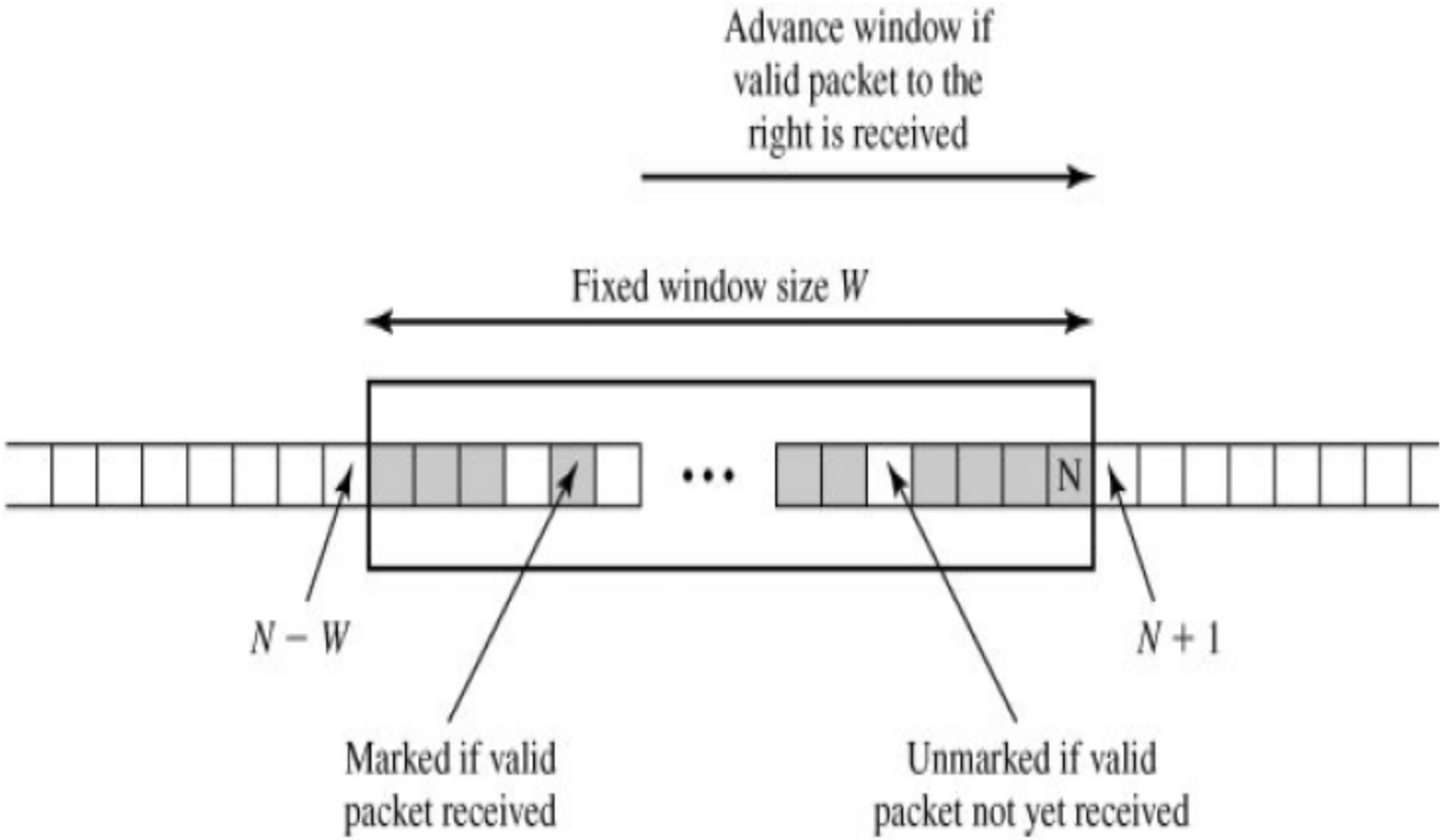
- provides support for data integrity & authentication of IP packets
  - end system/router can authenticate user/application
  - prevents address spoofing attacks
- based on use of a MAC
- parties must share a secret key

Authentication Header (AH)



Anti-Replay Service

- Anti-replay service is designed to overcome the problems faced due to replay attacks in which an intruder intervenes the packet being transferred.
- The Sequence Number field is designed to thwart such attacks.
- When a new SA is established, the sender initializes a sequence number counter to 0.
- Each time that a packet is sent on this SA, the sender increments the counter and places the value in the Sequence Number field
- The right edge of the window represents the highest sequence number, N, so far received for a valid packet



Antireplay Mechanism

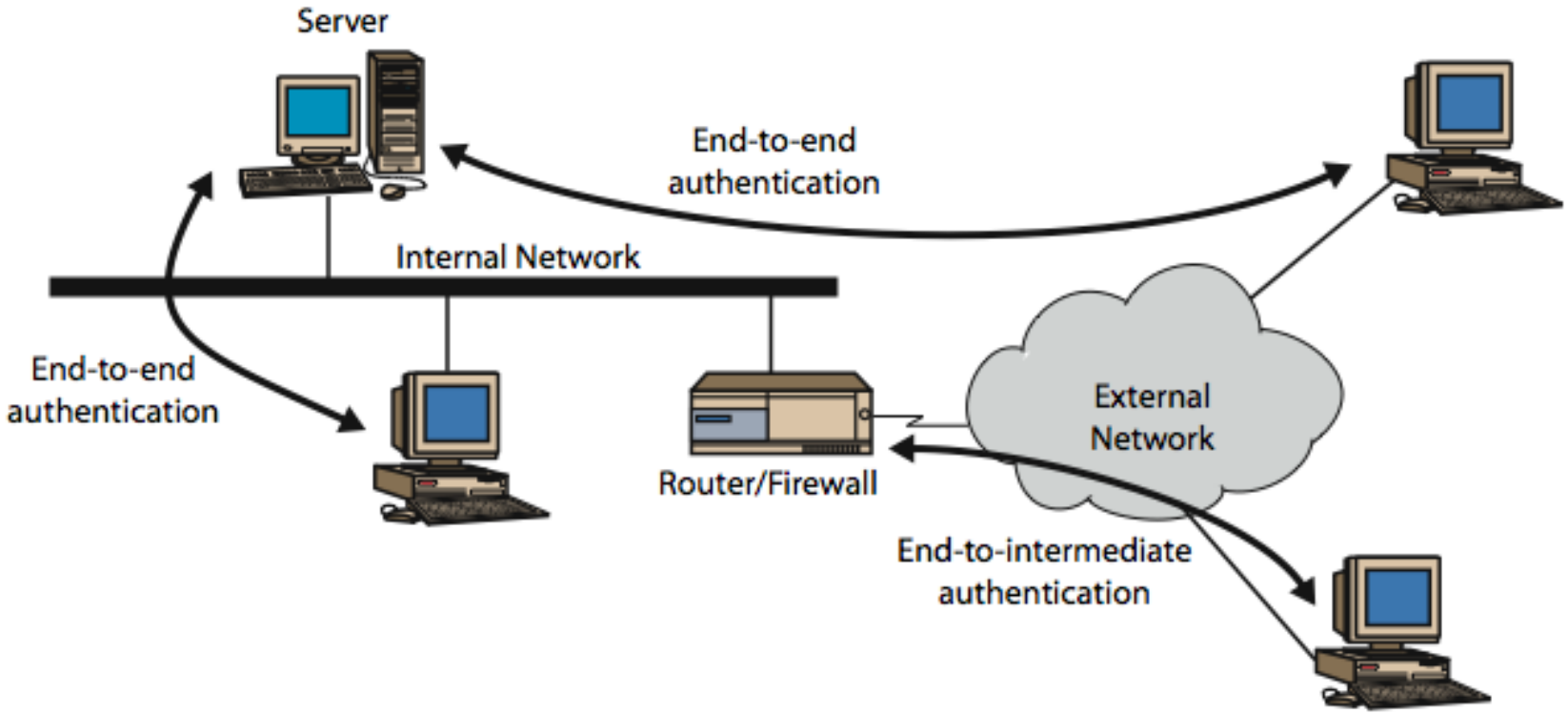
Anti-Replay Service

- 1. If the received packet falls within the window and is new, the MAC is checked. If the packet is authenticated, the corresponding slot in the window is marked.

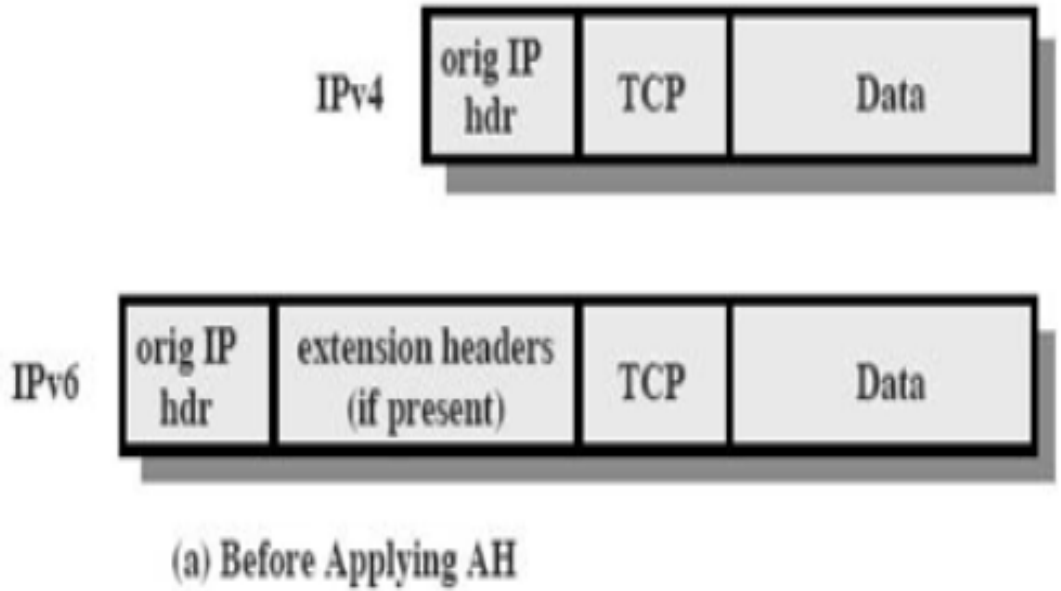
- 2. If the received packet is to the right of the window and is new, the MAC is checked. If the packet is authenticated, the window is advanced so that this sequence number is the right edge of the window, and the corresponding slot in the window is marked

- 3. If the received packet is to the left of the window, or if authentication fails, the packet is discarded; this is an auditable event.

Transport and Tunnel Modes

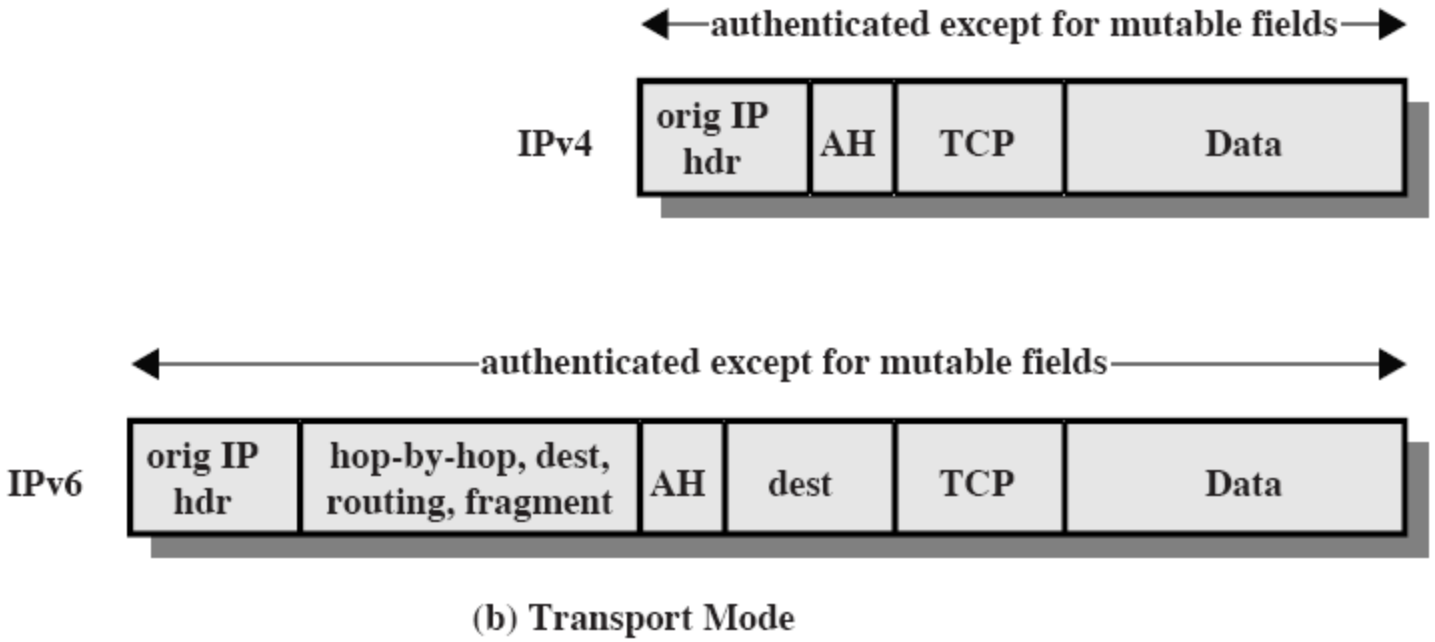


TRANSPORT MODE AH



(a) Before Applying AH

TRANSPORT MODE AH



(b) Transport Mode

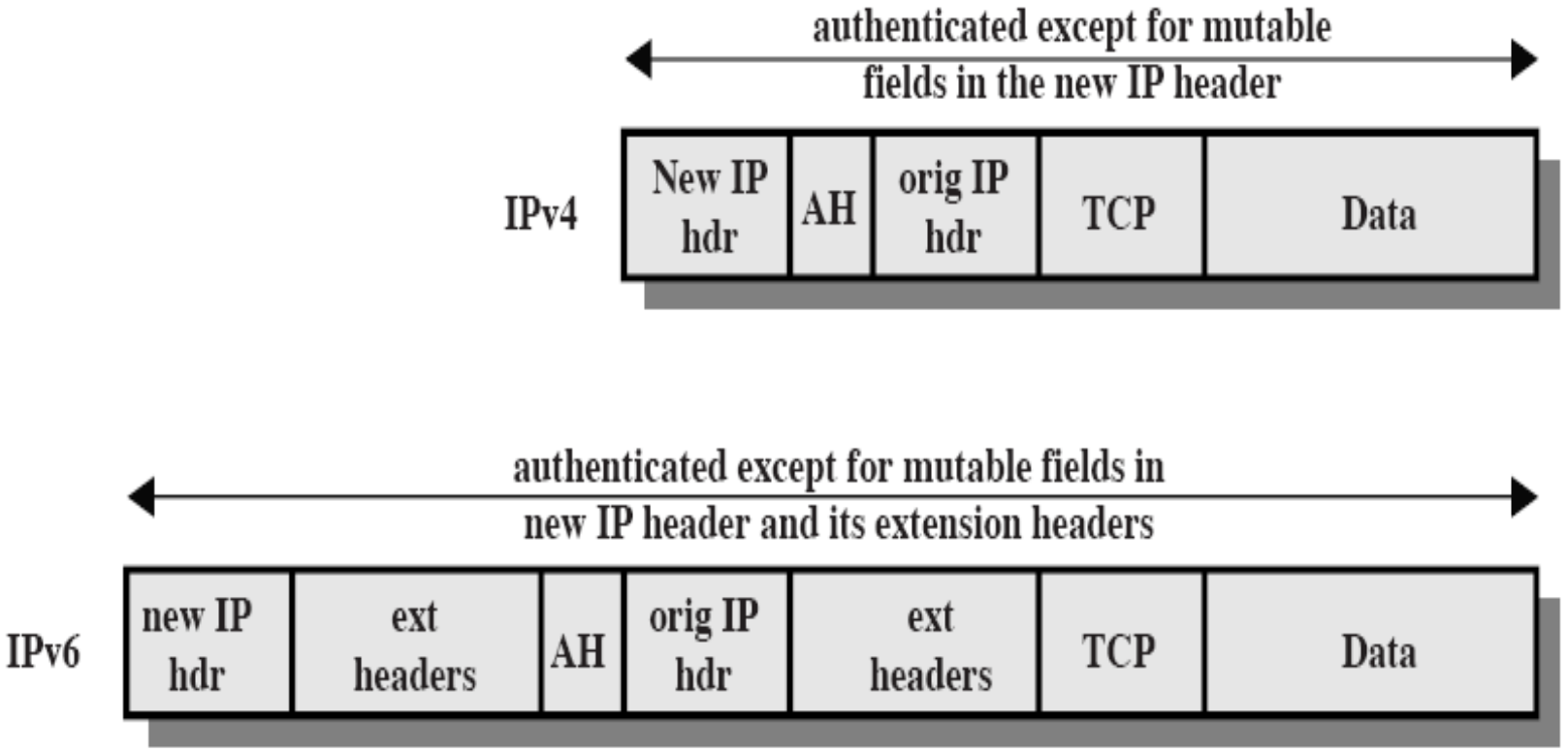
TRANSPORT MODE AH

- For transport mode AH using IPv4, the AH is inserted after the original IP header and before the IP payload (e.g., a TCP segment)
- Authentication covers the entire packet, excluding mutable fields in the IPv4 header that are set to zero for MAC calculation.

- In the context of IPv6, AH is viewed as an end-to-end payload; that is, it is not examined or processed by intermediate routers

- Again, authentication covers the entire packet, excluding mutable fields that are set to zero for MAC calculation

Tunnel Mode AH



Tunnel Mode AH

- For tunnel mode AH, the entire original IP packet is authenticated, and the AH is inserted between the original IP header and a new outer IP header.

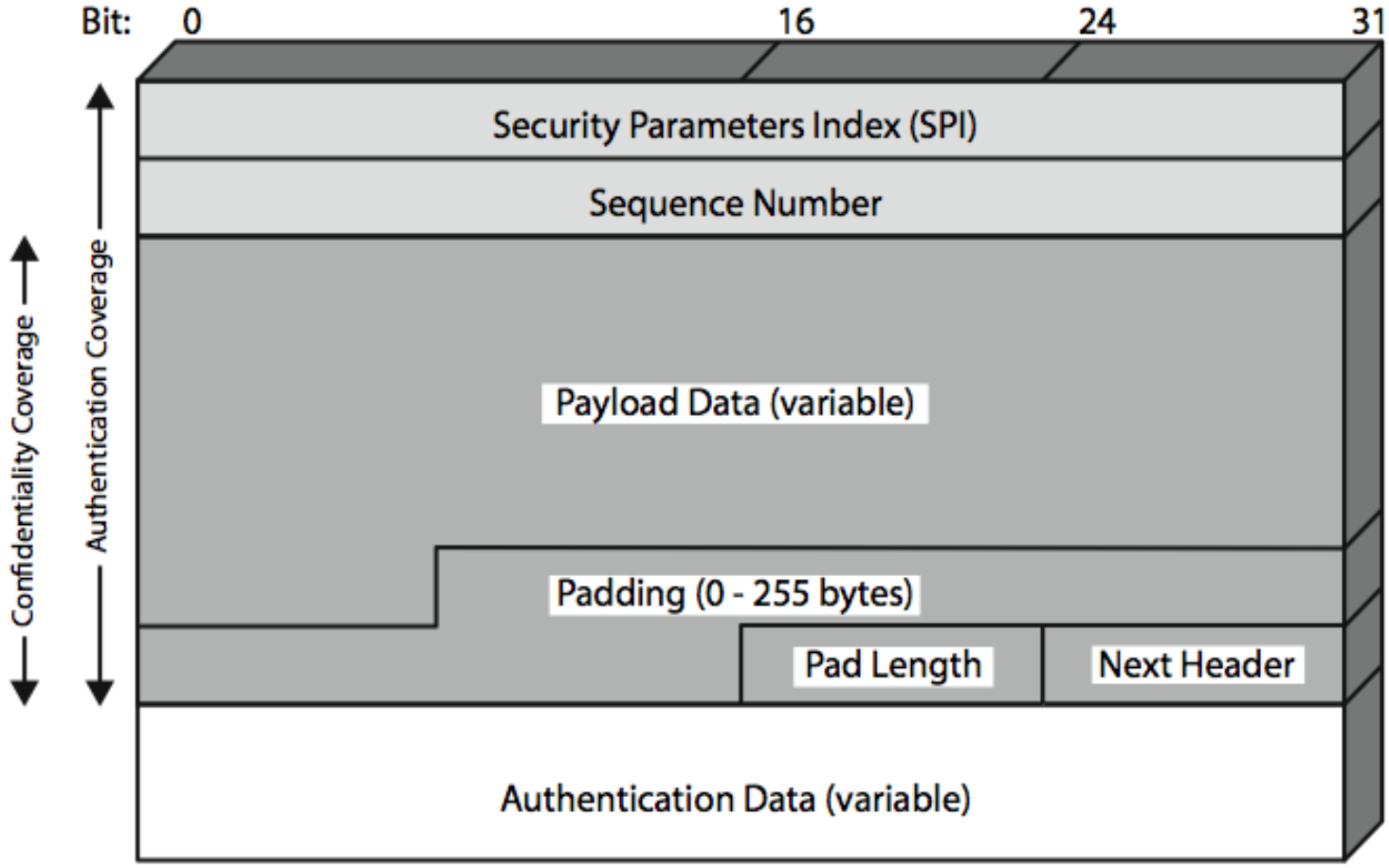
- The inner IP header carries the ultimate source and destination addresses, while an outer IP header may contain different IP addresses

- With tunnel mode, the entire inner IP packet, including the entire inner IP header is protected by AH

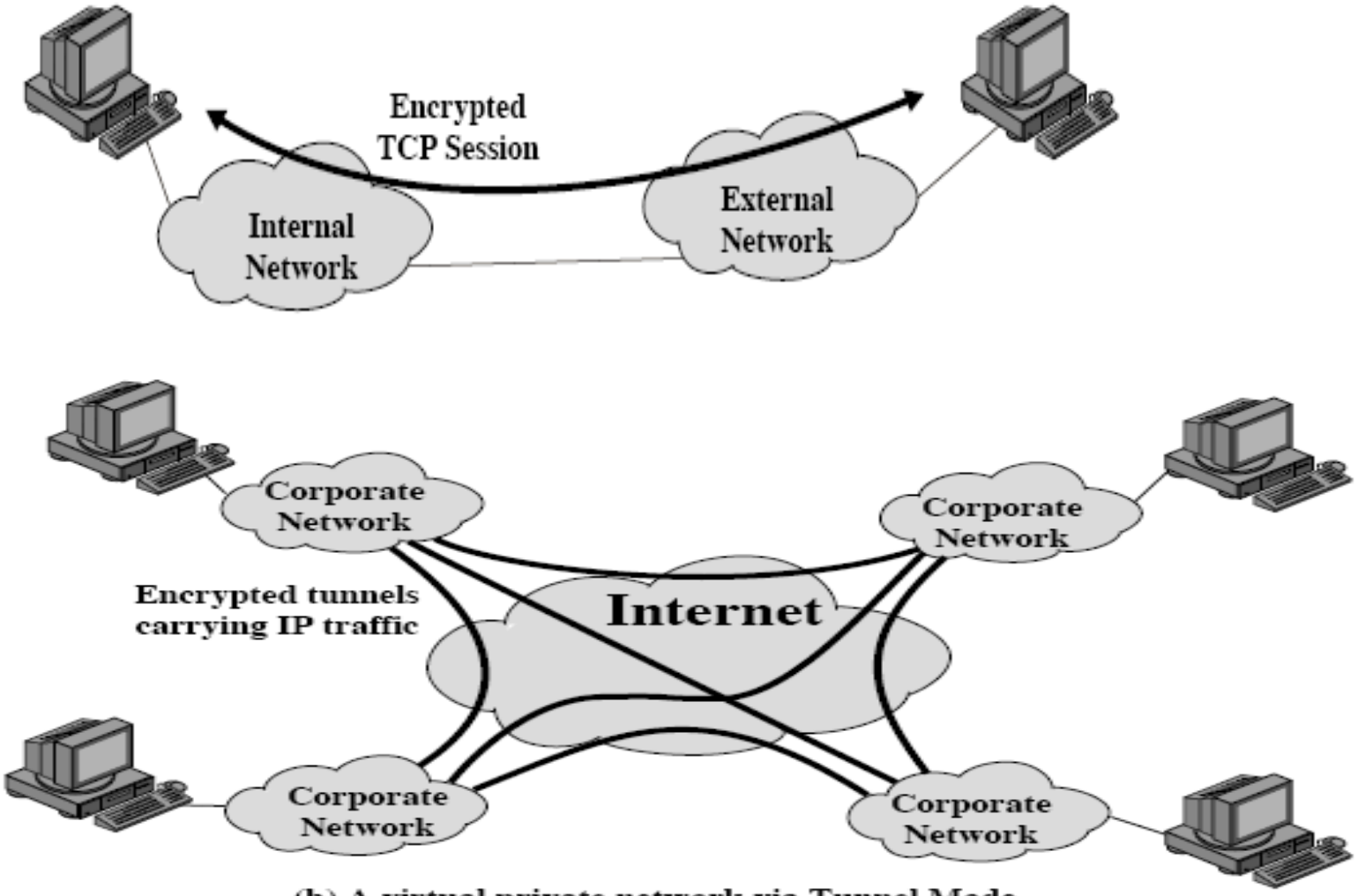
Encapsulating Security Payload

- provides message content confidentiality & limited traffic flow confidentiality
- can optionally provide the same authentication services as AH
- supports range of ciphers, modes, padding
  - incl. DES, Triple-DES, RC5, IDEA, CAST etc
  - CBC & other modes
  - padding needed to fill block size, fields, for traffic flow

Encapsulating Security Payload

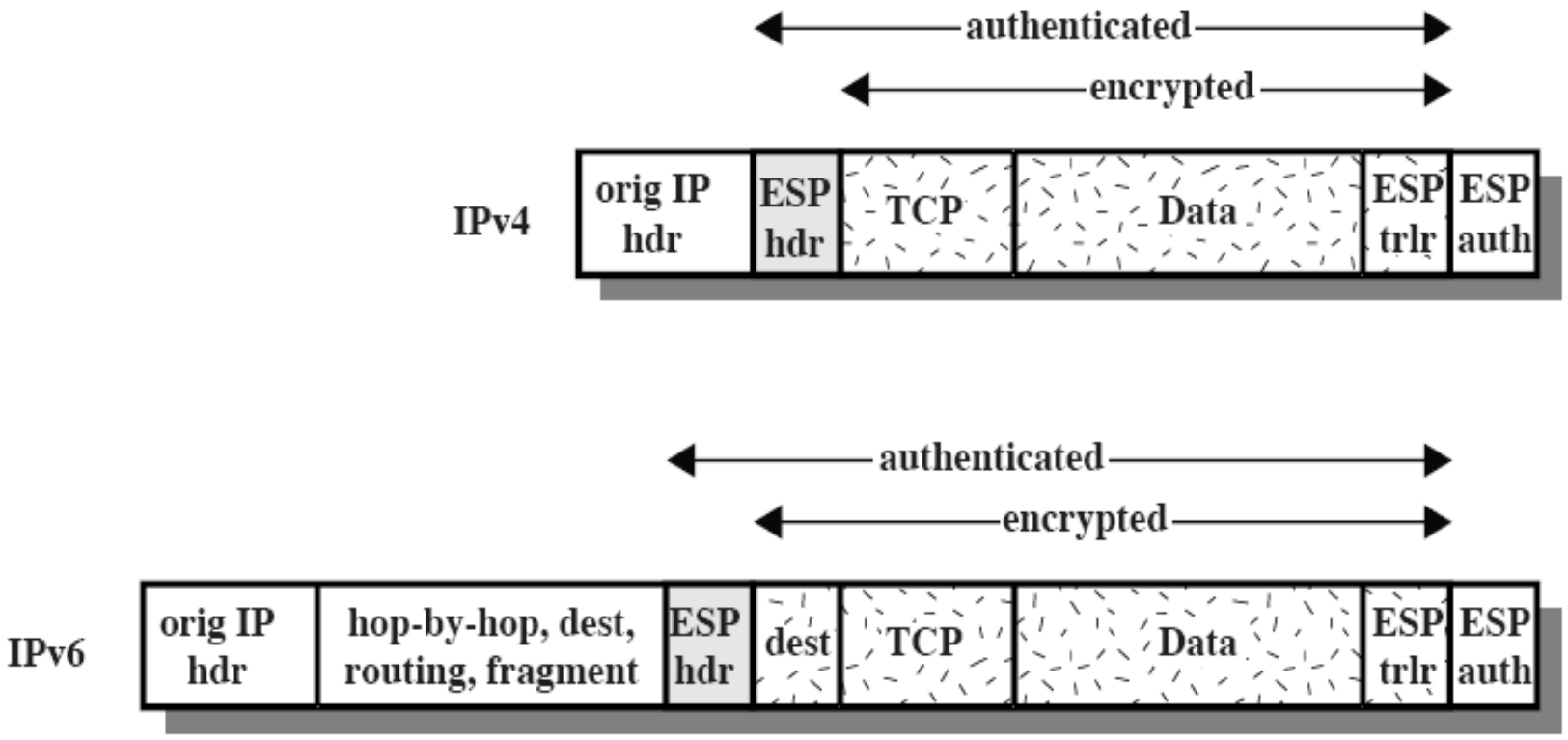


Transport and Tunnel Mode ESP

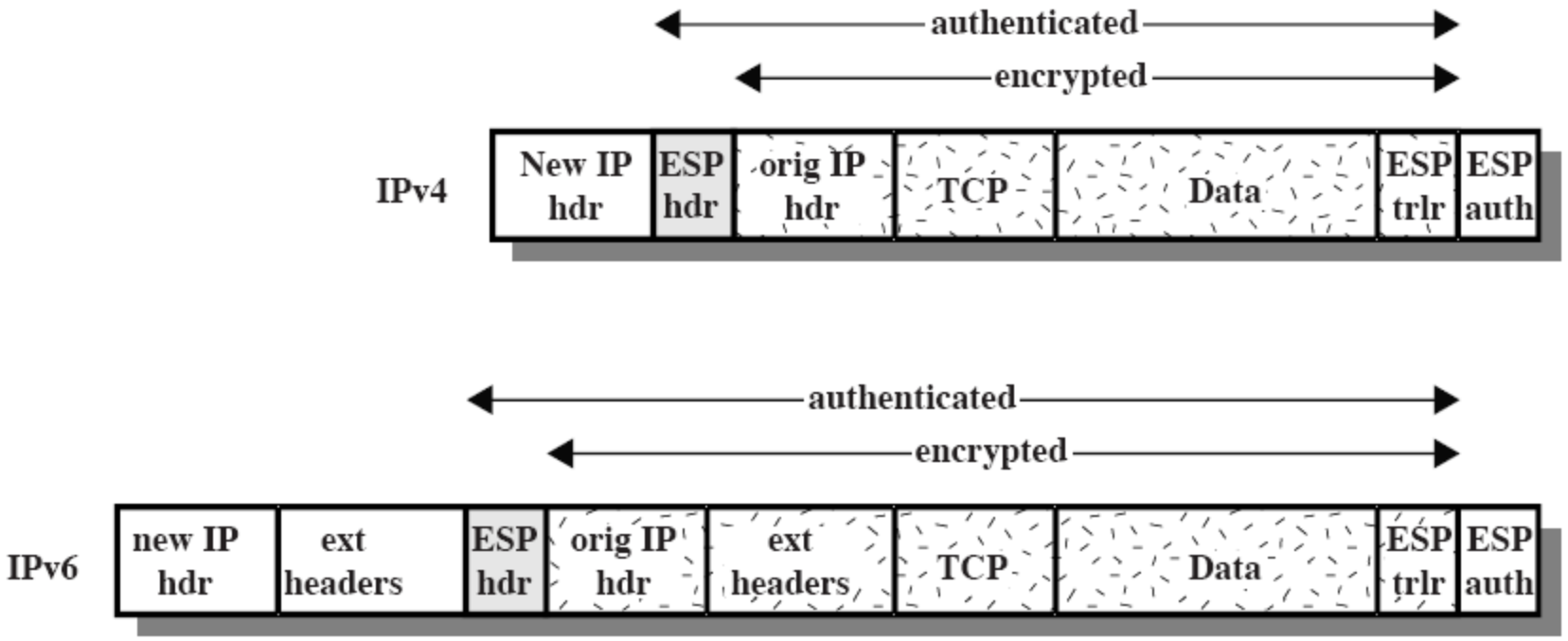


(b) A virtual private network via Tunnel Mode

Transport Mode ESP



Tunnel Mode ESP

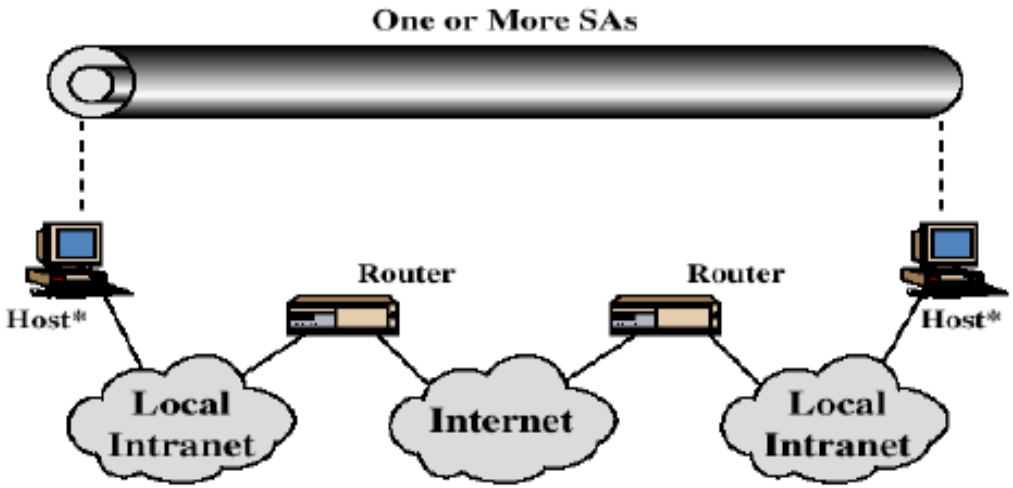


Tunnel Mode ESP

- Tunnel mode ESP is used to encrypt an entire IP packet.
- For this mode, the ESP header is prefixed to the packet and then the packet plus the ESP trailer is encrypted.
- This method can be used to counter traffic analysis

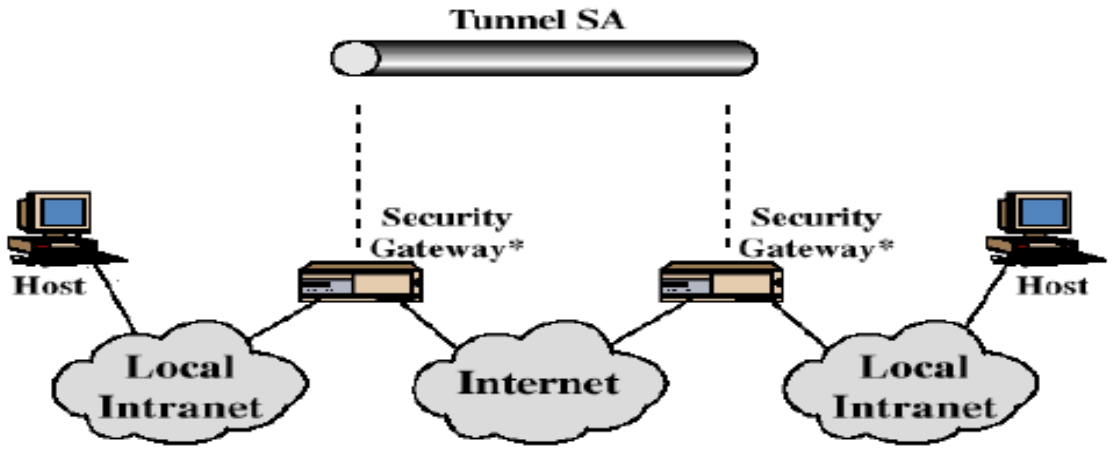
Combining Security Associations

case:-1



(a) Case 1

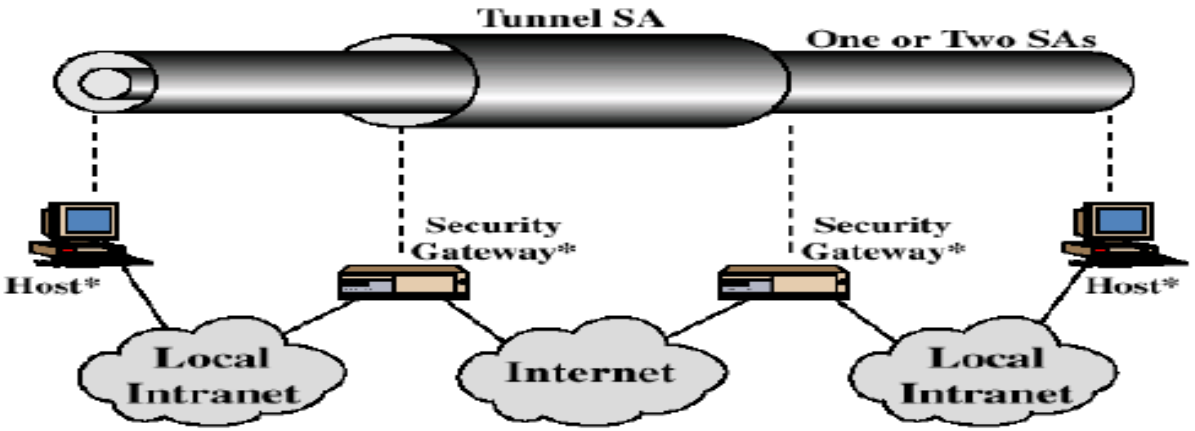
Case:-2



(b) Case 2

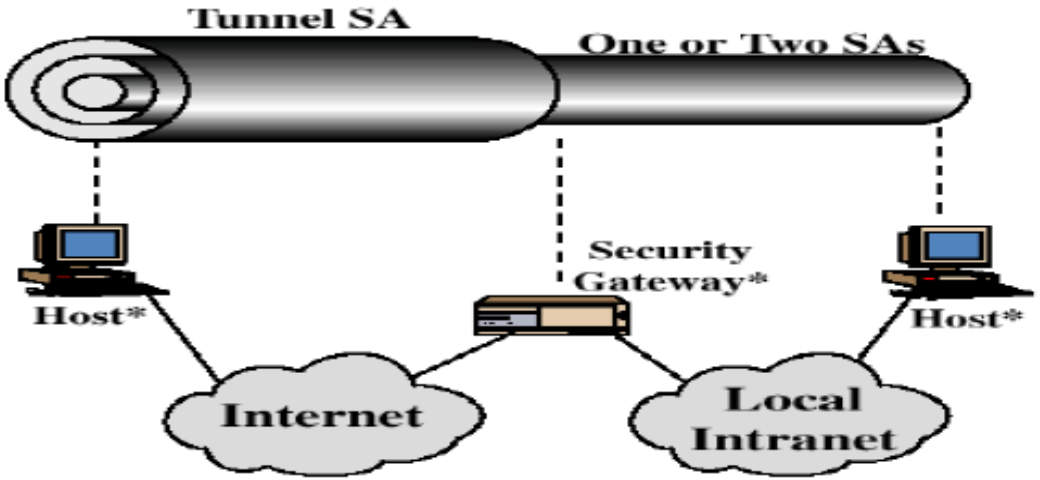
Combining Security Associations

Case-3:-



(c) Case 3

Case:-4



(d) Case 4

Key Management

- handles key generation & distribution
- typically need 2 pairs of keys
  - 2 per direction for AH & ESP
- manual key management
  - Sys admin manually configures every system
- automated key management
  - automated system for on demand creation of keys for SA's in large systems
  - has Oakley & ISAKMP elements

Oakley Key Determination Protocol

- Oakley is a refinement of the Diffie-Hellman key exchange algorithm.
- The Diffie-Hellman algorithm has two attractive features:
- Secret keys are created only when needed.
- The exchange requires no pre-existing infrastructure other than an agreement on the global parameters.

- Oakley is designed to retain the advantages of Diffie-Hellman while countering its weaknesses.

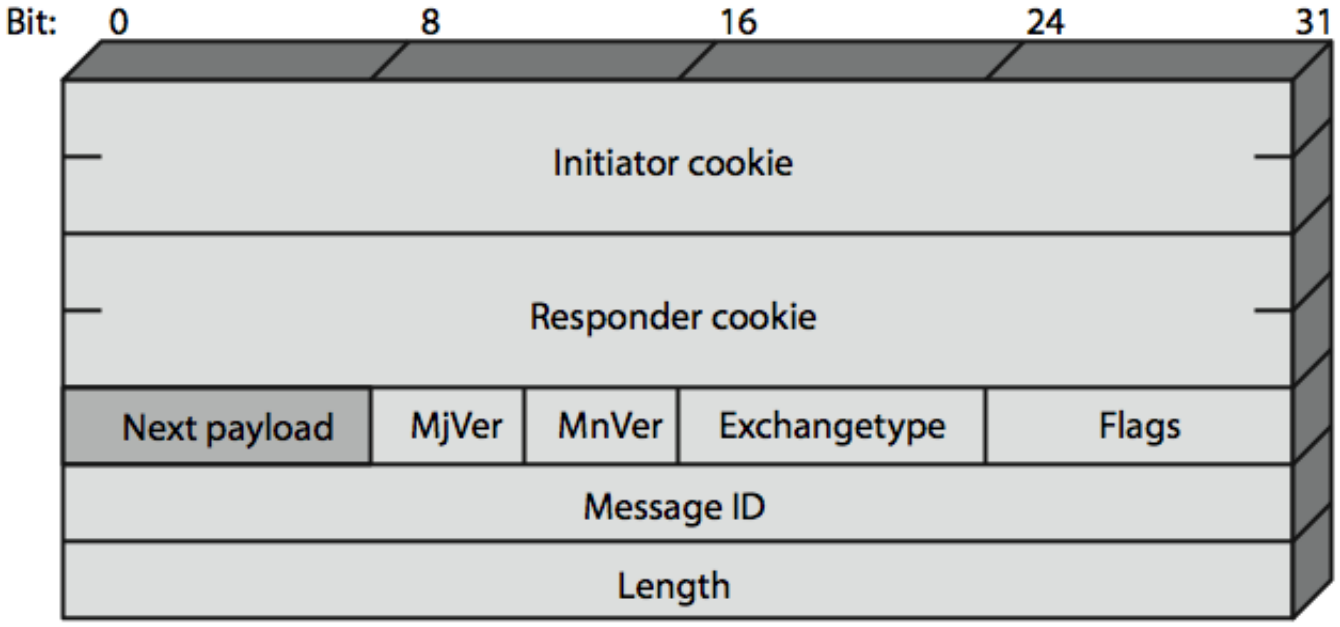
Features of Oakley

- The Oakley algorithm is characterized by five important features:
- 1. It employs a mechanism known as cookies to thwart clogging attacks.
- 2. It enables the two parties to negotiate a group; this, in essence, specifies the global parameters of the Diffie-Hellman key exchange.
- 3. It uses nonces to ensure against replay attacks.
- 4. It enables the exchange of Diffie-Hellman public key values.
- 5. It authenticates the Diffie-Hellman exchange to thwart man-in-the-middle attacks.

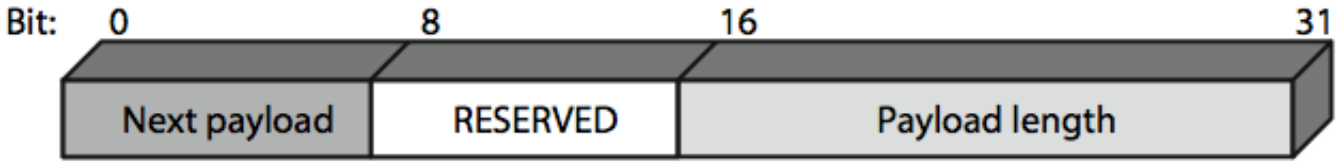
ISAKMP

- Internet Security Association and Key Management Protocol
- provides framework for key management
- defines procedures and packet formats to establish, negotiate, modify, & delete SAs
- independent of key exchange protocol, encryption alg, & authentication method

ISAKMP



(a) ISAKMP Header



(b) Generic Payload Header

UNIT-V

Web security

- Web security considerations
- The web is increasingly serving as a highly visible outlet for corporate and as platform for business transactions
- Web browsers are very easy to use
- Web servers are relatively easy to configure and manage
- The underlying software is extraordinarily complex
- Web server can be exploited as a launching pad into the corporations entire computer complex
- Casual users are not necessarily aware of the security risks that exist.

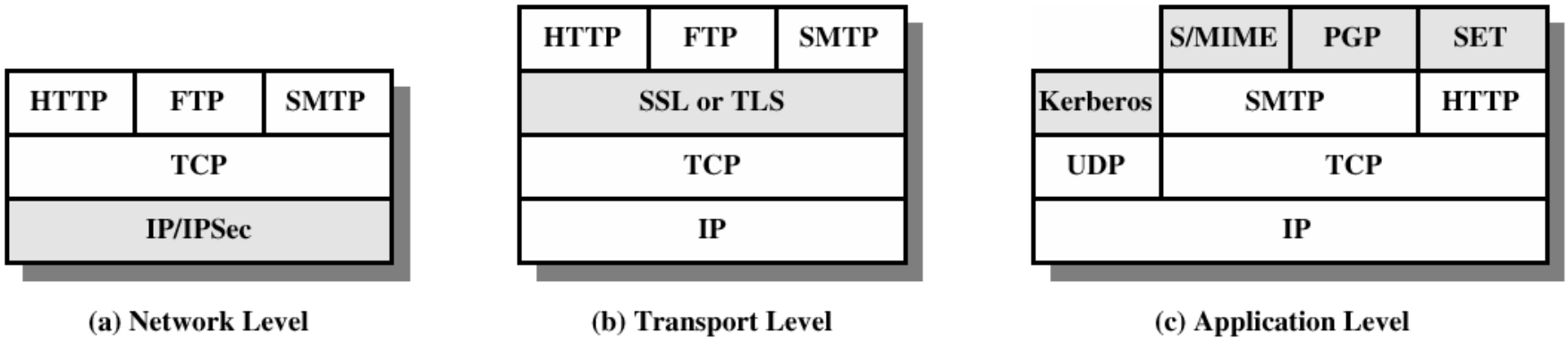
Web security threats

- One way to group these threats is in terms of passive and active attacks
- Passive attacks include eaves dropping on network traffic between browser and server
- Active attacks include impersonating another user, altering messages in transit between client and server
- Another way to classify web security threats is in terms of location of the threat
- Web server, web browser and network traffic between browser and server

Web traffic security approaches

- One way to provide web security is to use IP security
- Ip security includes filtering capability so that only selected traffic need incur the overhead of IP sec processing(fig a)
- Another solution is to implement security just above Tcp
- The example of this approach is the SECURE SOCKET LAYER(SSL OR TLS (transport layer security)
- SSL can be embedded in specific packages . For example Netscape and Microsoft explorer browsers come equipped with SSL
- Most web servers have implemented the protocol

Security facilities in the TCP/IP protocol stack



(a) Network Level

(b) Transport Level

(c) Application Level

SSL and TLS

- SSL was originated by Netscape
- TLS working group was formed within IETF
- First version of TLS can be viewed as an SSLv3.1

SSL Architecture

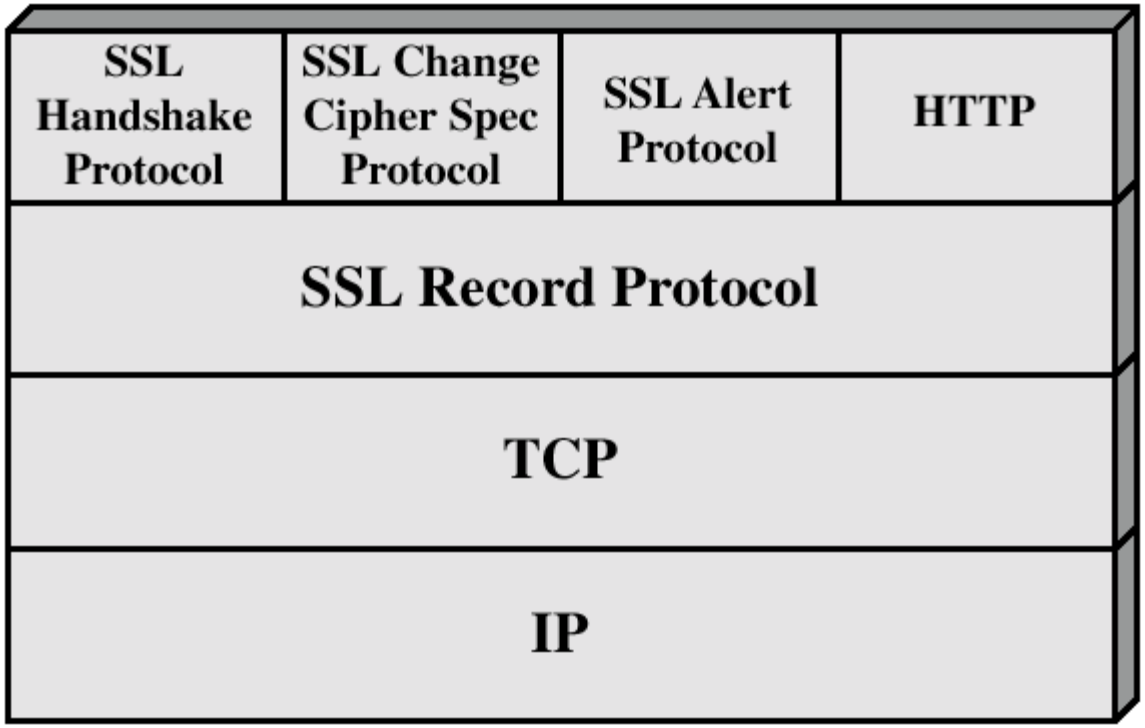
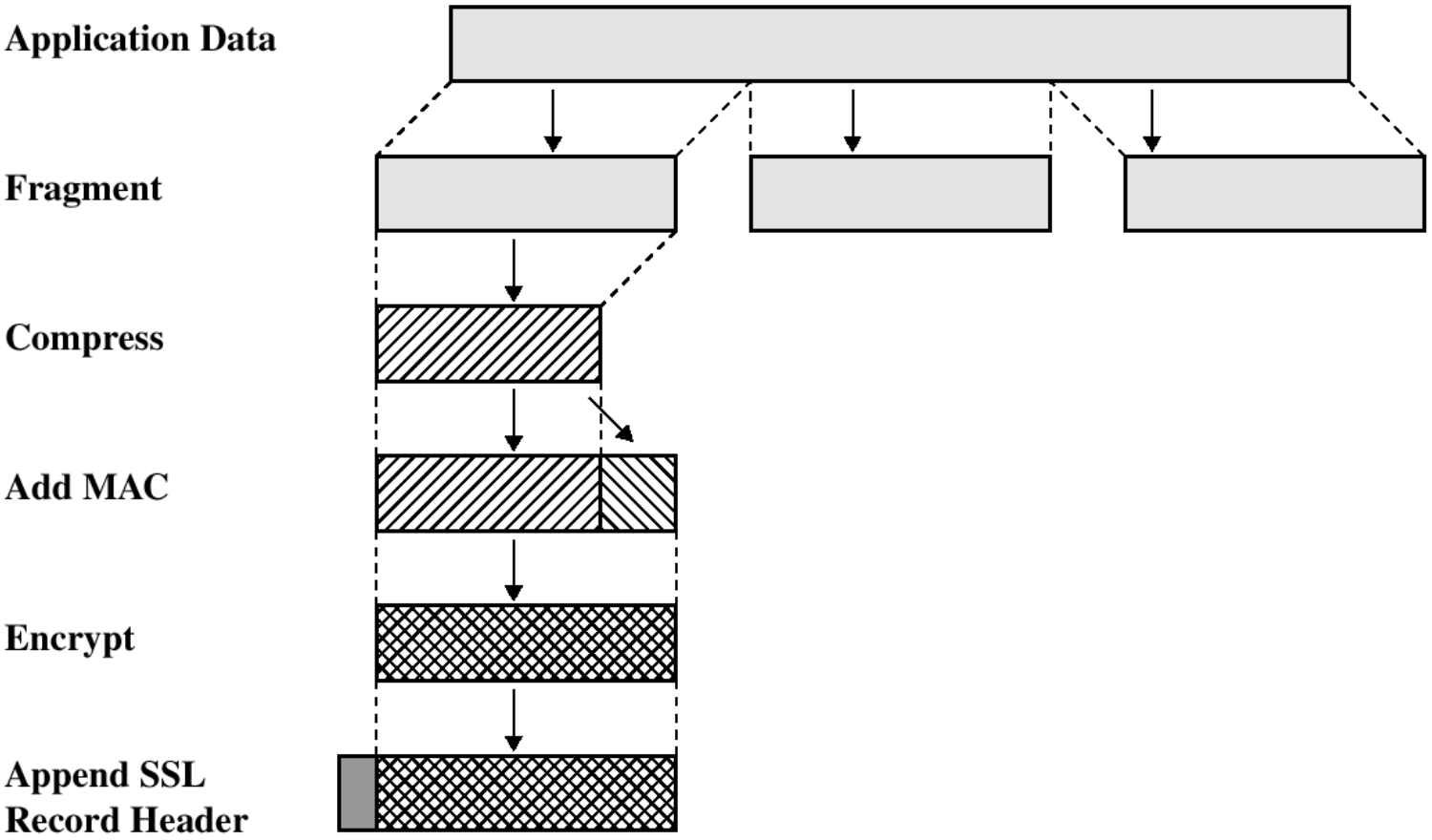
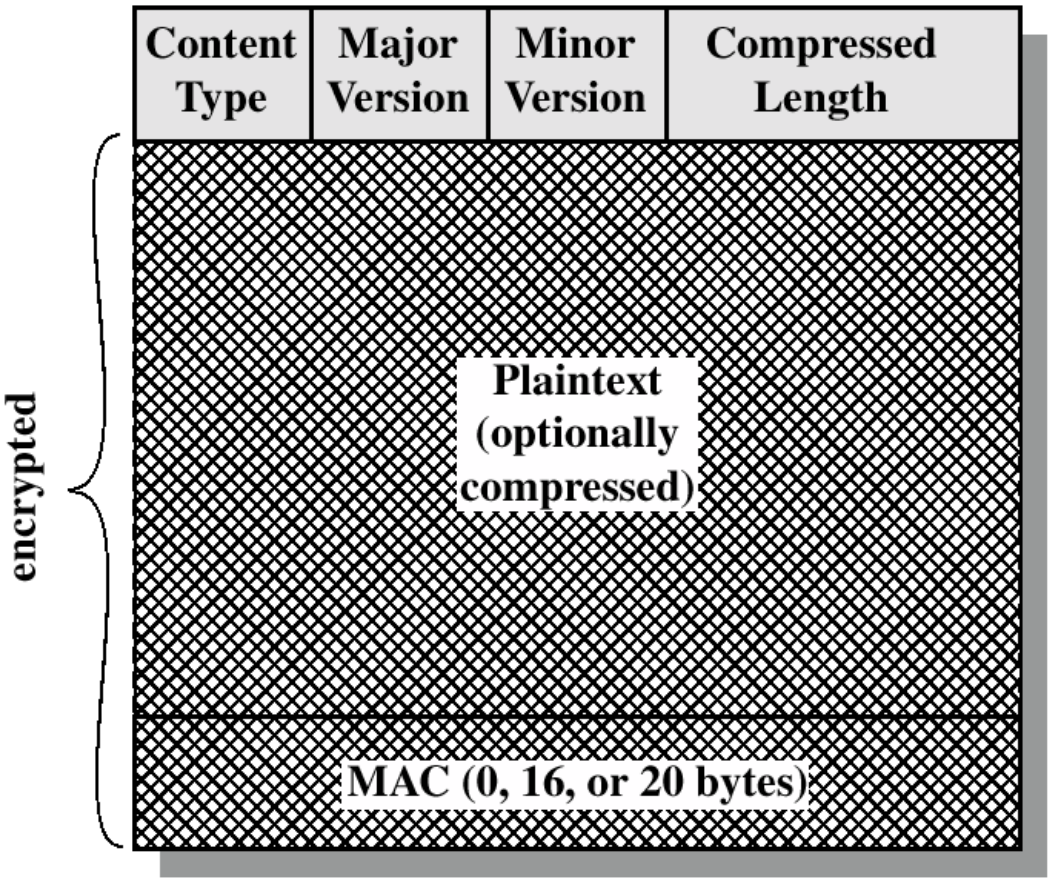


Figure 7.2 SSL Protocol Stack

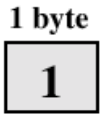
SSL Record Protocol Operation



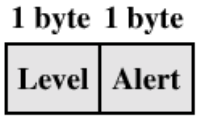
SSL Record Format



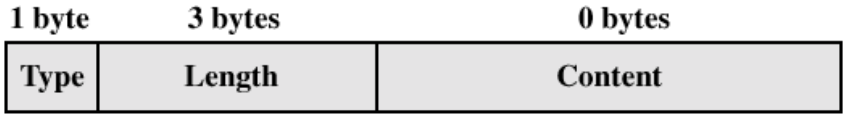
SSL Record Protocol Payload



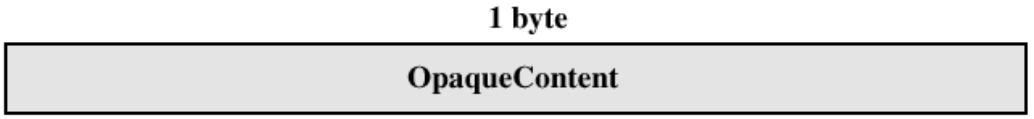
(a) Change Cipher Spec Protocol



(b) Alert Protocol



(c) Handshake Protocol



(d) Other Upper-Layer Protocol (e.g., HTTP)

Handshake Protocol

- The most complex part of SSL.
- Allows the server and client to authenticate each other.
- Negotiate encryption, MAC algorithm and cryptographic keys.
- Used before any application data are transmitted.

Transport Layer Security

- The same record format as the SSL record format.
- Defined in RFC 2246.
- Similar to SSLv3.
- Differences in the:
  - version number
  - message authentication code
  - pseudorandom function
  - alert codes
  - cipher suites
  - client certificate types
  - certificate_verify and finished message
  - cryptographic computations
  - padding

Secure Electronic Transactions

- An open encryption and security specification.
- Protect credit card transaction on the Internet.
- Companies involved:
  - MasterCard, Visa, IBM, Microsoft, Netscape, RSA, Terisa and Verisign
- Not a payment system.
- Set of security protocols and formats.

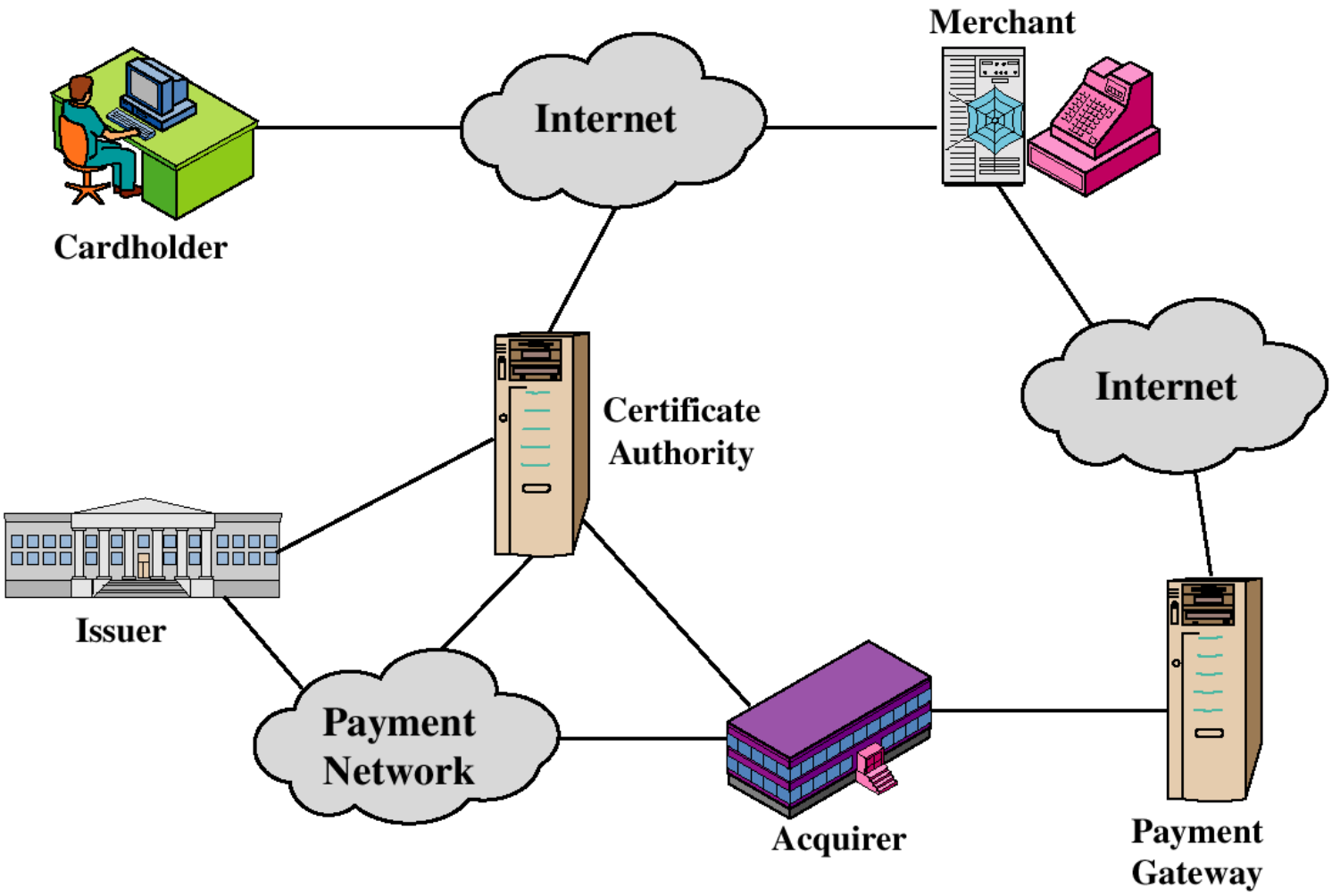
SET Services

- Provides a secure communication channel in a transaction.
- Provides tust by the use of X.509v3 digital certificates.
- Ensures privacy.

SET Overview

- Key Features of SET:
  - Confidentiality of information
  - Integrity of data
  - Cardholder account authentication
  - Merchant authentication

SET Participants

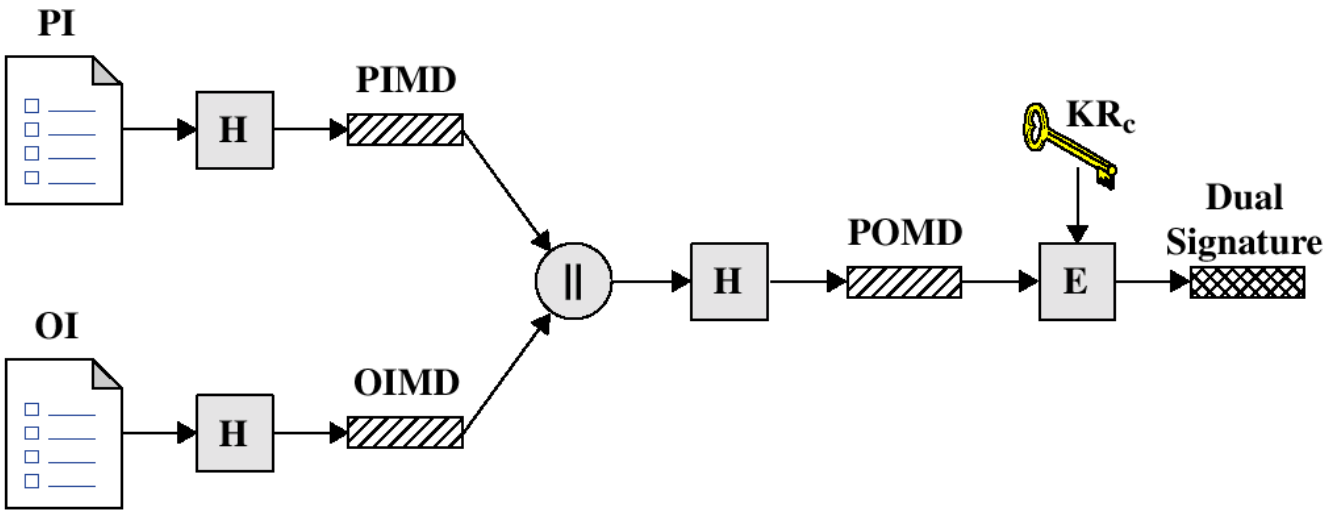


Sequence of events for transactions

1. The customer opens an account.
2. The customer receives a certificate.
3. Merchants have their own certificates.
4. The customer places an order.
5. The merchant is verified.
6. The order and payment are sent.
7. The merchant request payment authorization.
8. The merchant confirm the order.
9. The merchant provides the goods or service.
10. The merchant requests payments.

Dual Signature

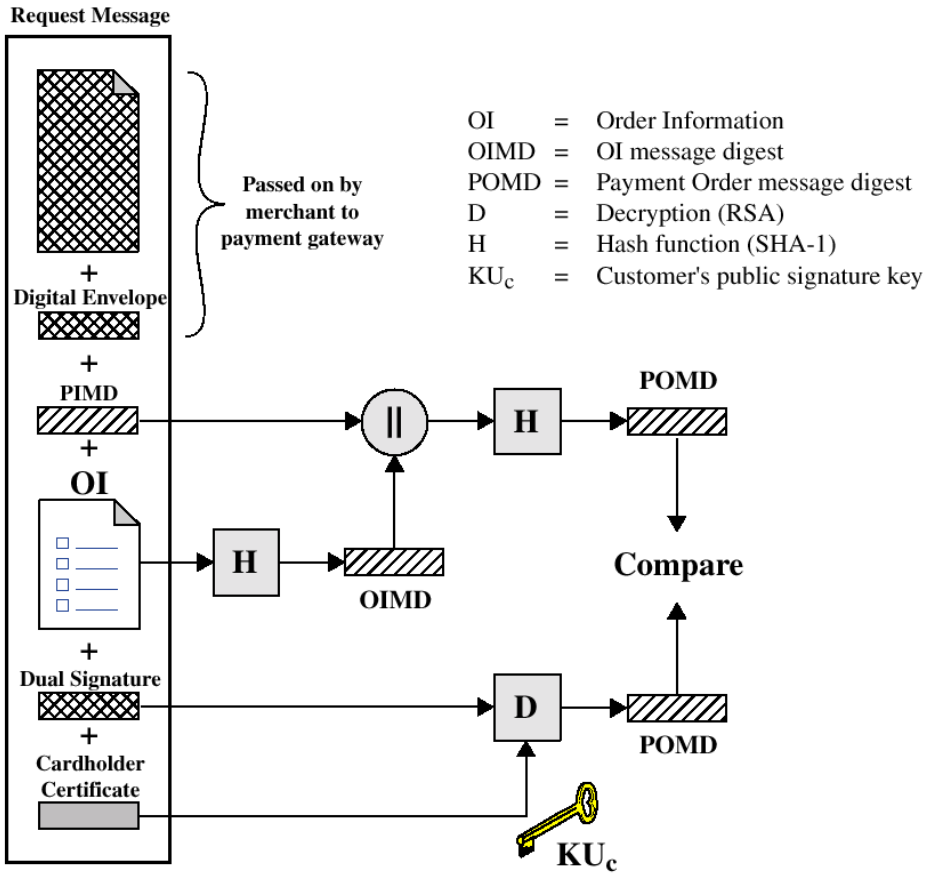
$$DS = E_{KR_c}[H(H(PI) \| H(OI))]$$



PI = Payment Information
OI = Order Information
H = Hash function (SHA-1)
|| = Concatenation

PIMD = PI message digest
OIMD = OI message digest
POMD = Payment Order message digest
E = Encryption (RSA)
KR_c = Customer's private signature key

Payment processing



Merchant Verifies Customer Purchase Request

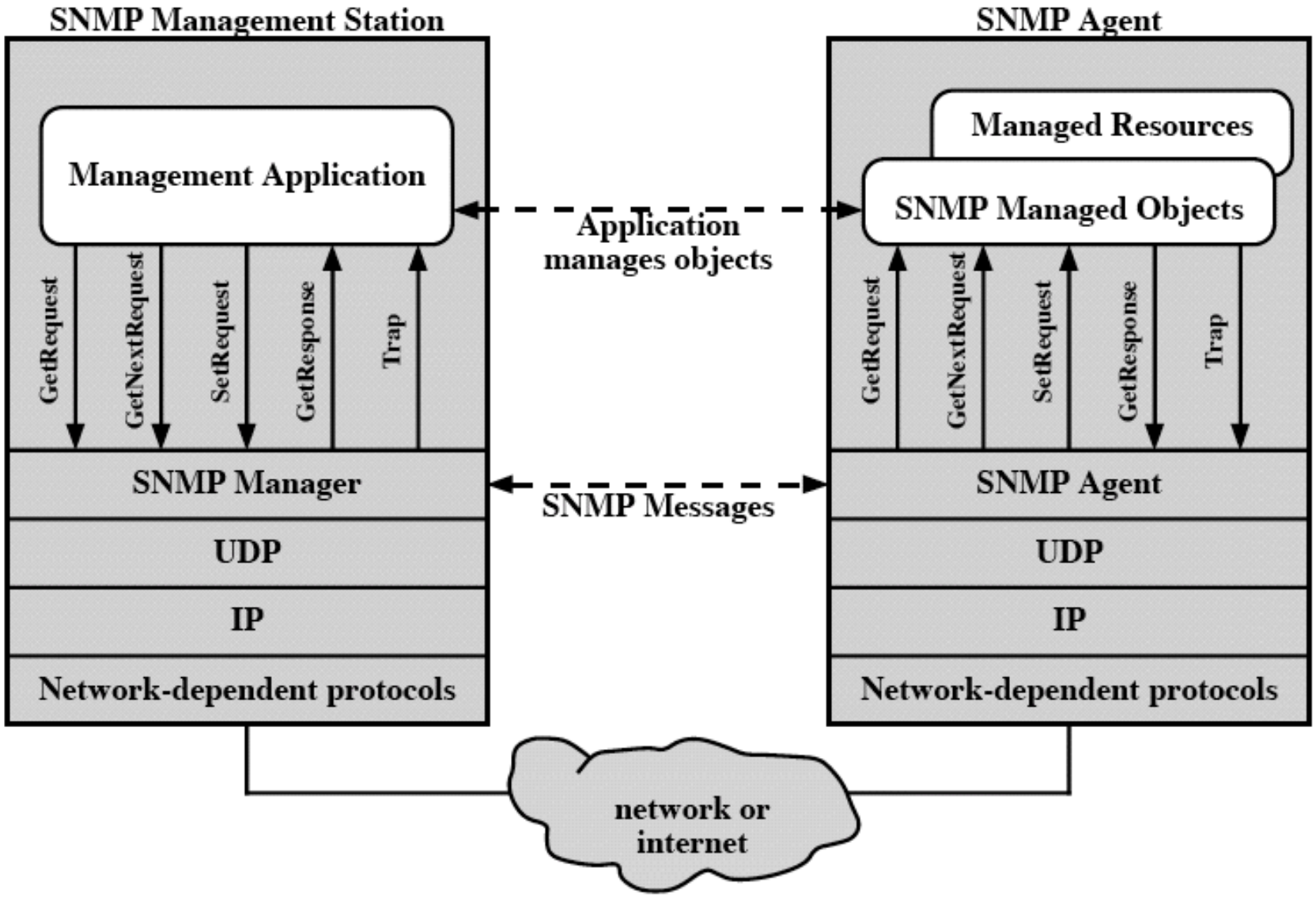
Payment processing

- Payment Authorization:
  - Authorization Request
  - Authorization Response
- Payment Capture:
  - Capture Request
  - Capture Response

Network Management Architecture

- A network management system is a collection of tools for network monitoring and control. that is integrated in the following senses
- The model of network management that is used for SNMP includes the following key elements
- Management station
- Management agent
- Management information base
- Network management protocol

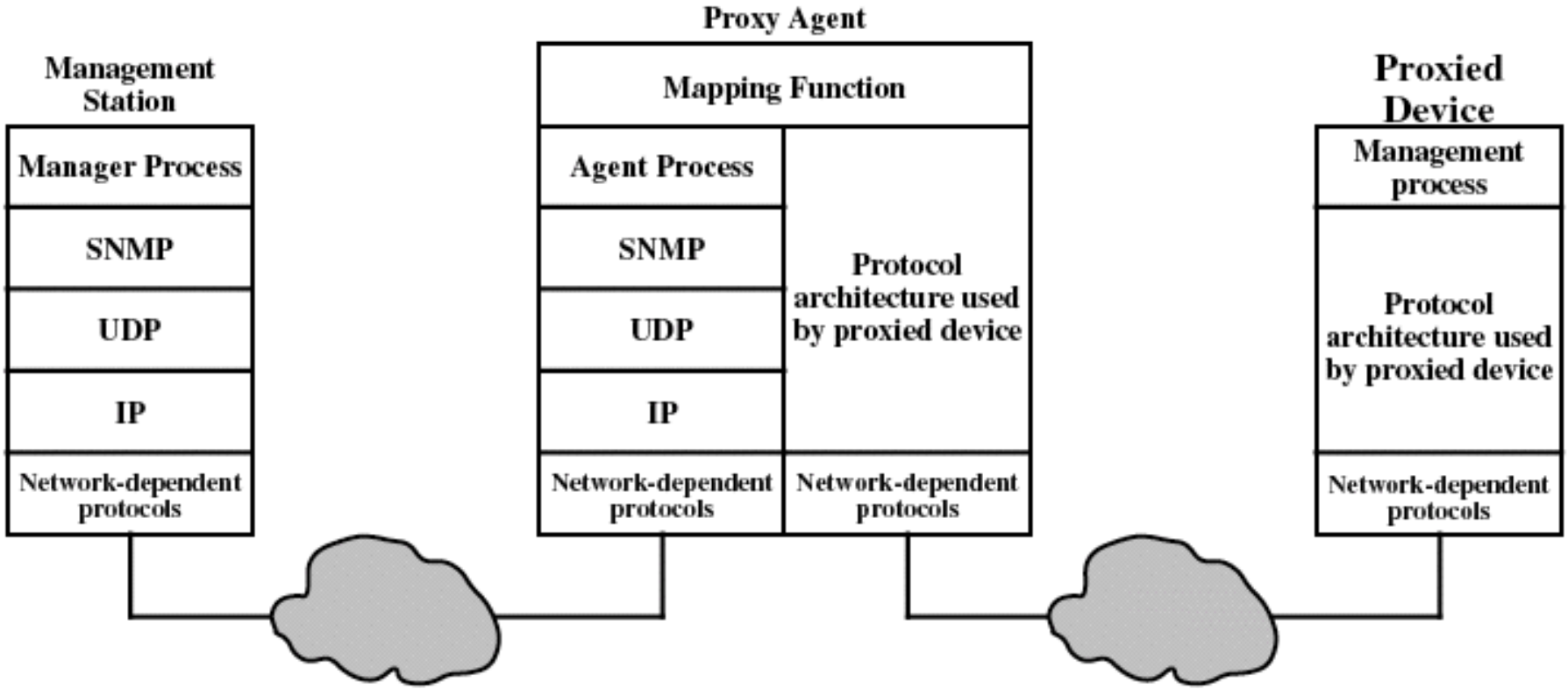
The Role of SNMP



PROTOCOL CONTEXT OF SNMP

- From a management station : three types of SNMP messages are issued on behalf of a management applications:
- GET REQUEST,GET NEXT REQUEST AND SET REQUEST
- All three messages are acknowledged by the agent in the form of a GET RESPONSE MESSAGE, which is passed up to management application.
- In addition, an agent may issue a trap message in response to an event that affects the MIB and the underlying managed resources

Proxy Configuration



Proxy Configuration

- Proxy concept was developed to accommodate devices that do not implement SNMP.
- A SNMP agent acts as a proxy for one or more other devices. it acts on behalf of those devices
- The MANAGEMENT STATION sends a query concerning a device to its proxy agent
- The PROXY AGENT converts each query into the management protocol that is used by the device
- When the agent receives a reply to a query, it passes that reply back to MANAGEMENT STATION

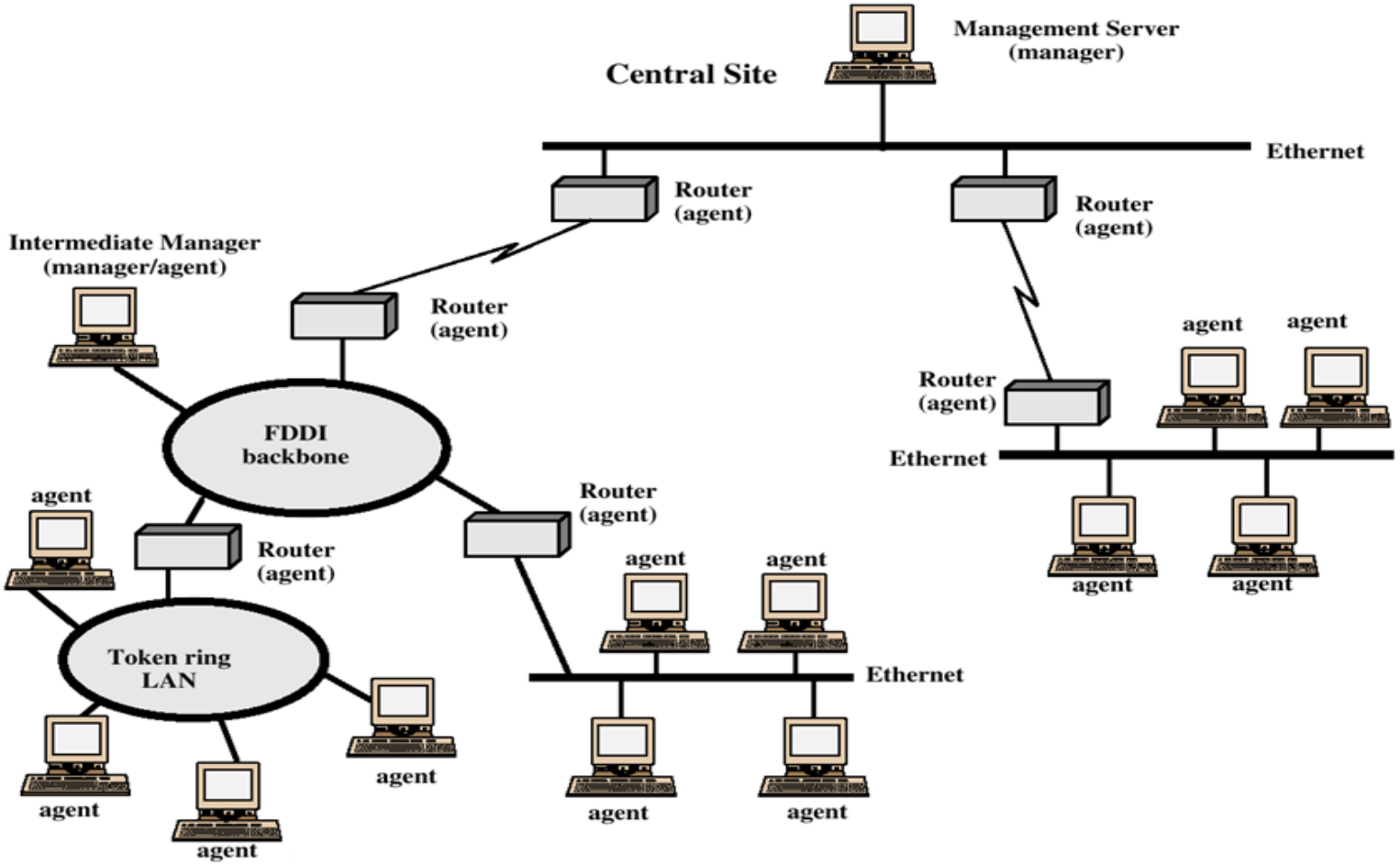
SNMPV2

- SNMP is very simple, that made users relay heavily to manage the ever expanding networks resulting in the deficiencies becoming apparent
- Lack of support for distributed network management
- Functional deficiencies
- Security deficiencies
- SNMP V2 addresses the first two categories of deficiencies and security deficiencies

Distributed network management

- In a decentralized network management scheme, multiple top level management stations are present, referred as MANAGEMENT SERVERS
- MANAGEMENT SERVERS , manages a portion of total pool of agents
- For many agents, the management server delegates responsibility to an intermediate manager
- The intermediate manager plays the role of manger to monitor and control the agents under its responsibility

Distributed Network Management Configuration



Functional enhancements

- Both protocols are defined in terms of a set of commands that are communicated as PROTOCOL DATA UNITS(PDU)
- In the case of SNMP V2 it includes all of five commands found in SNMP V1 plus two new ones
- INFORM COMMAND which is sent by one MS to another MS. including information regarding the sender
- GET BULK ,which allows a manager to retrieve a large block of data at one time

Comparison of SNMPv1 & SNMPv2

SNMPv1 PDU	SNMPv2 PDU	Direction	Description
GetRequest	GetRequest	Manager to agent	Request value for each listed object
Get nextRequest	Get next Request	Manager to agent	Request next value for each listed object
------	GetBulkRequest	Manager to agent	Request multiple values
SetRequest	SetRequest	Manager to agent	Set value for each listed object
------	InformRequest	Manager to manager	Transmit unsolicited information
GetResponse	Response	Agent to manager or Manager to manager(SNMPv2)	Respond to manager request
Trap	SNMPv2-Trap	Agent to manager	Transmit unsolicited information

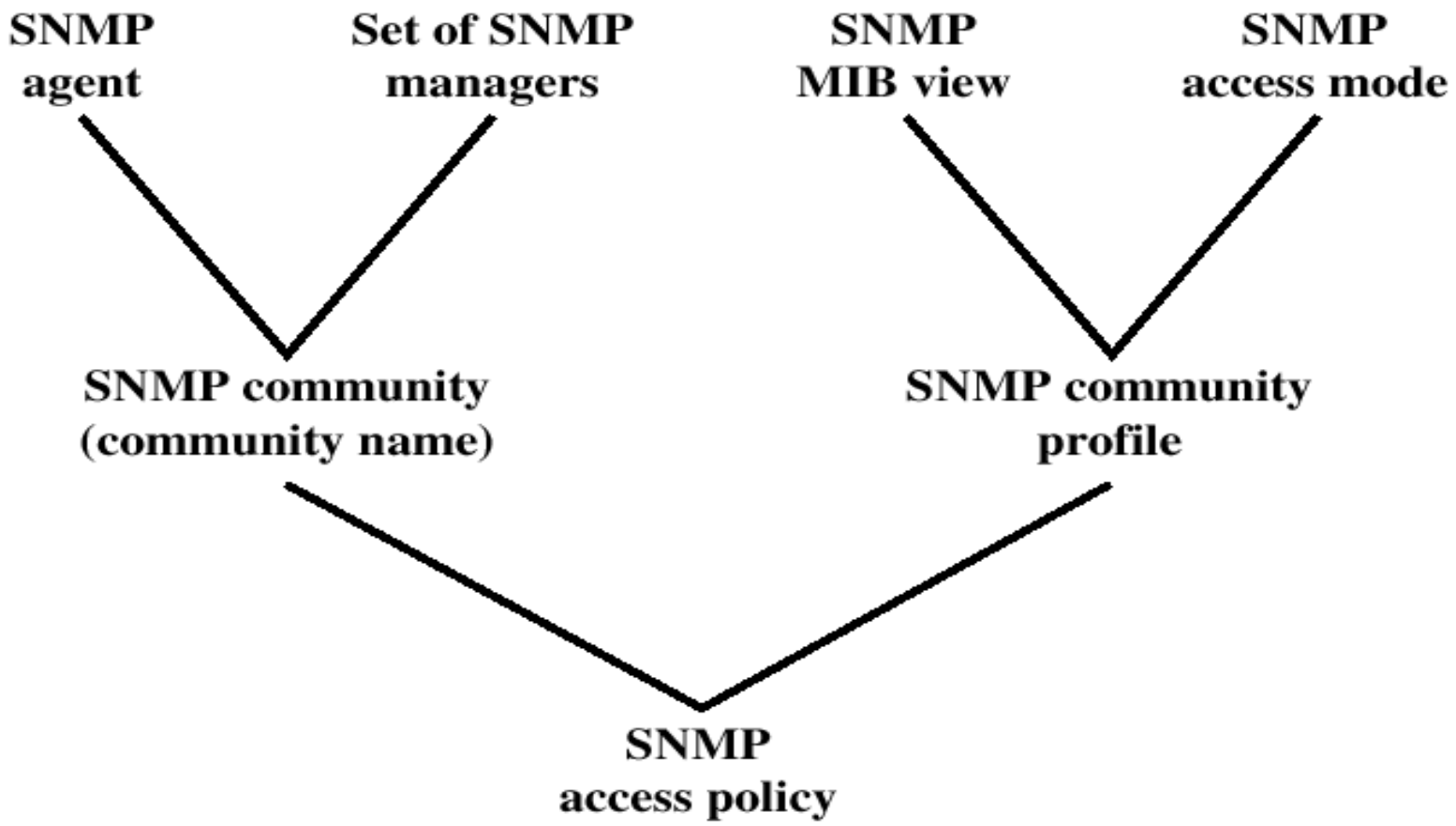
SNMP COMMUNITIES

- SNMP network management involves a one-to-many relationship between a manager and a set of agents. and vice versa.
- Three aspects of agent control are:
- Authentication service
- Access policy
- Proxy service
- A SNMP community is a relationship between an agent and a set of managers that defines authentication, access control& proxy characteristics

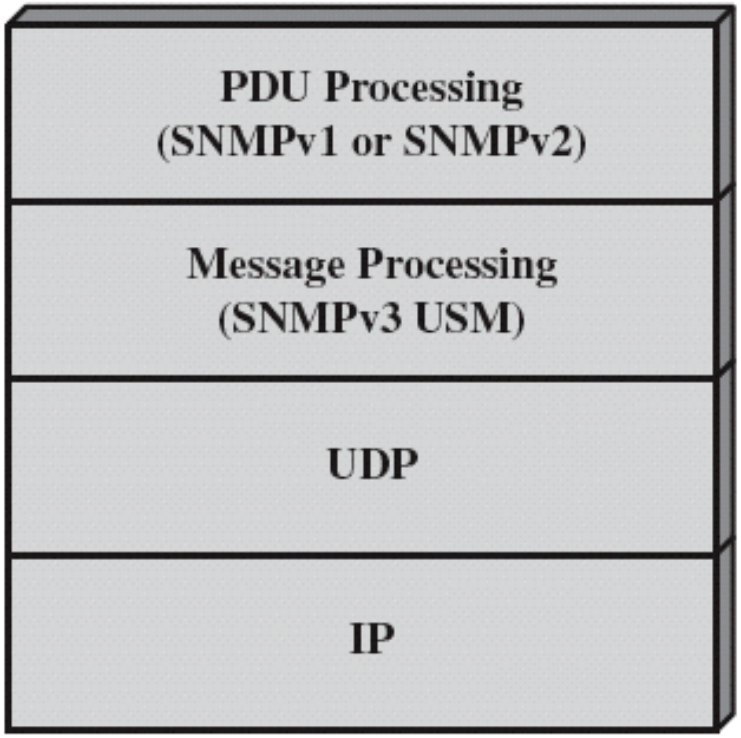
SNMP COMMUNITIES

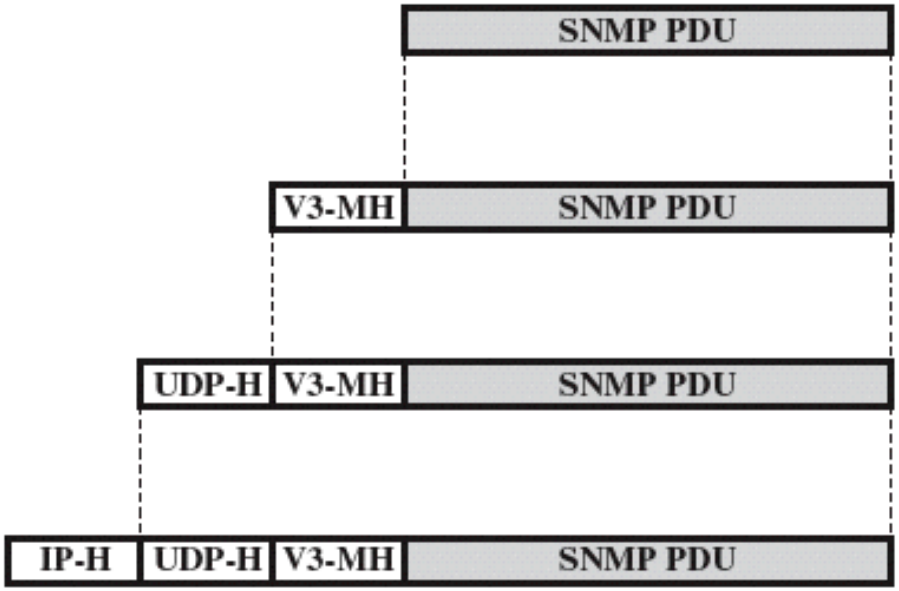
- SNMP AUTHENTICATION SERVICE
- every SNMP message from a manager includes a community name (used as password)– very primitive
- Most agents only allow GET operations
- SNMP ACCESS POLICY
- An agent can provide different categories of MIB access using the concepts of SNMP MIB VIEW AND ACCESS MODE
- SNMP MIB VIEW
- A subset of objects within a MIB
- Different MIB views may be defined for each community
- The set of objects in a view need to belong to single sub tree

SNMPv1 Administrative Concepts



SNMP Protocol Architecture



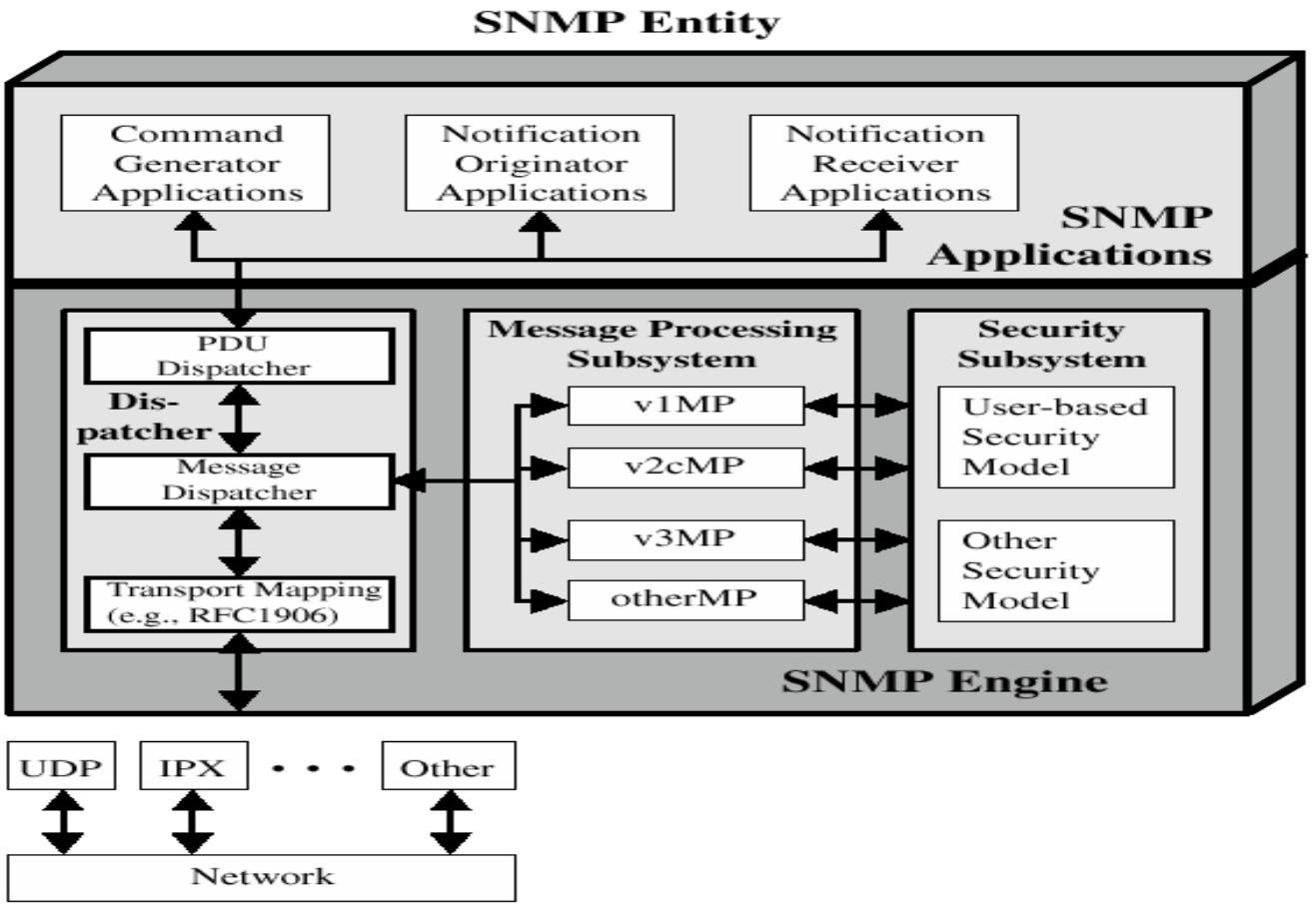


IP-H = IP header
UDP-H = UDP header
V3-MH = SNMPv3 message header
PDU = Protocol data unit

SNMP V 3

- SNMP v3 defines a security capability to be used in conjunction with SNMP V1 OR V2
- SNMP V3 is not a stand alone replacement for either SNMP V1 OR V2
- Information is exchanged between a management station and an agent in the form of SNMP message
- Security – related processing occurs at the message level. For example SNMP V3 specifies a user security model(USM) that makes use of fields in message header

Traditional SNMP Manager



SNMP ARCHITECTURE

- Each entity implements a portion of SNMP capability and may act as a agent node , and a manager or both
- Each SNMP entity consists of collection of modules that interact with each other to provide services& single SNMP engine

- Role of SNMP entity(management or agent) is determined by which modules are implemented in that entity

- An SNMP engine implements functions for sending/receiving messages, authenticating& enc/dec messages& controlling access to managed objects

SNMP ENGINE FUNCTIONS

- SNMP ENGINE PERFOMES 2 FUNCTIONS
- ACCEPTING OUTGOING PDUS FROM SNMP APPLICATIONS:
- Performs necessary processing, including inserting authentication codes and encrypting& then encapsulates the PDUs into message transmission
- ACCEPTING INCOMING SNMP MESSAGES FROM TRANSPORT LAYER:
- Performs necessary processing, including authentication and decryption& then extracts the PDUs from messages and sends to SNMP application

SNMP Manger-engine components

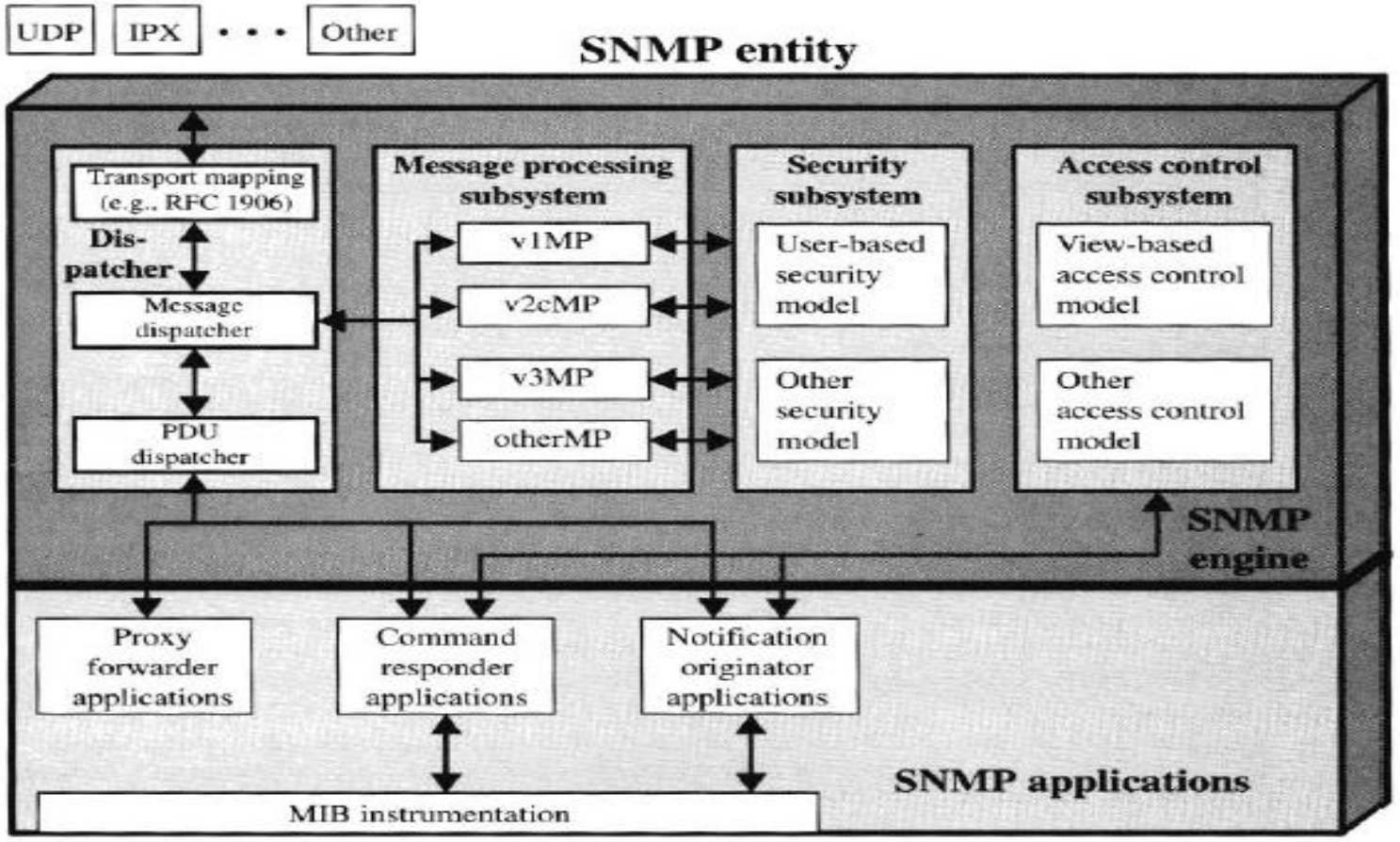
- In a traditional manager, the SNMP engine contains a Dispatcher, a Message processing subsystem & a security sub system

- The Dispatcher is simply a traffic manager, accepting both outgoing PDU'S and incoming PDU'S

- The message processing subsystem accepts the incoming messages from the dispatcher, processes each message header &returns enclosed PDU to the dispatcher

- The security subsystem performs authentication

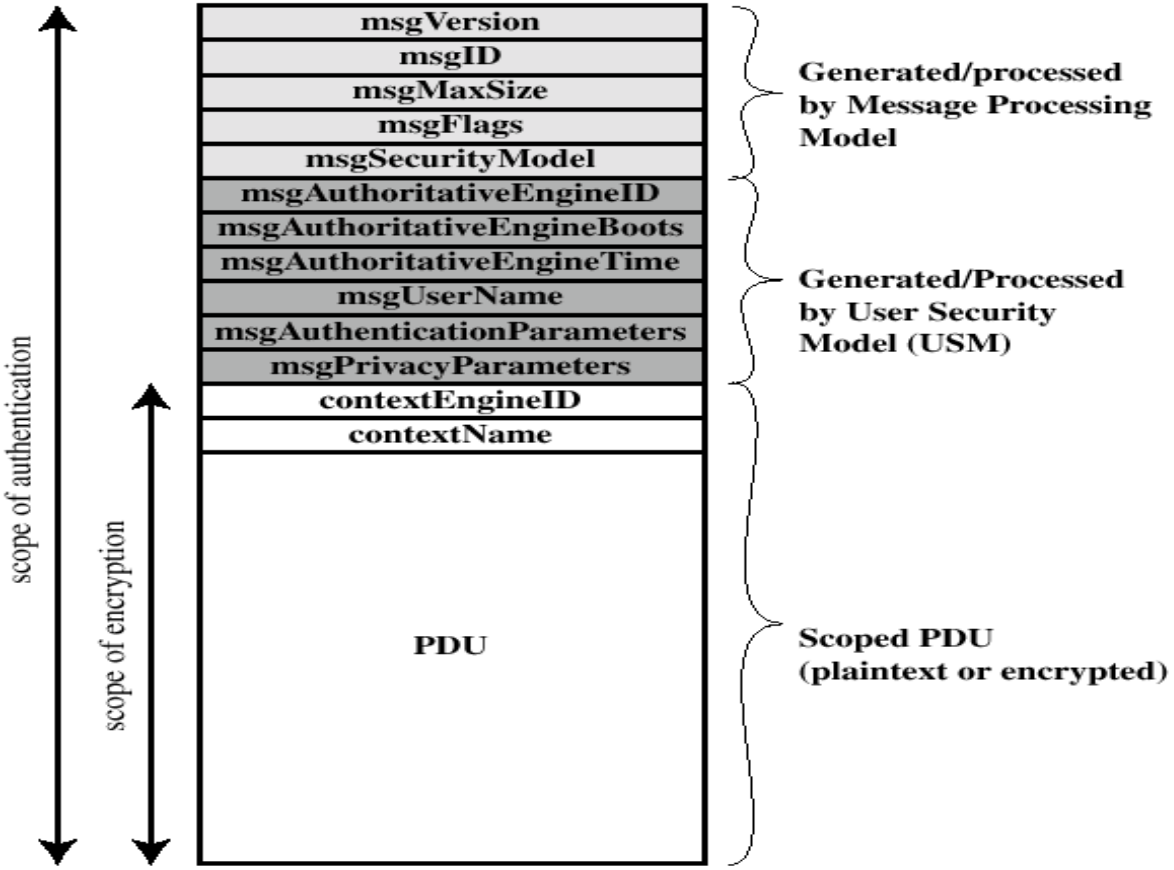
SNMP AGENT


UDP
IPX
Other
SNMP entity
Transport mapping (e.g., RFC 1906)
Dis-patcher
Message dispatcher
PDU dispatcher
Message processing subsystem
v1MP
v2cMP
v3MP
otherMP
Security subsystem
User-based security model
Other security model
Access control subsystem
View-based access control model
Other access control model
SNMP engine
Proxy forwarder applications
Command responder applications
Notification originator applications
SNMP applications
MIB instrumentation

SNMP AGENT

- The traditional SNMP agent contain three types of applications
- Command responder applications: it provides access to management data. these applications respond to incoming request by retrieving managed objects and response PDU
- Notification originator applications: it initiates asynchronous messages. in case of a traditional agent. the SNMP v2 trap or v1 trap PDU is used for this application
- Proxy forwarder applications: it forwards messages between entities

SNMP v3 Message Format with USM



MESSAGE PROCESSING MODEL

- The next six fields shows security parameters used by the USM
- Finally the PDU, together with the CONTEXT ENGINE ID and CONTEXT NAME , constitutes a scoped PDU used for PDU processing
- This model is responsible for accepting PDU'S from dispatcher encapsulating them in messages & invoking the USM to process the security related parameters in the message header

USM (USER SECURITY MODEL)

- Not intended to secure against following threats

- Denial of service(Dos)
- An attacker may prevent exchanges between manager and a agent

- Traffic analysis
- An attacker may observe the general pattern of traffic between manager and agents