EC3401 NETWORKS AND SECURITY

UNIT I NETWORK MODELS AND DATALINK LAYER 9

Overview of Networks and its Attributes – Network Models – OSI, TCP/IP, Addressing – Introduction to Datalink Layer – Error Detection and Correction – Ethernet(802.3)- Wireless LAN – IEEE 802.11, Bluetooth – Flow and Error Control Protocols – HDLC – PPP.

Overview of Networks and its Attributes

Data communications are the exchange of data between two devices via some form of transmission medium such as a wire cable. For data communications to occur, the communicating devices must be part of a communication system made up of a combination of hardware (physical equipment) and software (programs)

Fundamental characteristics of Data communication systems:

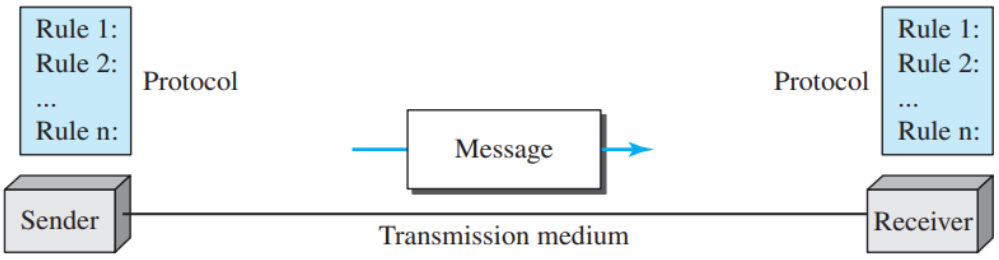
1. Delivery. The system must deliver data to the correct destination. Data must be received by the only by that device

2. Accuracy. The system must deliver the data accurately. Data that have been altered in transmission and left uncorrected are unusable.

3. Timeliness. The system must deliver data in a timely manner. Data delivered late are useless. In the case of video and audio, timely delivery means delivering data as they are produced, in the same order that they are produced, and without significant delay. This kind of delivery is called real-time transmission.

4. Jitter. Jitter refers to the variation in the packet arrival time. It is the uneven delay in the delivery of audio or video packets.

Components of Data communication systems



1. Message. The message is the information (data) to be communicated. Popular forms of information include text, numbers, pictures, audio, and video.

2. Sender. The sender is the device that sends the data message. It can be a computer, workstation, telephone handset, video camera, and so on.

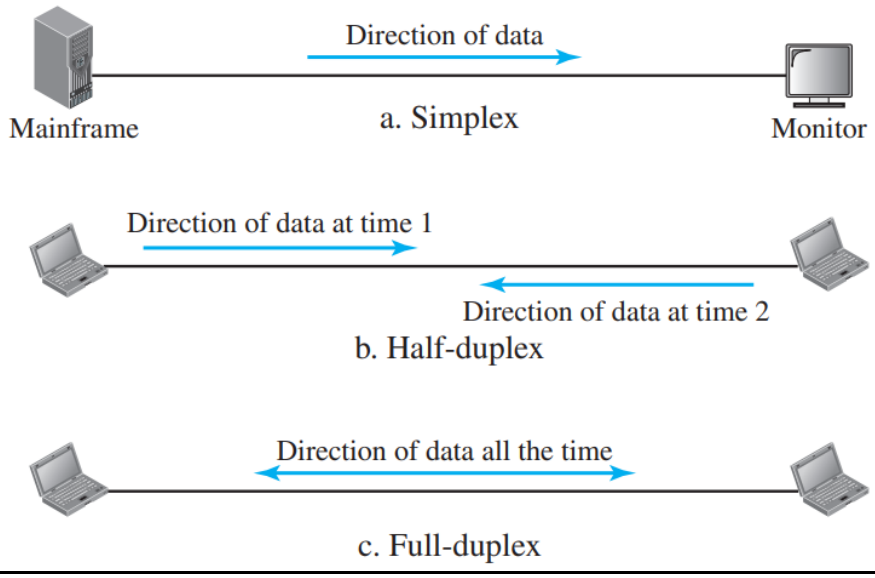
3. Receiver. The receiver is the device that receives the message. It can be a computer, workstation, telephone handset, television, and so on.

4. Transmission medium. The transmission medium is the physical path by which a message travels from sender to receiver. Some examples of transmission media include twisted-pair wire, coaxial cable, fiber-optic cable, and radio waves.

5. Protocol. A protocol is a set of rules that govern data communications. It represents an agreement between the communicating devices

Data Flow

Communication between two devices can be simplex, half-duplex, or full-duplex



Simplex

In simplex mode, the communication is unidirectional, as on a one-way street.
Eg: Monitor, Keyboard

Half-Duplex

In half-duplex mode, each station can both transmit and receive, but not at the same time.
Eg: Walkie-talkies and CB (citizens band) radios

Full-Duplex

In full-duplex mode (also called duplex), both stations can transmit and receive simultaneously
Eg: Telephone, Mobile Phone

NETWORKS

A network is the interconnection of a set of devices capable of communication
Host is a large computer
Eg: Desktop, laptop, workstation, cellular phone, or security system

Network Criteria

Performance: Performance can be measured in many ways, including transit time and response time. Transit time is the amount of time required for a message to travel from one device to another. Response time is the elapsed time between an inquiry and a response.
Eg: The number of users, the type of transmission medium, the capabilities of the connected hardware, and the efficiency of the software
Performance is evaluated by two networking metrics: throughput and delay.

Reliability

Network reliability is measured by the frequency of failure, the time it takes a link to recover from a failure, and the network's robustness in a catastrophe.

Security

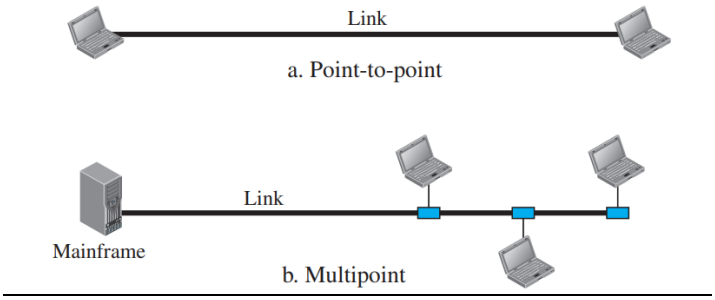
Network security issues include protecting data from unauthorized access, protecting data from damage and development, and implementing policies and procedures for recovery from breaches and data losses.

Physical Structures

A network is two or more devices connected through links. A link is a communications pathway that transfers data from one device to another

Point-to-Point: A point-to-point connection provides a dedicated link between two devices

Multipoint: A multipoint (also called multidrop) connection is one in which more than two specific devices share a single link

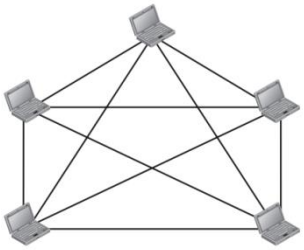


Physical Topology

The term physical topology refers to the way in which a network is laid out physically. It is a geometric representation of the relationship of all the links and linking devices (usually called nodes) to one another. There are four basic topologies possible: mesh, star, bus, and ring.

Mesh Topology

In a mesh topology, every device has a dedicated point-to-point link to every other device. The term dedicated means that the link carries traffic only between the two devices it connects.

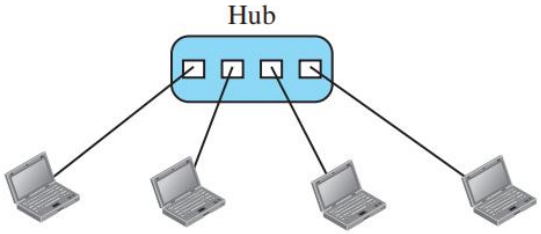


The use of dedicated links guarantees that each connection can carry its own data load, thus eliminating the traffic problems. If one link becomes unusable, it does not incapacitate the entire system. High privacy and security

The main disadvantages of a mesh are related to the amount of cabling and the number of I/O ports required.

Star Topology

In a star topology, each device has a dedicated point-to-point link only to a central controller, usually called a hub. The devices are not directly linked to one another



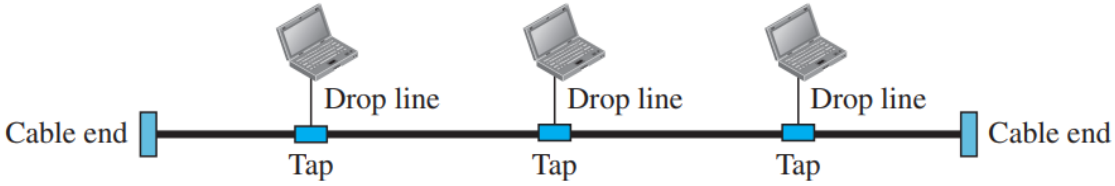
If one link fails, only that link is affected.

One big disadvantage of a star topology is the dependency of the whole topology on one single point, the hub

The star topology is used in local-area networks (LANs)

Bus Topology

One long cable acts as a backbone to link all the devices in a network

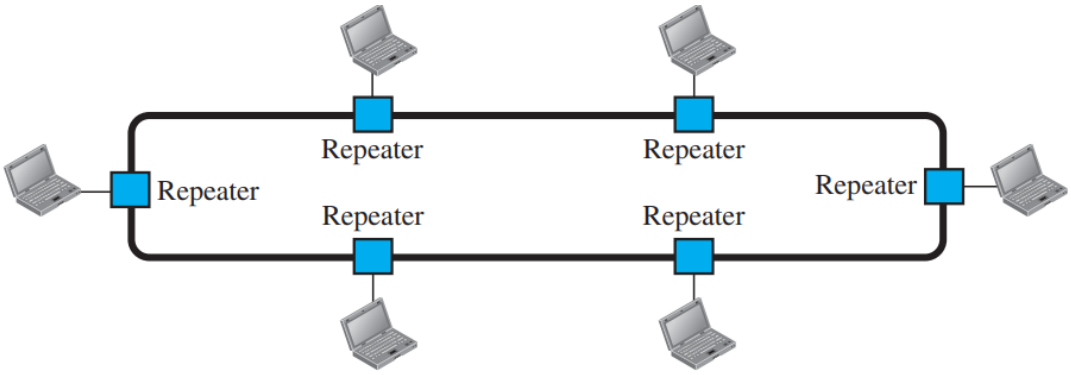


Advantages of a bus topology include ease of installation

Disadvantages include difficult reconnection and fault isolation.

Ring Topology

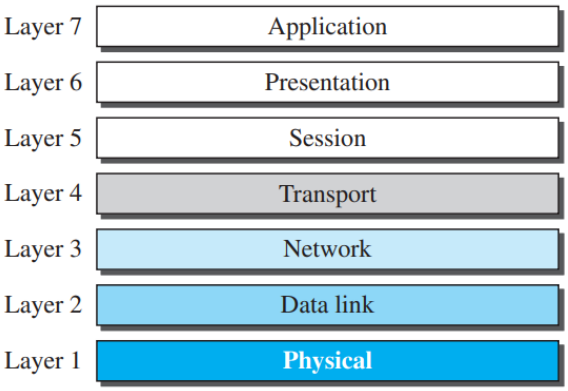
In a ring topology, each device has a dedicated point-to-point connection with only the two devices on either side of it. A signal is passed along the ring in one direction, from device to device, until it reaches its destination.



In a simple ring, a break in the ring (such as a disabled station) can disable the entire network. This weakness can be solved by using a dual ring or a switch capable of closing off the break.

THE OSI MODEL:

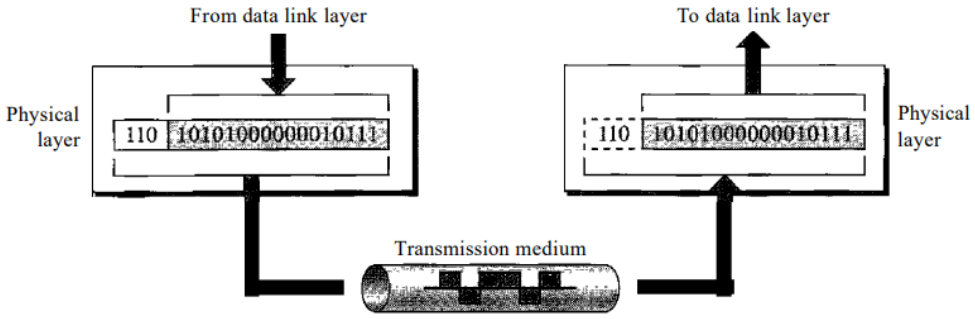
- Established in 1947, the International Organization for Standardization (ISO) is a multinational body dedicated to worldwide agreement on international standards.
- An ISO standard that covers all aspects of network communications is the Open Systems Interconnection (OSI) model. It was first introduced in the late 1970s.
- The purpose of the OSI model is to show how to facilitate communication between different systems without requiring changes to the logic of the underlying hardware and software
- The OSI model is not a protocol; it is a model for flexible, robust, and interoperation
- It consists of seven separate but related layers, each of which defines a part of the process of moving information across a network



Layers in the OSI Model:

Physical Layer:

- Physical Layer The physical layer coordinates the functions required to carry a bit stream over a physical medium.
- It deals with the mechanical and electrical specifications of the interface and transmission medium

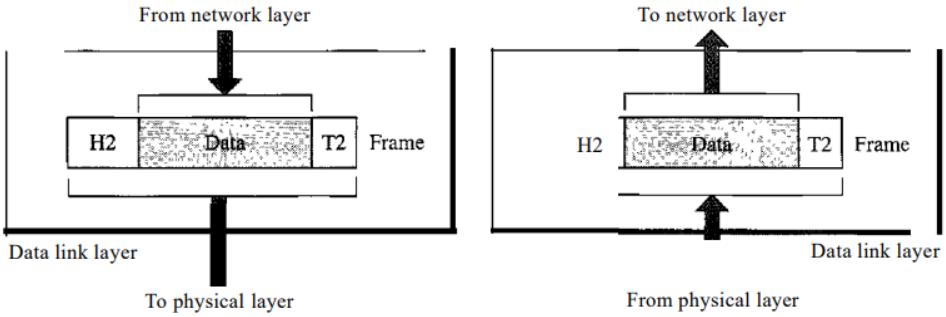


Responsibilities of Physical Layer:

- **Data rate.**
  The transmission rate-the number of bits sent each second-is also defined by the physical layer.
- **Synchronization of bits.**
  The sender and receiver not only must use the same bit rate but also must be synchronized at the bit level.
- **Line configuration**.
  It defines the connection is whether the connection is point-to-point or point to multipoint like broadcasting
- **Physical topology.**
  The physical topology defines how devices are connected to make a network. Devices can be connected by using a mesh topology , a star topology , a ring topology , a bus topology, or a hybrid topology (
- **Transmission mode.**
  The physical layer also defines the direction of transmission between two devices: simplex, half-duplex, or full-duplex.

Data Link Layer:

The data link layer transforms the physical layer, a raw transmission facility, to a reliable link.
It makes the physical layer appear error-free to the upper layer



- **Framing:**
  The data link layer divides the stream of bits received from the network layer into manageable data units called frames.
- **Physical addressing.**
  If frames are to be distributed to different systems on the network, the data link layer adds a header to the frame to define the sender and/or receiver of the frame.
- **Flow control**.

If the rate at which the data are absorbed by the receiver is less than the rate at which data are produced in the sender, the data link layer imposes a flow control mechanism to avoid overwhelming the receiver.

- **Error control.**
  The data link layer adds reliability to the physical layer by adding mechanisms to detect and retransmit damaged or lost frames.
  It also uses a mechanism to recognize duplicate frames.
- **Access control**.
  When two or more devices are connected to the same link, data link layer protocols are necessary to determine which device has control over the link at any given time

Network Layer

- The network layer is responsible for the source-to-destination delivery of a packet, possibly across multiple networks (links).
- The data link layer oversees the delivery of the packet between two systems on the same network (links), the network layer ensures that each packet gets from its point of origin to its final destination.
- If two systems are connected to the same link, there is usually no need for a network layer.

Resposibilities of Network Layer

Logical addressing:

The physical addressing implemented by the data link layer handles the addressing problem locally. If a packet passes the network boundary, we need another addressing system to help distinguish the source and destination systems.

The network layer adds a header to the packet coming from the upper layer that, among other things, includes the logical addresses of the sender and receiver.

Routing.

When independent networks or links are connected to create internetworks (network of networks) or a large network, the connecting devices route or switch the packets to their final destination.

Transport Layer

- The transport layer is responsible for process-to-process delivery of the entire message.
- A process is an application program running on a host. Whereas the network layer oversees source-to-destination delivery of individual packets, it does not recognize any relationship between those packets.
- It treats each one independently, as though each piece belonged to a separate message, whether or not it does.
- The transport layer, on the other hand, ensures that the whole message arrives intact and in order, overseeing both error control and flow control at the source-to-destination level.

Responsibilities of Transport Layer:

- **Service-point addressing.**
  Computers often run several programs at the same time.
  For this reason, source-to-destination delivery means delivery not only from one computer to the next but also from a specific process (running program) on one computer to a specific process (running program) on the other. The transport layer header must therefore include a type of address called a service-point address (or port address).
  The network layer gets each packet to the correct computer; the transport layer gets the entire message to the correct process on that computer.
- **Segmentation and reassembly.**

A message is divided into transmittable segments, with each segment containing a sequence number. These numbers enable the transport layer to reassemble the message correctly upon arriving at the destination and to identify and replace packets that were lost in transmission.

- **Connection control.**
  The transport layer can be either connectionless or connectionoriented.
  A connectionless transport layer treats each segment as an independent packet and delivers it to the transport layer at the destination machine.
  A connectionoriented transport layer makes a connection with the transport layer at the destination machine first before delivering the packets.
  After all the data are transferred, the connection is terminated.
- **Flow control.**
  Like the data link layer, the transport layer is responsible for flow control.
  However, flow control at this layer is performed end to end rather than across a single link.
- **Error control.**
  Like the data link layer, the transport layer is responsible for error control.
  However, error control at this layer is performed process-toprocess rather than across a single link.
  Error correction is usually achieved through retransmission

Session Layer

- The services provided by the first three layers (physical, data link, and network) are not sufficient for some processes.
- The session layer is the network dialog controller.
- It establishes, maintains, and synchronizes the interaction among communicating systems

Responsibilities of Session Layer:

- **Dialog control.**
  The session layer allows two systems to enter into a dialog.
  It allows the communication between two processes to take place in either halfduplex (one way at a time) or full-duplex (two ways at a time) mode.
- **Synchronization.**
  The session layer allows a process to add checkpoints, or synchronization points, to a stream of data. For example, if a system is sending a file of 2000 pages, it is advisable to insert checkpoints after every 100 pages to ensure that each 100-page unit is received and acknowledged independently. In this case, if a crash happens during the transmission of page 523, the only pages that need to be resent after system recovery are pages 501 to 523. Pages previous to 501 need not be resent. Figure 2.12 illustrates the relationship of the session layer to the transport and presentation layers.

  Presentation Layer

  The presentation layer is concerned with the syntax and semantics of the information exchanged between two systems
- **Translation.** The processes (running programs) in two systems are usually exchanging information in the form of character strings, numbers, and so on. The information must be changed to bit streams before being transmitted. Because different computers use different encoding systems, the presentation layer is responsible for interoperability between these different encoding methods. The presentation layer at the sender changes the information from its sender-dependent format into a common format
- **Encryption.** To carry sensitive information, a system must be able to ensure privacy
- **Compression.** Data compression reduces the number of bits contained in the information

Application Layer

- The application layer enables the user, whether human or software, to access the network.
- It provides user interfaces and support for services such as electronic mail, remote file access and transfer, shared database management, and other types of distributed information services

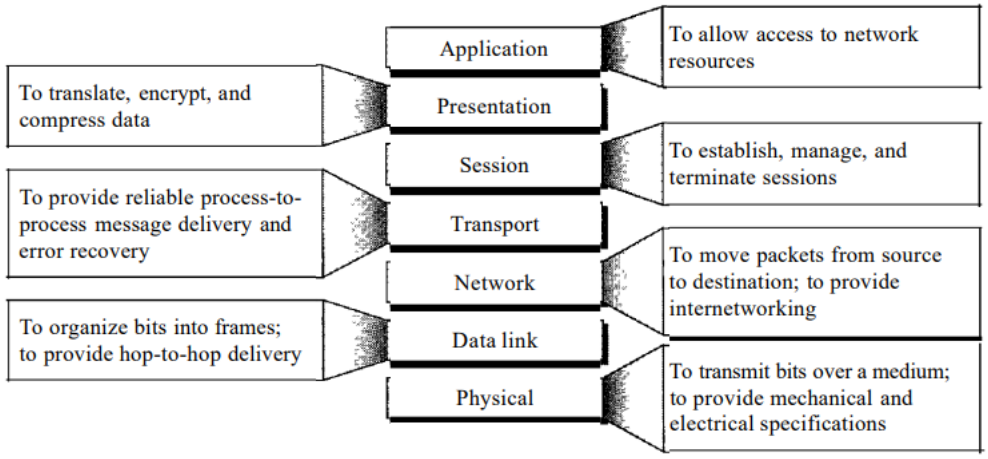
Network virtual terminal.

A network virtual terminal is a software version of a physical terminal, and it allows a user to log on to a remote host.

File transfer, access, and management.

Mail services. This application provides the basis for e-mail forwarding and storage.

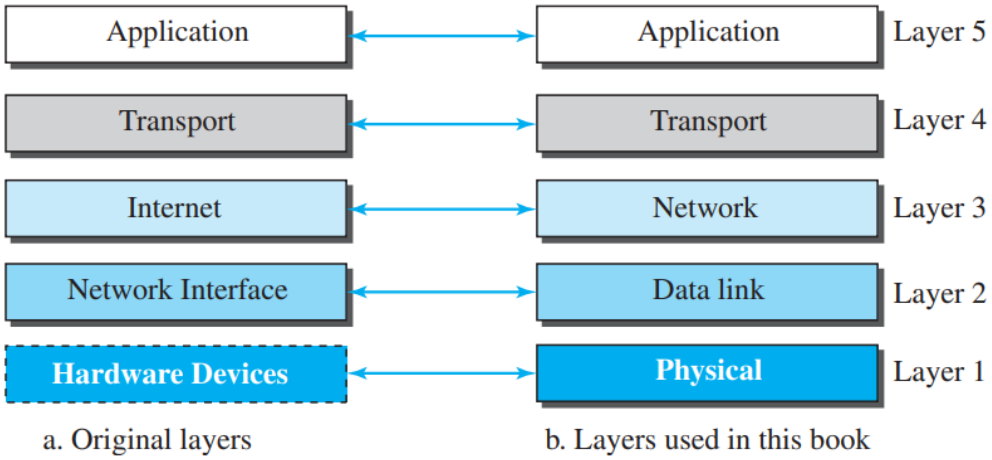
Directory services. This application provides distributed database sources and access for global information about various objects and services.

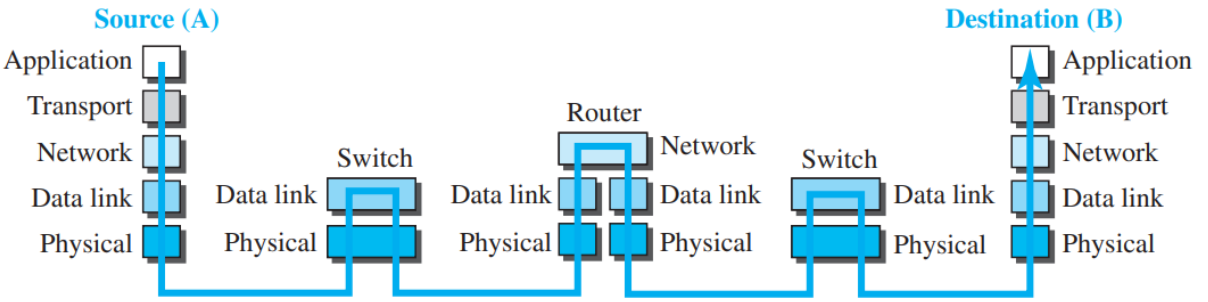


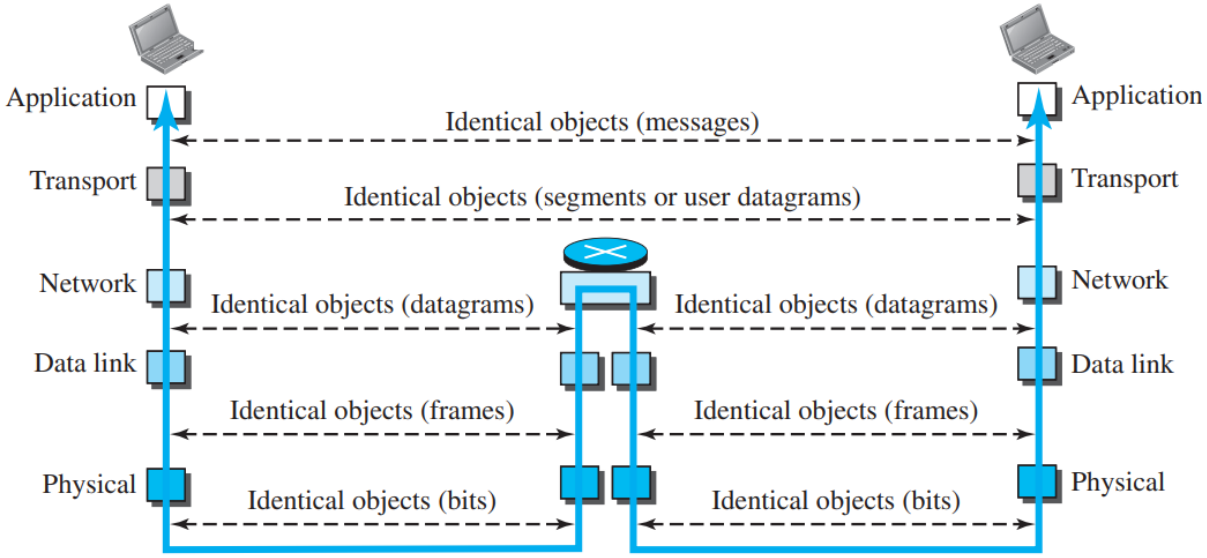
TCP/IP is a protocol suite

- A set of protocols organized in different layers) used in the Internet today.
- It is a hierarchical protocol made up of interactive modules, each of which provides a specific functionality.
- Each upper level protocol is supported by the services provided by one or more lower level protocols. The original TCP/IP protocol suite was defined as four software layers built upon the hardware

Layers in TCP/IP Protocol Suite







Physical Layer

- The physical layer is responsible for carrying individual bits in a frame across the link.
- It is the lowest level in the TCP/IP protocol suite
- The communication between two devices at the physical layer is still a logical communication because there is another, hidden layer, the transmission media, under the physical layer.
- Two devices are connected by a transmission medium (cable or air).
- The transmission medium does not carry bits; it carries electrical or optical signals. So the bits received in a frame from the data-link layer are transformed and sent through the transmission media, but we can think that the logical unit between two physical layers in two devices is a bit. There are several protocols that transform a bit to a signal.

Data-link Layer

- There may be several overlapping sets of links that a datagram can travel from the host to the destination.
- The routers are responsible for choosing the best links. However, when the next link to travel is determined by the router, the data-link layer is responsible for taking the datagram and moving it across the link.
- The link can be a wired LAN with a link-layer switch, a wireless LAN, a wired WAN, or a wireless WAN. We can also have different protocols used with any link type.

- TCP/IP does not define any specific protocol for the data-link layer. It supports all the standard and proprietary protocols.
- Some data link-layer protocols provide complete error detection and correction, some provide only error correction

Network Layer

- The network layer is responsible for creating a connection between the source computer and the destination computer.
- The communication at the network layer is host-to-host. However, since there can be several routers from the source to the destination, the routers in the path are responsible for choosing the best route for each packet.
- Since there can be several routers from the source to the destination, the routers in the path are responsible for choosing the best route for each packet
- The network layer in the Internet includes the main protocol, Internet Protocol (IP), that defines the format of the packet, called a datagram at the network layer
- IP is a connectionless protocol that provides no flow control, no error control, and no congestion control services
- A routing protocol does not take part in routing (it is the responsibility of IP), but it creates forwarding tables for routers to help them in the routing process.
- The network layer also has some auxiliary protocols that help IP in its delivery and routing tasks.
- The Internet Control Message Protocol (ICMP) helps IP to report some problems when routing a packet.
- The Internet Group Management Protocol (IGMP) is another protocol that helps IP in multitasking.
- The Dynamic Host Configuration Protocol (DHCP) helps IP to get the network-layer address for a host.
- The Address Resolution Protocol (ARP) is a protocol that helps IP to find the link-layer address of a host or a router when its network-layer address is given.

Transport Layer

- The transport layer at the source host gets the message from the application layer, encapsulates it in a transportlayer packet and sends it, through the logical connection, to the transport layer at the destination host.
- The transport layer is responsible for giving services to the application layer
- The transport layer should be independent of the application layer
- The main protocol, Transmission Control Protocol (TCP), is a connection-oriented protocol that first establishes a logical connection between transport layers at two hosts before transferring data. It creates a logical pipe between two TCPs for transferring a stream of bytes
- TCP provides flow control and error control Mechanisms
- User Datagram Protocol (UDP), is a connectionless protocol that transmits user datagrams without first creating a logical connection
- UDP is a simple protocol that does not provide flow, error, or congestion control.

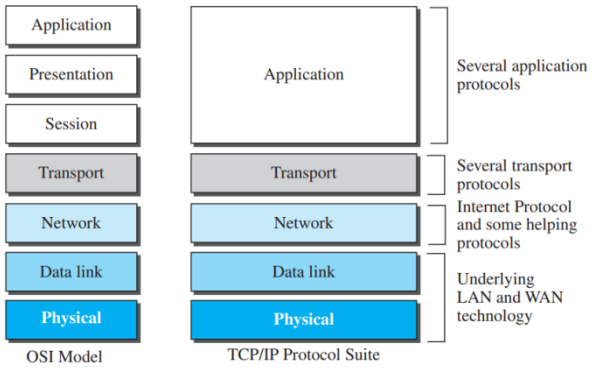
Application Layer

- The two application layers exchange messages between each other as though there were a bridge between the two layers.

- To communicate, a process sends a request to the other process and receives a response.
- Process-to-process communication is the duty of the application layer. The a user can also create a pair of processes to be run at the two hosts
- The Hypertext Transfer Protocol (HTTP) is a vehicle for accessing the World Wide Web (WWW).
- The Simple Mail Transfer Protocol (SMTP) is the main protocol used in electronic mail (e-mail) service.
- The File Transfer Protocol (FTP) is used for transferring files from one host to another.
- The Terminal Network (TELNET) and Secure Shell (SSH) are used for accessing a site remotely.
- The Simple Network Management Protocol (SNMP) is used by an administrator to manage the Internet at global and local levels.
- The Domain Name System (DNS) is used by other protocols to find the network-layer address of a computer

OSI versus TCP/IP:

- There are two layers, session and presentation, are missing from the TCP/IP protocol suite.
- These two layers were not added to the TCP/IP protocol suite after the publication of the OSI model



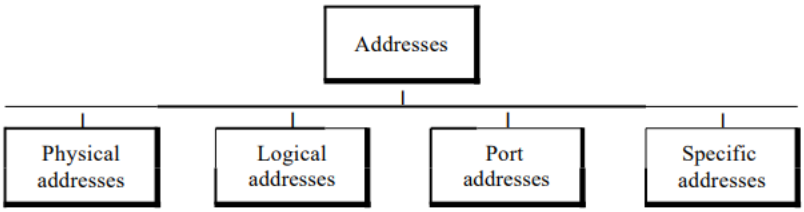
- TCP/IP has more than one transport-layer protocol. Some of the functionalities of the session layer are available in some of the transport-layer protocols.
- Second, the application layer is not only one piece of software. Many applications can be developed at this layer
- If some of the functionalities mentioned in the session and presentation layers are needed for a particular application, they can be included in the development of that piece of software.

Lack of OSI Model's Success:

- First, OSI was completed when TCP/IP was fully in place and a lot of time and money had been spent on the suite
- Second, some layers in the OSI model were never fully defined. For example, although the services provided by the presentation and the session layers were listed in the document, actual protocols for these two layers were not fully defined.
- Third, when OSI was implemented by an organization in a different application, it did not show a high enough level of performance

Addressing:

There are four levels of addresses are used in an internet employing the TCP/IP protocols: physical (link) addresses, logical (IP) addresses, port addresses, and specific addresses



Physical Address or Link Address:

- It is the address of a node as defined by its LAN or WAN.
- It is included in the frame used by the data link layer.
- It is the lowest-level address.
- The physical addresses have authority over the network (LAN or WAN).
- The size and format of these addresses vary depending on the network.

  Eg: Ethernet uses a 6-byte (48-bit) physical address that is imprinted on the network interface card (NIC).

07:01:02:01:2C:4B

A 6-byte (12 hexadecimal digits) physical address

Logical Address:

- Physical addresses are not adequate in an internetwork environment where different networks can have different address formats.
- A universal addressing system is needed in which each host can be identified uniquely, regardless of the underlying physical network.
- The logical addresses are designed for this purpose.
- A logical address in the Internet is currently a 32-bit address that can uniquely define a host connected to the Internet
- The physical addresses will change from hop to hop, but the logical addresses usually remain the same.

Port Addresses:

- The IP address and the physical address are necessary for a quantity of data to travel from a source to the destination host. H
- However, arrival at the destination host is not the final objective of data communications on the Internet.
- Today, computers are devices that can run multiple processes at the same time.
- The end objective of Internet communication is a process communicating with another process
- Port address is used to address different processes for receiving the data simultaneously

Specific Addresses

- Some applications have user-friendly addresses that are designed for that specific address.

  Eg: E-mail address (for example, forouzan@fhda.edu)

  Universal Resource Locator (URL) (for example, www.mhhe.com).

Data Link Layer:

- Data link control functions include framing, flow and error control, and softwareimplemented protocols that provide smooth and reliable transmission of frames between nodes
- The duty scope of the data-link layer is node-to-node.
- When a packet is travelling in the Internet, the data-link layer of a node (host or router) is responsible for delivering a datagram to the next node in the path.
- For this purpose, the data-link layer of the sending node needs to encapsulate the datagram received from the network in a frame, and the data-link layer of the receiving node needs to decapsulate the datagram from the frame.

Framing:

- A packet at the data-link layer is normally called a frame.
- The first service provided by the data-link layer is framing
- The data-link layer at each node needs to encapsulate the datagram in a frame before sending it to the next node
- The node also needs to decapsulate the datagram from the frame received on the logical channel.
- Framing in the data link layer separates a message from one source to a destination, or from other messages to other destinations, by adding a sender address and a destination address.
- The destination address defines where the packet is to go; the sender address helps the recipient acknowledge the receipt

Flow Control:

- Flow control refers to a set of procedures used to restrict the amount of data that the sender can send before waiting for acknowledgment.
- The flow of data must not be allowed to overwhelm the receiver.
- Any receiving device has a limited speed at which it can process incoming data and a limited amount of memory in which to store incoming data.
- The receiving device must be able to inform the sending device before those limits are reached and to request that the transmitting device send fewer frames or stop temporarily.
- Incoming data must be checked and processed before they can be used.

Error Control:

- At the sending node, a frame in a data-link layer needs to be changed to bits, transformed to electromagnetic signals, and transmitted through the transmission media.
- At the receiving node, electromagnetic signals are received, transformed to bits, and put together to create a frame.
- Since electromagnetic signals are susceptible to error, a frame is susceptible to error. The error needs first to be detected.
- After detection, it needs to be either corrected at the receiver node or discarded and retransmitted by the sending node.
- Error control in the data link layer is based on automatic repeat request, which is the retransmission of data

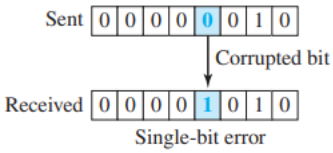
Error Detection and Correction:

Types of Errors

Whenever bits flow from one point to another, they are subject to unpredictable changes because of interference. This interference can change the shape of the signal.

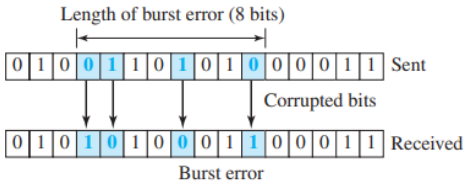
Single bit Error:

In this only 1 bit of a given data unit (such as a byte, character, or packet) is changed from 1 to 0 or from 0 to 1.



Burst Error:

The term burst error means that 2 or more bits in the data unit have changed from 1 to 0 or from 0 to 1.



A burst error is more likely to occur than a single-bit error because the duration of the noise signal is normally longer than the duration of 1 bit, which means that when noise affects data, it affects a set of bits.

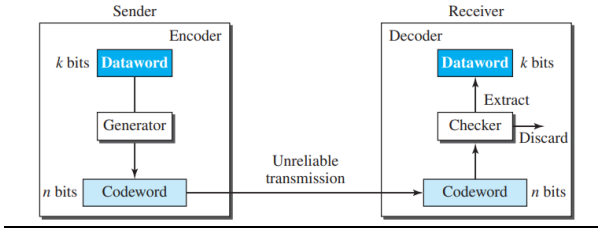
Redundancy:

The central concept in detecting or correcting errors is redundancy. To be able to detect or correct errors, we need to send some extra bits with our data. These redundant bits are added by the sender and removed by the receiver. The number of errors and the size of the message are important factors. There are two coding techniques are commonly available- Block coding and Convolutional coding

BLOCK CODING:

- In block coding, the total message is divided into blocks, each of k bits, called **data words.**
- There are r redundant bits are added to each block to make the length n = k + r. The resulting n-bit blocks are called **code words.**
- With k bits, we can create a combination of 2^k data words; with n bits, we can create a combination of 2^n code words.
- Since n > k, the number of possible code words is larger than the number of possible data words. The block coding process is one-to-one; the same dataword is always encoded as the same code word. This means that we have $2^n - 2^k$ code words that are not used.
- If the receiver receives an invalid code word, this indicates that the data was corrupted during transmission.
- The sender creates codewords out of data words by using a generator that applies the rules and procedures of encoding.
- Each code word sent to the receiver may change during transmission. If the received code word is the same as one of the valid code words, the word is accepted; the corresponding data word is extracted for use.

- If the received codeword is not valid, it is discarded. However, if the code word is corrupted during transmission but the received word still matches a valid code word, the error remains undetected



Hamming Distance:

- **The Hamming distance between two words is the number of differences between the corresponding bits**.The Hamming distance between two words x and y as d(x, y).
- It is distance between the received code word and the sent code word is the number of bits that are corrupted during transmission.
  Eg: If the codeword 00000 is sent and 01101 is received, 3 bits are in error and the Hamming distance between the two is d(00000, 01101) = 3.
- The Hamming distance can easily be found by applying the XOR operation (⊕) on the two words and count the number of 1s in the result.
  Eg: . The Hamming distance d(000, 011) is 2 because (000 ⊕ 011) is 011 (two 1s). 2.
  The Hamming distance d(10101, 11110) is 3 because (10101 ⊕ 11110) is 01011 (three 1s).
- In a set of codewords, the **minimum Hamming distance** is the smallest Hamming distance between all possible pairs of code words
- If s errors occur during transmission, the Hamming distance between the sent codeword and received code word is s. If our system is to detect up to s errors, the minimum distance between the valid codes must be (s + 1).

$$d_{min} = s + 1.$$

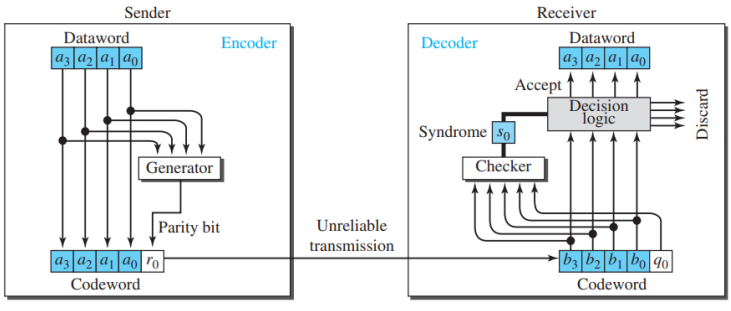
Linear Block Codes:

- Almost all block codes used today belong to a subset of block codes called linear block codes.
- In this an exclusive OR (addition modulo-2) of two valid code words creates another valid code word.

Parity-Check Code

- It is the most familiar error-detecting code is the parity-check code.
- This code is a linear block code.
- In this code, a k-bit data word is changed to an n-bit code word where n = k + 1. The extra bit, called the parity bit, is selected to make the total number of 1s in the code word even.
- Parity check code is a single bit error detection code

Dataword	*Codeword*
0000	**00000**
0001	**00011**
0010	**00101**
0011	**00110**
0100	**01001**
0101	**01010**



Cyclic codes:

- Cyclic codes are special linear block codes with one extra property.
- In a cyclic code, if a code word is cyclically shifted (rotated), the result is another code word.

  Eg: if 1011000 is a code word and we cyclically left-shift, then 0110001 is also a Code word.

$b_1 = a_0$ $b_2 = a_1$ $b_3 = a_2$ $b_4 = a_3$ $b_5 = a_4$ $b_6 = a_5$ $b_0 = a_6$

Cyclic Redundancy Check:

- A subset of cyclic codes called the cyclic redundancy check (CRC), which is used in networks such as LANs and WANs
- It is based on binary division.

Sender Side:

- A string of n 0's is appended to the data unit to be transmitted.
- Here, n is one less than the number of bits in CRC generator.
- Binary division is performed of the resultant string with the CRC generator.
- After division, the remainder so obtained is called as CRC.
- It may be noted that CRC also consists of n bits.
- The newly formed code word (Original data + CRC) is transmitted to the receiver.

At receiver side:

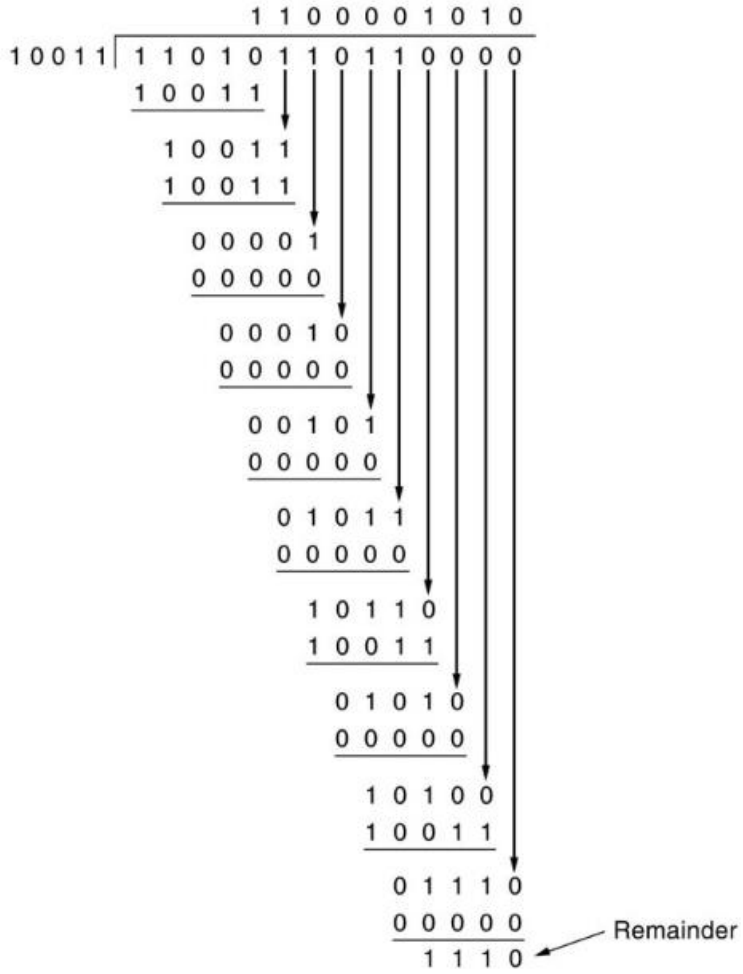
- The transmitted code word is received.
- The received code word is divided with the same CRC generator.
- On division, the remainder so obtained is checked.

Problems:

A bit stream 1101011011 is transmitted using the standard CRC method. The generator polynomial is x4+x+1. What is the actual bit string transmitted?

The generator polynomial G(x) = x4 + x + 1 is encoded as 10011.

Clearly, the generator polynomial consists of 5 bits. So, a string of 4 zeroes is appended to the bit stream to be transmitted. The resulting bit stream is 11010110110000.



Now,

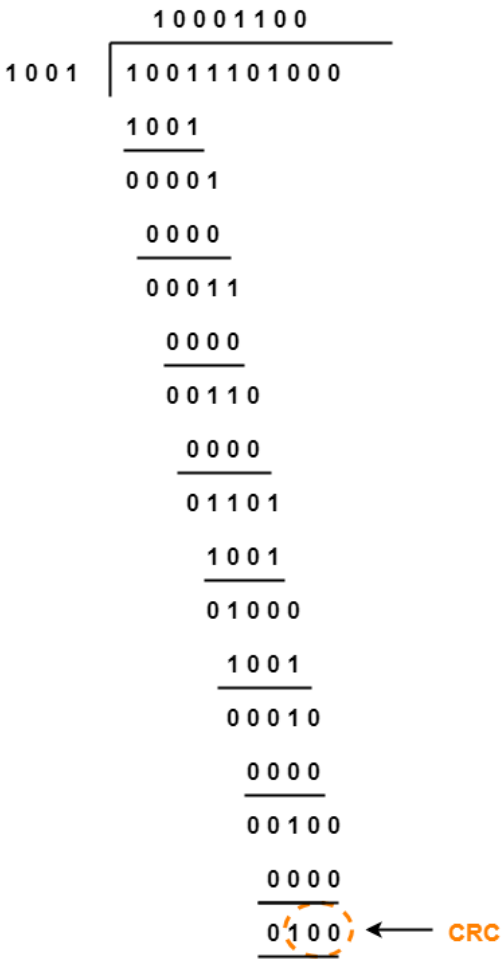
- The code word to be transmitted is obtained by replacing the last 4 zeroes of 11010110110000 with the CRC.
- Thus, the code word transmitted to the receiver = 11010110111110.

A bit stream 10011101 is transmitted using the standard CRC method. The generator polynomial is x3+1.

1. What is the actual bit string transmitted?

The generator polynomial G(x) = x3 + 1 is encoded as 1001.
Clearly, the generator polynomial consists of 4 bits.
So, a string of 3 zeroes is appended to the bit stream to be transmitted.
The resulting bit stream is 10011101000.

Now, the binary division is performed as



From here, CRC = 100.

Now,

- The code word to be transmitted is obtained by replacing the last 3 zeroes of 10011101000 with the CRC.
- Thus, the code word transmitted to the receiver = 10011101100.

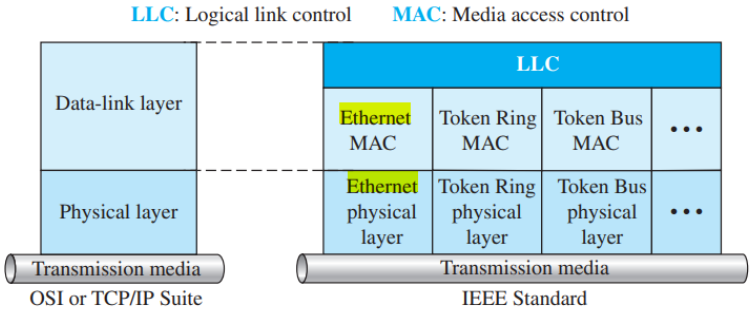
Ethernet Protocol:

- In 1985, the Computer Society of the IEEE started a project, called Project 802, to set standards to enable intercommunication among equipment from a variety of manufacturers
- Project 802 does not seek to replace any part of the OSI model or TCP/IP protocol suite. Instead, it is a way of specifying functions of the physical layer and the data-link layer of major LAN protocols
- The IEEE has subdivided the data-link layer into two sub layers: logical link control (LLC) and media access control (MAC). IEEE has also created several physical-layer standards for different LAN protocols

Logical Link Control (LLC)

- The data link control handles framing, flow control, and error control.

- In IEEE Project 802, flow control, error control, and part of the framing duties are collected into one sub layer called the logical link control (LLC).
- Framing is handled in both the LLC sub layer and the MAC sub layer.
- The LLC provides a single link-layer control protocol for all IEEE LANs.

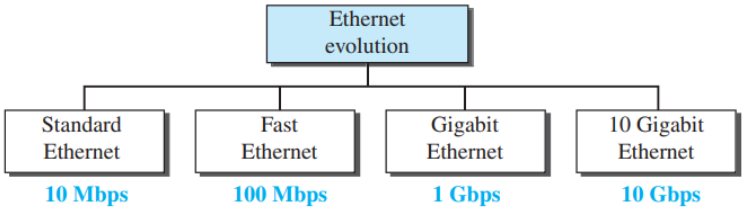


Media Access Control (MAC)

- IEEE Project 802 has created a sub layer called media access control that defines the specific access method for each LAN.

Ethernet Evolution

- The Ethernet LAN was developed in the 1970s by Robert Metcalfe and David Boggs.
- It has gone through four generations: Standard Ethernet (10 Mbps), Fast Ethernet (100 Mbps), Gigabit Ethernet (1 Gbps), and 10 Gigabit Ethernet (10 Gbps),



Standard Ethernet:

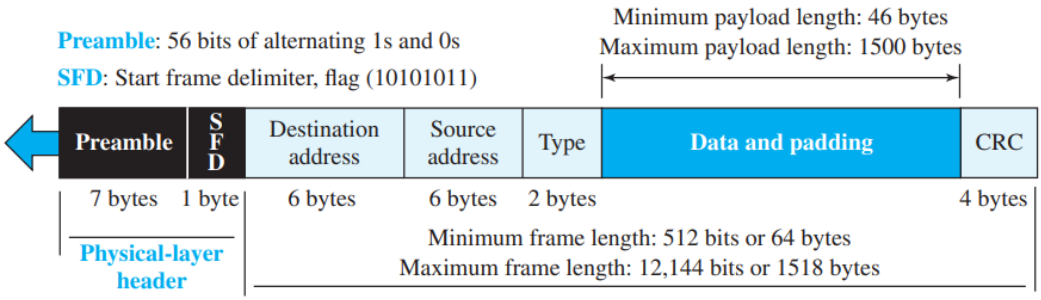
The original Ethernet technology with the data rate of 10 Mbps as the Standard Ethernet

Connectionless and Unreliable Service

- Ethernet provides a connectionless service, which means each frame sent is independent of the previous or next frame.
- Ethernet has no connection establishment or connection termination phases.
- The sender sends a frame whenever it has it; the receiver may or may not be ready for it.
- The sender may overwhelm the receiver with frames, which may result in dropping frames. If a frame drops, the sender will not know about it.
- If the transport layer is also a connectionless protocol, such as UDP, the frame is lost and salvation may only come from the application layer.
- However, if the transport layer is TCP, the sender TCP does not receive acknowledgment for its segment and sends it again.
- Ethernet is also unreliable like IP and UDP.

Frame Format

The Ethernet frame contains **seven fields**



Preamble

- This field contains 7 bytes (56 bits) of alternating 0s and 1s that alert the receiving system to the coming frame and enable it to synchronize its clock if it's out of synchronization.
- The pattern provides only an alert and a timing pulse.
- The 56-bit pattern allows the stations to miss some bits at the beginning of the frame.
- The preamble is actually added at the physical layer and is not (formally) part of the frame

Start frame delimiter (SFD)

- This field (1 byte: 10101011) signals the beginning of the frame.
- The SFD warns the station or stations that this is the last chance for synchronization.
- The last 2 bits are (11)2 and alert the receiver that the next field is the destination address.
- This field is actually a flag that defines the beginning of the frame.
- It needs to remember that an Ethernet frame is a variable-length frame.
- It needs a flag to define the beginning of the frame. The SFD field is also added at the physical layer.

Destination address (DA):

- This field is six bytes (48 bits) and contains the linklayer address of the destination station or stations to receive the packet.
- When the receiver sees its own link-layer address, or a multicast address for a group that the receiver is a member of, or a broadcast address, it decapsulates the data from the frame and passes the data to the upper layer protocol defined by the value of the type field.

Source address (SA).

- This field is also six bytes and contains the link-layer address of the sender of the packet.

Type.

- This field defines the upper-layer protocol whose packet is encapsulated in the frame.
- It is used for multiplexing and demultiplexing.

Data

- This field carries data encapsulated from the upper-layer protocols.
- It is a minimum of 46 and a maximum of 1500 bytes.
- If the data coming from the upper layer is more than 1500 bytes, it should be fragmented and encapsulated in more than one frame.
- If it is less than 46 bytes, it needs to be padded with extra 0s. A padded data frame is delivered to the upper-layer protocol as it is (without removing the padding), which means that it is the responsibility of the upper layer to remove or, in the case of the sender, to add the padding.
- The upper-layer protocol needs to know the length of its data.

Addressing

- Each station on an Ethernet network has its own network interface card (NIC).
- The NIC fits inside the station and provides the station with a link-layer address.
- The Ethernet address is 6 bytes (48 bits), normally written in hexadecimal notation, with a colon between the bytes.

4A:30:10:21:10:1A

Access Method

- The standard Ethernet chose CSMA/CD with 1-persistent method
- Assume station A has a frame to send to station D.
- Station A first should check whether any other station is sending (carrier sense).
- Station A measures the level of energy on the medium (for a short period of time, normally less than 100 µs).
- If there is no signal energy on the medium, it means that no station is sending (or the signal has not reached station A).
- Station A interprets this situation as idle medium.
- It starts sending its frame.
- On the other hand, if the signal energy level is not zero, it means that the medium is being used by another station.
- Station A continuously monitors the medium until it becomes idle for 100 µs. It then starts sending the frame.
- However, station A needs to keep a copy of the frame in its buffer until it is sure that there is no collision.
- The medium sensing does not stop after station A has started sending the frame. Station A needs to send and receive continuously.

Efficiency of standard Ethernet:

The efficiency of the Ethernet is defined as the ratio of the time used by a station to send data to the time the medium is occupied by this station.

$$\text{Efficiency} = 1 / (1 + 6.4 \times a)$$

"a" is the number of frames that can fit on the medium. It can be calculated as

a = (propagation delay)/(transmission delay)

In the Standard Ethernet with the transmission rate of 10 Mbps, we assume that the length of the medium is 2500 m and the size of the frame is 512 bits. The propagation speed of a signal in a cable is normally 2×10^8 m/s.

Propagation delay = $2500/(2 \times 10^8) = 12.5$ µs	**Transmission delay = $512/(10^7) = 51.2$ µs**
$a = 12.5/51.2 = 0.24$	**Efficiency = 39%**

Wireless LAN:

- Wireless communication is one of the fastest-growing technologies. The demand for connecting devices without the use of cables is increasing everywhere. Wireless LANs can be found on college campuses, in office buildings, and in many public areas.

Architectural Comparison:

Activity/Category	Wireless Network	Wired Network
Freedom of movement for users	Users can access network from anywhere within range.	Users location limited by need to use cable and/or connect to a port.
Sharing Files	Easier with wireless network as you do not need to be cabled to network, though transfer speeds may be slower.	Generally less convenient as you have to be cabled in, but transfer speeds often faster.
Cables	Far less complicated, disruptive, and untidy cabling needed.	Lots of cables and ports needed which can be a headache.
Business	For businesses dealing with public, customers like and often expect wireless, so wireless can increase income.	Wired networks are not convenient for public use, but sometimes acceptable for a traditional office.
Connection speeds	Usually slower than wired.	Usually faster than wireless.
Security	Less secure than wired. Both bandwidth and information can sometimes be accessed.	More secure than wireless.
Set up	Upgrading to a wireless network can be difficult and expensive.	Can also be difficult and expensive to set up.

Characteristics of Wireless LAN:

Attenuation

- The strength of electromagnetic signals decreases rapidly because the signal disperses in all directions; only a small portion of it reaches the receiver. The situation becomes worse with mobile senders that operate on batteries and normally have small power supplies.

Interference

- A receiver may receive signals not only from the intended sender, but also from other senders if they are using the same frequency band.

Multipath Propagation

- A receiver may receive more than one signal from the same sender because electromagnetic waves can be reflected back from obstacles such as walls, the ground, or objects.
- The result is that the receiver receives some signals at different phases (because they travel different paths). This makes the signal less recognizable

Error

- The errors and error detection are more serious issues in a wireless network than in a wired network.
- Error level is measured using signal-to-noise ratio (SNR). If SNR is high, it means that the signal is stronger than the noise (unwanted signal), so we may be able to convert the signal to actual data.
- On the other hand, when SNR is low, it means that the signal is corrupted by the noise and the data cannot be recovered.

Access Control

- The Standard Ethernet uses the CSMA/CD algorithm. In this method, each host contends to access the medium and sends its frame if it finds the medium idle.
- If a collision occurs, it is detected and the frame is sent again. Collision detection in CSMA/CD serves two purposes. If a collision is detected, it means that the frame has not been received and needs to be resent. If a collision is not detected, it is a kind of acknowledgment that the frame was received.

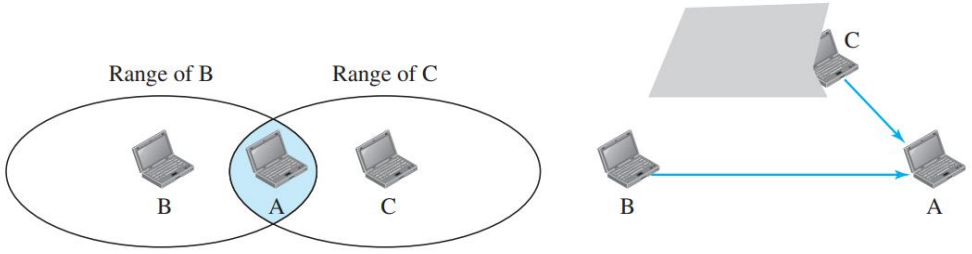
The CSMA/CD algorithm does not work in wireless LANs for three reasons:

- To detect a collision, a host needs to send and receive at the same time, which means the host needs to work in a duplex mode. Wireless hosts do not have enough power to do so (the power is supplied by batteries). They can only send or receive at one time
- Because of the hidden station problem, in which a station may not be aware of another station's transmission due to some obstacles or range problems, collision may occur but not be detected

  Example:

  Consider the circuit shown in Figure. Station B has a transmission range shown by the left oval (sphere in space); every station in this range can hear any signal transmitted by station B. Station C has a transmission range shown by the right oval (sphere in space); every station located in this range can hear any signal transmitted by C.

  Station C is outside the transmission range of B; likewise, station B is outside the transmission range of C. Station A, however, is in the area covered by both B and C; it can hear any signal transmitted by B or C. The figure also shows that the hidden station problem may also occur due to an obstacle.



a. Stations B and C are not in each other's range.

b. Stations B and C are hidden from each other.

  Assume that station B is sending data to station A. In the middle of this transmission, station C also has data to send to station A. However, station C is out of B's range and transmissions from B cannot reach C. Therefore C thinks the medium is free. Station C sends its data to A, which results in a collision at A because this station is receiving data from both B and C. In this case, we say that stations B and C are hidden from each other with respect to A. Hidden stations can reduce the capacity of the network because of the possibility of collision
- The distance between stations can be great. Signal fading could prevent a station at one end from hearing a collision at the other end.

IEEE 802.11 PROJECT(or) WIRELESS ETHERNET:

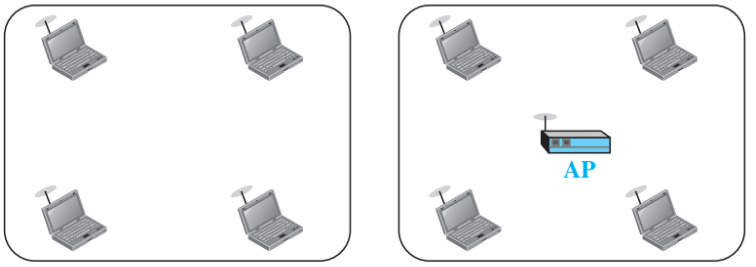
- IEEE has defined the specifications for a wireless LAN, called IEEE 802.11, which covers the physical and data-link layers
- The term WiFi (Wireless Fidelity) as a synonym for wireless LAN

Architecture

- The standard defines two kinds of services: the basic service set (BSS) and the extended service set (ESS).

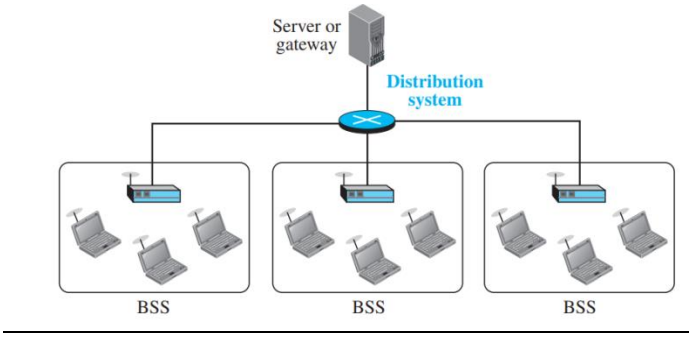
Basic Service Set :

- A basic service set is made of stationary or mobile wireless stations and an optional central base station, known as the access point (AP).
- The BSS without an AP is a stand-alone network and cannot send data to other BSSs. It is called an **ad hoc architecture.**
- A BSS with an AP is sometimes referred to as an **Infrastructure BSS**



Extended Service Set

- An extended service set (ESS) is made up of two or more BSSs with APs.
- In this case, the BSSs are connected through a distribution system, which is a wired or a wireless network.
- The distribution system connects the APs in the BSSs.
- IEEE 802.11 does not restrict the distribution system; it can be any IEEE LAN such as an Ethernet.
- The mobile stations are normal stations inside a BSS.
- The stationary stations are AP stations that are part of a wired LAN.



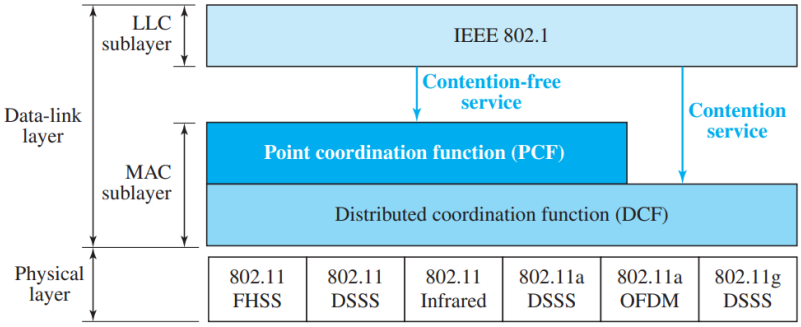
Station Types

IEEE 802.11 defines three types of stations based on their mobility in a wireless LAN: no-transition, BSS-transition, and ESS-transition mobility.

- A station with no-transition mobility is either stationary (not moving) or moving only inside a BSS.
- A station with BSS-transition mobility can move from one BSS to another, but the movement is confined inside one ESS.
- A station with ESS-transition mobility can move from one ESS to another

MAC Sublayer:

IEEE 802.11 defines two MAC sublayers: the distributed coordination function (DCF) and point coordination function (PCF).



Distributed Coordination Function:

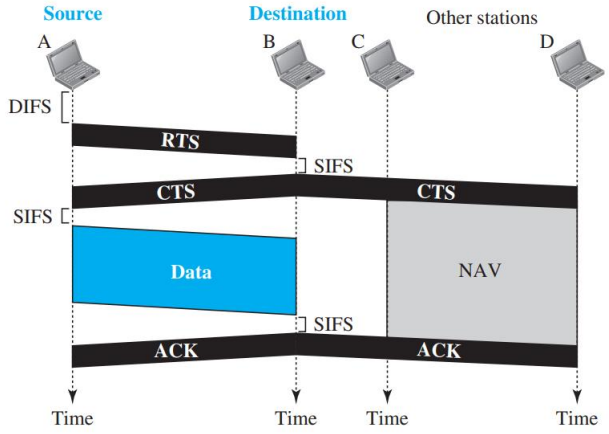
- One of the two protocols defined by IEEE at the MAC sub layer is called the distributed coordination function (DCF).
- DCF uses CSMA/CA as the access method

Frame Exchange Time Line:

1. Before sending a frame, the source station senses the medium by checking the energy level at the carrier frequency.

a. The channel uses a persistence strategy with backoff until the channel is idle.

b. After the station is found to be idle, the station waits for a period of time called the distributed interframe space (DIFS); then the station sends a control frame called the request to send (RTS).

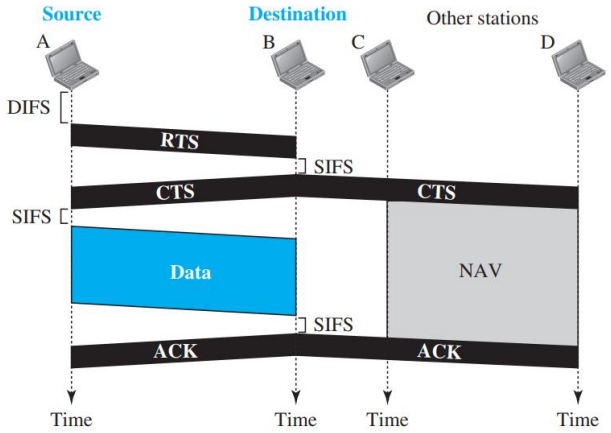


2. After receiving the RTS and waiting a period of time called the short interframe space (SIFS), the destination station sends a control frame, called the clear to send (CTS), to the source station. This control frame indicates that the destination station is ready to receive data.

3. The source station sends data after waiting an amount of time equal to SIFS.

4. The destination station, after waiting an amount of time equal to SIFS, sends an acknowledgment to show that the frame has been received. Acknowledgment is needed in this protocol because the station does not have any means to check for the successful arrival of its data at the destination. On the other hand, the lack of collision in CSMA/CD is a kind of indication to the source that data have arrived.

When a station sends an RTS frame, it includes the duration of time that it needs to occupy the channel. The stations that are affected by this transmission create a timer called a **Network Allocation Vector (NAV) that shows how much time must pass before these stations are allowed to check the channel for idleness.**

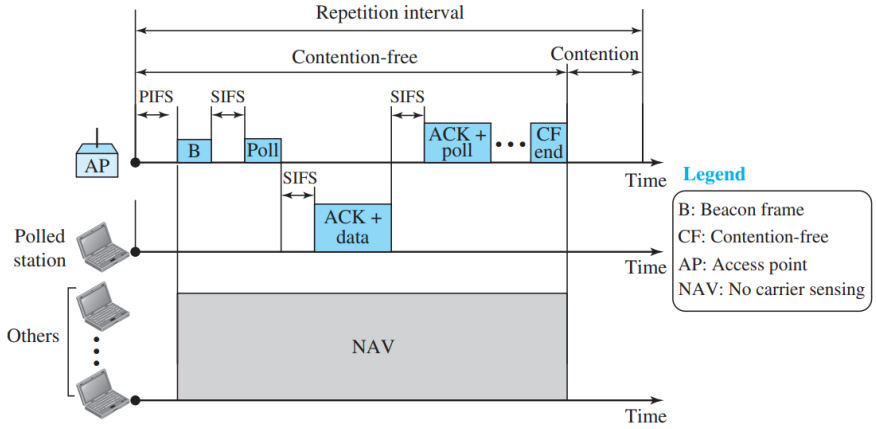


Collision During Handshaking

- Two or more stations may try to send RTS frames at the same time. These control frames may collide. However, because there is no mechanism for collision detection, the sender assumes there has been a collision if it has not received a CTS frame from the receiver.

Point Coordination Function (PCF):

- The point coordination function (PCF) is an optional access method that can be implemented in an infrastructure network
- It is implemented on top of the DCF and is used mostly for time-sensitive transmission.
- The AP performs polling for stations that are capable of being polled. The stations are polled one after another, sending any data they have to the AP.

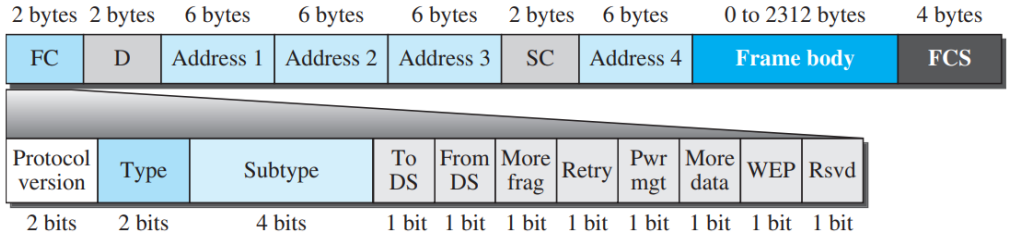


- To give priority to PCF over DCF, another interframe space, PIFS, has been defined.
- PIFS (PCF IFS) is shorter than DIFS.
- Due to the priority of PCF over DCF, stations that only use DCF may not gain access to the medium.
- To prevent this, a repetition interval has been designed to cover both contention-free PCF and contention-based DCF traffic. The repetition interval, which is repeated continuously, starts with a special control frame, called a beacon frame.
- When the stations hear the beacon frame, they start their NAV for the duration of the contention-free period of the repetition interval

- At the end of the contention-free period, the PC sends a CF end (contention-free end) frame to allow the contention-based stations to use the medium.

Frame Format

The MAC layer frame consists of nine fields



Frame control (FC).

The FC field is 2 bytes long and defines the type of frame and some control information.

Field	*Explanation*
Version	Current version is 0
Type	Type of information: management (00), control (01), or data (10)
Subtype	Subtype of each type (see Table 15.2)
To DS	Defined later
From DS	Defined later
More frag	When set to 1, means more fragments
Retry	When set to 1, means retransmitted frame
Pwr mgt	When set to 1, means station is in power management mode
More data	When set to 1, means station has more data to send
WEP	Wired equivalent privacy (encryption implemented)
Rsvd	Reserved

D.This field defines the duration of the transmission that is used to set the value of NAV. In one control frame, it defines the ID of the frame.

Addresses. There are four address fields, each 6 bytes long.

Sequence control. This field, often called the SC field, defines a 16-bit value. The first four bits define the fragment number; the last 12 bits define the sequence number, which is the same in all fragments.

Frame body. This field, which can be between 0 and 2312 bytes, contains information based on the type and the subtype defined in the FC field.

FCS. The FCS field is 4 bytes long and contains a CRC-32 error-detection sequence

Frame Types

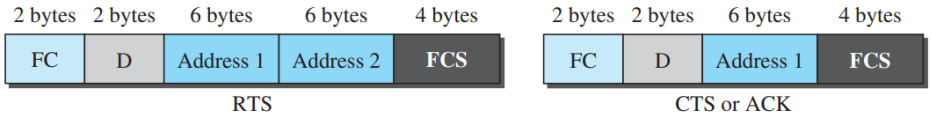
- A wireless LAN defined by IEEE 802.11 has three categories of frames: management frames, control frames, and data frames.

Management Frames

Management frames are used for the initial communication between stations and access points.

Control Frames

Control frames are used for accessing the channel and acknowledging frames.



Data Frames

Data frames are used for carrying data and control information.

BLUETOOTH

- Bluetooth is a wireless LAN technology designed to connect devices of different functions such as telephones, notebooks, computers ,cameras, printers, and even coffee makers when they are at a short distance from each other.
- A Bluetooth LAN is an ad hoc network, which means that the network is formed spontaneously; the devices, sometimes called gadgets, find each other and make a network called a piconet.
- A Bluetooth LAN can even be connected to the Internet if one of the gadgets has this capability. A Bluetooth LAN, by nature, cannot be large.

Applications:

- Peripheral devices such as a wireless mouse or keyboard can communicate with the computer through this technology.
- Monitoring devices can communicate with sensor devices in a small health care center.
- Home security devices can use this technology to connect different sensors to the main security controller.
- Conference attendees can synchronize their laptop computers at a conference.

Bluetooth was originally started as a project by the Ericsson Company. It is named for Harald Blaatand, the king of Denmark (940-981) who united Denmark and Norway.

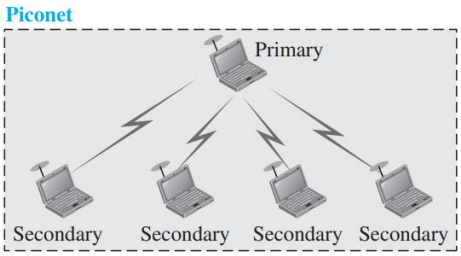
Today, **Bluetooth technology is the implementation of a protocol defined by the IEEE 802.15 standard.** The standard defines a wireless personal-area network (PAN) operable in an area the size of a room or a hall.

Architecture

Bluetooth defines two types of networks: piconet and scatternet.

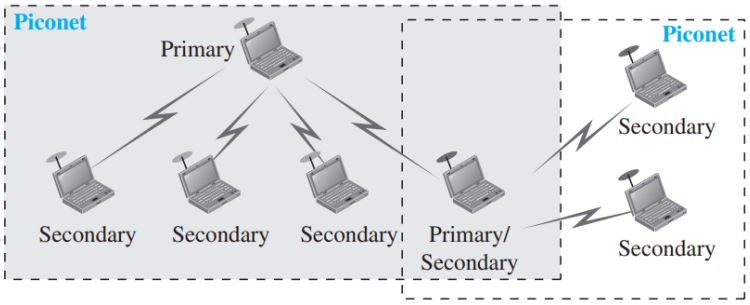
Piconets :

- A Bluetooth network is called a piconet, or a small net.
- A piconet can have up to eight stations, one of which is called the primary; the rest are called secondaries.
- All the secondary stations synchronize their clocks and hopping sequence with the primary.
- A piconet can have only one primary station.
- The communication between the primary and secondary stations can be one-to-one or one-to-many
- Although a piconet can have a maximum of seven secondaries, additional secondaries can be in the parked state.
- A secondary in a parked state is synchronized with the primary, but cannot take part in communication until it is moved from the parked state to the active state



Scatternet:

- Piconets can be combined to form what is called a scatternet.
- A secondary station in one piconet can be the primary in another piconet.
- This station can receive messages from the primary in the first piconet (as a secondary) and, acting as a primary, deliver them to secondaries in the second piconet.
- A station can be a member of two piconets



Bluetooth Layers

L2CAP :The Logical Link Control and Adaptation Protocol, or L2CAP (L2 here means LL), is roughly equivalent to the LLC sublayer in LANs. It is used for data exchange on an ACL link; SCO channels do not use L2CAP

- The 16-bit length field defines the size of the data, in bytes, coming from the upper layers. Data can be up to 65,535 bytes.
- The channel ID (CID) defines a unique identifier for the virtual channel created at this level.
- The L2CAP has specific duties: multiplexing, segmentation and reassembly, quality of service (QoS), and group management.

TDMA:

- Bluetooth uses a form of TDMA that is called TDD-TDMA (time-division duplex TDMA).
- TDD-TDMA is a kind of half-duplex communication in which the sender and receiver send and receive data, but not at the same time (half-duplex)
- The communication for each direction uses different hops. This is similar to walkie-talkies using different carrier frequencies

Links

Two types of links can be created between a primary and a secondary: SCO links and ACL links.

SCO:

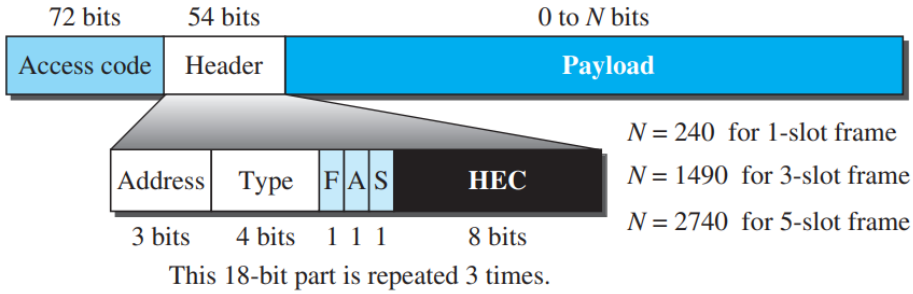
- A synchronous connection-oriented (SCO) link is used when avoiding latency (delay in data delivery) is more important than integrity (error-free delivery).
- In an SCO link, a physical link is created between the primary and a secondary by reserving specific slots at regular intervals.
- The basic unit of connection is two slots, one for each direction. If a packet is damaged, it is never retransmitted

ACL:

- An asynchronous connectionless link (ACL) is used when data integrity is more important than avoiding latency.
- In this type of link, if a payload encapsulated in the frame is corrupted, it is retransmitted.
- A secondary returns an ACL frame in the available odd-numbered slot if the previous slot has been addressed to it. ACL can use one, three, or more slots and can achieve a maximum data rate of 721 kbps

Frame Format

- A frame in the baseband layer can be one of three types: **one-slot, three-slot, or fiveslot**. A slot, as we said before, is 625 µs.
- In a one-slot frame exchange, 259 µs is needed for hopping and control mechanisms. This means that a one-slot frame can last only 625 − 259, or 366 µs.
- With a 1-MHz bandwidth and 1 bit/Hz, the size of a oneslot frame is 366 bits.
- A three-slot frame occupies three slots. However, since 259 µs is used for hopping, the length of the frame is 3 × 625 − 259 = 1616 µs or 1616 bits. A device that uses a three-slot frame remains at the same hop for three slots.
- A five-slot frame also uses 259 bits for hopping, which means that the length of the frame is 5 × 625 − 259 = 2866 bits.



Access code. This 72-bit field normally contains synchronization bits and the identifier of the primary to distinguish the frame of one piconet from that of another.

Header. This 54-bit field is a repeated 18-bit pattern. Each pattern has the following subfields:

a. **Address-**The 3-bit address subfield can define up to seven secondary (1 to 7). If the address is zero, it is used for broadcast communication from the primary to all secondaries.
b. **Type-**The 4-bit type subfield defines the type of data coming from the upper layers.
c. **F.** This 1-bit subfield is for flow control. When set (1), it indicates that the device is unable to receive more frames (buffer is full).
d. **A.** This 1-bit subfield is for acknowledgment. Bluetooth uses Stop-and-Wait ARQ; 1 bit is sufficient for acknowledgment.

e. **S**. This 1-bit subfield holds a sequence number. Bluetooth uses Stop-and-Wait ARQ; 1 bit is sufficient for sequence numbering.

f. **HEC.** The 8-bit header error correction subfield is a checksum to detect errors in each 18-bit header section. The header has three identical 18-bit sections. The receiver compares these three sections, bit by bit. If each of the corresponding bits is the same, the bit is accepted; if not, the majority opinion rules. This is a form of forward error correction. This double error control is needed because the nature of the communication, via air, is very noisy. Note that there is no retransmission in this sublayer.

Payload. This subfield can be 0 to 2740 bits long. It contains data or control information coming from the upper layers.

Band Bluetooth uses a 2.4-GHz ISM band divided into 79 channels of 1 MHz each.

FHSS Bluetooth uses the frequency-hopping spread spectrum (FHSS) method in the physical layer to avoid interference from other devices or other networks

Modulation To transform bits to a signal, Bluetooth uses a sophisticated version of FSK, called GFSK (FSK with Gaussian bandwidth filtering)

Flow and Error Control Protocols:

The Data Link Control (DLC) protocol can be either connectionless or connection-oriented.

Connectionless Protocol:

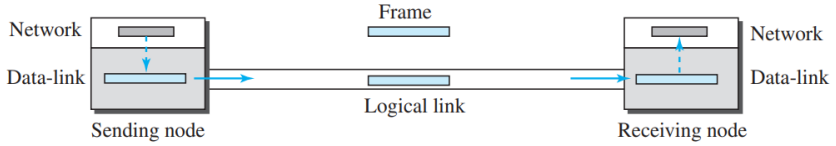
- In a connectionless protocol, frames are sent from one node to the next without any relationship between the frames; each frame is independent.
- The frames are not numbered and there is no sense of ordering.
- Most of the data-link protocols for LANs are connectionless protocols.

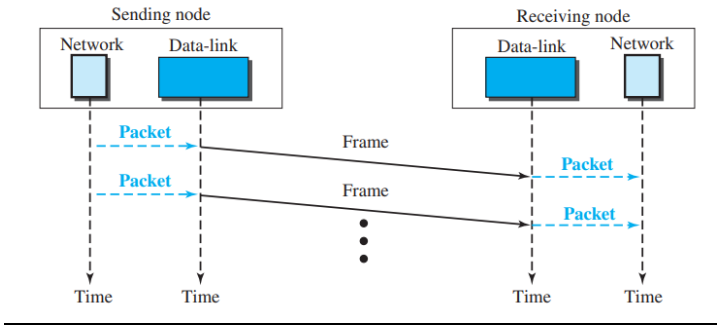
Connection-Oriented Protocol

- In a connection-oriented protocol, a logical connection should first be established between the two nodes (setup phase).
- After all frames that are somehow related to each other are transmitted (transfer phase), the logical connection is terminated (teardown phase).
- In this type of communication, the frames are numbered and sent in order.
- If they are not received in order, the receiver needs to wait until all frames belonging to the same set are received and then deliver them in order to the network layer.
- Connectionoriented protocols are rare in wired LANs, but we can see them in some point-to-point protocols, some wireless LANs, and some WANs.

Simple Protocol

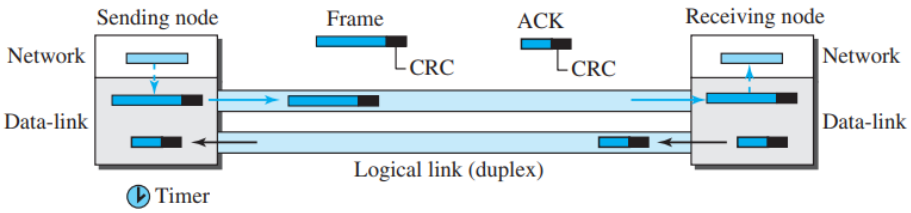
- First protocol is a simple protocol with neither flow nor error control.
- The receiver can immediately handle any frame it receives. In other words, the receiver can never be overwhelmed with incoming frames
- The sender site should not send a frame until its network layer has a message to send.
- The receiver site cannot deliver a message to its network layer until a frame arrives.





Stop-and-Wait Protocol

- Our second protocol is called the Stop-and-Wait protocol, which uses both flow and error control
- In this protocol, the sender sends one frame at a time and waits for an acknowledgment before sending the next one.
- To detect corrupted frames, we need to add a CRC to each data frame.
- When a frame arrives at the receiver site, it is checked. If its CRC is incorrect, the frame is corrupted and silently discarded.
- The silence of the receiver is a signal for the sender that a frame was either corrupted or lost. Every time the sender sends a frame, it starts a timer.
- If an acknowledgment arrives before the timer expires, the timer is stopped and the sender sends the next frame.
- If the timer expires, the sender resends the previous frame, assuming that the frame was either lost or corrupted. This means that the sender needs to keep a copy of the frame until its acknowledgment arrives.
- When the corresponding acknowledgment arrives, the sender discards the copy and sends the next frame if it is ready.



Sender States

The sender is initially in the ready state, but it can move between the ready and blocking state

Ready State:

- When the sender is in this state, it is only waiting for a packet from the network layer. If a packet comes from the network layer, the sender creates a frame, saves a copy of the frame, starts the only timer and sends the frame.
- The sender then moves to the blocking state.

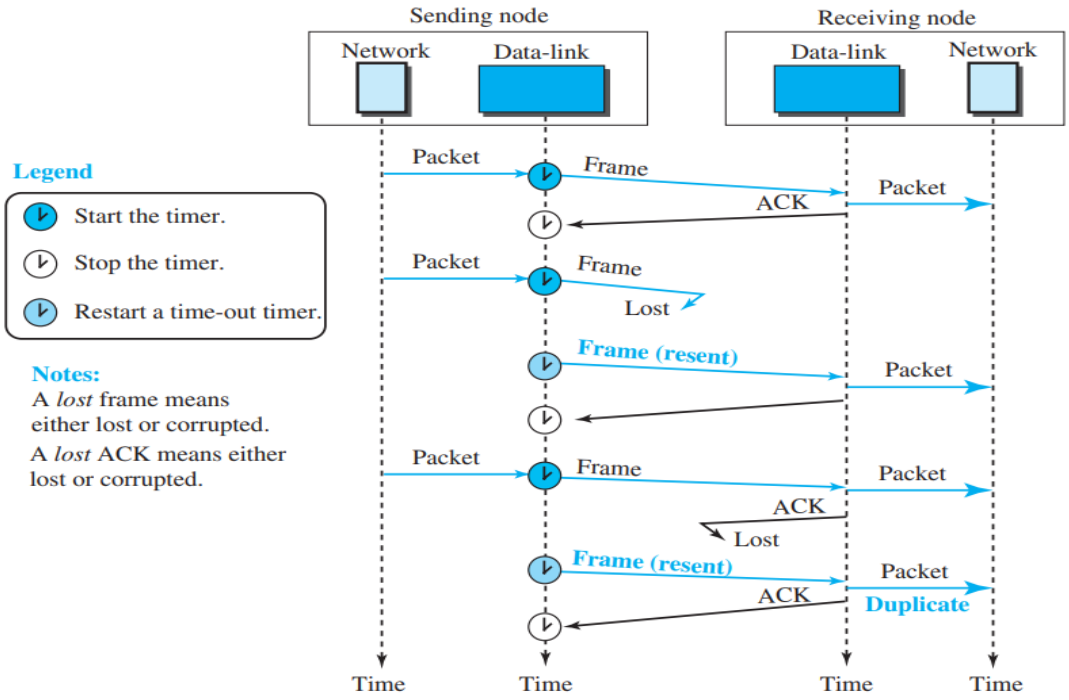
Blocking State:

When the sender is in this state, three events can occur:

a. If a time-out occurs, the sender resends the saved copy of the frame and restarts the timer.

b. If a corrupted ACK arrives, it is discarded.

c. If an error-free ACK arrives, the sender stops the timer and discards the saved copy of the frame. It then moves to the ready state.



Receiver

- The receiver is always in the ready state. Two events may occur:

  a. If an error-free frame arrives, the message in the frame is delivered to the network layer and an ACK is sent.

  b. If a corrupted frame arrives, the frame is discarded.

HDLC (High-level Data Link Control)

HDLC is a bit-oriented protocol for communication over point-to-point and multipoint links.

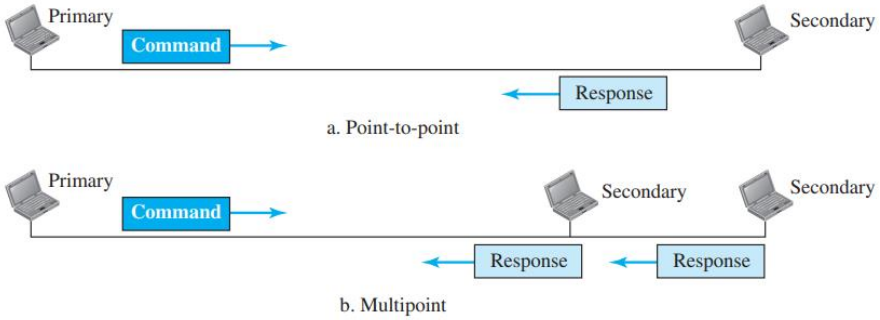
It implements the Stop-and-Wait protocol

Configurations and Transfer Modes

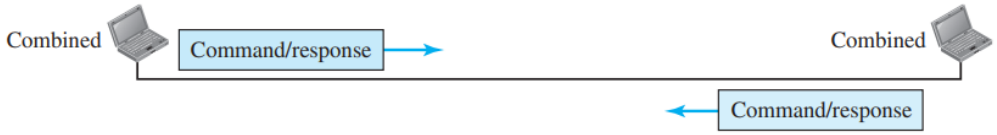
HDLC provides two common transfer modes that can be used in different configurations:

Normal response mode (NRM) and asynchronous balanced mode (ABM).

In normal response mode (NRM), the station configuration is unbalanced. Assume one primary station and multiple secondary stations. A primary station can send commands; a secondary station can only respond. The NRM is used for both point-to-point and multipoint links



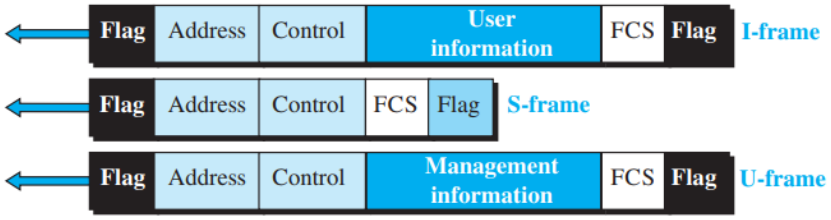
In ABM, the configuration is balanced. The link is point-to-point, and each station can function as a primary and a secondary



Framing

HDLC defines three types of frames:
information frames (I-frames), supervisory frames (S-frames), and
unnumbered frames (U-frames).

- Each type of frame serves as an envelope for the transmission of a different type of message. Iframes are used to data-link user data and control information relating to user data (piggybacking).
- S-frames are used only to transport control information. U-frames are reserved for system management.
- Information carried by U-frames is intended for managing the link itself



Flag field

- This field contains synchronization pattern 01111110, which identifies both the beginning and the end of a frame.

Address field

- This field contains the address of the secondary station. If a primary station created the frame, it contains a to address. If a secondary station creates the frame, it contains a from address. The address field can be one byte or several bytes long, depending on the needs of the network.

Control field.

- The control field is one or two bytes used for flow and error control.

Information field.

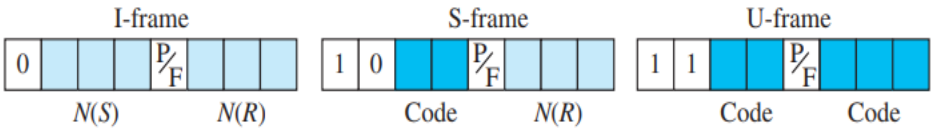
- The information field contains the user's data from the network layer or management information. Its length can vary from one network to another.

FCS field.

- The frame check sequence (FCS) is the HDLC error detection field. It can contain either a 2- or 4-byte CRC.

Control Field:

The control field determines the type of frame and defines its functionality



Control Field for I-Frames:

- I-frames are designed to carry user data from the network layer.
- In addition, they can include flow- and error-control information (piggybacking).
- The subfields in the control field are used to define these functions.
- The first bit defines the type. If the first bit of the control field is 0, this means the frame is an I-frame.
- The next 3 bits, called N(S), define the sequence number of the frame.
- The last 3 bits, called N(R), correspond to the acknowledgment number when piggybacking is used.
- The single bit between N(S) and N(R) is called the P/F bit.
- The P/F field is a single bit with a dual purpose. It has meaning only when it is set (bit = 1) and can mean poll or final.
- It means poll when the frame is sent by a primary station to a secondary (when the address field contains the address of the receiver).
- It means final when the frame is sent by a secondary to a primary (when the address field contains the address of the sender).

Control Field for S-Frames

Supervisory frames are used for flow and error control whenever piggybacking is either impossible or inappropriate.

S-frames do not have information fields.

If the first 2 bits of the control field are 10, this means the frame is an S-frame. The last 3 bits, called N(R), correspond to the acknowledgment number (ACK) or negative acknowledgment number (NAK), depending on the type of S-frame.

The 2 bits called code are used to define the type of S-frame itself.

With 2 bits, there are four types of S-frames possible

- **Receive ready (RR).** If the value of the code subfield is 00, it is an RR S-frame. This kind of frame acknowledges the receipt of a safe and sound frame or group of frames. In this case, the value of the N(R) field defines the acknowledgment number
- **Receive not ready (RNR).** If the value of the code subfield is 10, it is an RNR Sframe. It acknowledges the receipt of a frame or group of frames, and it announces that the receiver is busy and cannot receive more frames. It acts as a kind of congestion-control mechanism by asking the sender to slow down. The value of N(R) is the acknowledgment number.
- **Reject (REJ).** If the value of the code subfield is 01, it is an REJ S-frame. This is a NAK frame, but not like the one used for Selective Repeat ARQ. The value of N(R) is the negative acknowledgment number.
- **Selective reject (SREJ)**. If the value of the code subfield is 11, it is an SREJ Sframe. This is a NAK frame used in Selective Repeat ARQ. The value of N(R) is the negative acknowledgment number.

Control Field for U-Frames

- Unnumbered frames are used to exchange session management and control information between connected devices.
- U-frames contain an information field, but one used for system management information, not user data.
- U-frame codes are divided into two sections: a 2-bit prefix before the P/ F bit and a 3-bit suffix after the P/F bit. Together, these two segments (5 bits) can be used to create up to 32 different types of U-frames.

POINT-TO-POINT PROTOCOL (PPP)

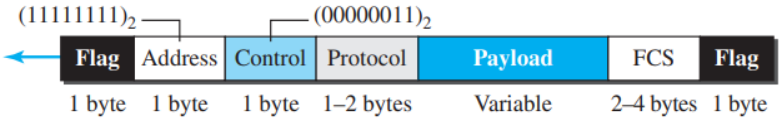
One of the most common protocols for point-to-point access is the Point-to-Point Protocol (PPP). Today, millions of Internet users who need to connect their home computers to the server of an Internet service provider use PPP

Services Provided by PPP:

- PPP defines the format of the frame to be exchanged between devices.
- It also defines how two devices can negotiate the establishment of the link and the exchange of data. PPP is designed to accept payloads from several network layers.
- Authentication is also provided in the protocol, but it is optional.
- The new version of PPP, called Multilink PPP, provides connections over multiple links.

Framing

PPP uses a character-oriented (or byte-oriented) frame.



Address. The address field in this protocol is a constant value and set to 11111111 (broadcast address).

Control. This field is set to the constant value 00000011.PPP does not provide any flow control. Error control is also limited to error detection.

Protocol. The protocol field defines what is being carried in the data field: either user data or other information. This field is by default 2 bytes long, but the two parties can agree to use only 1 byte.

Payload field. This field carries either the user data or other information. The data field is a sequence of bytes with the default of a maximum of 1500 bytes; but this can be changed during negotiation. The data field is byte-stuffed if the flag byte pattern appears in this field. Because there is no field defining the size of the data field, padding is needed if the size is less than the maximum default value or the maximum negotiated value.

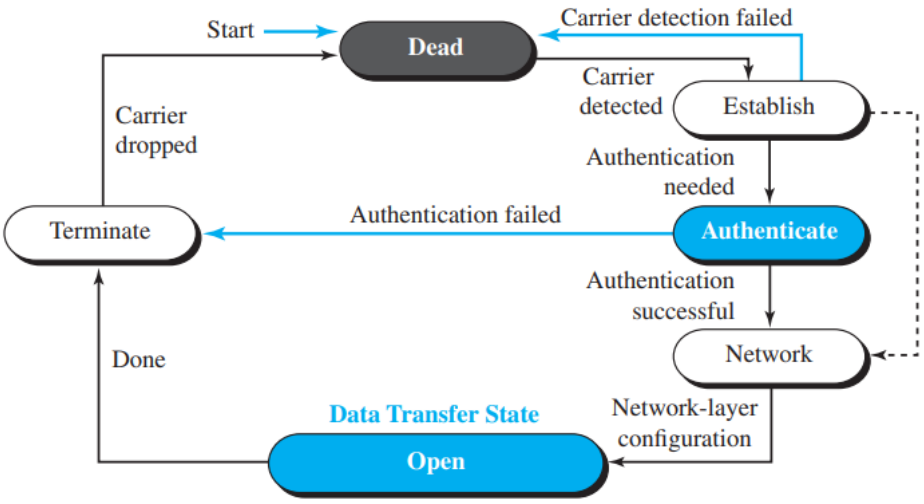
FCS. The frame check sequence (FCS) is simply a 2-byte or 4-byte standard CRC

Byte Stuffing

Since PPP is a byte-oriented protocol, the flag in PPP is a byte that needs to be escaped whenever it appears in the data section of the frame. The escape byte is 01111101, which means that every time the flag like pattern appears in the data, this extra byte is stuffed to tell the receiver that the next byte is not a flag.

Transition Phases

- A PPP connection goes through phases which can be shown in a transition phase diagram
- The transition diagram, which is an FSM, starts with the dead state.
- In this state, there is no active carrier and the line is quiet.
- When one of the two nodes starts the communication, the connection goes into the establish state. In this state, options are negotiated between the two parties.



- If the two parties agree that they need authentication (for example, if they do not know each other), then the system needs to do authentication (an extra step); otherwise, the parties can simply start communication.

EC3401 NETWORKS AND SECURITY

UNIT II NETWORK LAYER PROTOCOLS **9**

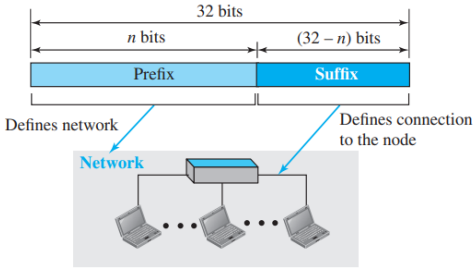
Network Layer – IPv4 Addressing – Network Layer Protocols (IP, ICMP and Mobile IP)- Unicast and Multicast Routing – Intradomain and Interdomain Routing Protocols – IPv6 Addresses – IPv6 – Datagram Format - Transition from IPv4 to IPv6.

Network Layer:

- The network layer is involved at the source host, destination host, and all routers in the path.
- At the source host (Alice), the network layer accepts a packet from a transport layer, encapsulates the packet in a datagram, and delivers the packet to the data-link layer
- At the destination host (Bob), the datagram is decapsulated, and the packet is extracted and delivered to the corresponding transport layer

IPv4 Addressing:

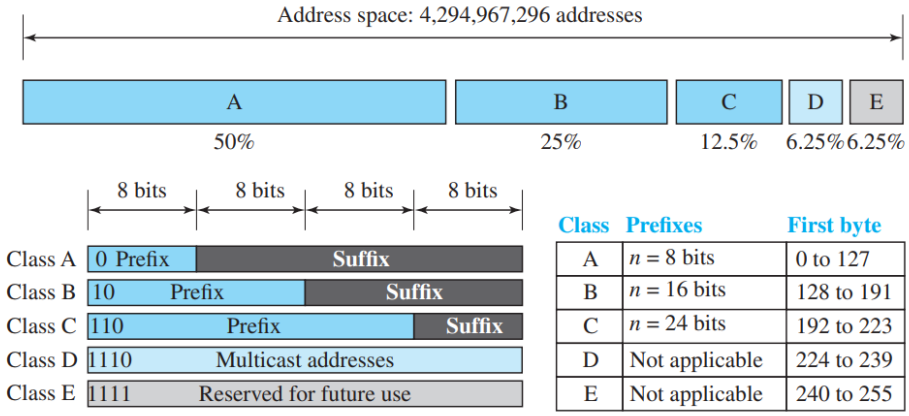
- An IPv4 address is a 32-bit address that uniquely and universally defines the connection of a device (for example, a computer or a router) to the Internet
- IPv4 addresses are unique. They are unique in the sense that each address defines one, and only one, connection to the Internet.
- Two devices on the Internet can never have the same address at the same time
- A 32-bit IPv4 address is also hierarchical, but divided only into two parts. The first part of the address, called the prefix, defines the network; the second part of the address, called the suffix, defines the node (connection of a device to the Internet)
- The prefix length is n bits and the suffix length is (32 − n) bits.
- A prefix can be fixed length or variable length.
- The scheme which uses fixed length prefix is called as classful addressing and the scheme which uses variable-length network prefix is referred to as classless addressing



Classful Addressing

- When the Internet started, an IPv4 address was designed with a fixed-length prefix, but to accommodate both small and large networks, three fixed-length prefixes were designed instead of one (n = 8, n = 16, and n = 24). The whole address space was divided into five classes (class A, B, C, D, and E).This scheme is referred to as classful addressing
- In class A, the network length is 8 bits, but since the first bit, which is 0, defines the class, we can have only seven bits as the network identifier. This means there are only 2^7 = 128 networks in the world that can have a class A address.

- In class B, the network length is 16 bits, but since the first two bits, which are (10)2, define the class, we can have only 14 bits as the network identifier. This means there are only 2^{14} = 16,384 networks in the world that can have a class B address.
- All addresses that start with (110)2 belong to class C. In class C, the network length is 24 bits, but since three bits define the class, we can have only 21 bits as the network identifier. This means there are 2^{21} = 2,097,152 networks in the world that can have a class C address.
- Class D is not divided into prefix and suffix. It is used for multicast addresses. All addresses that start with 1111 in binary belong to class E.
- As in Class D, Class E is not divided into prefix and suffix and is used as reserve

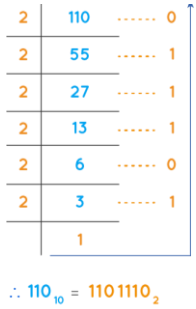


Class	Prefixes	First byte
A	n = 8 bits	0 to 127
B	n = 16 bits	128 to 191
C	n = 24 bits	192 to 223
D	Not applicable	224 to 239
E	Not applicable	240 to 255

Example:

1. Rewrite the following IP addresses using binary notation:

a. 110.11.5.88



Likewise convert everything:

a. 110.11.5.88- 01101110.00001011.00000101.01011000

b. 12.74.16.18

00001100 01001010 00010000 00010010

c. 201.24.44.32

11001001 00011000 00101100 00100000

2. Find the class of the following classful IP addresses:
a.) 130.34.54.12
130 is between 128 and 191 => Class B
b.) 200.34.2.1
200 is between 192 and 223 => Class C
c.) 245.34.2.8
245 is between 240 and 254 => Class E

3.Find the class of each address.
a. 00000001 00001011 00001011 11101111
b. 11000001 10000011 00011011 11111111
c. 14.23.120.8
d. 252.5.15.111
Solution
a. The first bit is O. This is a class A address.
b. The first 2 bits are 1; the third bit is O. This is a class C address.
c. The first byte is 14 (between 0 and 127); the class is A.
d. The first byte is 252 (between 240 and 255); the class is E

Address Depletion:

- In the Internet if the addresses were not distributed properly, the Internet was faced with the problem of the addresses being rapidly used up.
- For an Example consider class A address in this 128 organizations connected and each organization allowed to use with 16,777,216 nodes (2^{32}). Since there may be only a few organizations that are this large, most of the addresses in this class were wasted.
- Class B addresses was designed for midsize organizations, but many of the addresses in this class also remained unused.
- Class C addresses have a completely different flaw in design. The number of addresses that can be used in each network (256) was so small that most companies were not comfortable using a block in this address class. C
- Class E addresses were almost never used, wasting the whole class.

Subnetting and Supernetting:

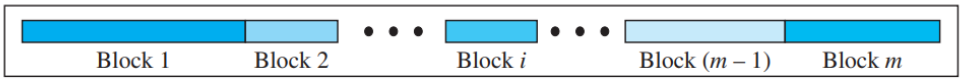
- In subnetting, a class A or class B block is divided into several subnets. Each subnet has a larger prefix length than the original network.
- If a network in class A is divided into four subnets, each subnet has a prefix of nsub = 10. At the same time, if all of the addresses in a network are not used, subnetting allows the addresses to be divided among several organizations.
- This idea did not work because most large organizations were not happy about dividing the block and giving some of the unused addresses to smaller organizations.
- While subnetting was devised to divide a large block into smaller ones, supernetting was devised to combine several class C blocks into a larger block to be attractive

Advantage of Classful Addressing:

- For the given an address, we can easily find the class of the address and, since the prefix length for each class is fixed, we can find the prefix length immediately.

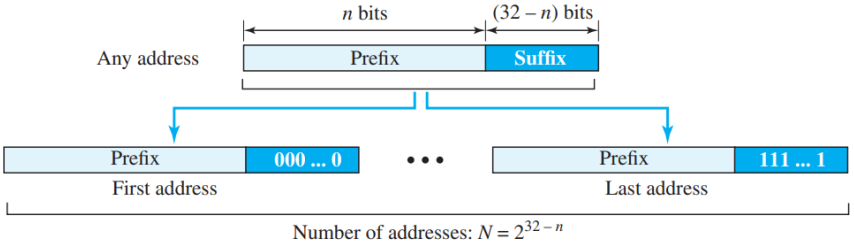
Classless Addressing:

- Subnetting and supernetting in classful addressing did not really solve the address depletion problem. With the growth of the Internet, it was clear that a larger address space was needed as a long-term solution
- In classless addressing, the whole address space is divided into variable length blocks. The prefix in an address defines the block
- Theoretically, there is a block of 2^0, 2^1 , 2^2 , ..., 2^{32} addresses.
- An organization can be granted one block of addresses



Address space

- The prefix length in classless addressing is variable it ranges from 0 to 32
- A small prefix means a larger network; a large prefix means a smaller network.

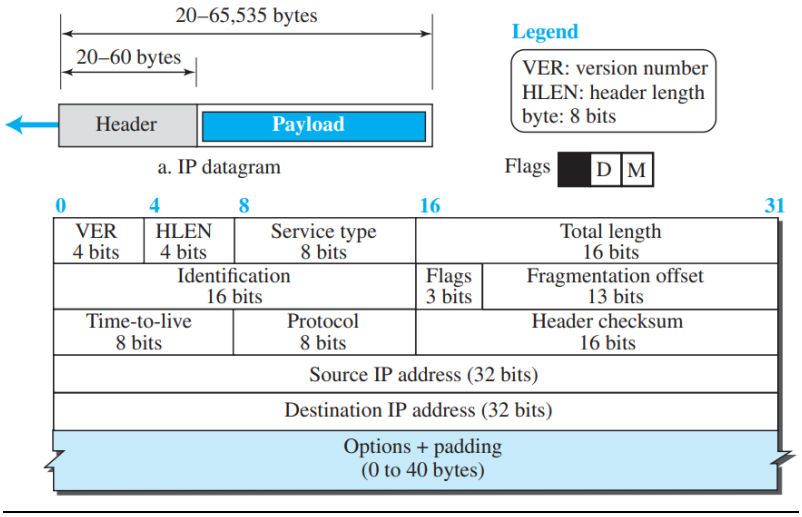


Internet Protocol version 4 (IPv4):

- Internet Protocol version 4 (IPv4), is responsible for packetizing, forwarding, and delivery of a packet at the network layer
- IPv4 is an unreliable datagram protocol—a best-effort delivery service.
- The term best-effort means that IPv4 packets can be corrupted, be lost, arrive out of order, or be delayed, and may create congestion for the network.
- IPv4 is also a connectionless protocol that uses the datagram approach.
- Packets used by the IP are called datagrams.
- Each datagram is handled independently, and each datagram can follow a different route to the destination

Datagram Format:

- IPv4 defines the format of a packet in which the data coming from the upper layer or other protocols are encapsulated
- A datagram is a variable-length packet consisting of two parts: **header and payload (data).** The **header is 20 to 60 bytes** in length and contains information essential to routing and delivery.



Version Number: The 4-bit version number (VER) field defines the version of the IPv4 protocol

Header Length:

- The 4-bit header length (HLEN) field defines the total length of the datagram header in 4-byte words. The IPv4 datagram has a variable-length header.
- When a device receives a datagram, it needs to know when the header stops and the data, which is encapsulated in the packet, starts.
- However, to make the value of the header length (number of bytes) fit in a 4-bit header length, the total length of the header is calculated as 4-byte words.
- The total length is divided by 4 and the value is inserted in the field. The receiver needs to multiply the value of this field by 4 to find the total length.

Service:

- In the original design of the IP header, this field was referred to as type of service (TOS), which defined how the datagram should be handled.

Total Length.

- This 16-bit field defines the total length (header plus data) of the IP datagram in bytes.
- A 16-bit number can define a total length of up to 65,535 (when all bits are 1s). However, the size of the datagram is normally much less than this.
- This field helps the receiving device to know when the packet has completely arrived.
- To find the length of the data coming from the upper layer, subtract the header length from the total length.
- The header length can be found by multiplying the value in the HLEN field by 4.

Length of data = total length − (HLEN) × 4

Identification, Flags, and Fragmentation Offset.

- These three fields are related to the fragmentation of the IP datagram when the size of the datagram is larger than the underlying network can carry.

Time-to-live.

- Due to some malfunctioning of routing protocols a datagram may be circulating in the Internet, visiting some networks over and over without reaching the destination.
- This may create extra traffic in the Internet.

- The time-to-live (TTL) field is used to control the maximum number of hops visited by the datagram.
- When a source host sends the datagram, it stores a number in this field.
- This value is approximately two times the maximum number of routers between any two hosts.
- Each router that processes the datagram decrements this number by one. If this value, after being decremented, is zero, the router discards the datagram

Protocol

- In TCP/IP, the data section of a packet, called the payload, carries the whole packet from another protocol.
- A datagram can also carry a packet from other protocols that directly use the service of the IP, such as some routing protocols or some auxiliary protocols

Header checksum

- IP is not a reliable protocol; it does not check whether the payload carried by a datagram is corrupted during the transmission.
- IP puts the burden of error checking of the payload on the protocol that owns the payload, such as UDP or TCP.
- The datagram header, however, is added by IP, and its error-checking is the responsibility of IP.

Source and Destination Addresses.

- These 32-bit source and destination address fields define the IP address of the source and destination respectively. The source host should know its IP address

Options

- A datagram header can have up to 40 bytes of options. Options can be used for network testing and debugging.
- Although options are not a required part of the IP header, option processing is required of the IP software.

Payload

- Payload, or data, is the main reason for creating a datagram. Payload is the packet coming from other protocols that use the service of IP

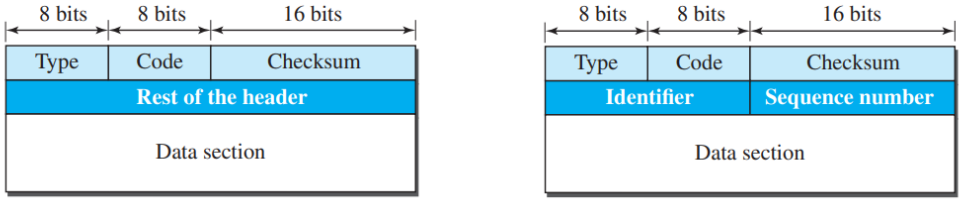
ICMPv4 Internet Control Message Protocol version 4

- The IPv4 has no error-reporting or error-correcting mechanism
- The IP protocol also lacks a mechanism for host and management queries. A host sometimes needs to determine if a router or another host is alive
- The Internet Control Message Protocol version 4 (ICMPv4) has been designed to compensate for the above two deficiencies. It is a companion to the IP protocol.
- It is a companion to the IP protocol. ICMP itself is a network-layer protocol. However, its messages are not passed directly to the data-link layer as would be expected
- When an IP datagram encapsulates an ICMP message, the value of the protocol field in the IP datagram is set to 1 to indicate that the IP payroll is an ICMP message

Messages:

- ICMP messages are divided into two broad categories: **error-reporting messages and query messages.** The error-reporting messages report problems that a router or a host (destination) may encounter when it processes an IP packet.
- The query messages, which occur in pairs, help a host or a network manager get specific information from a router or another host. For example, nodes can discover their neighbors.

- An ICMP message has an 8-byte header and a variable-size data section
- The first field, ICMP type, defines the type of the message. The code field specifies the reason for the particular message type. The last common field is the checksum field. The rest of the header is specific for each message type.



Type and code values

Error-reporting messages	Query messages
03: Destination unreachable (codes 0 to 15)	08 and 00: Echo request and reply (only code 0)
04: Source quench (only code 0)	13 and 14: Timestamp request and reply (only code 0)
05: Redirection (codes 0 to 3)	
11: Time exceeded (codes 0 and 1)	
12: Parameter problem (codes 0 and 1)	

- The data section in error messages carries information for finding the original packet that had the error. In query messages, the data section carries extra information based on the type of query

Error Reporting Messages

- Since IP is an unreliable protocol, one of the main responsibilities of ICMP is to report some errors that may occur during the processing of the IP datagram.
- ICMP does not correct errors, it simply reports them.
- Error correction is left to the higher-level protocols.
- Error messages are always sent to the original source because the only information available in the datagram about the route is the source and destination IP addresses.
- ICMP uses the source IP address to send the error message to the source (originator) of the datagram.
- To make the error-reporting process simple, ICMP follows some rules in reporting messages

Destination Unreachable

- The most widely used error message is the destination unreachable (type 3).
- This message uses different codes (0 to 15) to define the type of error message and the reason why a datagram has not reached its final destination

Source Quench

- It informs the sender that the network has encountered congestion and the datagram has been dropped; the source needs to slow down sending more datagrams

Redirection Message

- The redirection message (type 5) is used when the source uses a wrong router to send out its message.
- The router redirects the message to the appropriate router, but informs the source that it needs to change its default router in the future. The IP address of the default router is sent in the message.

Parameter Problem

- A parameter problem message (type 12) can be sent when either there is a problem in the header of a datagram (code 0) or some options are missing or cannot be interpreted (code 1).

Query Messages

- Query messages in ICMP can be used independently without relation to an IP datagram.
- Query messages are used to probe or test the liveliness of hosts or routers in the Internet, find the one-way or the round-trip time for an IP datagram between two devices, or even find out whether the clocks in two devices are synchronized.
- Naturally, query messages come in pairs: request and reply. The echo request (type 8) and the echo reply (type 0) pair of messages are used by a host or a router to test the liveliness of another host or router.
- A host or router sends an echo request message to another host or router; if the latter is alive, it responds with an echo reply message
- The timestamp request (type 13) and the timestamp reply (type 14) pair of messages are used to find the round-trip time between two devices or to check whether the clocks in two devices are synchronized.
- The timestamp request message sends a 32-bit number, which defines the time the message is sent.
- The timestamp reply resends that number, but also includes two new 32-bit numbers representing the time the request was received and the time the response was sent.

MOBILE IP

- Mobile IP, the extension of IP protocol that allows mobile computers to be connected to the Internet at any location where the connection is possible

Addressing-Stationary Hosts:

- The original IP addressing was based on the assumption that a host is stationary, attached to one specific network.
- A router uses an IP address to route an IP datagram. An IP address has two parts: a prefix and a suffix. The prefix associates a host with a network. For example, the IP address 10.3.4.24/8 defines a host attached to the network 10.0.0.0/8.
- The host in the Internet does not have an address that it can carry with itself from one place to another.
- The address is valid only when the host is attached to the network.
- If the network changes, the address is no longer valid.

Mobile Hosts:

- When a host moves from one network to another, the IP addressing structure needs to be modified. Several solutions have been proposed to solve this issue

Changing the Address:

- One simple solution to address the mobile host problem is to change its address as it goes to the new network.

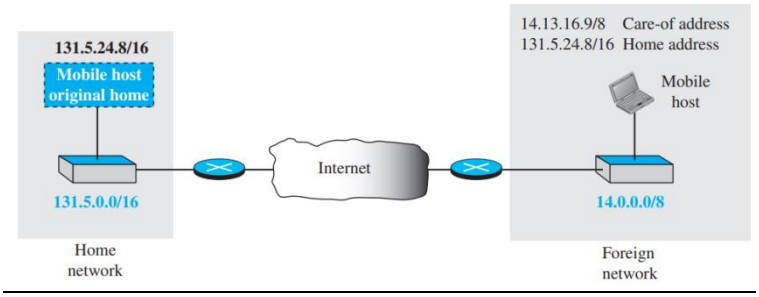
There are many drawbacks with this approach

- The configuration files would need to be changed.
- Each time the computer moves from one network to another, it must be rebooted.
- The DNS tables need to be revised so that every other host in the Internet is aware of the change.

- The host roams from one network to another during a transmission, the data exchange will be interrupted.

Two Addresses

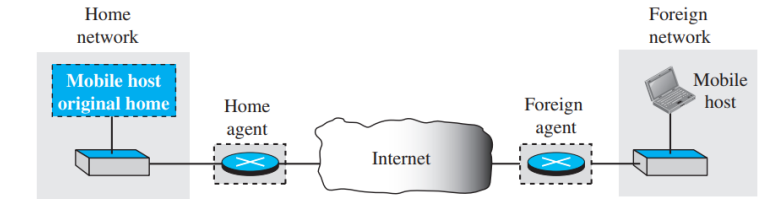
- The approach that is more feasible is the use of two addresses. The host has its original address, called the **home address**, and a temporary address, called the **care-of address**.
- The home address is permanent; it associates the host with its home network, the network that is the permanent home of the host. The care-of address is temporary.
- When a host moves from one network to another, the care-of address changes; it is associated with the foreign network, the network to which the host moves.



- Mobile IP has two addresses for a mobile host: one home address and one care-of address. The home address is permanent; the care-of address changes as the mobile host moves from one network to another

Agents

- To make the change of address transparent to the rest of the Internet requires a home agent and a foreign agent.



Home Agent

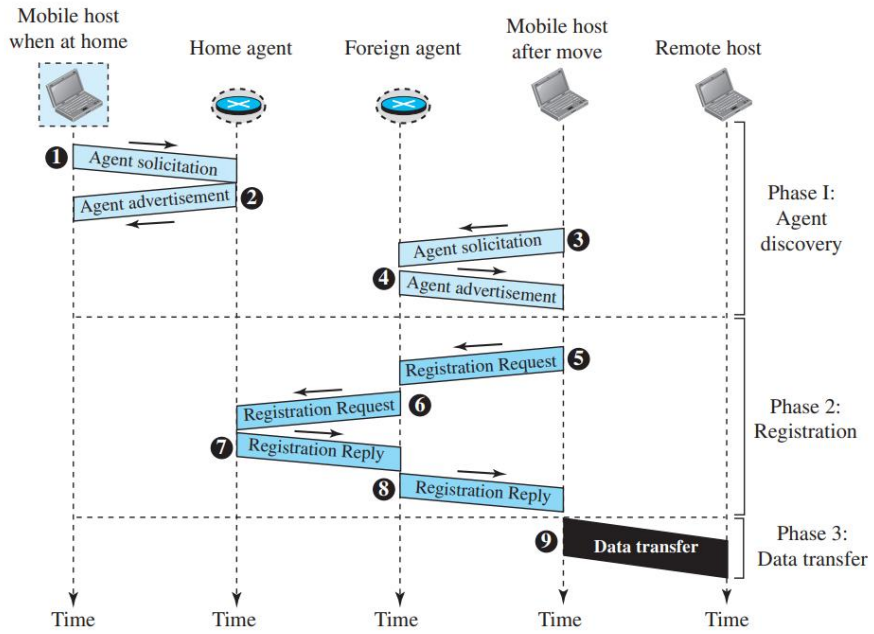
- The home agent is usually a router attached to the home network of the mobile host. The home agent acts on behalf of the mobile host when a remote host sends a packet to the mobile host.
- The home agent receives the packet and sends it to the foreign agent.

Foreign Agent

- The foreign agent is usually a router attached to the foreign network. The foreign agent receives and delivers packets sent by the home agent to the mobile host.
- When the mobile host acts as a foreign agent, the care-of address is called a collocated care-of address.

Three Phases

- To communicate with a remote host, a mobile host goes through three phases: agent discovery, registration, and data transfer



Agent Discovery

- The first phase in mobile communication, agent discovery, consists of two sub phases. A mobile host must discover (learn the address of) a home agent before it leaves its home network. A mobile host must also discover a foreign agent after it has moved to a foreign network

Agent Advertisement

- When a router advertises its presence on a network using an ICMP router advertisement, it can append an agent advertisement to the packet if it acts as an agent

ICMP Advertisement message			
Type	Length	Sequence number	
Lifetime		Code	Reserved
Care-of addresses (foreign agent only)			

Type. The 8-bit type field is set to 16.
Length. The 8-bit length field defines the total length of the extension message (not the length of the ICMP advertisement message).
Sequence number. The 16-bit sequence number field holds the message number. The recipient can use the sequence number to determine if a message is lost.
Lifetime. The lifetime field defines the number of seconds that the agent will accept requests. If the value is a string of 1s, the lifetime is infinite.
Code. The code field is an 8-bit flag in which each bit is set (1) or unset (0).
Care-of Addresses. This field contains a list of addresses available for use as careof addresses

Agent Solicitation

- When a mobile host has moved to a new network and has not received agent advertisements, it can initiate an agent solicitation

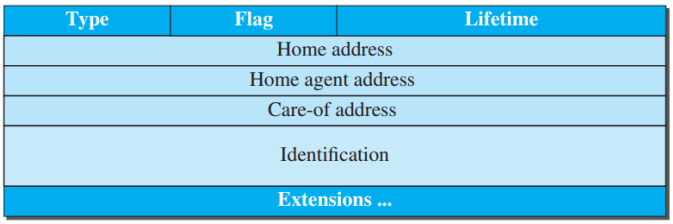
Bit	Meaning
0	Registration required. No collocated care-of address.
1	Agent is busy and does not accept registration at this moment.
2	Agent acts as a home agent.
3	Agent acts as a foreign agent.
4	Agent uses minimal encapsulation.
5	Agent uses generic routing encapsulation (GRE).
6	Agent supports header compression.
7	Unused (0).

Registration

- The second phase in mobile communication is registration. After a mobile host has moved to a foreign network and discovered the foreign agent, it must register.

Registration Request

- A registration request is sent from the mobile host to the foreign agent to register its care-of address and also to announce its home address and home agent address



Lifetime.

This field defines the number of seconds the registration is valid. If the field is a string of 0s, the request message is asking for deregistration. If the field is a string of 1s, the lifetime is infinite.

Home address. This field contains the permanent (first) address of the mobile host.

Home agent address. This field contains the address of the home agent.

Care-of address. This field is the temporary (second) address of the mobile host.

Identification. This field contains a 64-bit number that is inserted into the request by the mobile host and repeated in the reply message. It matches a request with a reply.

Extensions. Variable length extensions are used for authentication

Unicast Routing:

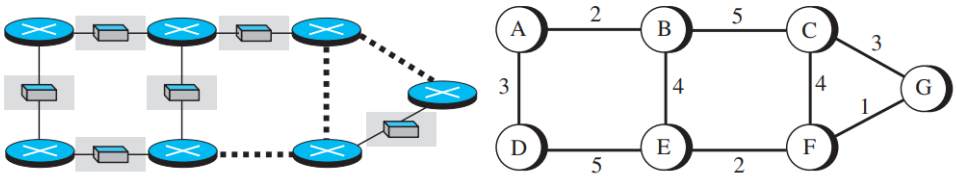
- In unicast routing, a packet is routed, hop by hop, from its source to its destination by the help of forwarding tables.
- The source host needs no forwarding table because it delivers its packet to the default router in its local network.
- The destination host needs no forwarding table because it receives the packet from its default router in its local network.
- Routing a packet from its source to its destination means routing the packet from a source router to a destination router.

Least-Cost Routing

When an internet is modeled as a weighted graph, one of the ways to interpret the best route from the source router to the destination router is to find the least cost between the two.

That is, the source router chooses a route to the destination router in such a way that the total cost for the route is the least cost among all possible routes.

- Consider the figure (below) assume that the best route between A and E is determined. There are two possible routes–one is from A-**D-E with the cost of 8 and the other is A-B-E, with the cost of 6**. Among that the path has **least cost is A-B-E with the cost of 6 has been choosen**.
- This means that each router needs to find the least-cost route between itself and all the other routers to be able to route a packet towards the destination.



Routing Table:

- To route a packet in the network a host or a router has a routing table with an entry for each destination, or a combination of destinations, to route IP packets.
- The routing table can be either static or dynamic.

Static Routing Table

- A static routing table contains information entered manually.
- The administrator enters the route for each destination into the table. When a table is created, it cannot update automatically when there is a change in the Internet.
- The table must be manually altered by the administrator.
- A static routing table can be used in a small internet that does not change very often.

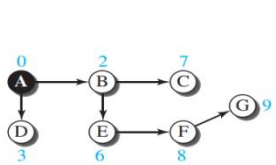
Dynamic Routing Table

- A dynamic routing table is updated periodically by using one of the dynamic routing protocols such as RIP, OSPF, or BGP.
- Whenever there is a change in the Internet, such as a shutdown of a router or breaking of a link, the dynamic routing protocols update all the tables in the routers automatically.

Distance Vector Routing:

- In distance vector routing, the least-cost route between any two nodes is the route with minimum distance
- In distance-vector routing, the first thing each node creates is its own least-cost tree with the limited information it has about its immediate neighbors.
- The incomplete trees are exchanged between immediate neighbors to make the trees more and more complete and to represent the whole internet.
- In distance-vector routing, a router continuously tells all of its neighbors what it knows about the whole internet
- A least-cost tree is a combination of least-cost paths from the root of the tree to all destinations.
- These paths are graphically fix together to form the tree.
- Distance-vector routing unfixes these paths and creates a distance vector, a one-dimensional array to represent the tree

Example: Consider the graph shown in figure

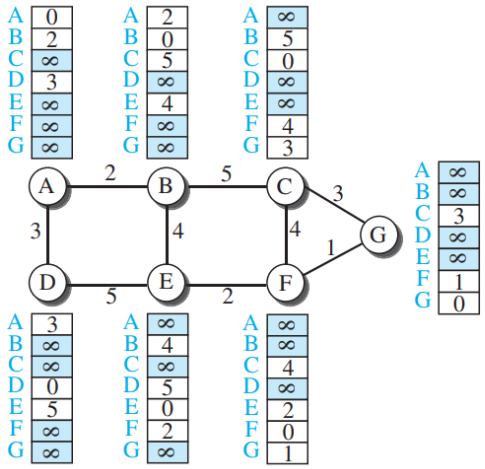


a. Tree for node A

	A
A	0
B	2
C	7
D	3
E	6
F	8
G	9

b. Distance vector for node A

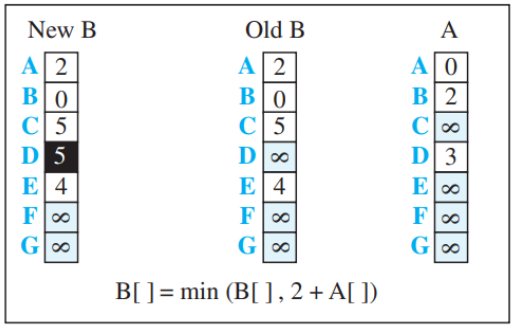
- The node sends some greeting messages to identify the immediate neighbors and the distance between itself and each neighbor.
- It then makes a simple distance vector by inserting the distances in the corresponding cells and leaves the value of other cells not connected as infinity.



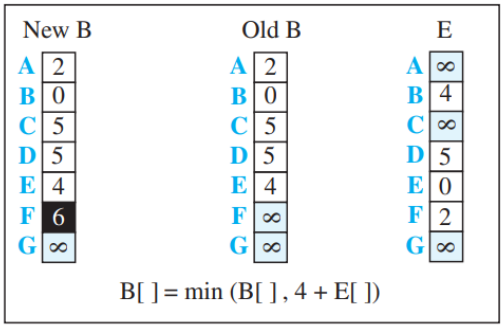
- After each node has created its vector, it sends a copy of the vector to all its immediate neighbors.
- After a node receives a distance vector from a neighbor, it updates its distance vector using the Bellman-Ford equation

$$D_{xy} = \min\{D_{xy}, (c_{xz} + D_{zy})\}$$

- Consider the figure, In the first event, node A has sent its vector to node B. Node B updates its vector using the cost $C_{BA} = 2$. In the second event, node E has sent its vector to node B. Node B updates its vector using the cost $C_{EA} = 4$.



a. First event: B receives a copy of A's vector.



b. Second event: B receives a copy of E's vector.

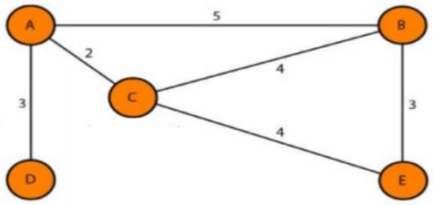
- After the first event, node B has one improvement in its vector: its least cost to node D has changed from infinity to 5 (via node A). After the second event, node B has one more improvement in its vector; its least cost to node F has changed from infinity to 6 (via node E).

Count to Infinity:

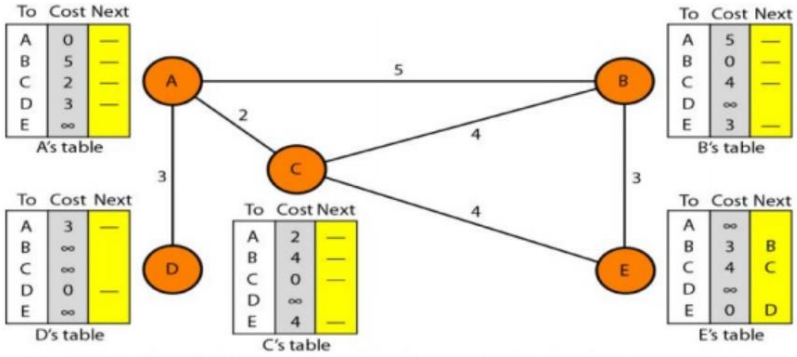
- A problem with distance-vector routing is that any decrease in cost propagates quickly, but any increase in cost will propagate slowly.

- For a routing protocol to work properly, if a link is broken, every other router should be aware of it immediately, but in distance-vector routing, this takes some time. The problem is referred to as count to infinity.

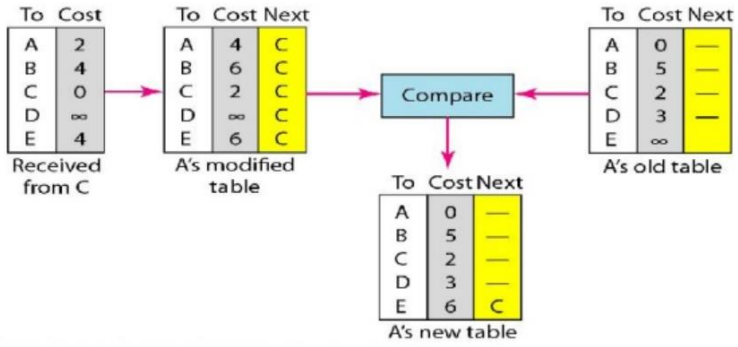
Example:
Consider the graph shown in Figure



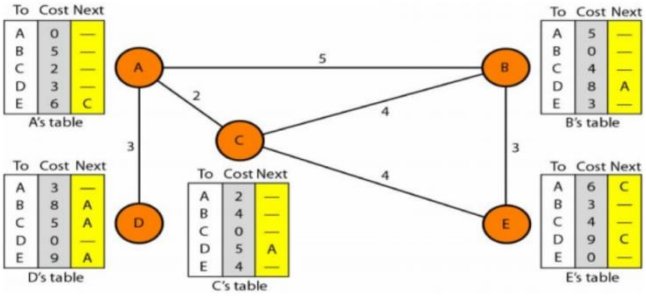
Initialization Table:



Updating Distance vector routing:



Distance vector routing table:

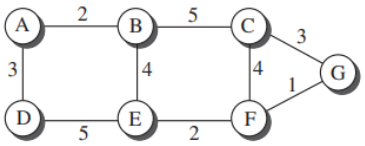


Link state Routing:

- A routing algorithm that directly creates least-cost trees and forwarding tables is link-state (LS) routing.
- This method uses the term link-state to define the characteristic of a link (an edge) that represents a network in the internet.
- In this algorithm the cost associated with an edge defines the state of the link.
- Links with lower costs are preferred to links with higher costs; if the cost of a link is infinity, it means that the link does not exist or has been broken.

Link-State Database (LSDB):

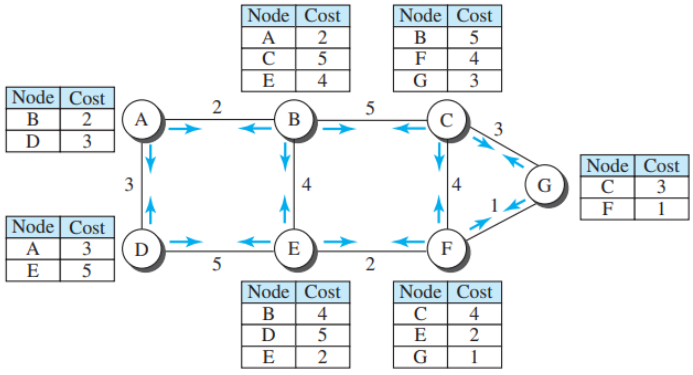
- To create a least-cost tree with this method, each node needs to have a complete map of the network, which means it needs to know the state of each link.
- The collection of states for all links is called the link-state database (LSDB). There is only one LSDB for the whole internet This process of creating LSDB is called flooding.
- Each node can send some greeting messages to all its immediate neighbors to collect two pieces of information for each neighboring node: the identity of the node and the cost of the link. The combination of these two pieces of information is called the LS packet (LSP)
- When a node receives an LSP from one of its interfaces, it compares the LSP with the copy it may already have. If the newly arrived LSP is older than the one it has (by checking the sequence number), it discards the LSP. If it is newer or the first one received, the node discards the old LSP (if there is one) and keeps the received one.
- It then sends a copy of it out of each interface except the one from which the packet arrived



a. The weighted graph

	A	B	C	D	E	F	G
A	0	2	∞	3	∞	∞	∞
B	2	0	5	∞	4	∞	∞
C	∞	5	0	∞	∞	4	3
D	3	∞	∞	0	5	∞	∞
E	∞	4	∞	5	0	2	∞
F	∞	∞	4	∞	2	0	1
G	∞	∞	3	∞	∞	1	0

b. Link state database

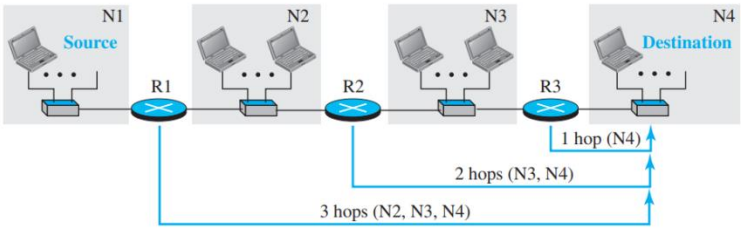


- In the distance-vector routing algorithm, each router tells its neighbors what it knows about the whole internet. In the link-state routing algorithm, each router tells the whole internet what it knows about its neighbors.

Intradomain Protocols:

Routing Information Protocol (RIP):

- The Routing Information Protocol (RIP) is one of the most widely used intradomain routing protocols based on the distance-vector routing algorithm
- RIP was started as part of the Xerox Network System (XNS), but it was the Berkeley Software Distribution (BSD) version of UNIX.
- A router in this protocol basically implements the distance-vector routing algorithm
- First, since a router in an AS needs to know how to forward a packet to different networks (subnets) in an AS, RIP routers advertise the cost of reaching different networks instead of reaching other nodes in a theoretical graph.
- Second, to make the implementation of the cost simpler, the cost is defined as the number of hops, which means the number of networks (subnets) a packet needs to travel through from the source router to the final destination host



Forwarding Tables

- A forwarding table in RIP is a three-column table in which the first column is the address of the destination network, the second column is the address of the next router to which the packet should be forwarded, and the third column is the cost (the number of hops) to reach the destination network.

Forwarding table for R1

Destination network	Next router	Cost in hops
N1	——	1
N2	——	1
N3	R2	2
N4	R2	3

Forwarding table for R2

Destination network	Next router	Cost in hops
N1	R1	2
N2	——	1
N3	——	1
N4	R3	2

Forwarding table for R3

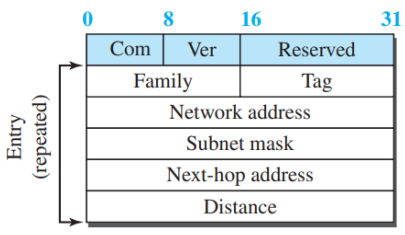
Destination network	Next router	Cost in hops
N1	R2	3
N2	R2	2
N3	——	1
N4	——	1

RIP Implementation:

- RIP is implemented as a process that uses the service of UDP on the well-known port number 520.
- RIP has gone through two versions: RIP-1 and RIP-2. The second version is backward compatible with the first section; it allows the use of more information in the RIP messages that were set to 0 in the first version.

RIP Messages:

- Two RIP processes, a client and a server, like any other processes, need to exchange messages



Fields

Com: Command, request (1), response (2)
Ver: Version, current version is 2
Family: Family of protocol, for TCP/IP value is 2
Tag: Information about autonomous system
Network address: Destination address
Subnet mask: Prefix length
Next-hop address: Address length
Distance: Number of hops to the destination

- RIP has two types of messages: request and response. A request message is sent by a router that has just come up or by a router that has some time-out entries. A request message can ask about specific entries or all entries.
- A response (or update) message can be either solicited or unsolicited. A solicited response message is sent only in answer to a request message. It contains information about the destination specified in the corresponding request message

RIP Algorithm:

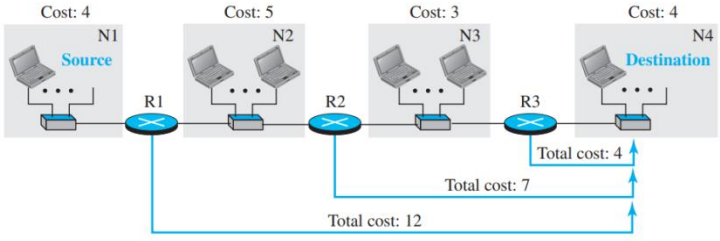
RIP implements the same algorithm as the distance-vector routing algorithm. There are some changes added in that algorithm

- Instead of sending only distance vectors, a router needs to send the whole contents of its forwarding table in a response message.
- The receiver adds one hop to each cost and changes the next router field to the address of the sending router. We call each route in the modified forwarding table the received route and each route in the old forwarding table the old route.
- The received router selects the old routes as the new ones except in the following three cases:

  1. If the received route does not exist in the old forwarding table, it should be added to the route.
  2. If the cost of the received route is lower than the cost of the old one, the received route should be selected as the new one.
  3. If the cost of the received route is higher than the cost of the old one, but the value of the next router is the same in both routes, the received route should be selected as the new one.
  4. The new forwarding table needs to be sorted according to the destination route

Open Shortest Path First (OSPF):

- OSPF is an open protocol, which means that the specification is a public document.
- Like RIP, the cost of reaching a destination from the host is calculated from the source router to the destination network.
- However, each link (network) can be assigned a weight based on the throughput, round-trip time, reliability, and so on. An administration can also decide to use the hop count as the cost.



Forwarding Tables

- Each OSPF router can create a forwarding table after finding the shortest-path tree between itself and the destination using Dijkstra's algorithm
- Compared with RIP, which is normally used in small ASs, OSPF was designed to be able to handle routing in a small or large autonomous system.
- However, the formation of shortest-path trees in OSPF requires that all routers flood the whole AS with their LSPs to create the global LSDB.

- Although this may not create a problem in a small AS, it may have created a huge volume of traffic in a large AS.
- To prevent this, the AS needs to be divided into small sections called areas. Each area acts as a small independent domain for flooding LSPs. OSPF uses two level of hierarchy in routing: the first level is the autonomous system, the second is the area.

Forwarding table for R1

Destination network	Next router	Cost
N1	——	4
N2	——	5
N3	R2	8
N4	R2	12

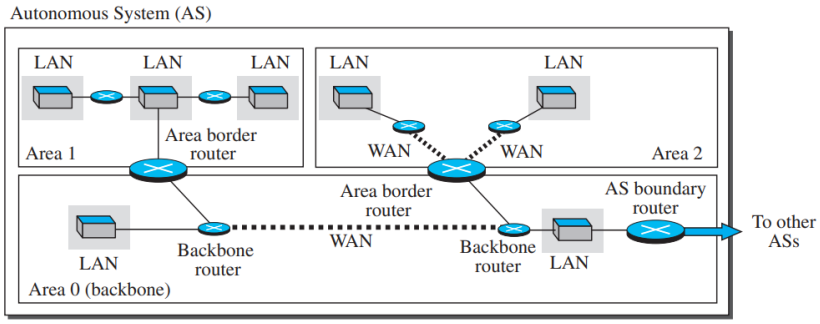
Forwarding table for R2

Destination network	Next router	Cost
N1	R1	9
N2	——	5
N3	——	3
N4	R3	7

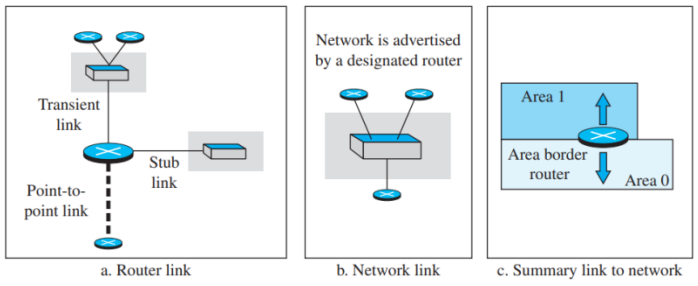
Forwarding table for R3

Destination network	Next router	Cost
N1	R2	12
N2	R2	8
N3	——	3
N4	——	4

- The routers in the backbone area are responsible for passing the information collected by each area to all other areas



- OSPF is based on the link-state routing algorithm, which requires that a router advertise the state of each link to all neighbors for the formation of the LSDB.
- There are five types of link-state advertisements are there: router link, network link, summary link to network, summary link to AS border router, and external link.

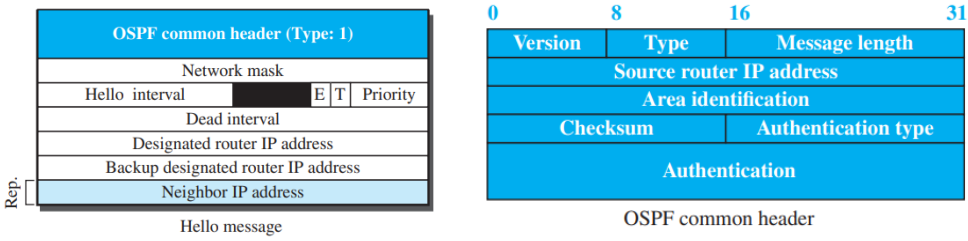


OSPF Messages:

OSPF is a very complex protocol; it uses five different types of messages.

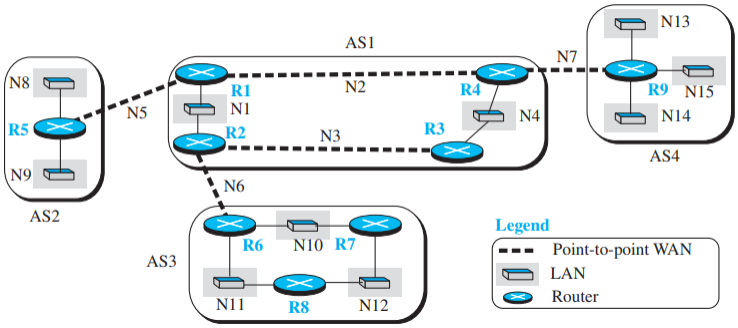
- The hello message (type 1) is used by a router to introduce itself to the neighbors and announce all neighbors that it already knows.
- The database description message (type 2) is normally sent in response to the hello message to allow a newly joined router to acquire the full LSDB.
- The linkstate request message (type 3) is sent by a router that needs information about a specific LS.

- The link-state update message (type 4) is the main OSPF message used for building the LSDB.



Border Gateway Protocol Version 4 (BGP4):

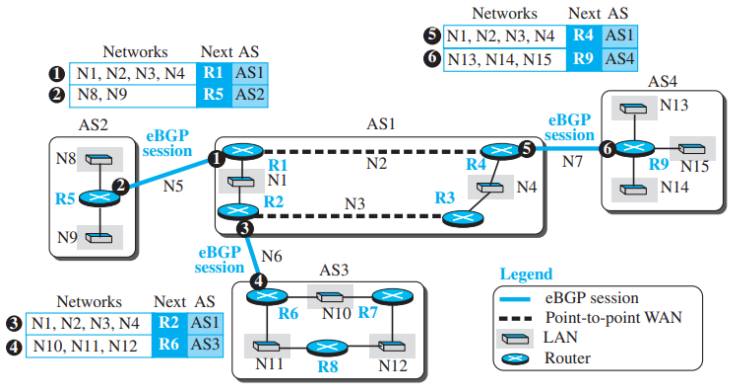
- The Border Gateway Protocol version 4 (BGP4) is the only interdomain routing protocol used in the Internet today. BGP4 is based on the path-vector algorithm.
- BGP, and in particular BGP4, is a complex protocol. AS2, AS3, and AS4 are stub autonomous systems; AS1 is a transient one.
- Each router in each AS knows how to reach a network that is in its own AS, but it does not know how to reach a network in another AS.



- The BGP may be external BGP (eBGP) or internal BGP (iBGP)

Operation of External BGP (eBGP)

- When the software is installed on two routers, they try to create a TCP connection using the well-known port 179.
- A pair of client and server processes continuously communicates with each other to exchange messages.
- The two routers that run the BGP processes are called BGP peers or BGP speakers.

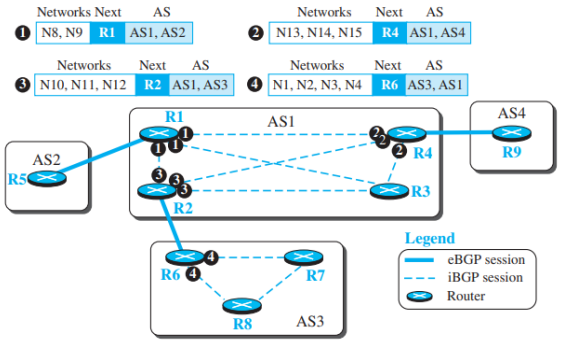


- The eBGP variation of BGP allows two physically connected border routers in two different ASs to form pairs of eBGP speakers and exchange messages.

- However, there is a need for a logical TCP connection to be created over the physical connection to make the exchange of information possible.
- Each logical connection in BGP parlance is referred to as a session.

Operation of Internal BGP (iBGP):

- The iBGP protocol is similar to the eBGP protocol in that it uses the service of TCP on the well-known port 179, but it creates a session between any possible pair of routers inside an autonomous system. However, some points should be made clear.
- First, if an AS has only one router, there cannot be an iBGP session. For example, we cannot create an iBGP session inside AS2 or AS4 in our internet.
- Second, if there are n routers in an autonomous system, there should be [n × (n − 1) / 2] iBGP sessions in that autonomous system to prevent loops in the system.
- Each router needs to advertise its own reachability to the peer in the session instead of flooding what it receives from another peer in another session



- The first message (numbered 1) is sent by R1 announcing that networks N8 and N9 are reachable through the path AS1-AS2, but the next router is R1.
- This message is sent, through separate sessions, to R2, R3, and R4. Routers R2, R4, and R6 do the same thing but send different messages to different destinations.
- The interesting point is that, at this stage, R3, R7, and R8 create sessions with their peers, but they actually have no message to send.
- After R1 receives the update message from R2, it combines the reachability information about AS3 with the reachability information it already knows about AS1 and sends a new update message to R5.
- Now R5 knows how to reach networks in AS1 and AS3. The process continues when R1 receives the update message from R4

Networks	Next	Path
N8, N9	R5	AS1, AS2
N10, N11, N12	R2	AS1, AS3
N13, N14, N15	R4	AS1, AS4

Path table for R1

Networks	Next	Path
N8, N9	R1	AS1, AS2
N10, N11, N12	R6	AS1, AS3
N13, N14, N15	R1	AS1, AS4

Path table for R2

Networks	Next	Path
N8, N9	R2	AS1, AS2
N10, N11, N12	R2	AS1, AS3
N13, N14, N15	R4	AS1, AS4

Path table for R3

Networks	Next	Path
N8, N9	R1	AS1, AS2
N10, N11, N12	R1	AS1, AS3
N13, N14, N15	R9	AS1, AS4

Path table for R4

Networks	Next	Path
N1, N2, N3, N4	R1	AS2, AS1
N10, N11, N12	R1	AS2, AS1, AS3
N13, N14, N15	R1	AS2, AS1, AS4

Path table for R5

Networks	Next	Path
N1, N2, N3, N4	R2	AS3, AS1
N8, N9	R2	AS3, AS1, AS2
N13, N14, N15	R2	AS3, AS1, AS4

Path table for R6

Networks	Next	Path
N1, N2, N3, N4	R6	AS3, AS1
N8, N9	R6	AS3, AS1, AS2
N13, N14, N15	R6	AS3, AS1, AS4

Path table for R7

Networks	Next	Path
N1, N2, N3, N4	R6	AS3, AS1
N8, N9	R6	AS3, AS1, AS2
N13, N14, N15	R6	AS3, AS1, AS4

Path table for R8

Networks	Next	Path
N1, N2, N3, N4	R4	AS4, AS1
N8, N9	R4	AS4, AS1, AS2
N10, N11, N12	R4	AS4, AS1, AS3

Path table for R9

IPv6 ADDRESSING:

- The main reason for migration from IPv4 to IPv6 is the small size of the address space in IPv4.
- An IPv6 address is 128 bits or 16 bytes (octets) long, four times the address length in IPv4
- A computer normally stores the address in binary, but it is clear that 128 bits cannot easily be handled by humans
- Binary notation is used when the addresses are stored in a computer.
- The colon hexadecimal notation (or colon hex for short) divides the address into eight sections, each made of four hexadecimal digits separated by colons

Abbreviation

- Although an IPv6 address, even in hexadecimal format, is very long, many of the digits are zeros.
- The leading zeros of a section can be omitted. Using this form of abbreviation, 0074 can be written as 74, 000F as F, and 0000 as 0. Note that 3210 cannot be abbreviated.
- Further abbreviation, often called zero compression, can be applied to colon hex notation if there are consecutive sections consisting of zeros only. We can remove all the zeros and replace them with a double semicolon.

FDEC:0:0:0:0:BBFF:0:FFFF ⟶ FDEC::BBFF:0:FFFF

- IPv6 uses hierarchical addressing. The address space of IPv6 contains 2^{128} addresses. This address space is 296 times the IPv4 address—definitely no address depletion

Address Types

In IPv6, a destination address can belong to one of three categories: unicast, anycast, and multicast.

Unicast Address

A unicast address defines a single interface (computer or router).

The packet sent to a unicast address will be routed to the intended recipient.

Anycast Address

- An anycast address defines a group of computers that all share a single address.
- A packet with an anycast address is delivered to only one member of the group, the most reachable one. An anycast communication is used, for example, when there are several servers that can respond to an inquiry.
- The request is sent to the one that is most reachable. The hardware and software generate only one copy of the request; the copy reaches only one of the servers.
- IPv6 does not designate a block for anycasting; the addresses are assigned from the unicast block.

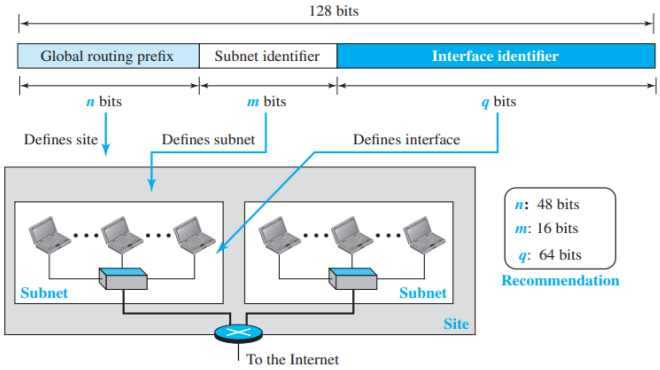
Multicast Address

- A multicast address also defines a group of computers. However, there is a difference between anycasting and multicasting.
- In anycasting, only one copy of the packet is sent to one of the members of the group; in multicasting each member of the group receives a copy.

Global Unicast Addresses

- The block in the address space that is used for unicast (one-to-one) communication between two hosts in the Internet is called the global unicast address block.
- CIDR for the block is 2000::/3, which means that the three leftmost bits are the same for all addresses in this block (001).

- The size of this block is 2125 bits, which is more than enough for Internet expansion for many years to come.
- An address in this block is divided into three parts: global routing prefix (n bits), subnet identifier (m bits), and interface identifier (q bits)



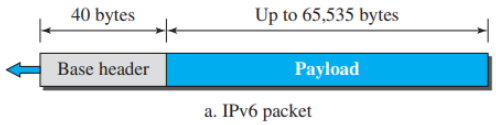
IPv6 PROTOCOL:

The change of the IPv6 address size requires the change in the IPv4 packet format.

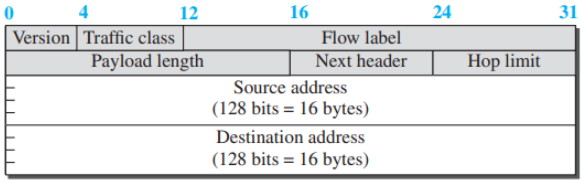
- **Better header format**. IPv6 uses a new header format in which options are separated from the base header and inserted, when needed, between the base header and the data.
- **New options.** IPv6 has new options to allow for additional functionalities.
- **Allowance for extension.** IPv6 is designed to allow the extension of the protocol if required by new technologies or applications.
- **Support for resource allocation.** In IPv6, the type-of-service field has been removed, but two new fields, traffic class and flow label, have been added to enable the source to request special handling of the packet. This mechanism can be used to support traffic such as real-time audio and video.
- **Support for more security.** The encryption and authentication options in IPv6 provide confidentiality and integrity of the packet.

Packet Format:

Each packet is composed of a base header followed by the payload. The base header occupies 40 bytes, whereas payload can be up to 65,535 bytes of information.



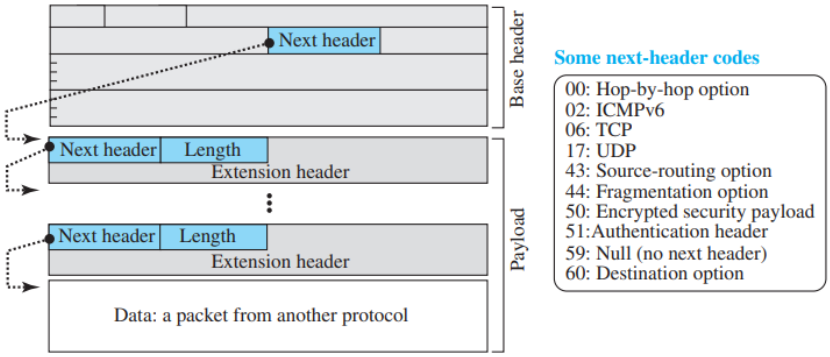
a. IPv6 packet



b. Base header

- **Version.** The 4-bit version field defines the version number of the IP. For IPv6, the value is 6.
- **Traffic class.** The 8-bit traffic class field is used to distinguish different payloads with different delivery requirements. It replaces the type-of-service field in IPv4.

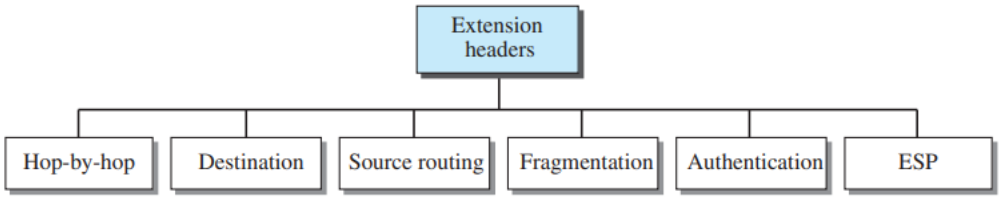
- **Flow label**. The flow label is a 20-bit field that is designed to provide special handling for a particular flow of data.
- **Payload length.** The 2-byte payload length field defines the length of the IP datagram excluding the header.
- **Next header**. The next header is an 8-bit field defining the type of the first extension header or the type of the data that follows the base header in the datagram.
- **Hop limit**. The 8-bit hop limit field serves the same purpose as the TTL field in IPv4.
- **Source and destination addresses.** The source address field is a 16-byte (128-bit) Internet address that identifies the original source of the datagram. The destination address field is a 16-byte (128-bit) Internet address that identifies the destination of the datagram.
- **Payload.** The payload field in IPv6 differ from IPv4



The payload in IPv6 means a combination of zero or more extension headers (options) followed by the data from other protocols.

Extension Header

- An IPv6 packet is made of a base header and some extension headers.
- The length of the base header is fixed at 40 bytes. However, to give more functionality to the IP datagram, the base header can be followed by up to six extension headers.
- These are hop-by-hop option, source routing, fragmentation, authentication, encrypted security payload, and destination option.



Hop-by-Hop - The hop-by-hop option is used when the source needs to pass information to all routers visited by the datagram

Destination Option The destination option is used when the source needs to pass information to the destination only

Source Routing The source routing extension header combines the concepts of the strict source route and the loose source route options of IPv4.

Authentication The authentication extension header has a dual purpose: it validates the message sender and ensures the integrity of data

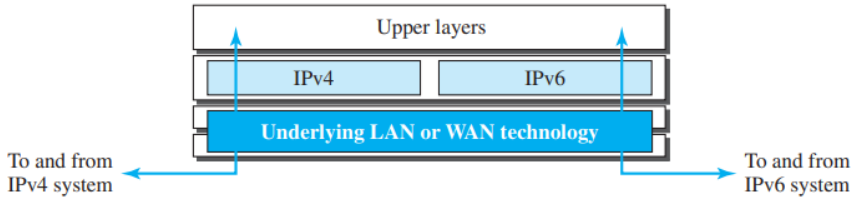
Encrypted Security Payload The encrypted security payload (ESP) is an extension that provides confidentiality and guards against eavesdropping.

TRANSITION FROM IPv4 TO IPv6:

There are three strategies have been devised for transition: dual stack, tunneling, and header translation

Dual Stack:

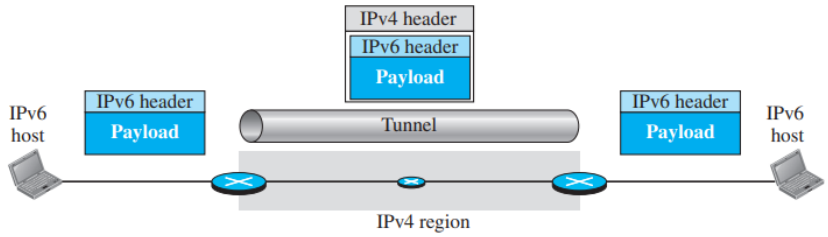
- It is recommended that all hosts, before migrating completely to version 6, have a dual stack of protocols during the transition. In other words, a station must run IPv4 and IPv6 simultaneously until all the Internet uses IPv6



- To determine which version to use when sending a packet to a destination, the source host queries the DNS. If the DNS returns an IPv4 address, the source host sends an IPv4 packet. If the DNS returns an IPv6 address, the source host sends an IPv6 packet.

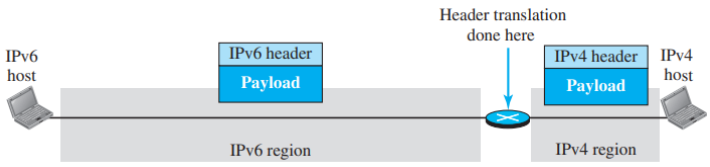
Tunneling

- Tunneling is a strategy used when two computers using IPv6 want to communicate with each other and the packet must pass through a region that uses IPv4.
- To pass through this region, the packet must have an IPv4 address. So the IPv6 packet is encapsulated in an IPv4 packet when it enters the region, and it leaves its capsule when it exits the region. It seems as if the IPv6 packet enters a tunnel at one end and emerges at the other end.



Header

- Translation Header translation is necessary when the majority of the Internet has moved to IPv6 but some systems still use IPv4.
- The sender wants to use IPv6, but the receiver does not understand IPv6. Tunneling does not work in this situation because the packet must be in the IPv4 format to be understood by the receiver.



EC3401 NETWORKS AND SECURITY

UNIT III TRANSPORT AND APPLICATION LAYERS

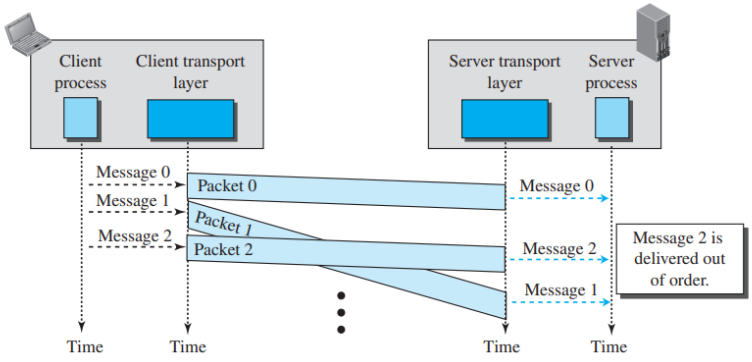
Transport Layer Protocols – UDP and TCP Connection and State Transition Diagram - Congestion Control and Avoidance (DEC bit, RED)- QoS - Application Layer Paradigms – Client – Server Programming – Domain Name System – World Wide Web, HTTP, Electronic Mail.

The transport layer is located between the application layer and the network layer. It provides a process-to-process communication between two application layers, one at the local host and the other at the remote host.

A transport-layer protocol, like a network-layer protocol, can provide two types of services: connectionless and connection-oriented

Connectionless Service:

- In a connectionless service, the source process (application program) needs to divide its message into chunks of data of the size acceptable by the transport layer and deliver them to the transport layer one by one.
- The transport layer treats each chunk as a single unit without any relation between the chunks.
- When a chunk arrives from the application layer, the transport layer encapsulates it in a packet and sends it.
- Since there is no dependency between the packets at the transport layer, the packets may arrive out of order at the destination and will be delivered out of order to the server process



- The situation would be worse if one of the packets were lost. Since there is no numbering on the packets, the receiving transport layer has no idea that one of the messages has been lost. It just delivers two chunks of data to the server process.

Connection-Oriented Service:

- In a connection-oriented service, the client and the server first need to establish a logical connection between them. The data exchange can only happen after the connection establishment. After data exchange, the connection needs to be torn down.

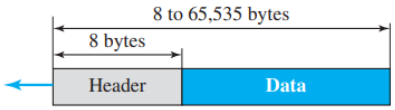
USER DATAGRAM PROTOCOL

- The User Datagram Protocol (UDP) is a connectionless, unreliable transport protocol.

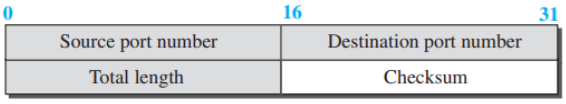
- It does not add anything to the services of IP except for providing process-to-process communication instead of host-to-host communication.
- UDP is a very simple protocol using a minimum of overhead.
- If a process wants to send a small message and does not care much about reliability, it can use UDP. Sending a small message using UDP takes much less interaction between the sender and receiver

User Datagram

- UDP packets, called user datagrams, have a fixed-size header of 8 bytes made of four fields, each of 2 bytes (16 bits).
- The first two fields define the source and destination port numbers.
- The third field defines the total length of the user datagram, header plus data.
- The 16 bits can define a total length of 0 to 65,535 bytes. However, the total length needs to be less because a UDP user datagram is stored in an IP datagram with the total length of 65,535 bytes
- The last field can carry the optional checksum



a. UDP user datagram



b. Header format

Example:

The following is the content of a UDP header in hexadecimal format. **CB84000D001C001C**

a. What is the source port number?

b. What is the destination port number?

c. What is the total length of the user datagram?

d. What is the length of the data?

Solution

a. The source port number is the first four hexadecimal digits (CB84)16, which means that the source port number is 52100.

b. The destination port number is the second four hexadecimal digits (000D)16, which means that the destination port number is 13.

c. The third four hexadecimal digits (001C)16 define the length of the whole UDP packet as 28 bytes.

d. The length of the data is the length of the whole packet minus the length of the header, or 28 − 8 = 20 bytes.

UDP Services:

Process-to-Process Communication:

UDP provides process-to-process communication using socket addresses, a combination of IP addresses and port numbers

Connectionless Services

- UDP provides a connectionless service.
- There is no relationship between the different user datagrams even if they are coming from the same source process and going to the same destination program.
- The user datagrams are not numbered

Flow Control

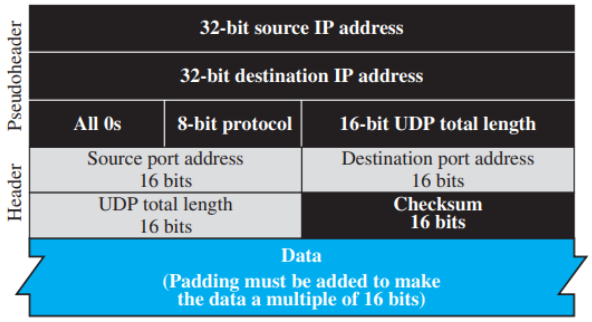
- UDP is a very simple protocol. There is no flow control, and hence no window mechanism. The receiver may overflow with incoming messages.

Error Control

- There is no error control mechanism in UDP except for the checksum.
- The sender does not know if a message has been lost or duplicated.
- When the receiver detects an error through the checksum, the user datagram is silently discarded.

Checksum

- UDP checksum calculation includes three sections: a pseudoheader, the UDP header, and the data coming from the application layer.
- The pseudoheader is the part of the header of the IP packet in which the user datagram is to be encapsulated with some fields filled with 0s
- If the checksum does not include the pseudo header, a user datagram may arrive safe and sound. However, if the IP header is corrupted, it may be delivered to the wrong host.
- If the checksum does not include the pseudo header, a user datagram may arrive safe and sound. However, if the IP header is corrupted, it may be delivered to the wrong host.



Applications of UDP:

- UDP is suitable for a process that requires simple request-response communication with little concern for flow and error control.
- UDP is a suitable transport protocol for multicasting.
- UDP is used for management processes such as SNMP
- UDP is used for some route updating protocols such as Routing Information Protocol (RIP)

- UDP is normally used for interactive real-time applications that cannot tolerate uneven delay between sections of a received message

TRANSMISSION CONTROL PROTOCOL:

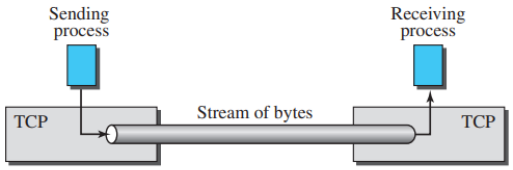
Transmission Control Protocol (TCP) is a connection-oriented, reliable protocol. TCP explicitly defines connection establishment, data transfer, and connection teardown phases to provide a connection-oriented service.

TCP Services:

Process-to-Process Communication: As with UDP, TCP provides process-to-process communication using port numbers.

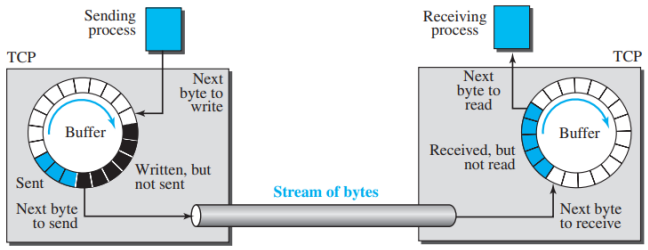
Stream Delivery Service:

- TCP allows the sending process to deliver data as a stream of bytes and allows the receiving process to obtain data as a stream of bytes. TCP creates an environment in which the two processes seem to be connected by an imaginary "tube" that carries their bytes across the Internet



Sending and Receiving Buffers:

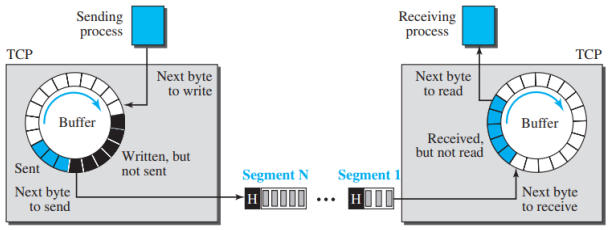
- Because the sending and the receiving processes may not necessarily write or read data at the same rate, TCP needs buffers for storage. There are two buffers, the sending buffer and the receiving buffer, one for each direction
- At the sender, the buffer has three types of chambers. The white section contains empty chambers that can be filled by the sending process (producer). The colored area holds bytes that have been sent but not yet acknowledged. The TCP sender keeps these bytes in the buffer until it receives an acknowledgment
- TCP may be able to send only part of this shaded section. This could be due to the slowness of the receiving process or to congestion in the network
- The operation of the buffer at the receiver is simpler. The circular buffer is divided into two areas. The white area contains empty chambers to be filled by bytes received from the network. The colored sections contain received bytes that can be read by the receiving process.



Segments:

- At the transport layer, TCP groups a number of bytes together into a packet called a segment

- TCP adds a header to each segment and delivers the segment to the network layer for transmission
- The segments are encapsulated in an IP datagram and transmitted. This entire operation is transparent to the receiving process.



Full-Duplex Communication: TCP offers full-duplex service, where data can flow in both directions at the same time

Multiplexing and Demultiplexing: TCP performs multiplexing at the sender and demultiplexing at the receiver

Connection-Oriented Service TCP: When a process at site A wants to send to and receive data from another process at site B, the following three phases occur:

1. The two TCP's establish a logical connection between them.
2. Data are exchanged in both directions.
3. The connection is terminated.

Numbering System: Although the TCP software keeps track of the segments being transmitted or received, there is no field for a segment number value in the segment header. Instead, there are two fields, called the sequence number and the acknowledgment number. These two fields refer to a byte number and not a segment number.

Byte Number: TCP numbers all data bytes that are transmitted in a connection. Numbering is independent in each direction. When TCP receives bytes of data from a process, TCP stores them in the sending buffer and numbers them. The numbering does not necessarily start from 0. Instead, TCP chooses an arbitrary number between 0 and 232 − 1 for the number of the first byte.

Sequence Number: After the bytes have been numbered, TCP assigns a sequence number to each segment that is being sent. The sequence number, in each direction, is defined as follows:

1. The sequence number of the first segment is the ISN (initial sequence number), which is a random number.

2. The sequence number of any other segment is the sequence number of the previous segment plus the number of bytes (real or imaginary) carried by the previous segment.

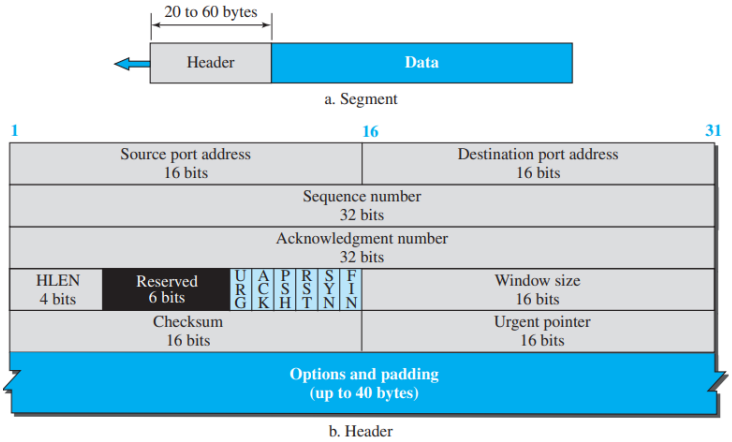
TCP Segment Format:

- A packet in TCP is called a segment. The segment consists of a header of 20 to 60 bytes, followed by data from the application program. The header is 20 bytes if there are no options and up to 60 bytes if it contains options.

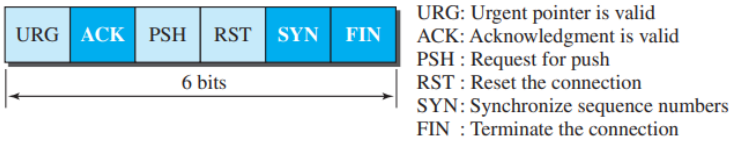
Source port address. This is a 16-bit field that defines the port number of the application program in the host that is sending the segment.

Destination port address. This is a 16-bit field that defines the port number of the application program in the host that is receiving the segment.

Sequence number. This 32-bit field defines the number assigned to the first byte of data contained in this segment. The sequence number tells the destination which byte in this sequence is the first byte in the segment.
Acknowledgment number. This 32-bit field defines the byte number that the receiver of the segment is expecting to receive from the other party. If the receiver of the segment has successfully received byte number x from the other party, it returns x + 1 as the acknowledgment number. Acknowledgment and data can be piggybacked together.
Header length. This 4-bit field indicates the number of 4-byte words in the TCP header. The length of the header can be between 20 and 60 bytes. Therefore, the value of this field is always between 5 (5 × 4 = 20) and 15 (15 × 4 = 60).



Control. This field defines 6 different control bits or flags. These bits enable flow control, connection establishment and termination, connection abortion, and the mode of data transfer in TCP.



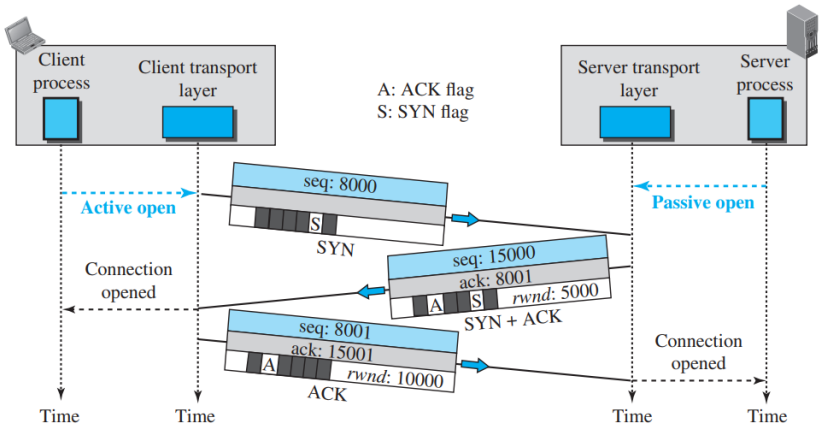
Window size. This field defines the window size of the sending TCP in bytes. The length of this field is 16 bits, which means that the maximum size of the window is 65,535 bytes
Checksum. This 16-bit field contains the checksum
Urgent pointer. This 16-bit field, which is valid only if the urgent flag is set, is used when the segment contains urgent data
TCP Connection:
TCP is connection-oriented. A connection-oriented transport protocol establishes a logical path between the source and destination. All of the segments belonging to a message are then sent over this logical path.
Using a single logical pathway for the entire message facilitates the acknowledgment process as well as retransmission of damaged or lost frames
In TCP, connection-oriented transmission requires three phases: **connection establishment, data transfer, and connection termination**
Connection Establishment:

- TCP transmits data in full-duplex mode.
- When two TCPs in two machines are connected, they are able to send segments to each other simultaneously.

- This implies that each party must initialize communication and get approval from the other party before any data are transferred.

Three-Way Handshaking:

- The connection establishment in TCP is called three-way handshaking. An application program, called the client, wants to make a connection with another application program, called the server, using TCP as the transport-layer protocol.
- The process starts with the server. The server program tells its TCP that it is ready to accept a connection. This request is called a passive open
- The client program issues a request for an active open. A client that wishes to connect to an open server tells its TCP to connect to a particular server

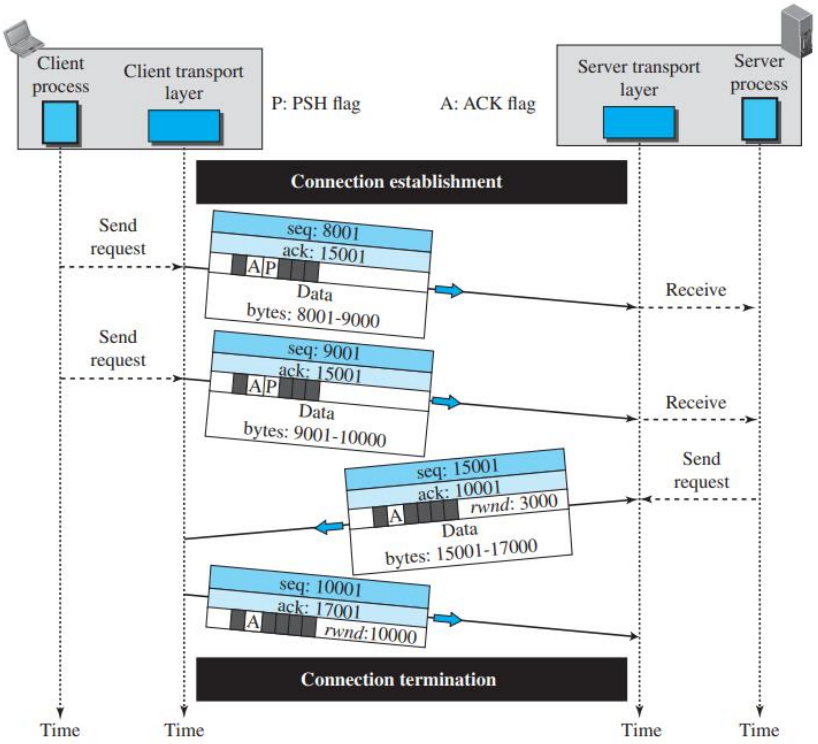


- The client sends the first segment, a SYN segment, in which only the SYN flag is set. This segment is for synchronization of sequence numbers. This sequence number is called the initial sequence number (ISN). This segment does not contain an acknowledgment number
- The SYN segment is a control segment and carries no data. However, it consumes one sequence number because it needs to be acknowledged.
- The server sends the second segment, a SYN + ACK segment with two flag bits set as: SYN and ACK. This segment has a dual purpose. First, it is a SYN segment for communication in the other direction. The server uses this segment to initialize a sequence number for numbering the bytes sent from the server to the client.
- The server also acknowledges the receipt of the SYN segment from the client by setting the ACK flag and displaying the next sequence number it expects to receive from the client. Because the segment contains an acknowledgment, it also needs to define the receive window size, rwnd (to be used by the client)
- The client sends the third segment. This is just an ACK segment. It acknowledges the receipt of the second segment with the ACK flag and acknowledgment number field

Data Transfer

- After connection is established, bidirectional data transfer can take place. The client and server can send data and acknowledgments in both directions
- In the diagram shown in figure the client sends 2,000 bytes of data in two segments. The server then sends 2,000 bytes in one segment. The client sends one more segment. The first three segments carry both data and acknowledgment, but the last segment carries only an acknowledgment because there is no more data to be sent

- The data segments sent by the client have the PSH (push) flag set so that the server TCP knows to deliver data to the server process as soon as they are received

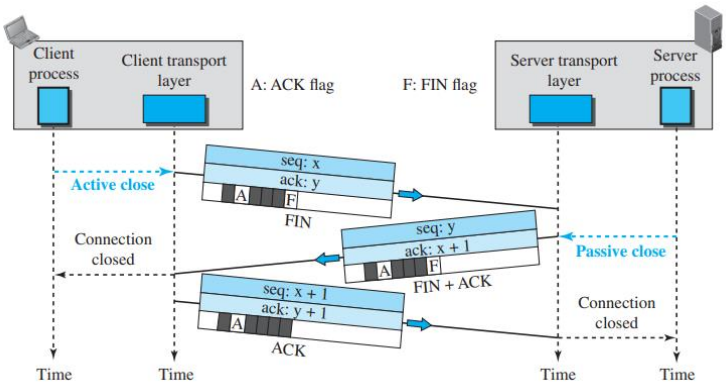


Connection Termination:

- Either of the two parties involved in exchanging data (client or server) can close the connection, although it is usually initiated by the client
- Most implementations today allow two options for connection termination: three-way handshaking and four-way handshaking with a half-close option.

Three-Way Handshaking:

- In this situation, the client TCP, after receiving a close command from the client process, sends the first segment, a FIN segment in which the FIN flag is set.
- The server TCP, after receiving the FIN segment, informs its process of the situation and sends the second segment, a FIN + ACK segment, to confirm the receipt of the FIN segment
- The client TCP sends the last segment, an ACK segment, to confirm the receipt of the FIN segment from the TCP server. This segment contains the acknowledgment number, which is one plus the sequence number received in the FIN segment from the server. This segment cannot carry data and consumes no sequence numbers.

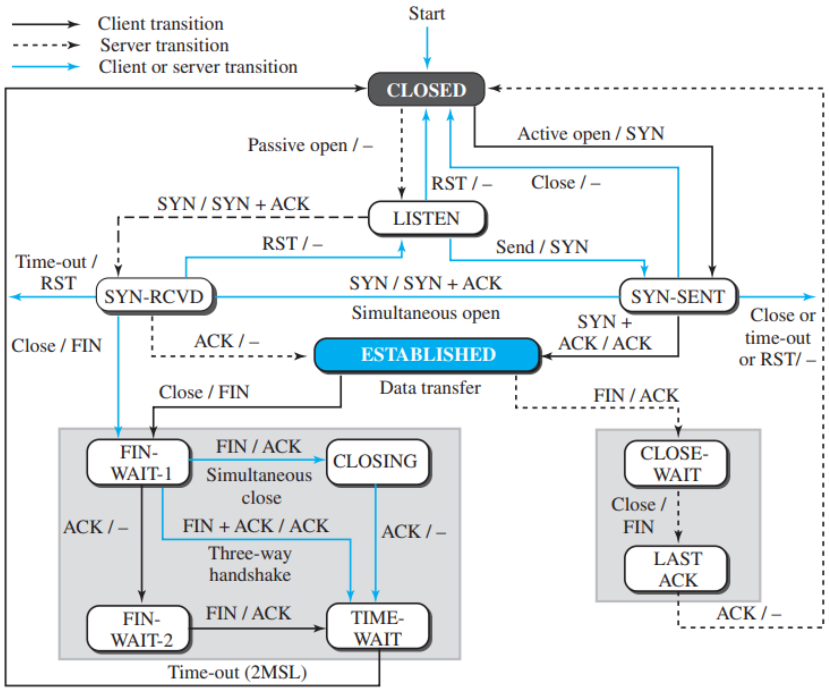


Connection Reset

- TCP at one end may deny a connection request, may abort an existing connection, or may terminate an idle connection

State Transition Diagram

- To keep track of all the different events happening during connection establishment, connection termination, and data transfer, TCP is specified as the finite state machine (FSM)
- The two FSMs used by the TCP client and server combined in one diagram. The rounded-corner rectangles represent the states. The transition from one state to another is shown using directed lines. Each line has two strings separated by a slash. The first string is the input, what TCP receives. The second is the output, what TCP sends.



- The dotted black lines in the figure represent the transition that a server normally goes through; the solid black lines show the transitions that a client normally goes through. However, in some situations, a server transitions through a solid line or a client transitions through a dotted line. The colored lines show special situations

State	Description
CLOSED	No connection exists
LISTEN	Passive open received; waiting for SYN
SYN-SENT	SYN sent; waiting for ACK
SYN-RCVD	SYN + ACK sent; waiting for ACK
ESTABLISHED	Connection established; data transfer in progress
FIN-WAIT-1	First FIN sent; waiting for ACK
FIN-WAIT-2	ACK to first FIN received; waiting for second FIN
CLOSE-WAIT	First FIN received, ACK sent; waiting for application to close
TIME-WAIT	Second FIN received, ACK sent; waiting for 2MSL time-out
LAST-ACK	Second FIN sent; waiting for ACK
CLOSING	Both sides decided to close simultaneously

TCP Congestion Control:

- In flow control the size of the send window is controlled by the receiver using the value of rwnd, which is advertised in each segment traveling in the opposite direction.
- The use of this strategy guarantees that the receive window is never overflowed with the received bytes
- However, does not mean that the intermediate buffers, buffers in the routers, do not become congested. A router may receive data from more than one sender. No matter how large the buffers of a router may be, it may be overwhelmed with data, which results in dropping some segments sent by a specific TCP sender
- TCP needs to worry about congestion in the middle because many segments lost may seriously affect the error control.
- To control the number of segments to transmit, TCP uses another variable called a congestion window, cwnd, whose size is controlled by the congestion situation in the network
- The cwnd variable and the rwnd variable together define the size of the send window in TCP. The first is related to the congestion in the middle .The second is related to the congestion at the end. The actual size of the window is the minimum of these two.

Actual window size = minimum (rwnd, cwnd)

Congestion Detection:

- The TCP sender uses the occurrence of two events as signs of congestion in the network: time-out and receiving three duplicate ACKs.
- The first is the time-out. If a TCP sender does not receive an ACK for a segment or a group of segments before the time-out occurs, it assumes that the corresponding segment or segments are lost and the loss is due to congestion
- When a TCP receiver sends a duplicate ACK, it is the sign that a segment has been delayed, but sending three duplicate ACKs is the sign of a missing segment, which can be due to congestion in the network.
- When a receiver sends three duplicate ACKs, it means that one segment is missing, but three segments have been received. The network is either slightly congested or has recovered from the congestion

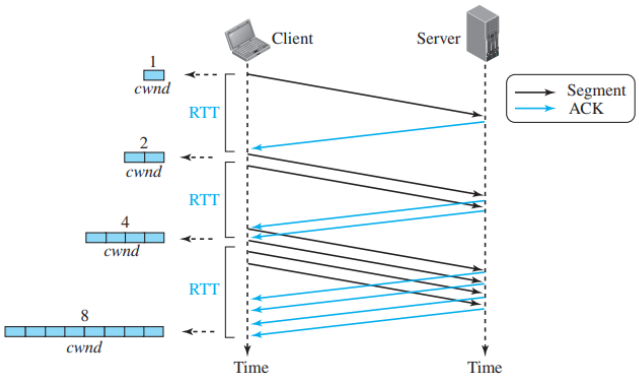
Congestion Policies:

- TCP's general policy for handling congestion is based on three algorithms: slow start, congestion avoidance, and fast recovery

Slow Start (Exponential Increase):

- The slow-start algorithm is based on the idea that the size of the congestion window (cwnd) starts with one maximum segment size (MSS), but it increases one MSS each time an acknowledgment arrives.

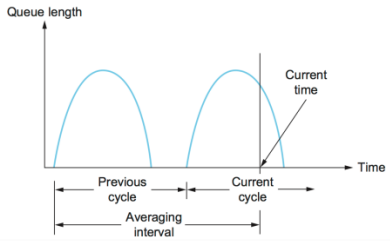
- The algorithm starts slowly, but grows exponentially. The rwnd is much larger than cwnd, so that the sender window size always equals cwnd.
- The sender starts with cwnd = 1. This means that the sender can send only one segment. After the first ACK arrives, the acknowledged segment is purged from the window, which means there is now one empty segment slot in the window.



- The size of the congestion window is also increased by 1 because the arrival of the acknowledgement is a good sign that there is no congestion in the network.
- The size of the window is now 2. After sending two segments and receiving two individual acknowledgments for them, the size of the congestion window now becomes 4, and so on.
- The size of the congestion window in this algorithm is a function of the number of ACKs arrived and can be determined as follows.
- If an ACK arrives, cwnd = cwnd + 1.
- A slow start cannot continue indefinitely. There must be a threshold to stop this phase. The sender keeps track of a variable named ssthresh (slow-start threshold). When the size of the window in bytes reaches this threshold, slow start stops and the next phase starts

DEC bit

- It is first mechanism was developed for use on the Digital Network Architecture (DNA), a connectionless network with a connection-oriented transport protocol.
- The idea is to more evenly split the responsibility for congestion control between the routers and the end nodes.
- Each router monitors the load it is experiencing and explicitly notifies the end nodes when congestion is about to occur. This notification is implemented by setting a binary congestion bit in the packets that flow through the router, hence the name DECbit.
- The destination host then copies this congestion bit into the ACK it sends back to the source. Finally, the source adjusts its sending rate so as to avoid congestion.
- A single congestion bit is added to the packet header. A router sets this bit in a packet if its average queue length is greater than or equal to 1 at the time the packet arrives.
- This average queue length is measured over a time interval that spans the last busy+idle cycle, plus the current busy cycle.
- Figure shows the queue length at a router as a function of time. Essentially, the router calculates the area under the curve and divides this value by the time interval to compute the average queue length.
- Using a queue length of 1 as the trigger for setting the congestion bit is a trade-off between significant queuing (and hence higher throughput) and increased idle time (and hence lower delay). In other words, a queue length of 1 seems to optimize the power function.



Random Early Detection(RED)

- A second mechanism, called random early detection (RED), is similar to the DECbit scheme in that each router is programmed to monitor its own queue length and, when it detects that congestion is imminent, to notify the source to adjust its congestion window.
- The first is that rather than explicitly sending a congestion notification message to the source, RED is most commonly implemented such that it implicitly notifies the source of congestion by dropping one of its packets.
- The source is, therefore, effectively notified by the subsequent timeout or duplicate ACK.
- In case you haven't already guessed, RED is designed to be used in conjunction with TCP, which currently detects congestion by means of timeouts.
- As the "early" part of the RED acronym suggests, the gateway drops the packet earlier than it would have to, so as to notify the source that it should decrease its congestion window sooner than it would normally have.
- In other words, the router drops a few packets before it has exhausted its buffer space completely, so as to cause the source to slow down, with the hope that this will mean it does not have to drop lots of packets later on.

Quality of service (QoS):

- It is an internetworking issue that refers to a set of techniques and mechanisms that guarantee the performance of the network to deliver predictable service to an application program

QoS Characteristics:

Reliability :

Reliability is a characteristic that a flow needs in order to deliver the packets safe and sound to the destination. Lack of reliability means losing a packet or acknowledgment, which entails retransmission

Delay:

Source-to-destination delay is another flow characteristic. The applications can tolerate delay in different degrees. In this case, telephony, audio conferencing, video conferencing, and remote logging need minimum delay, while delay in file transfer or e-mail is less important.

Jitter:

Jitter is the variation in delay for packets belonging to the same flow. For example, if four packets depart at times 0, 1, 2, 3 and arrive at 20, 21, 22, 23, all have the same delay, 20 units of time

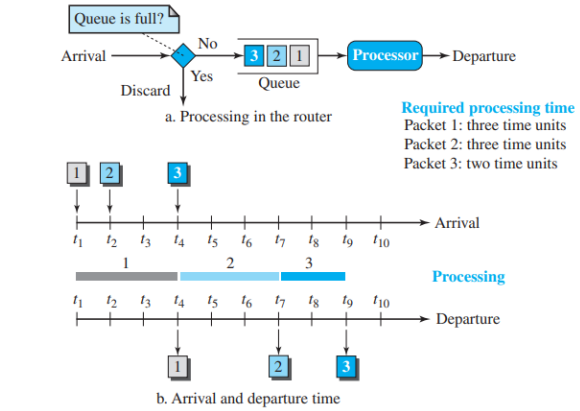
Flow Control to Improve QoS:

Scheduling:

- Treating packets in the Internet based on their required level of service can mostly happen at the routers. It is at a router that a packet may be delayed, suffer from jitters, be lost, or be assigned the required bandwidth.
- A good scheduling technique treats the different flows in a fair and appropriate manner. Several scheduling techniques are designed to improve the quality of service- FIFO queuing, priority queuing, and weighted fair queuing.

FIFO Queuing

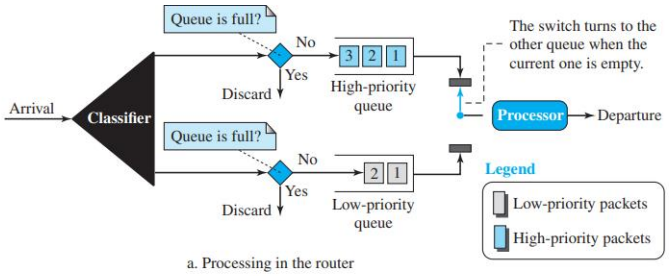
- In first-in, first-out (FIFO) queuing, packets wait in a buffer (queue) until the node (router) is ready to process them.
- If the average arrival rate is higher than the average processing rate, the queue will fill up and new packets will be discarded.
- A larger packet definitely may need a longer processing time. In the figure, packets 1 and 2 need three time units of processing, but packet 3, which is smaller, needs two time units.
- This means that packets may arrive with some delays but depart with different delays. If the packets belong to the same application, this produces jitters. If the packets belong to different applications, this also produces jitters for each application.



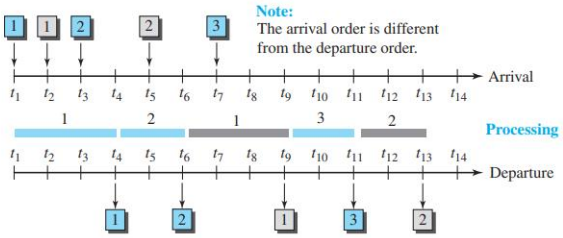
- FIFO queuing is the default scheduling in the Internet.
- With FIFO queuing, all packets are treated the same in a packet-switched network.
- The bandwidth allocated for each application depends on how many packets arrive at the router in a period of time.

Priority Queuing:

- Queuing delay in FIFO queuing often degrades quality of service in the network. A frame carrying real-time packets may have to wait a long time behind a frame carrying a small file.
- In priority queuing, packets are first assigned to a priority class. Each priority class has its own queue. The packets in the highest-priority queue are processed first. Packets in the lowest-priority queue are processed last.
- A priority queue can provide better QoS than the FIFO queue because higher-priority traffic, such as multimedia, can reach the destination with less delay
- If there is a continuous flow in a high-priority queue, the packets in the lower-priority queues will never have a chance to be processed. This is a condition called **starvation**. Severe starvation may result in dropping of some packets of lower priority.



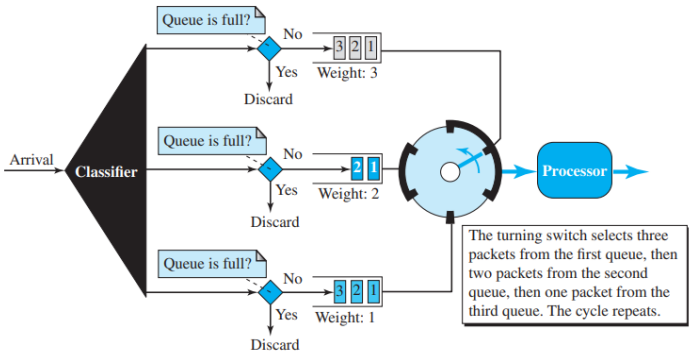
a. Processing in the router



b. Arrival and departure time

Weighted Fair Queuing :

- A better scheduling method is weighted fair queuing.
- In this technique, the packets are still assigned to different classes and admitted to different queues.
- The queues, however, are weighted based on the priority of the queues; higher priority means a higher weight.



- The system processes packets in each queue in a round-robin fashion with the number of packets selected from each queue based on the corresponding weight.
- For example, if the weights are 3, 2, and 1, three packets are processed from the first queue, two from the second queue, and one from the third queue.
- In weighted fair queuing, each class may receive a small amount of time in each time period. In other words, a fraction of time is devoted to serve each class of packets, but the fraction depends on the priority of the class

Client Server Programming:

In a client-server paradigm, communication at the application layer is between two running application programs called processes: a client and a server.

A client is a running program that initializes the communication by sending a request; a server is another application program that waits for a request from a client.

The server handles the request received from a client, prepares a result, and sends the result back to the client.

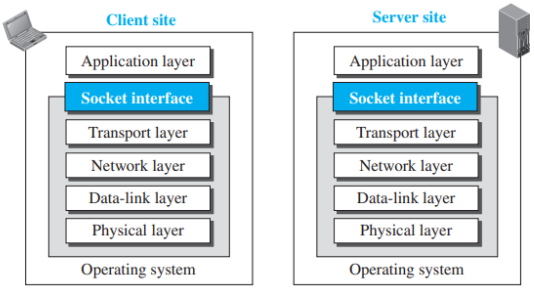
The server program needs to be started before start running the client program. The lifetime of server is infinite and client is finite

Application Programming Interface

A computer manufacturer needs to build the first four layers of the suite in the operating system and include an API.

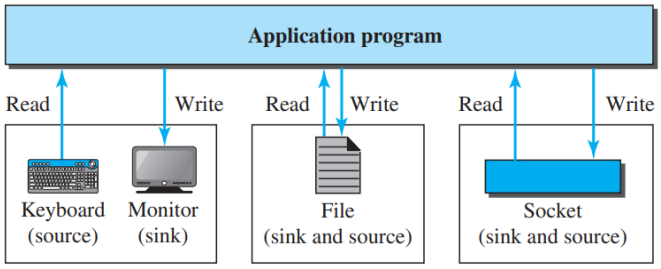
In this way, the processes running at the application layer are able to communicate with the operating system when sending and receiving messages through the Internet.

Several APIs have been designed for communication. Three among them are common: **socket interface, Transport Layer Interface (TLI), and STREAM.**

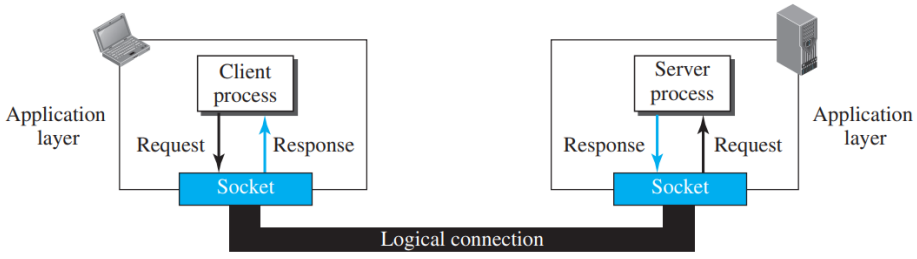


Socket Interface:

- Socket is supposed to behave like a terminal or a file. It is an abstraction.
- It is an object that is created and used by the application program.



- The client thinks that the socket is the entity that receives the request and gives the response; the server thinks that the socket is the one that has a request and needs the response.
- If we create two sockets, one at each end, and define the source and destination addresses correctly, we can use the available instructions to send and receive data



- The interaction between a client and a server is two-way communication.
- In a two-way communication, there are pair of addresses needed: local (sender) and remote (receiver).

- The local address in one direction is the remote address in the other direction and vice versa. Since communication in the client-server paradigm is between two sockets, we need a pair of socket addresses for communication: a local socket address and a remote socket address.

World Wide Web

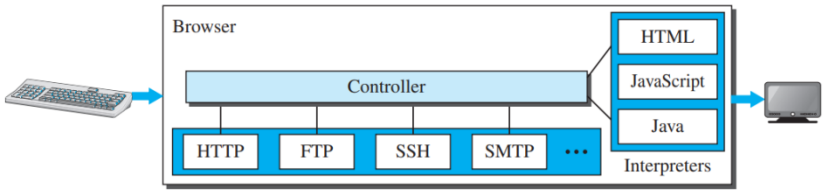
- The idea of the Web was first proposed by Tim Berners-Lee in 1989 at CERN in Europe. The commercial Web started in the early 1990s.
- The Web today is a repository of information in which the documents, called web pages, are distributed all over the world and related documents are linked together.
- The idea was to use a machine that automatically retrieved another document stored in the system when a link to it appeared in the document.
- The Web implemented this idea electronically to allow the linked document to be retrieved when the link was clicked by the user.
- Today, the term hypertext, coined to mean linked text documents, has been changed to hypermedia, to show that a web page can be a text document, an image, an audio file, or a video file.

Architecture:

- The WWW today is a distributed client-server service, in which a client using a browser can access a service using a server.
- However, the service provided is distributed over many locations called sites. Each site holds one or more web pages.
- Each web page, however, can contain some links to other web pages in the same or other sites.
- A web page can be simple or composite. A simple web page has no links to other web pages; a composite web page has one or more links to other web pages. Each web page is a file with a name and address.

Web Client (Browser):

- A variety of vendors offer commercial browsers that interpret and display a web page, and all of them use nearly the same architecture.
- Each browser usually consists of three parts: a controller, client protocols, and interpreters.



- The controller receives input from the keyboard or the mouse and uses the client programs to access the document.
- After the document has been accessed, the controller uses one of the interpreters to display the document on the screen.

- The interpreter can be HTML, Java, or JavaScript, depending on the type of document. Some commercial browsers include Internet Explorer, Netscape Navigator, and Firefox.

Web Server

- The web page is stored at the server. Each time a request arrives, the corresponding document is sent to the client.
- To improve efficiency, servers normally store requested files in a cache in memory.
- A server can also become more efficient through multithreading or multiprocessing.
- In this case, a server can answer more than one request at a time. Some popular web servers include **Apache and Microsoft Internet Information Server**.

Uniform Resource Locator (URL)

- A web page, as a file, needs to have a unique identifier to distinguish it from other web pages. To define a web page, we need three identifiers: host, port, and path.
- However, before defining the web page, it needs to tell the browser what clientserver application or protocol required for usage

  Protocol. The first identifier is the abbreviation for the client-server program that we need in order to access the web page.

  Host. The host identifier can be the IP address of the server or the unique name given to the server.

  Port. The port, a 16-bit integer, is normally predefined for the client-server application.

  Path. The path identifies the location and the name of the file in the underlying operating system.

  To combine these four pieces together, the uniform resource locator (URL) has been designed

protocol://host/path	Used most of the time
protocol://host:port/path	Used when port number is needed

Web Documents

- The documents in the WWW can be grouped into three broad categories: static, dynamic, and active.

Static Documents

- Static documents are fixed-content documents that are created and stored in a server.
- The client can get a copy of the document only.
- When a client accesses the document, a copy of the document is sent. The user can then use a browser to see the document.
- Static documents are prepared using one of several languages: HyperText Markup Language (HTML), Extensible Markup Language (XML), Extensible Style Language (XSL), and Extensible Hypertext Markup Language (XHTML).

Dynamic Documents

- A dynamic document is created by a web server whenever a browser requests the document. When a request arrives, the web server runs an application program or a script that creates the dynamic document.
- The server returns the result of the program or script as a response to the browser that requested the document.
- Because a fresh document is created for each request, the contents of a dynamic document may vary from one request to another.
- A very simple example of a dynamic document is the retrieval of the time and date from a server.

- The Common Gateway Interface (CGI) was used to retrieve a dynamic document in the past, some other options included are Java Server Pages (JSP), which uses the Java language for scripting, or Active Server Pages (ASP), a Microsoft product that uses Visual Basic language for scripting, or ColdFusion, which embeds queries in a Structured Query Language (SQL) database in the HTML document.

Active Documents:

- For many applications, we need a program or a script to be run at the client site. These are called active documents.
- When a browser requests an active document, the server sends a copy of the document or a script. The document is then run at the client site. One way to create an active document is to use Java applets, a program written in Java on the server.

Hyper Text Transfer Protocol (HTTP)

- The Hyper Text Transfer Protocol (HTTP) is used to define how the client-server programs can be written to retrieve web pages from the Web.
- An HTTP client sends a request; an HTTP server returns a response.
- The server uses the port number 80; the client uses a temporary port number.
- HTTP uses the services of TCP, a connection-oriented and reliable protocol.
- The client and server do not needs to worry about errors in messages exchanged or loss of any message, because the TCP is reliable and will take care of this matter

The hypertext concept embedded in web page documents may require several requests and responses.

- If the web pages, objects to be retrieved, are located on different servers, we do not have any other choice than to create a new TCP connection for retrieving each object.
- However, if some of the objects are located on the same server, we have two choices: to retrieve each object using a new TCP connection or to make a TCP connection and retrieve them all. The first method is referred to as a nonpersistent connection, the second as a persistent connection.

  Non persistent Connections:

  In a nonpersistent connection, one TCP connection is made for each request/response.

  1. The client opens a TCP connection and sends a request.
  2. The server sends the response and closes the connection.
  3. The client reads the data until it encounters an end-of-file marker; it then closes the connection.
     - If a file contains links to N different pictures in different files the connection must be opened and closed N + 1 times.
     - The nonpersistent strategy imposes high overhead on the server because the server needs N + 1 different buffers each time a connection is opened.

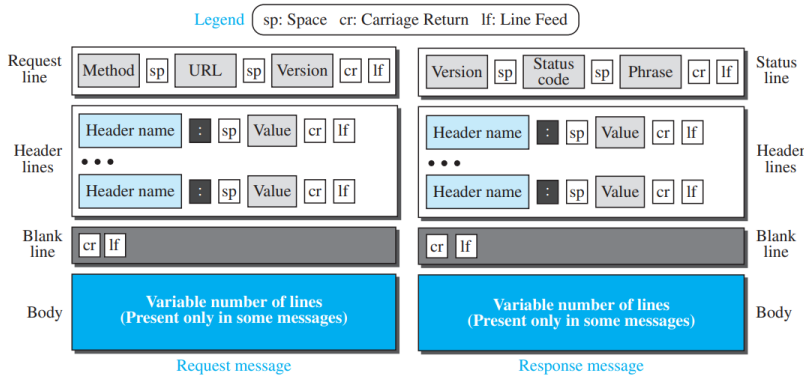
  Persistent Connections

     - In a persistent connection, the server leaves the connection open for more requests after sending a response.
     - The server can close the connection at the request of a client or if a time-out has been reached.
     - The sender usually sends the length of the data with each response. However, there are some occasions when the sender does not know the length of the data.

- In these cases, the server informs the client that the length is not known and closes the connection after sending the data so the client knows that the end of the data has been reached.
- Time and resources are saved using persistent connections.
- Only one set of buffers and variables needs to be set for the connection at each site. The round trip time for connection establishment and connection termination is saved.

Message Formats

- The HTTP protocol defines the format of the request and response messages.
- Each message is made of four sections. The first section in the request message is called the request line; the first section in the response message is called the status line.
- The other three sections have the same names in the request and response messages.

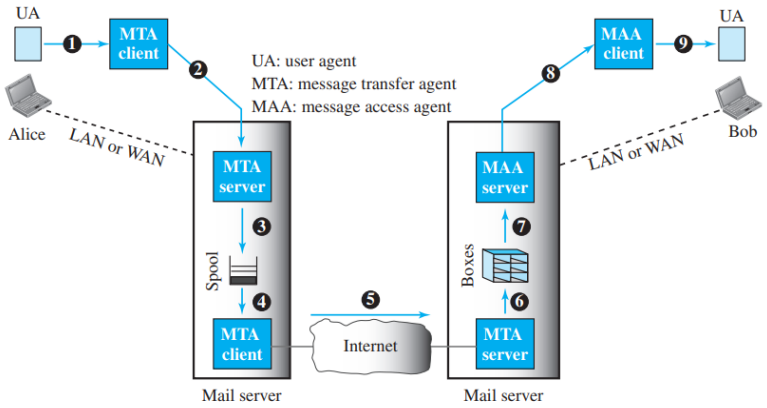


ELECTRONIC MAIL

- Electronic mail (or e-mail) allows users to exchange messages.
- In an application such as HTTP or FTP, the server program is running all the time, waiting for a request from a client.
- When the request arrives, the server provides the service. There is a request and there is a response.
- First, e-mail is considered a one-way transaction. When Alice sends an email to Bob, she may expect a response, but this is not a mandate.
- Bob may or may not respond. If he does respond, it is another one-way transaction. Second, it is neither feasible nor logical for Bob to run a server program and wait until someone sends an e-mail to him. Bob may turn off his computer when he is not using it.

Architecture:

- The sender and the receiver of the e-mail, Alice and Bob respectively, are connected via a LAN or a WAN to two mail servers.
- The administrator has created one mailbox for each user where the received messages are stored.
- A mailbox is part of a server hard drive, a special file with permission restrictions. Only the owner of the mailbox has access to it. The administrator has also created a queue (spool) to store messages waiting to be sent.
- A simple e-mail from Alice to Bob takes nine different steps. Alice and Bob use three different agents: a user agent (UA), a message transfer agent (MTA), and a message access agent (MAA).



- When Alice needs to send a message to Bob, she runs a UA program to prepare the message and send it to her mail server.
- The mail server at her site uses a queue (spool) to store messages waiting to be sent. The message, however, needs to be sent through the Internet from Alice's site to Bob's site using an MTA.
- Here two message transfer agents are needed: one client and one server. Like most client-server programs on the Internet, the server needs to run all the time because it does not know when a client will ask for a connection.
- The client, on the other hand, can be triggered by the system when there is a message in the queue to be sent.
- The user agent at the Bob site allows Bob to read the received message. Bob later uses an MAA client to retrieve the message from an MAA server running on the second server
- Bob cannot bypass the mail server and use the MTA server directly. To use the MTA server directly, Bob would need to run the MTA server all the time because he does not know when a message will arrive
- Second, note that Bob needs another pair of client-server programs: message access programs. This is because an MTA client-server program is a push program: the client pushes the message to the server. Bob needs a pull program

User Agent

- The first component of an electronic mail system is the user agent (UA). It provides service to the user to make the process of sending and receiving a message easier.
- A user agent is a software package (program) that composes, reads, replies to, and forwards messages. It also handles local mailboxes on the user computers. There are two types of user agents: command-driven and GUI-based. Commanddriven user agents belong to the early days of electronic mail.
- They are still present as the underlying user agents. A command-driven user agent normally accepts a onecharacter command from the keyboard to perform its task
- Modern user agents are GUI-based. They contain graphical user interface (GUI) components that allow the user to interact with the software by using both the keyboard and the mouse

Sending Mail

- To send mail, the user, through the UA, creates mail that looks very similar to postal mail. It has an envelope and a message. The envelope usually contains the sender address, the receiver address, and other information.

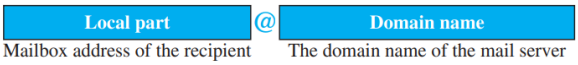
- The message contains the header and the body. The header of the message defines the sender, the receiver, the subject of the message, and some other information.
- The body of the message contains the actual information to be read by the recipient

Receiving Mail

- The user agent is triggered by the user .
- If a user has mail, the UA informs the user with a notice. If the user is ready to read the mail, a list is displayed in which each line contains a summary of the information about a particular message in the mailbox.
- The summary usually includes the sender mail address, the subject, and the time the mail was sent or received.
- The user can select any of the messages and display its contents on the screen.

Addresses

- To deliver mail, a mail handling system must use an addressing system with unique addresses.
- In the Internet, the address consists of two parts: a local part and a domain name, separated by an @ sign



- The local part defines the name of a special file, called the user mailbox, where all the mail received for a user is stored for retrieval by the message access agent.
- The second part of the address is the domain name. An organization usually selects one or more hosts to receive and send e-mail; they are sometimes called mail servers or exchangers.
- The domain name assigned to each mail exchanger either comes from the DNS database or is a logical name

EC3401 NETWORKS AND SECURITY

UNIT IV NETWORK SECURITY **9**

OSI Security Architecture – Attacks – Security Services and Mechanisms – Encryption –Advanced Encryption Standard – Public Key Cryptosystems – RSA Algorithm – Hash Functions – Secure Hash Algorithm – Digital Signature Algorithm

OSI Security Architecture

- International Telecommunication Union(ITU-T) Telecommunication Standardization Sector Recommended X.800, Security Architecture for OSI.
- It is a systematic approach. The OSI security architecture is useful to managers as a way of organizing the task of providing security.
- The OSI security architecture focuses on security attacks, mechanisms, and services. These can be defined briefly as

  Security attack: Any action that compromises the security of information owned by an organization.

  Security mechanism: A process that is designed to detect, prevent, or recover from a security attack.

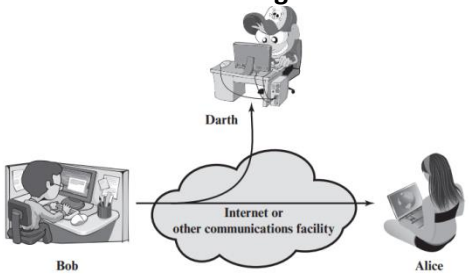
  Security service: A processing or communication service that enhances the security of the data processing systems and the information transfers of an organization.

Security Attacks:

- Security attacks mainly classified into two types. **Passive attacks and Active attacks**

Passive Attacks:

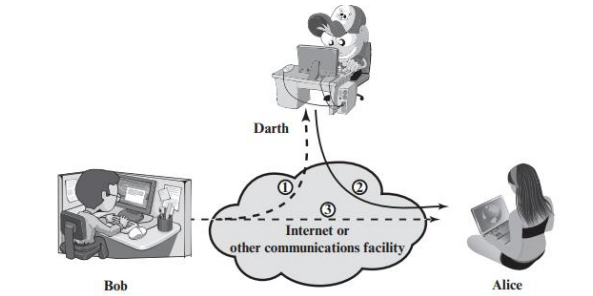
- Passive attacks are in the nature of listening and monitoring others transmissions. The goal of the opponent is to obtain information that is being transmitted.
- Two types of passive attacks are the release of **message contents and traffic analysis**.



- The release of message contents is easily understood. A telephone conversation, an electronic mail message, and a transferred file may contain sensitive or confidential information.
- A second type of passive attack, **traffic analysis**. Suppose that we had a way of masking the contents of messages or other information traffic so that opponents, even if they captured the message, could not extract the information from the message.
- Passive attacks are very difficult to detect, because they do not involve any alteration of the data.
- Typically, the message traffic is sent and received in an apparently normal fashion, and neither the sender nor receiver is aware that a third party has read the messages or observed the traffic pattern.

Active attacks

- It involves some modification of the data stream or the creation of a false stream and can be subdivided into four categories: masquerade, replay, modification of messages, and denial of service.



Masquerade:

It takes place when one entity pretends to be a different entity.

A masquerade attack usually includes one of the other forms of active attack.

Replay:

It involves the passive capture of a data unit and its subsequent retransmission to produce an unauthorized effect (paths 1, 2, and 3 active).

Modification of messages: It simply means that some portion of a legitimate message is altered, or that messages are delayed or reordered, to produce an unauthorized effect (paths 1 and 2 active).

Denial of service: It prevents or inhibits the normal use or management of communications facilities. This attack may have a specific target. Another form of service denial is the disruption of an entire network, either by disabling the network or by overloading it with messages so as to degrade performance.

Security Services:

- X.800 defines a security service as a service that is provided by a protocol layer of communicating open systems and that ensures adequate security of the systems or of data transfers.
- 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:

- The authentication service is concerned with assuring that a communication is authentic.
- In the case of a single message, such as a warning or alarm signal, the function of the authentication service is to assure the recipient that the message is from the source that it claims to be from.
- In the case of an ongoing interaction, such as the connection of a terminal to a host, two aspects are involved. First, at the time of connection initiation, the service assures that the two entities are authentic, that is, that each is the entity that it claims to be.
- Second, the service must assure that the connection is not interfered with in such a way that a third party can masquerade as one of the two legitimate parties for the purposes of unauthorized transmission or reception.

Two specific authentication services are defined in X.800:

■ **Peer entity authentication:** Provides for the corroboration of the identity of a peer entity in an association. Two entities are considered peers if they implement to same protocol in different systems; for example two TCP modules in two communicating systems. Peer entity

authentication is provided for use at the establishment of, or at times during the data transfer phase of, a connection. It attempts to provide confidence that an entity is not performing either a masquerade or an unauthorized replay of a previous connection.

■ **Data origin authentication:** Provides for the corroboration of the source of a data unit. It does not provide protection against the duplication or modification of data units. This type of service supports applications like electronic mail, where there are no prior interactions between the communicating entities

AUTHENTICATION

The assurance that the communicating entity is the one that it claims to be.

Peer Entity Authentication
Used in association with a logical connection to provide confidence in the identity of the entities connected.

Data-Origin Authentication
In a connectionless transfer, provides assurance that the source of received data is as claimed.

ACCESS CONTROL

The prevention of unauthorized use of a resource (i.e., this service controls who can have access to a resource, under what conditions access can occur, and what those accessing the resource are allowed to do).

DATA CONFIDENTIALITY

The protection of data from unauthorized disclosure.

Connection Confidentiality
The protection of all user data on a connection.

Connectionless Confidentiality
The protection of all user data in a single data block.

Selective-Field Confidentiality
The confidentiality of selected fields within the user data on a connection or in a single data block.

Traffic-Flow Confidentiality
The protection of the information that might be derived from observation of traffic flows.

DATA INTEGRITY

The assurance that data received are exactly as sent by an authorized entity (i.e., contain no modification, insertion, deletion, or replay).

Connection Integrity with Recovery
Provides for the integrity of all user data on a connection and detects any modification, insertion, deletion, or replay of any data within an entire data sequence, with recovery attempted.

Connection Integrity without Recovery
As above, but provides only detection without recovery.

Selective-Field Connection Integrity
Provides for the integrity of selected fields within the user data of a data block transferred over a connection and takes the form of determination of whether the selected fields have been modified, inserted, deleted, or replayed.

Connectionless Integrity
Provides for the integrity of a single connectionless data block and may take the form of detection of data modification. Additionally, a limited form of replay detection may be provided.

Selective-Field Connectionless Integrity
Provides for the integrity of selected fields within a single connectionless data block; takes the form of determination of whether the selected fields have been modified.

NONREPUDIATION

Provides protection against denial by one of the entities involved in a communication of having participated in all or part of the communication.

Nonrepudiation, Origin
Proof that the message was sent by the specified party.

Nonrepudiation, Destination
Proof that the message was received by the specified party.

Data Confidentiality:

- Data Confidentiality is the protection of transmitted data from passive attacks.
- With respect to the content of a data transmission, several levels of protection can be identified. The broadest service protects all user data transmitted between two users over a period of time.
- Narrower forms of this service can also be defined, including the protection of a single message or even specific fields within a message.
- The other aspect of confidentiality is 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:

- Data integrity can apply to a stream of messages, a single message, or selected fields within a message. The most useful and straightforward approach is total stream protection.
- A connection-oriented integrity service, one that deals with a stream of messages, assures that messages are received as sent with no duplication, insertion, modification, reordering, or replays.
- The destruction of data is also covered under this service.
- Thus, the connection-oriented integrity service addresses both message stream modification and denial of service.
- A connectionless integrity service, one that deals with individual messages without regard to any larger context, generally provides protection against message modification only.
- Because the integrity service relates to active attacks, we are concerned with detection rather than prevention.
- If a violation of integrity is detected, then the service may simply report this violation, and some other portion of software or human intervention is required to recover from the violation. Alternatively, there are mechanisms available to recover from the loss of integrity of data.

Security Mechanisms:

- The mechanisms are divided into those that are implemented in a specific protocol layer, such as TCP or an application-layer protocol, and those that are not specific to any particular protocol layer or security service.

Encipherment or Encryption

The use of mathematical algorithms to transform data into a form that is not readily intelligible. The transformation and subsequent recovery of the data depend on an algorithm and zero or more encryption keys.

Digital Signature:

Data appended to, or a cryptographic transformation of, a data unit that allows a recipient of the data unit to prove the source and integrity of the data unit and protect against forgery (e.g., by the recipient)

Access Control -A variety of mechanisms that enforce access rights to resources.

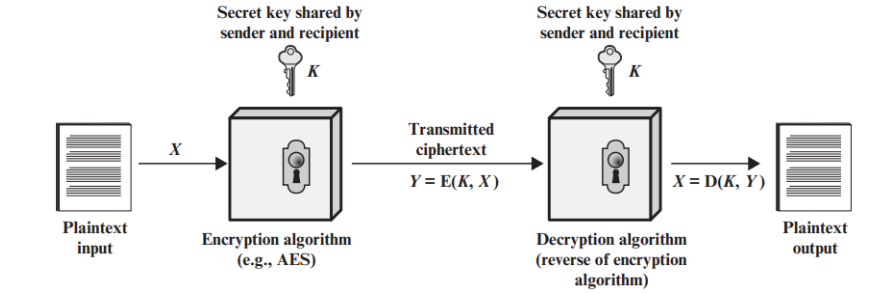
Data Integrity -A variety of mechanisms used to assure the integrity of a data unit or stream of data units

Security Label- The marking bound to a resource that names or designates the security attributes of that resource.

Event Detection -Detection of security-relevant events.

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 Deals with requests from mechanisms, such as event handling and management functions, and takes recovery actions.

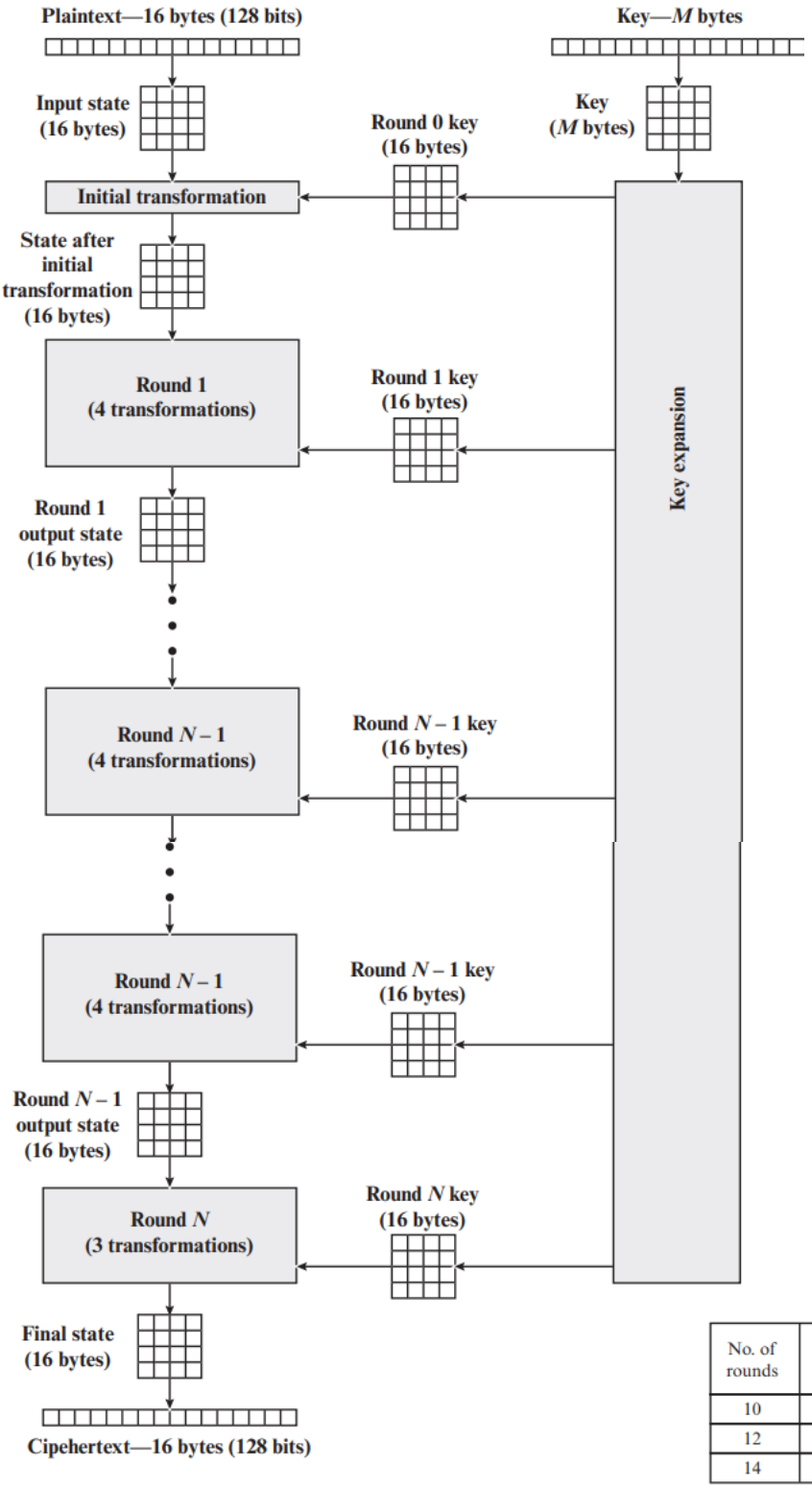
Encryption:
An original message is known as the **plaintext**, while the coded message is called the **ciphertext**. The process of converting from plaintext to ciphertext is known as **enciphering or encryption.** restoring the plaintext from the ciphertext is **deciphering or decryption**. The many schemes used for **encryption constitute the area of study known as cryptography**.
Symmetric Cipher Model:
Plaintext: This is the original intelligible message or data that is fed into the algorithm as input.
Encryption algorithm: The encryption algorithm performs various substitutions and transformations on the plaintext.
Secret key: The secret key is also input to the encryption algorithm. The key is a value independent of the plaintext and of the algorithm. The algorithm will produce a different output depending on the specific key being used at the time.
Ciphertext: This is the scrambled message produced as output. It depends on the plaintext and the secret key. For a given message, two different keys will produce two different ciphertexts.
Decryption algorithm: This is essentially the encryption algorithm run in reverse. It takes the ciphertext and the secret key and produces the original plaintext



The Advanced Encryption Standard (AES)

- Advanced Encryption Standard was published by the National Institute of Standards and Technology (NIST) in 2001.
- AES is widely used today as it is a much stronger than Data Encryption standard despite being harder to implement
- In AES, all operations are performed on 8-bit bytes. In particular, the arithmetic operations of addition, multiplication, and division are performed over the finite field.
- The cipher takes a plaintext block size of 128 bits, or 16 bytes. The key length can be 16, 24, or 32 bytes (128, 192, or 256 bits).
- The algorithm is referred to as AES-128, AES-192, or AES-256, depending on the key length. The input to the encryption and decryption algorithms is a single 128-bit block.
- This block is depicted as a 4 * 4 square matrix of bytes. This block is copied into the State array, which is modified at each stage of encryption or decryption. After the final stage, State is copied to an output matrix.
- The key is depicted as a square matrix of bytes. This key is then expanded into an array of key schedule words

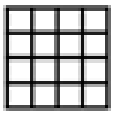
- Each word is four bytes, and the total key schedule is 44 words for the 128-bit key



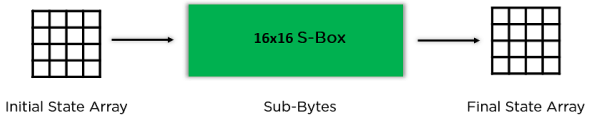
No. of rounds	Key Length (bytes)
10	16
12	24
14	32

- The cipher consists of N rounds, where the number of rounds depends on the key length: 10 rounds for a 16-byte key, 12 rounds for a 24-byte key, and 14 rounds for a 32-byte key
- The first N - 1 rounds consist of four distinct transformation functions: SubBytes, ShiftRows, MixColumns, and AddRoundKey, which are described subsequently. The final round contains only three transformations, and there is a initial single transformation

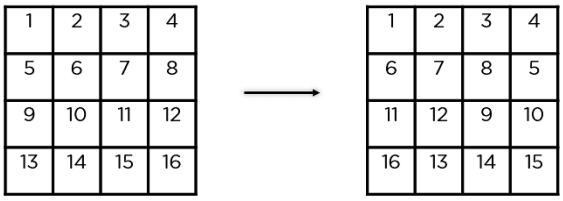
  Eg: AES considers each block as a 16 byte (4 byte x 4 byte = 128) grid in a column major arrangement.



- Each round comprises of 4 steps :SubBytes, Shift Rows, MixColumns, Add Round Key
- The last round doesn't have the MixColumns round.
- The SubBytes does the substitution and ShiftRows and MixColumns performs the permutation in the algorithm.
- **SubBytes :**
  This step implements the substitution.
- In this step each byte is substituted by another byte. Its performed using a lookup table also called the S-box.
- This substitution is done in a way that a byte is never substituted by itself and also not substituted by another byte which is a compliment of the current byte. The result of this step is a 16 byte (4 x 4) matrix like before.



- **ShiftRows :**
  This step is just as it sounds. Each row is shifted a particular number of times.
- The first row is not shifted
- The second row is shifted once to the left.
- The third row is shifted twice to the left.
  The fourth row is shifted thrice to the left. (A left circular shift is performed.)

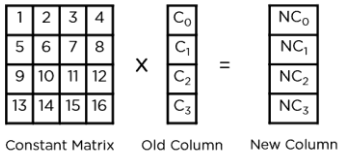


1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16

→

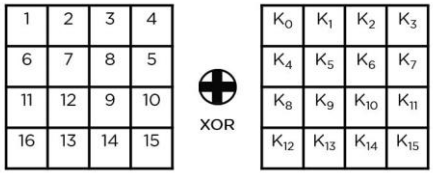
1	2	3	4
6	7	8	5
11	12	9	10
16	13	14	15

MixColumns :
This step is basically a matrix multiplication. Each column is multiplied with a specific matrix and thus the position of each byte in the column is changed as a result.
This step is skipped in the last round.

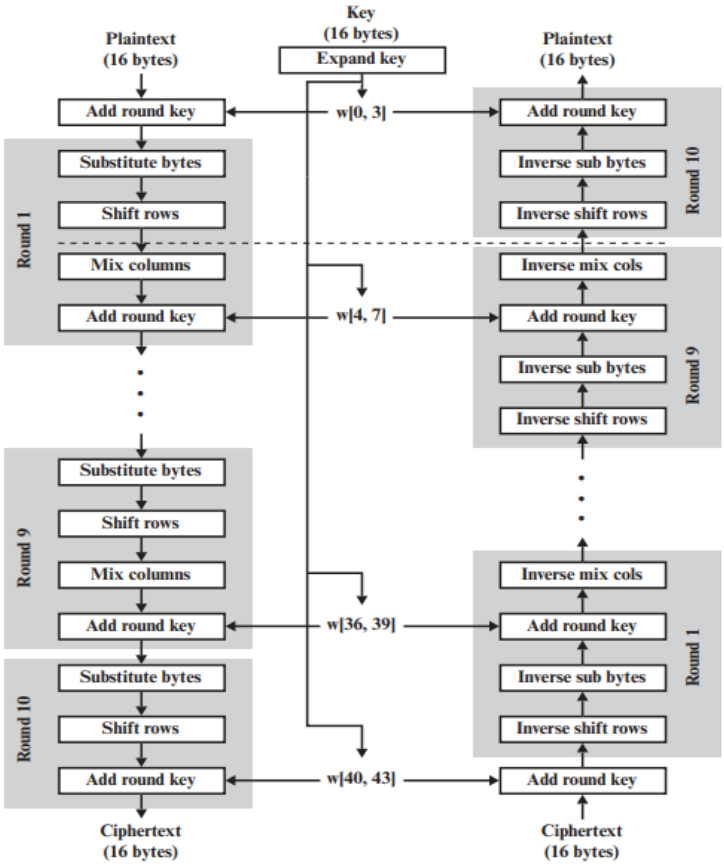


Add Round Keys :

The resultant output of the previous stage is XOR-ed with the corresponding round key. Here, the 16 bytes is not considered as a grid but just as 128 bits of data.



After all these rounds 128 bits of encrypted data is given back as output. This process is repeated until all the data to be encrypted undergoes this process.



Decryption :

The stages in the rounds can be easily undone as these stages have an opposite to it which when performed reverts the changes. Each 128 blocks goes through the 10,12 or 14 rounds depending on the key size.

The stages of each round in decryption is as follows:

- Add round key
- Inverse Mix Columns
- Shift Rows
- Inverse SubByte

The decryption process is the encryption process done in reverse .

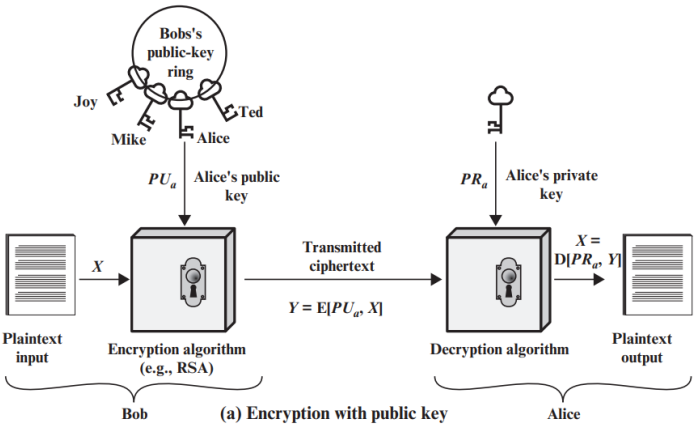
Public Key Cryptosystems:

Asymmetric algorithms rely on one key for encryption and a different but related key for decryption.

It is computationally infeasible to determine the decryption key given only knowledge of the cryptographic algorithm and the encryption key.

A public-key encryption scheme has six ingredients

- **Plaintext:** This is the readable message or data that is fed into the algorithm as input.
- **Encryption algorithm:** The encryption algorithm performs various transformations on the plaintext.



Public and private keys: This is a pair of keys that have been selected so that if one is used for encryption, the other is used for decryption. The exact transformations performed by the algorithm depend on the public or private key that is provided as input.

Ciphertext: This is the encrypted message produced as output. It depends on the plaintext and the key. For a given message, two different keys will produce two different cipher texts.

Decryption algorithm: This algorithm accepts the ciphertext and the matching key and produces the original plaintext.

1. Each user generates a pair of keys to be used for the encryption and decryption of messages.
2. Each user places one of the two keys in a public register or other accessible file. This is the public key. The companion key is kept private.
3. If Bob wishes to send a confidential message to Alice, Bob encrypts the message using Alice's public key.
4. When Alice receives the message, she decrypts it using her private key. No other recipient can decrypt the message because only Alice knows Alice's private key.

Conventional Encryption	Public-Key Encryption
Needed to Work: 1. The same algorithm with the same key is used for encryption and decryption. 2. The sender and receiver must share the algorithm and the key. *Needed for Security:* 1. The key must be kept secret. 2. It must be impossible or at least impractical to decipher a message if the key is kept secret. 3. Knowledge of the algorithm plus samples of ciphertext must be insufficient to determine the key.	*Needed to Work:* 1. One algorithm is used for encryption and a related algorithm for decryption with a pair of keys, one for encryption and one for decryption. 2. The sender and receiver must each have one of the matched pair of keys (not the same one). *Needed for Security:* 1. One of the two keys must be kept secret. 2. It must be impossible or at least impractical to decipher a message if one of the keys is kept secret. 3. Knowledge of the algorithm plus one of the keys plus samples of ciphertext must be insufficient to determine the other key.

Applications of Public Key cryptosystems:

Encryption/decryption: The sender encrypts a message with the recipient's public key, and the recipient decrypts the message with the recipient's private key.

Digital signature: The sender "signs" a message with its private key. Signing is achieved by a cryptographic algorithm applied to the message or to a small block of data that is a function of the message.

Key exchange: Two sides cooperate to exchange a session key, which is a secret key for symmetric encryption generated for use for a particular transaction and valid for a short period of time

RSA Algorithm:

- Rivest-Shamir-Adleman (RSA) scheme is a most widely accepted and implemented general-purpose approach to public-key encryption.
- The RSA scheme is a cipher in which the plaintext and ciphertext are integers between 0 and n - 1 for some n.
- A typical size for n is 1024 bits, or 309 decimal digits. That is, n is less than 2^{1024}.
- RSA makes use of an expression with exponentials. Plaintext is encrypted in blocks, with each block having a binary value less than some number n. That is, the block size must be less than or equal to $\log_2(n) + 1$

Steps in RSA Algorithm:

Step-1: Select two prime numbers p and q where $p \neq q$.

Step-2: Calculate $n = p * q$.

Step-3: Calculate $\Phi(n) = (p-1) * (q-1)$.

Step-4: Select e such that, e is relatively prime to $\Phi(n)$, i.e. $(e, \Phi(n)) = 1$ and $1 < e < \Phi(n)$

Step-5: Calculate $d = e^{-1} \bmod \Phi(n)$ or $ed = 1 \bmod \Phi(n)$.

Step-6: Public key = {e, n}, private key = {d, n}.

Step-7: Find out cipher text using the formula,

$C = P^e \bmod n$ where, $P < n$ where C = Cipher text, P = Plain text, e = Encryption key and n=block size.

Step-8: $P = C^d \bmod n$. Plain text P can be obtain using the given formula. where, d = decryption key

Step – 1: Select two prime numbers p and q where $p \neq q$.

Example, Two prime numbers p = 13, q = 11.

Step – 2: Calculate n = p * q.

Example, n = p * q = 13 * 11 = 143.

Step – 3: Calculate $\Phi(n) = (p-1) * (q-1)$.

Example, $\Phi(n) = (13 - 1) * (11 - 1) = 12 * 10 = 120$.

Step – 4: Select e such that, e is relatively prime to $\Phi(n)$, i.e. $(e, \Phi(n)) = 1$ and $1 < e < \Phi(n)$.

Example, Select e = 13, gcd (13, 120) = 1.

Step – 4: Select e such that, e is relatively prime to $\Phi(n)$, i.e. $(e, \Phi(n)) = 1$ and $1 < e < \Phi(n)$.

Example, Select e = 13, gcd (13, 120) = 1.

Step – 5: Calculate $d = e^{-1} \bmod \Phi(n)$ or $e * d = 1 \bmod \Phi(n)$

Example, Finding d: $e * d \bmod \Phi(n) = 1$

$13 * d \bmod 120 = 1$

$d = ((\Phi(n) * i) + 1) / e$

$d = (120 + 1) / 13 = 9.30$ ($\because i = 1$)

$d = (240 + 1) / 13 = 18.53$ ($\because i = 2$)

$d = (360 + 1) / 13 = 27.76$ ($\because i = 3$)

$d = (480 + 1) / 13 = 37$ ($\because i = 4$))

Step – 6: Public key = {e, n}, private key = {d, n}.

Example, Public key = {13, 143} and private key = {37, 143}.

Step – 7: Find out *cipher text* using the formula, $C = P^e \bmod n$ where, P < n.

Example, Plain text P = 13. (Where, P < n)

$C = P^e \bmod n = 13^{13} \bmod 143 = 52$.

Step – 8: $P = C^d \bmod n$. Plain text P can be obtain using the given formula.

Example, Cipher text C = 52

$P = C^d \bmod n = 52^{37} \bmod 143 = 13$.

Question: P and Q are two prime numbers. P=7, and Q=17. Take public key E=5. If plain text value is 6, then what will be cipher text value according to RSA algorithm? Again calculate plain text value from cipher text.

Solution:

1. Two prime numbers P=7, Q=17
2. n = P * Q = 17 * 7 = 119 **n = 119**
3. Φ(n) = (P–1) * (Q –1) = (17 – 1) * (7 – 1) = 16 * 6 = 96 **Φ(n) = 96**
4. Public key E = 5. **E= 5**
5. Calculate d = 77. d = ((Φ(n) * i) + 1) / e **d= 77**
   d = ((96*1)+1) / 5 = 19.4
   d = ((96*2)+1) / 5 = 38.6
   d = ((96*3)+1) / 5 = 57.8
   d = ((96*4)+1) / 5 = 77 (Stop finding d because getting integer value)
6. Public key = {e, n} = {5, 119}, private key = {d, n} = {77, 119}.
7. Plain text PT = 6, CT = PT^E mod n = 6^5 mod 119 = 41. **Cipher Text = 41**
8. Cipher text CT = 41, PT = CT^d mod n = 41^{77} mod 119 = 6. **Plain Text = 6**

Question: In a public key cryptosystem using RSA algorithm, user uses two prime numbers 5 and 7. He chooses 11 as Encryption key, find out decryption key. What will be the cipher text, if the plaintext is 2? Decrypt the cipher text, what will be the value of plain text?

Solution:

1. Two prime numbers p = 5, q = 7
2. n = p * q = 5 * 7 = 35 **n = 35**
3. Φ(n) = (p–1) * (q –1) = (5 – 1) * (7 – 1) = 4 * 6 = 24 **Φ(n) = 24**
4. Public key e = 11. **e= 11**
5. Calculate d = 11. d = ((Φ(n) * i) + 1) / e **d= 11**
6. Public key = {e, n} = {11, 35}, private key = {d, n} = {11, 35}.
7. Plain text P = 2, C = P^e mod n = 2^{11} mod 35 = 18. **Cipher Text = 18**
8. Cipher text C = 18, P = C^d mod n = 18^{11} mod 35 = 2. **Plain Text = 2**

Question: In a public key cryptosystem using RSA algorithm, user uses two prime numbers 5 and 7. He chooses 11 as Encryption key, find out decryption key. What will be the cipher text, if the plaintext is 2? Decrypt the cipher text, what will be the value of plain text?

Solution:

1. Two prime numbers p = 5, q = 7
2. n = p * q = 5 * 7 = 35 **n = 35**
3. Φ(n) = (p–1) * (q –1) = (5 – 1) * (7 – 1) = 4 * 6 = 24 **Φ(n) = 24**
4. Public key e = 11. **e= 11**
5. Calculate d = 11. d = ((Φ(n) * i) + 1) / e **d= 11**
6. Public key = {e, n} = {11, 35}, private key = {d, n} = {11, 35}.
7. Plain text P = 2, C = P^e mod n = 2^{11} mod 35 = 18. **Cipher Text = 18**
8. Cipher text C = 18, P = C^d mod n = 18^{11} mod 35 = 2. **Plain Text = 2**

Hash Function:

- Cryptographic Hash is a Hash function that takes random size input and yields a fixed-size output. It is easy to calculate but challenging to retrieve the original data.
- It is strong and difficult to duplicate the same hash with unique inputs and is a one-way function so revert is not possible. Hashing is also known by different names such as Digest, Message Digest, Checksum, etc

Properties of Cryptography Hash Function:

The ideal cryptographic hash function has the following main properties:

1. **Deterministic:** This means that the same message always results in the same hash.
2. **Quick:** It is quick to compute the hash value for any given message.
3. Avalanche Effect: This means that every minor change in the message results in a major change in the hash value.
4. **One-Way Function:** It is not possible to reverse the cryptographic hash function to get to the data.
5. **Collision Resistance:** It is infeasible to find two different messages that produce the same hash value.
6. **Pre-Image Resistance:** The hash value shouldn't be predictable from the given string and vice versa.

Applications of Hash Functions:

Message authentication:

It is a mechanism or service used to verify the integrity of a message.

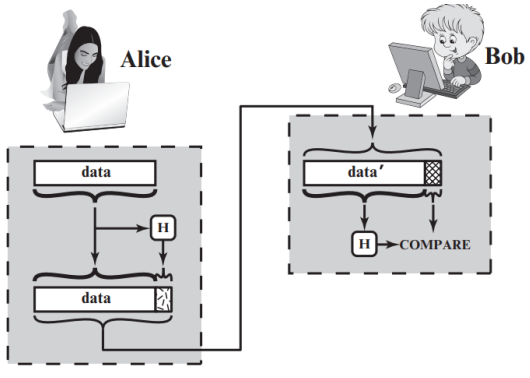
Message authentication assures that data received are exactly as sent (i.e., there is no modification, insertion, deletion, or replay).

When a hash function is used to provide message authentication, the hash function value is often referred to as a message digest.

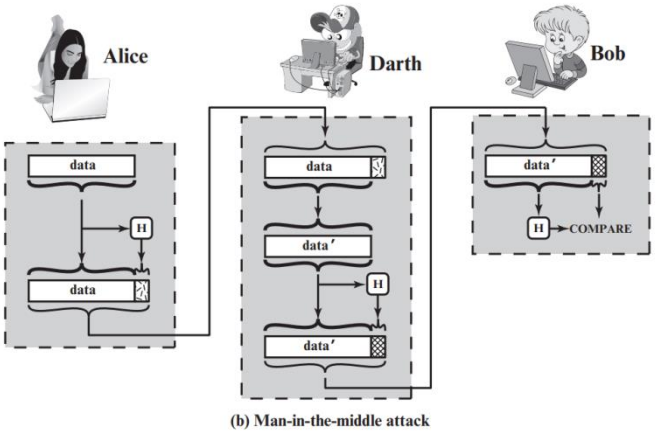
The sender computes a hash value as a function of the bits in the message and transmits both the hash value and the message. The receiver performs the same hash calculation on the message bits and compares this value with the incoming hash value.

If there is a mismatch, the receiver knows that the message has been altered. The hash value must be transmitted in a secure fashion.

That is, the hash value must be protected so that if an adversary alters or replaces the message, it is not feasible for adversary to also alter the hash value to fool the receiver. In this example, Alice transmits a data block and attaches a hash value. Darth intercepts the message, alters or replaces the data block, and calculates and attaches a new hash value. Bob receives the altered data with the new hash value and does not detect the change. To prevent this attack, the hash value generated by Alice must be protected.



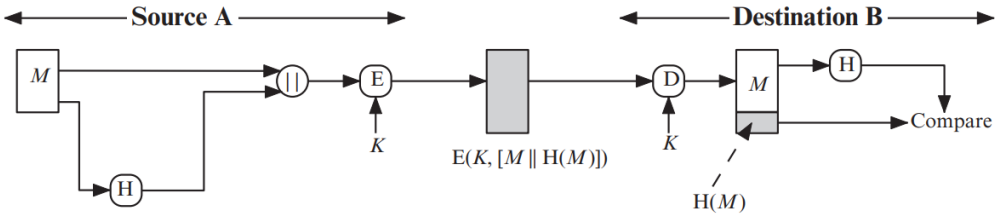
(a) Use of hash function to check data integrity



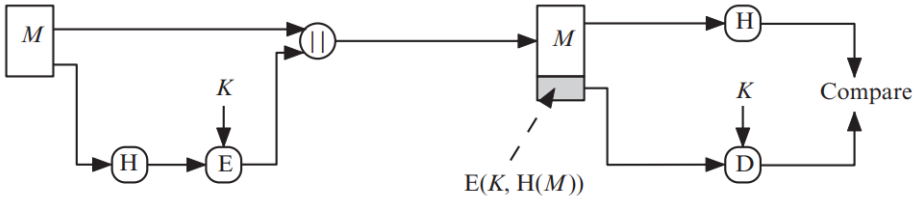
(b) Man-in-the-middle attack

A variety of ways in which a hash code can be used to provide message authentication, as follows.

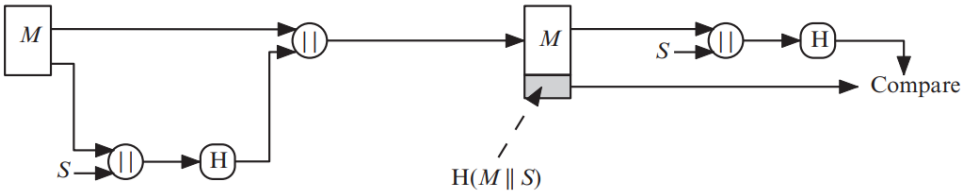
a. The message plus concatenated hash code is encrypted using symmetric encryption. Because only A and B share the secret key, the message must have come from A and has not been altered. The hash code provides the structure or redundancy required to achieve authentication. Because encryption is applied to the entire message plus hash code, confidentiality is also provided.



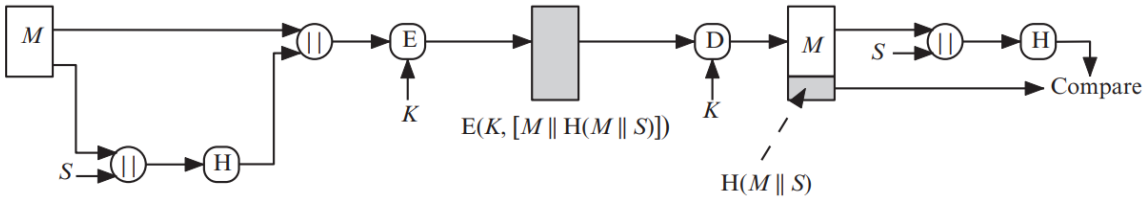
b. Only the hash code is encrypted, using symmetric encryption. This reduces the processing burden for those applications that do not require confidentiality



c. It is possible to use a hash function but no encryption for message authentication. The technique assumes that the two communicating parties share a common secret value S. A computes the hash value over the concatenation of M and S and appends the resulting hash value to M. Because B possesses S, it can recompute the hash value to verify. Because the secret value itself is not sent, an opponent cannot modify an intercepted message and cannot generate a false message.

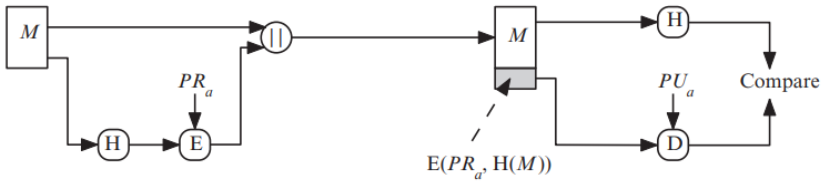


d. Confidentiality can be added to the approach of method (c) by encrypting the entire message plus the hash code.

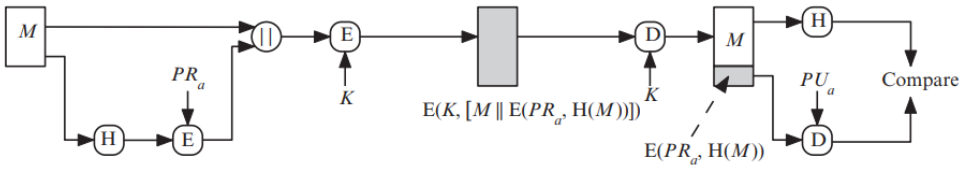


Digital Signature:

- An important application, which is similar to the message authentication application, is the digital signature. The operation of the digital signature is similar to that of the MAC. In the case of the digital signature, the hash value of a message is encrypted with a user's private key.
- Anyone who knows the user's public key can verify the integrity of the message that is associated with the digital signature. In this case, an attacker who wishes to alter the message would need to know the user's private key. The hash code is encrypted, using public-key encryption with the sender's private key.
- It also provides a digital signature, because only the sender could have produced the encrypted hash code. In fact, this is the essence of the digital signature technique.



- If confidentiality as well as a digital signature is desired, then the message plus the private-key-encrypted hash code can be encrypted using a symmetric secret key. This is a common technique.

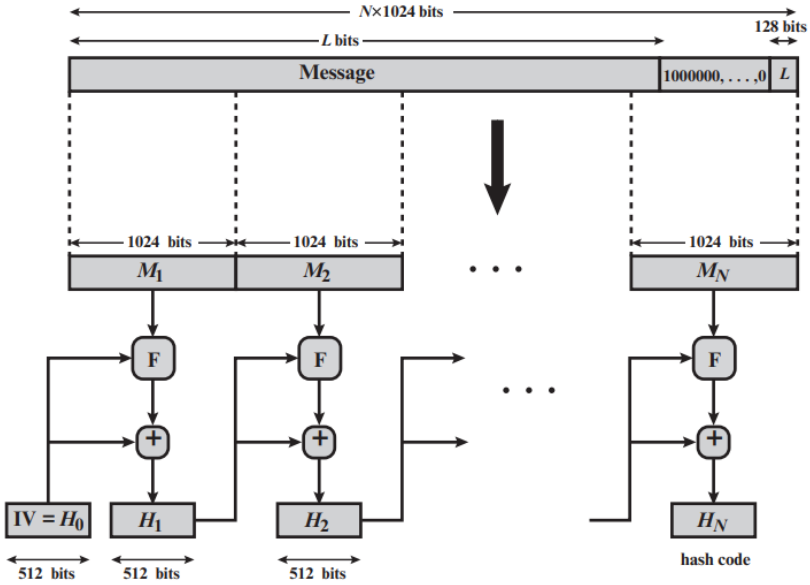


Secure Hash Algorithm:

- SHA developed by National Institute of Standards and Technology (NIST) and published as a federal information processing standard (FIPS 180) in 1993.
- SHA-1 produces a hash value of 160 bits. In 2002, NIST produced a revised version of the standard, FIPS 180-2, that defined three new versions of SHA, with hash value lengths of 256, 384, and 512 bits, known as SHA-256, SHA-384, and SHA-512, respectively.

SHA-512 Logic:

- The algorithm takes as input a message with a maximum length of less than 2128 bits and produces as output a 512-bit message digest. The input is processed in 1024-bit blocks.
  Step 1: Append padding bits. The message is padded so that its length is congruent to 896 modulo 1024 [length K 896(mod 1024)]. Padding is always added, even if the message is already of the desired length. Thus, the number of padding bits is in the range of 1 to 1024. The padding consists of a single 1 bit followed by the necessary number of 0 bits.



Step 2 Append length. A block of 128 bits is appended to the message. This block is treated as an unsigned 128-bit integer (most significant byte first) and contains the length of the original message in bits (before the padding). The outcome of the first two steps yields a message that is an integer multiple of 1024 bits in length. In the expanded message is represented as the sequence of 1024-bit blocks M_1, M_2….M_N, so that the total length of the expanded message is N * 1024 bits.

Step 3 Initialize hash buffer. A 512-bit buffer is used to hold intermediate and final results of the hash function. The buffer can be represented as eight 64-bit registers (a, b, c, d, e, f, g, h).

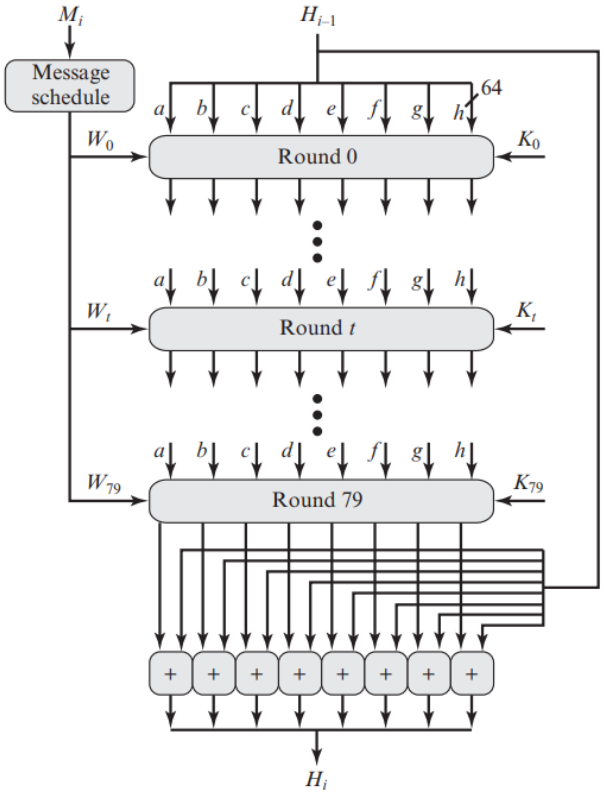
These registers are initialized to the following 64-bit integers (hexadecimal values):

a = 6A09E667F3BCC908 b = BB67AE8584CAA73B c = 3C6EF372FE94F82B
d = A54FF53A5F1D36F1 e= 510E527FADE682D1 f=9B05688C2B3E6C1F
g = 1F83D9ABFB41BD6B h = 5BE0CD19137E2179

These values are stored in big-endian format, which is the most significant byte of a word in the low-address (leftmost) byte position. These words were obtained by taking the first sixty-four bits of the fractional parts of the square roots of the first eight prime numbers

Step 4 Process message in 1024-bit (128-byte) blocks.

- The heart of the algorithm is a module that consists of 80 rounds.
- Each round takes as input the 512-bit buffer value, abcdefgh, and updates the contents of the buffer.
- At input to the first round, the buffer has the value of the intermediate hash value, Hi-1. Each round t makes use of a 64-bit value W_t, derived from the current 1024-bit block being processed (Mi). These values are derived using a message schedule described subsequently.
- Each round also makes use of an additive constant K_t , where 0 … t … 79 indicates one of the 80 rounds. These words represent the first 64 bits of the fractional parts of the cube roots of the first 80 prime numbers.
- The constants provide a “randomized” set of 64-bit patterns, which should eliminate any regularity in the input data.
- The output of the eightieth round is added to the input to the first round (Hi-1) to produce Hi.
- The addition is done independently for each of the eight words in the buffer with each of the corresponding words in Hi-1, using addition modulo 264.



Step 5 **Output.** After all N 1024-bit blocks have been processed, the output from the Nth stage is the 512-bit message digest.

We can summarize the behavior of SHA-512 as follows:

$$H_0 = \text{IV}$$
$$H_i = \text{SUM}_{64}(H_{i-1}, \text{abcdefgh}_i)$$
$$MD = H_N$$

where

IV = initial value of the abcdefgh buffer, defined in step 3

abcdefgh_i = the output of the last round of processing of the ith message block

N = the number of blocks in the message (including padding and length fields)

SUM_{64} = addition modulo 2^{64} performed separately on each word of the pair of inputs

MD = final message digest value

Advanced Encryption Standard Problems

Given the plaintext {000102030405060708090A0B0C0D0E0F} and the key {01010101010101010101010101010101}

a. Show the original contents of State, displayed as a 4×4 matrix.

b. Show the value of State after initial Add Round Key.

c. Show the value of State after Sub Bytes.

d. Show the value of State after Shift Rows

e. Show the value of State after Mix Columns

$$\text{state} = \begin{bmatrix} 00 & 04 & 08 & 0C \\ 01 & 05 & 09 & 0D \\ 02 & 06 & 0A & 0E \\ 03 & 07 & 0B & 0F \end{bmatrix}$$

Add 0^{th} round key

$$\text{Key} = \begin{bmatrix} 01 & 01 & 01 & 01 \\ 01 & 01 & 01 & 01 \\ 01 & 01 & 01 & 01 \\ 01 & 01 & 01 & 01 \end{bmatrix}$$

$$\begin{bmatrix} 00 & 04 & 08 & 0C \\ 01 & 05 & 09 & 0D \\ 02 & 06 & 0A & 0E \\ 03 & 07 & 0B & 0F \end{bmatrix} \oplus \begin{bmatrix} 01 & 01 & 01 & 01 \\ 01 & 01 & 01 & 01 \\ 01 & 01 & 01 & 01 \\ 01 & 01 & 01 & 01 \end{bmatrix} = \begin{bmatrix} 01 & 05 & 09 & 0D \\ 00 & 04 & 08 & 0C \\ 03 & 07 & 0B & 0F \\ 02 & 06 & 0A & 0E \end{bmatrix}$$

c. Byte Substitution

$$\begin{bmatrix} 01 & 05 & 09 & 0D \\ 00 & 04 & 08 & 0C \\ 03 & 07 & 0B & 0F \\ 02 & 06 & 0A & 0E \end{bmatrix} \rightarrow \begin{bmatrix} 7C & 6B & 01 & D7 \\ 63 & F2 & 30 & FE \\ 7B & C5 & 2B & 76 \\ 77 & 6F & 67 & AB \end{bmatrix}$$

d. Shifting Rows

$$\begin{bmatrix} 7C & 6B & 01 & D7 \\ 63 & F2 & 30 & FE \\ 7B & C5 & 2B & 76 \\ 77 & 6F & 67 & AB \end{bmatrix} \rightarrow \begin{bmatrix} 7C & 6B & 01 & D7 \\ F2 & 30 & FE & 63 \\ 2B & 76 & 7B & C5 \\ AB & 77 & 6F & 67 \end{bmatrix}$$

e. Mixing Column

$$\begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \begin{bmatrix} 7C & 6B & 01 & D7 \\ F2 & 30 & FE & 63 \\ 2B & 76 & 7B & C5 \\ AB & 77 & 6F & 67 \end{bmatrix} = \begin{bmatrix} 74 & E7 & 0F & A2 \\ 55 & E6 & 04 & 22 \\ 3E & 2E & B8 & 8C \\ F6 & 15 & 58 & 0B \end{bmatrix}$$

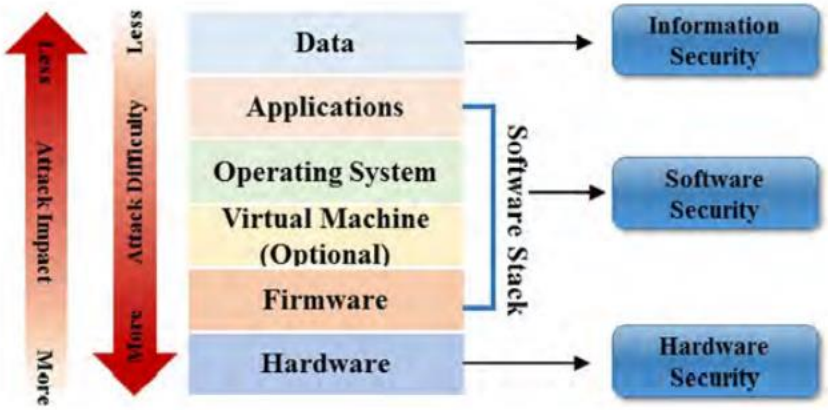
EC3401 NETWORKS AND SECURITY

UNIT V HARDWARE SECURITY **9**

Introduction to hardware security, Hardware Trojans, Side – Channel Attacks – Physical Attacks and Countermeasures – Design for Security. Introduction to Block chain Technology.

LAYERS OF A COMPUTING SYSTEM:

- Modern computing systems can be viewed as an organization consisting of multiple layers of abstraction.
- The hardware layer lies at the bottom of it, followed by the firmware that interfaces with the physical hardware layer.
- The firmware layer is followed by the software stack, comprising of an optional virtualization layer, the operating system (OS), and then the application layer.
- The data being processed by a computing system is stored in the hardware layer in volatile (for example, static or dynamic random access memory) or non-volatile (such as NAND or NOR flash) memory and accessed by the software layers.
- A system is connected to another system or to the Internet using networking mechanisms that are realized by a combination of hardware and software components.
- Computer security issues span all these layers



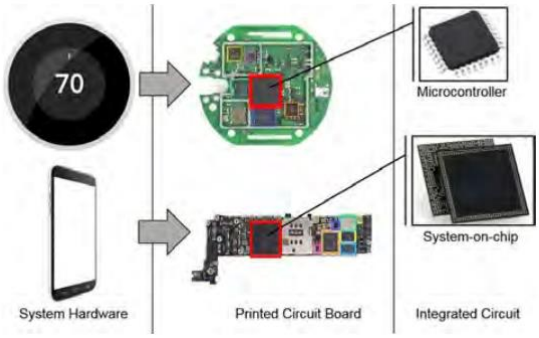
ELECTRONIC HARDWARE

The hardware in a computing system can, itself, be viewed as consisting of three layers

System-level hardware, that is, the integration of all physical components (such as PCBs, peripheral devices, and enclosures)

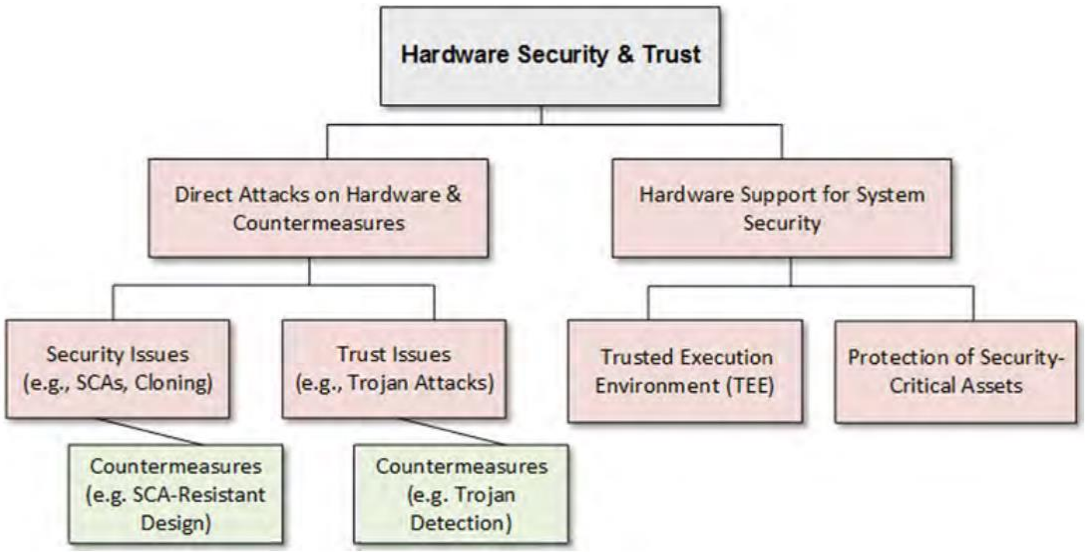
At the next level, one or more PCBs used which provide mechanical support and electrical connection to the electronic components that are required to meet the functional and performance requirements of a system.

At the bottom-most layer, we have active components (such as ICs, transistors, and relays), and passive electronic components.



HARDWARE SECURITY:

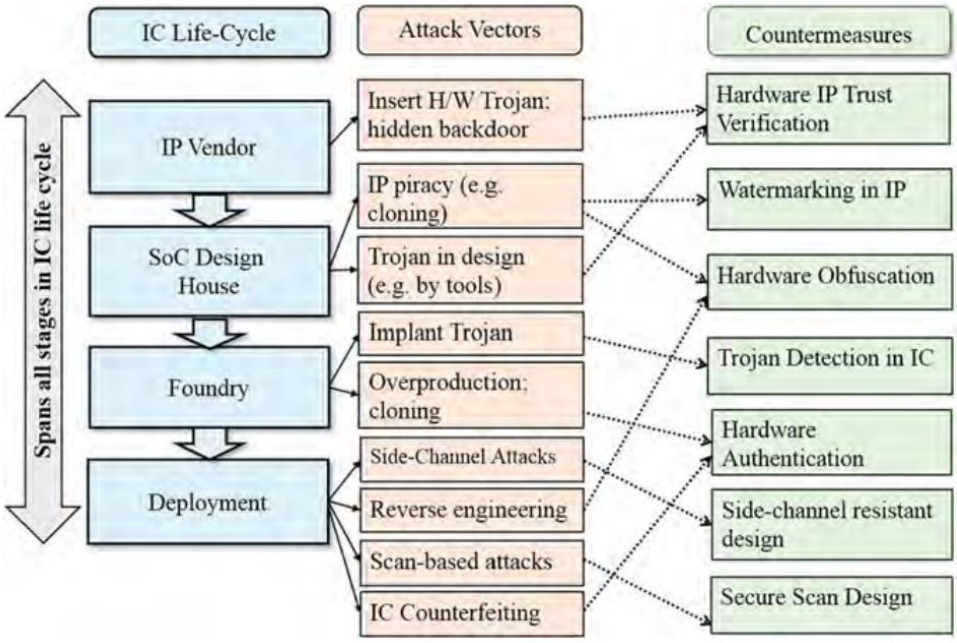
- Emerging trends in electronic hardware production, such as intellectual-property-based (IP-based) system on chip (SoC) design, and a long and distributed supply chain for manufacturing and distribution of electronic components—leading to reduced control of a chip manufacturer on the design and fabrication steps—have given rise to many growing security concerns
- Malicious modifications of ICs, also referred to as Hardware Trojan attacks , in an untrusted design house or foundry
- Another important aspect of hardware security relates to the hardware design, implementation, and validation to enable secure and reliable operation of the software stack.
- It deals with protecting sensitive assets stored in hardware from malicious software and network, and providing an appropriate level of isolation between secure and insecure data and code, in addition to providing separation between multiple user applications.



Hardware Trust:

- Hardware trust issues arise from involvement of untrusted entities in the life cycle of hardware, including untrusted IP or computer-aided design (CAD) tool vendors, and untrusted design, fabrication, test, or distribution facilities. These parties are capable of violating the trustworthiness of a hardware component or system.
- They can potentially cause deviations from intended functional behavior, performance, or reliability.

- Trust issues often lead to security concerns; for example, untrusted IP vendor can include malicious implant in a design, which can lead to denial of service (DoS), or information leakage attacks during field operation.



Hardware Trojans:

Hardware Trojans are malicious modifications to original circuitry inserted by adversaries to exploit hardware or to use hardware mechanisms to create backdoors in the design

Hardware Trojans have reportedly been used as 'kill switches' and backdoors in foreign military weapon system

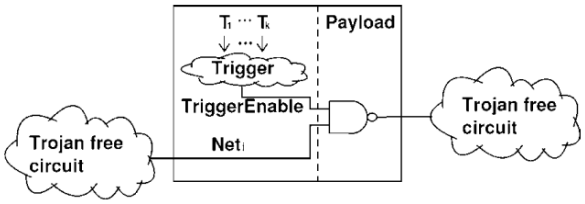
Detection of hardware Trojans is extremely difficult, for several reasons:

- Given the large number of soft, firm, and hard IP cores used in SoCs, and the high complexity of today's IP blocks, detecting a small malicious alteration is extremely difficult.
- Nanometer SoC feature sizes make detection by physical inspection and destructive reverse engineering very difficult, time consuming, and costly.
- Trojan circuits, by design, are typically activated under very specific conditions which makes them unlikely to be activated and detected using random or functional stimuli.
- Tests used to detect manufacturing faults, such as stuck-at and delay faults cannot guarantee detection of Trojans. Even when 100% fault coverage for all types of manufacturing faults is possible, there are no guarantees as far as Trojans are concerned.
- As physical feature sizes decrease because of improvements in lithography, process and environmental variations have an increasingly greater impact on the integrity of the circuit parametric behavior.

HARDWARE TROJAN STRUCTURE

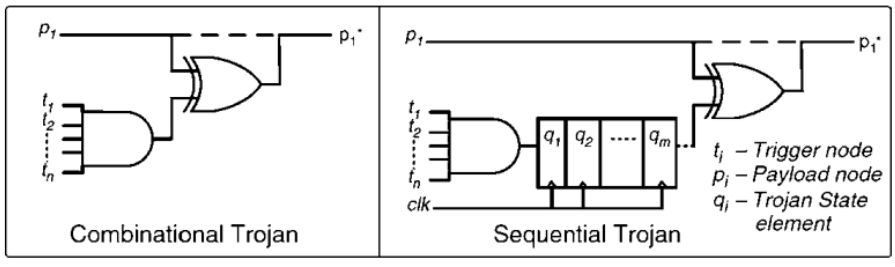
- The basic structure of a Trojan in a 3PIP (Party Intellectual Property) can include two main parts, **trigger and payload**
- A Trojan trigger is an optional part that monitors various signals and/or a series of events in the circuit.
- The payload usually taps signals from the original circuit and the output of the trigger.

- Once the trigger detects an expected event or condition, the payload is activated to perform malicious behavior. Typically, the trigger is expected to be activated under extremely rare conditions, so the payload remains inactive most of the time.
- When the payload is inactive, the IC acts like a Trojan-free circuit, making it difficult to detect the Trojan.
- Figure shows the basic structure of the Trojan at gate level. The trigger inputs (T1,T2,...,Tk) come from various nets in the circuit. The payload taps the original signal Neti from the original (Trojan-free) circuit and the output of the Trigger.
- Since the trigger is expected to be activated under rare condition, the payload output stays at the same value most of the time, Neti.
- However, when the trigger is active, that is, Trigger Enable is "0", the payload output will be different from Neti; this could result in injecting an erroneous value into the circuit and causing error at the output.



TROJAN MODELING

- In this model, it is assumed that a Trojan will be activated by rare circuit node conditions and will have its payload as a critical node in terms of functionality, but low observable node in terms of testing, to evade detection during normal functional testing.
- If the Trojan includes sequential elements, such as rare-event triggered counters, then the Trojan may be even harder to detect. Figure shows generic models for combinational and sequential Trojans.
- The trigger condition is an n-bit value at internal nodes, which is assumed to be rare enough to evade normal functional testing. The payload is defined as a node that is inverted when the Trojan is activated.

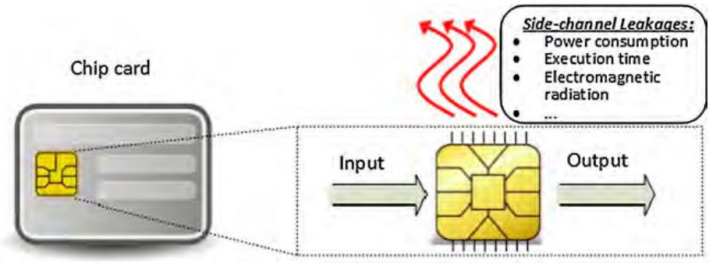


- To make it more difficult to detect, one might consider a sequential Trojan, which requires the rare event to repeat 2m times before the Trojan gets activated and inverts the payload node.
- The sequential state machine is considered in its simplest form to be a counter, and the effect of the output on the payload is considered to be an XOR function to have maximal impact.
- In more generic models, the counter can be replaced by any Finite State Machine (FSM) and the circuit can be modified as a function of Trojan output and the payload node.

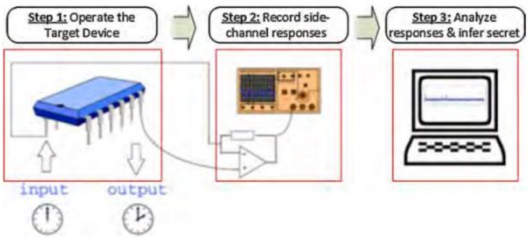
Side-channel attacks (SCA):

- It is a noninvasive attack that is based on targeting the implementation of a cryptographic algorithm rather than analyzing its statistical or mathematical weakness.

- These attacks exploit physical information leaking from various indirect sources or channels, such as, the target device's power consumption, electromagnetic (EM) radiation, or the time taken for a computation. These channels are referred to as "side channels".
- The information embedded in side-channel parameters depend on the intermediate values computed during the execution of a crypto-algorithm, and are correlated with the inputs and the secret key of the cipher
- An adversary can effectively extract the secret key by observing and analyzing side-channel parameters with relatively cheap equipment, and within a very short time span, ranging from a few minutes to a few hours.



- Figure illustrates how a device leaks side-channel information while operating. Common sidechannel attacks, such as power attacks, monitor the device's power consumption. Typically, this is done by incorporating a current path at V_{dd} or GND pin of a chip, which is performing the cryptographic operation, to record power dissipation for such an operation.
- The device's power consumption captures switching activity of the relevant transistors, which depends on inputs to a cryptographic function, such as the plaintext and the key.
- Simple power analysis (SPA) is a technique to directly interpret the collected traces of power consumption for a set of inputs.
- It requires relatively detailed knowledge about the implementation of a cryptographic algorithm and a skilled adversary to interpret secret key information by visually examining the power consumption.



Analysis of side channel effects

Side-Channel Attack	Measured Parameters	Analysis Methods	Countermeasures
Power Analysis	Current signature and power consumption patterns	Simple power analysis (SPA) Differential power analysis (DPA) Correlation power analysis (CPA)	Power consumption masking Power consumption hiding
EM Analysis	Intentional and non-Intentional electromagnetic emission	Simple EM analysis (SEMA) Differential EM analysis (DEMA)	EM emission shielding EM noise generation modules
Fault Analysis	Invalid outputs, underpowered behavior, and Laser/UV Glitching Responses	Comparative approach to analyze responses before and after fault insertion	Error detection schemes Anti-tamper protection modules
Timing Analysis	Operation delays, time elapsed when different input patterns are applied	Analysis to relate operation delay to the nature of the function	Randomized operational delay Fixed operational delay

Physical attacks and countermeasures:

Physical attacks are divided into three categories: noninvasive, semi-invasive, and invasive attacks.

Noninvasive attack:

- It does not require any initial preparations of the device under test, and will not physically harm the device during the attack.
- The attacker can either tap the wires to the device, or plug it into a test circuit for the analysis.

Invasive attack:

- It requires direct access to the internal components of the device, which normally requires a well-equipped and knowledgeable attacker to succeed.
- These attacks are more demanding and expensive, as feature sizes shrink, and device complexity increases.

Semi Invasive attack:

- There is a large gap between noninvasive and invasive attacks. Many attacks fall into this gap, called semi-invasive attacks.
- They are not very expensive as classical penetrative invasive attacks, but are as easily repeatable as noninvasive attacks.

Reverse Engineering:

- Reverse engineering (RE) is the process involving the thorough examination of an object to achieve a full understanding of its construction and/or functionality.
- RE is now widely used to clone, duplicate, or reproduce systems and devices in various security-critical applications, such as smartcards, smartphone, military, financial, and medical systems

Chip Level RE:

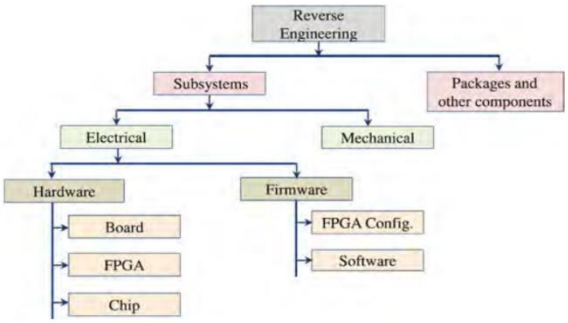
- A chip is an IC comprised of electronic devices that are fabricated using semiconductor material.
- A chip has package material, bond wires, a lead frame, and die. Each die has several metal layers, vias, interconnections, passivation, and active layers
- RE of chips can be nondestructive or destructive. X-ray tomography is a nondestructive method of RE that can provide layer-by-layer images of chips, and is often used for the analysis of internal vias, traces, wire bonding, capacitors, contacts, or resistors.

PCB-level RE:

- Electronic chips and components are mounted on a laminated nonconductive PCB and electrically interconnected using conductive copper traces and vias.
- The board might be single or multilayered, depending on the complexity of the electronic system.
- Delayering or x-ray imaging could be used to identify the connections, traces, and vias of the internal PCB layers.

System-level RE:

- Electronic systems are comprised of chips, PCBs, and firmware.
- A system's firmware includes the information about the system's operation and timing, and is typically embedded within nonvolatile memories (NVMs), such as ROM, EEPROM, and Flash



Equipments used for the analysis:

- Optical high/super-resolution microscopy (digital).
- Scanning electron microscopes.
- Transmission electron microscopes.
- Focused ion beam
- Scanning capacitance microscopy.
- High-resolution x-ray microscopy.

Board level PCB:

- The goal of board-level RE is to identify all components on the board and the connections between them.
- All of the components used in a design are called the bill of materials (BOM)
- Some electronic components mounted on the PCB can be identified easily through the use of IC markings, but fully custom and semicustom ICs are difficult to identify.
- Using standard off-the-shelf parts with silkscreen annotations will assist the RE process.

The IC Markings normally divided into four parts

• The first is the prefix, which is the code that is used to identify the manufacturer. It could be a one to a three-letter code, although a manufacturer might have several prefixes.

• The second part is the device code, which is used to identify a specific IC type.

• The next part is the suffix, which is used to identify the package type and temperature range. Manufacturers modify their suffixes frequently.

• A four-digit code is used for the date, where the first two digits identify the year and the last two identify the number of the week.

- If the IC marking is not readable, because it has faded away due to prior usage in the field or the manufacturer did not place a marking for security purposes, the reverse engineer could strip off the package, and read the die markings to identify the manufacturer and the chip's functionality

Probing Attack Targets

- It is essential for both attackers and countermeasure designers to determine which signals are more likely to be targeted in a probing attack. Such signals are termed as assets.

  Keys: Keys of an encryption module are archetypal assets. They are usually stored in nonvolatile memory on the chip. If the key is leaked, the root of trust it provides will become compromised, and could serve as a gateway to more serious attacks.

  Firmware and configuration bit stream: Electronic intellectual properties (IPs), such as low-level program instruction sets, manufacturer firmware, and FPGA configuration bit streams are often sensitive, mission critical, and/or contain trade secrets of the IP owner

  On-device protected data: Sensitive data, such as health and personal identifiable information, should be kept private

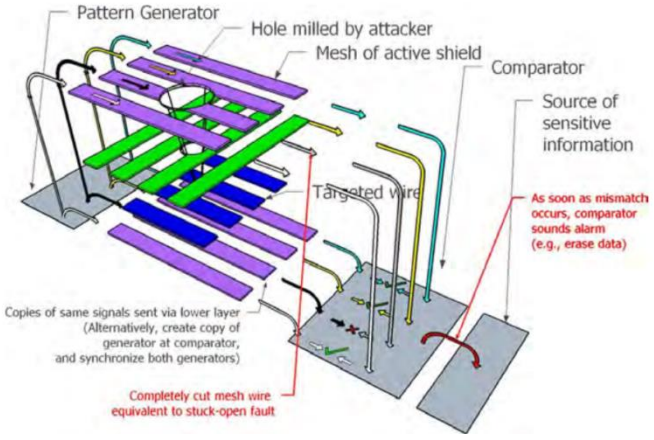
  Device configuration: Device configuration data control the access permissions to the device. They specify which services or resources can be accessed by each individual user

  Essential Steps of a Probing Attack:
    - Decapsulation
    - Reverse Engineering
    - Locating target wires
    - Reaching target wire and extracting information

Existing countermeasures and Limitations:

Active Shields

- In this approach, a shield which carries signals is placed on the top-most metal layer to detect holes milled by FIB.
- The shield is referred to as “active” because signals on these top layer wires are constantly monitored to detect if milling has cut them
- A digital pattern is generated from a pattern generator, transmitted through the shield wires on top-most metal layer, and then compared with a copy of itself transmitted from lower layer.
- If an attacker mills through the shield wires on top layer to reach target wire, the hole is expected to cut open one or more shield wires, thereby leading to a mismatch at the comparator and triggering an alarm signal to erase or stop generating sensitive information.



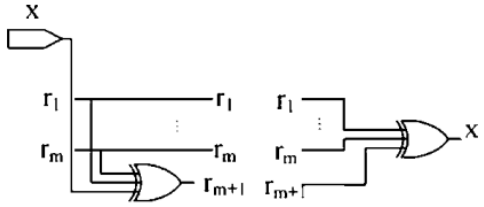
Analog Shields and Sensors

- Instead of generating, transmitting, and comparing digital patterns, analog shields monitor parametric disturbances with its mesh wires.
- In addition to shield designs, the probe attempt detector (PAD) also uses capacitance measurement on selected security critical wires to detect additional capacitance introduced by a metal probe.

- Compared to active shields, analog shields detect probing without test patterns and require less area overhead.
- The PAD technique is also unique in remaining effective against electrical probing from the back-side.
- The problem with analog sensors or shields is that analog measurements are less reliable due to process variations, a problem further exacerbated by feature scaling

t-Private Circuits

- The t-private circuit technique is proposed based on the assumption that the number of concurrent probe channels that an attacker could use is limited, and exhausting this resource deters an attack.
- In this technique, the circuit of a security-critical block is transformed so that at least t + 1 probes are required within one clock cycle to extract one bit of information.
- First, masking is applied to split computation into multiple separate variables, where an important binary signal x is encoded into t + 1 binary signals by XOR ing it with t independently generated random signals ($r_{t+1} = x \oplus r_1 \oplus \cdots \oplus r_t$)
- The computations on x are performed in its encoded form in the transformed circuit. x can be recovered (decoded) by computing $r_{t+1} = x \oplus r_1 \oplus \cdots \oplus r_t$.
- The major issue with t-private circuit is that the area overhead involved for the transformation is prohibitively expensive.



Input and Output decoder for masking in t-private circuits

Other Countermeasure Designs

- One known countermeasure that deters decapsulation stage of probing attacks is a light sensor that is sometimes included in a tamper-resistant design.
- Some other techniques include scrambling wires and avoiding repetitive patterns in shield mesh to impede the locating-targetwire stage of probing attacks.

Design for Security:

- The baseline architecture is typically derived from legacy architectures for previous products, adapted to account for the policies defined for the system under exploration.
- A SoC design may have a significant number of assets, often in the order of thousands, if not more.
- Not all assets are statically defined; many assets are created at different IPs during the system execution.
- During system execution, these modes are passed to the cryptographic engine, which generates the cryptographic keys for different IPs, and transmits them through the system network-onchip (NoC) to the respective IPs.
- Each participant in this process has sensitive assets during different phases of the system execution.
- The security architecture must account for any potential access to these assets at any point of execution, possibly under the relevant adversary model.
- There are different Trusted Execution Environment(TEE) frameworks specifically developed for SoC designs .Some are,

Samsung KNOX:

This architecture is specifically targeted toward smartphones, and provides secure separation features to enable information partition between business and personal content to coexist on the same system.

It permits hot swap between these two content worlds, for example without requiring system restart.

This architecture permits several system-level services, including the following:

• **Trusted boot** that is, preventing unauthorized OS and software from being loaded onto the device at startup.

• **Trust-zone-based integrity measurement architecture (TIMA)**, which continually monitors kernel integrity.

• **Security enhancement (SE)** for Android, an enforcement mechanism providing protection of system/user data based on confidentiality and integrity requirements through separation.

• **KNOX container**, which offers a secure environment in which protected business applications can run with guaranteed information separation from the rest of the device.

ARM Trust Zone

- Trust Zone technology is a system-wide approach to provide security on high-performance computing platforms.
- The Trust Zone implementation relies on partitioning the SoC's hardware and software resources, so that they exist in two worlds: secure and nonsecure.
- The hardware supports access control and permissions for the handling of secure/non secure applications, and the interaction and communication among them.
- The software supports secure system calls and interrupts for secure runtime execution in a multitasking environment.
- This protection extends to input/output (I/O), connected to the system bus via the Trust Zone enabled AMBA3 AXI bus fabric, which also manages memory compartmentalization.

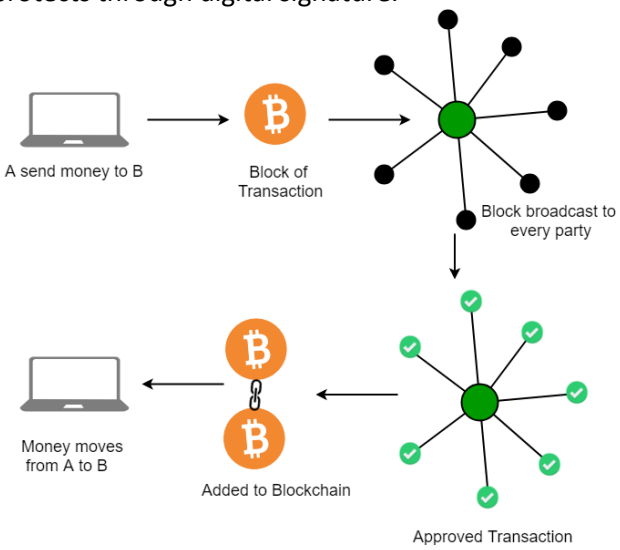
Intel SGX

- SGX is an architecture for providing a trusted execution environment provided by the underlying hardware to protect sensitive application and user programs or data against potentially malicious or tampered operating systems.
- SGX permits applications to initiate secure enclaves or containers, which serve as so-called "islands of trust".
- It is implemented as a set of new CPU instructions that can be used by applications to set aside such secure enclaves of code and data.
  This enables
  1) Applications to preserve the confidentiality and integrity of sensitive data without disrupting the ability of legitimate system software to manage the platform resources;
  2) End users to retain control of their platforms, applications, and services even in the presence of malicious system software.

Introduction to Block chain technology:

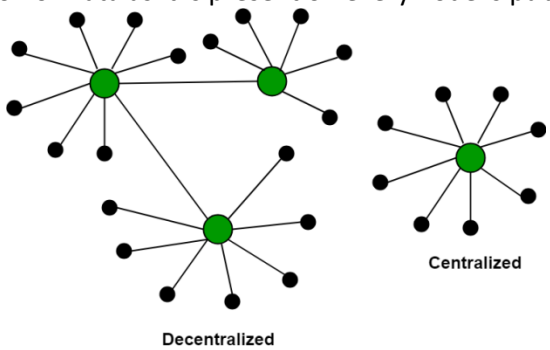
- Block chain is the backbone Technology of Digital CryptoCurrency BitCoin.
- The block chain is a distributed database of records of all transactions or digital event that have been executed and shared among participating parties.
- Each transaction verified by the majority of participants of the system. It contains every single record of each transaction.
- Bit Coin is the most popular crypto currency an example of the block chain.
- Bitcoin is a crypto currency and is used to exchange digital assets online.

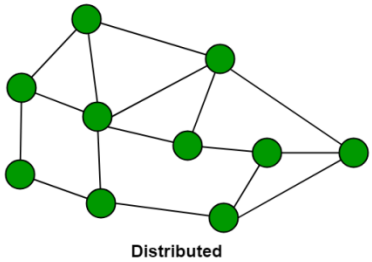
- Bitcoin uses cryptographic proof instead of third-party trust for two parties to execute transactions over the internet.
- Each transaction protects through digital signature.



Distributed Database:

There is no Central Server or System which keeps the data of the Block chain.
The data is distributed over Millions of Computers around the world which are connected to the Block chain.
This system allows Notarization of Data as it is present on every node is publicly verifiable.





A network of nodes:

- A node is a computer connected to the Block chain Network.
- Node gets connected with Block chain using the client.
- Client helps in validating and propagating transaction on to the Block chain.
- When a computer connects to the Blockchain, a copy of the Block chain data gets downloaded into the system and the node comes in sync with the latest block of data on Block chain.

- The Node connected to the Block chain which helps in the execution of a Transaction in return for an incentive is called Miners.

Benefits of Block chain Technology:

- **Time-saving:** No central Authority verification needed for settlements making the process faster and cheaper.
- **Cost-saving:** A Block chain network reduces expenses in several ways. No need for third-party verification. Participants can share assets directly. Intermediaries are reduced. Transaction efforts are minimized as every participant has a copy of shared ledger.
- **Tighter security:** No one can temper with Block chain Data as it shared among millions of Participant. The system is safe against cybercrimes and Fraud.
- **Collaboration:** It permits every party to interact directly with one another while not requiring third party negotiate.
- **Reliability:** Block chain certifies and verifies identities of every interested party. This removes double record, reducing rates and accelerates transactions.

Applications of Block chain:

- Leading Investment Banking Companies like Credit Suisse, JP Morgan Chase, Goldman Sachs and Citigroup have invested in Block chain and are experimenting to improve the banking experience and secure it.
- Following the Banking Sector, the Accountants are following the same path. Accountancy involves extensive data, including financial statements spreadsheets containing lots of personal and institutional data.
- Booking a Flight requires sensitive data ranging from the passenger's name, credit card numbers, immigration details, identification, destinations, and sometimes even accommodation and travel information. So the sensitive data can be secured using block chain technology. Russian Airlines are working towards the same.
    - Barclaysuses Block chain to streamline the Know Your Customer (KYC) and Fund Transfer processes while filling patents against these features.
    - Visauses Blockchain to deal with business to business payment services.
    - Unileveruses Block chain to track all their transactions in the supply chain and maintain the product's quality at every stage of the process.
    - Walmart has been using Block chain Technology for quite some time to keep track of their food items coming right from farmers to the customer.