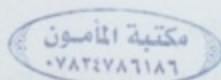




Computer Networks

3rd Stage

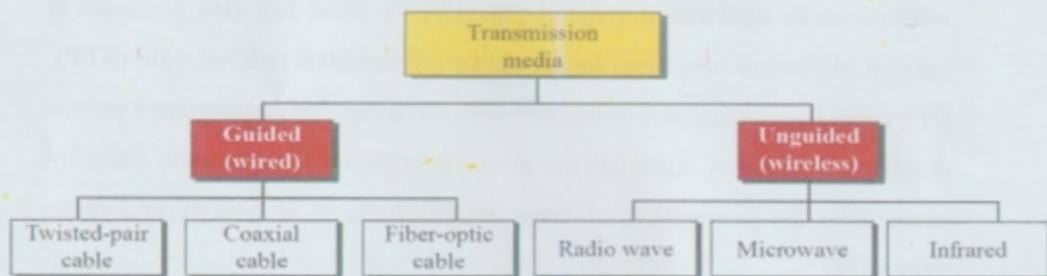


Part two

Dr. Saif Alansari

A transmission **medium** is defined as anything that can carry information from a source to a destination.

Transmission media can be divided into two broad categories: guided and unguided.

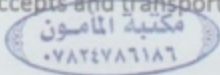


Classes of transmission media

1. GUIDED MEDIA

Guided media, which are those that provide a conduit from one device to another. Twisted-pair and coaxial cable use metallic (copper) conductors that accept and transport signals in the form of electric current.

Optical fiber is a cable that accepts and transports signals in the form of light.



a. Twisted-Pair Cable

A twisted pair consists of two conductors (normally copper), each with its own plastic insulation, twisted together.

One of the wires is used to carry signals to the receiver, and the other is used only as a ground reference.

In addition to the signal sent by the sender on one of the wires, interference (noise) and crosstalk may affect both wires and create unwanted signals.

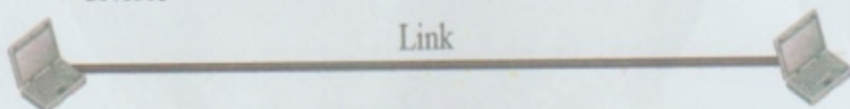
Type of Connection

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

There are two possible types of connections: point to point and multipoint.

1. Point to Point

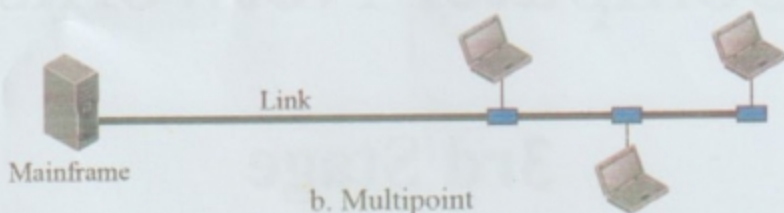
A point to point connection provides a dedicated link between two devices



a. Point-to-point

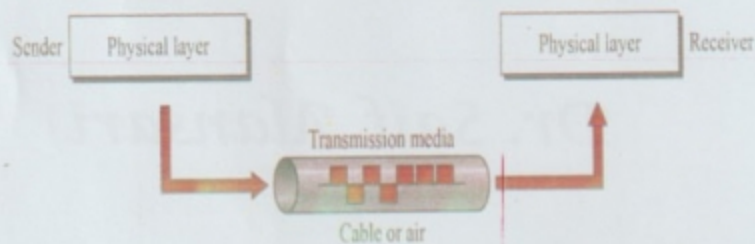
2. Multipoint

A multipoint connection is one in which more than two specific devices share a single link.



Transmission Media

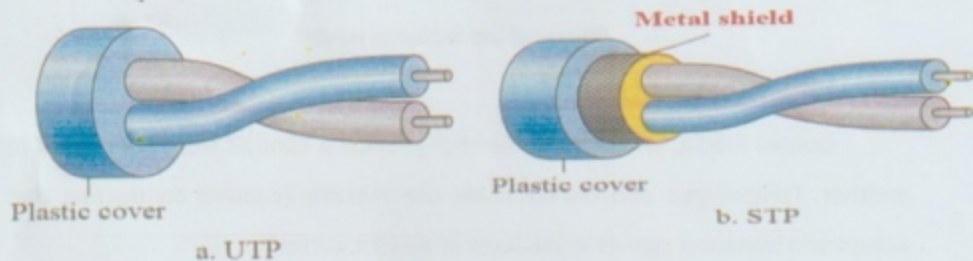
Transmission media are actually located below the physical layer and are directly controlled by the physical layer. You could say that transmission media belong to layer zero.





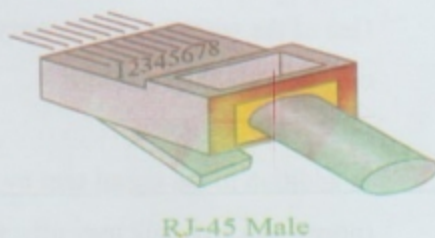
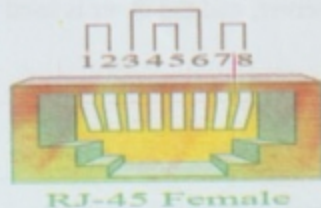
UTP VS STP

The most common twisted-pair cable used in communications is referred to as **unshielded twisted-pair (UTP)**. IBM has also produced a version of twisted-pair cable for its use called **shielded twisted-pair (STP)**. STP cable has a metal foil or braided mesh covering that encases each pair of insulated conductors. Although metal casing improves the quality of cable by preventing the penetration of noise or crosstalk, it is bulkier and more expensive.



Connectors

The most common UTP connector is RJ45 (RJ stands for registered jack). The RJ45 is a keyed connector, meaning the connector can be inserted in only one way.



Performance:

One way to measure the performance of twisted-pair cable is to compare attenuation versus frequency and distance.

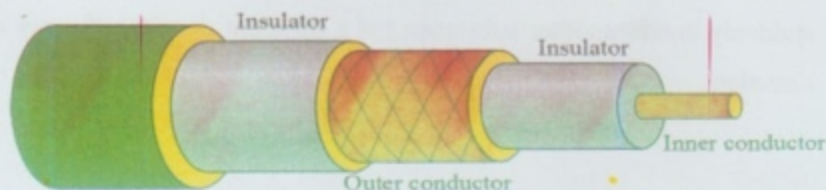
Applications:

Twisted-pair cables are used in telephone lines to provide voice and data channels. Local area networks, such as 10Base-T and 100Base-T, also use twisted-pair cables.

Base-T: is the Ethernet wiring standard for 10 Mbps (megabits per second) for a maximum distance of approximately 100 meters per segment over unshielded twisted pair cables terminated with RJ-45 connectors.

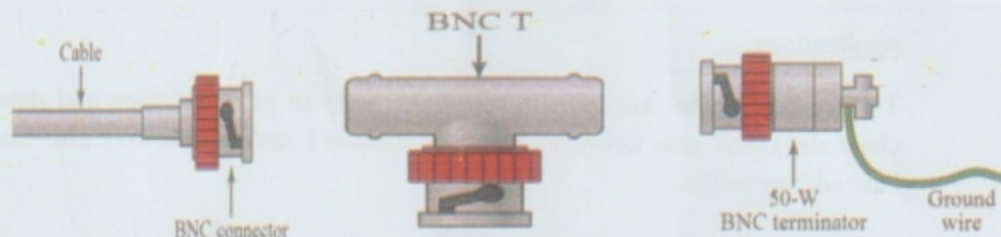
b. Coaxial Cable

- Coaxial cable carries signal of higher frequency ranges than those in twisted pair cable
- Instead of having two wires, coax has a central core conductor of solid or stranded wire (usually copper) enclosed in an insulating sheath, which is, in turn, encased in an outer conductor of metal foil, braid, or a combination of the two.
- The outer metallic wrapping serves both as a shield against noise and as the second conductor, which completes the circuit.
- This outer conductor is also enclosed in an insulating sheath, and the whole cable is protected by a plastic cover.



Connectors:

To connect coaxial cable to devices, we need coaxial connectors. The most common type of connector used today is (BNC) connector.



- **Performance:**

The attenuation is much higher in coaxial cables than in twisted pair cable. The signal weakens quickly and requires the frequent use of repeaters. Long distance transmission (500 m).

- **Applications:**

Coaxial cable was widely used in:

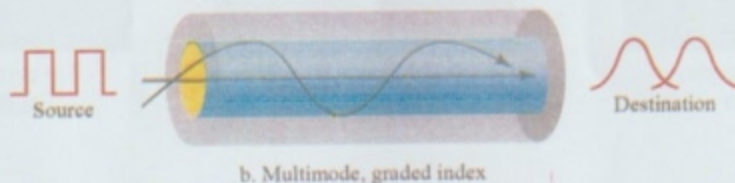
- ✓ Telephone network.
- ✓ Cable TV networks.
- ✓ Common application is traditional Ethernet LANs.

C. Fiber-Optic Cable

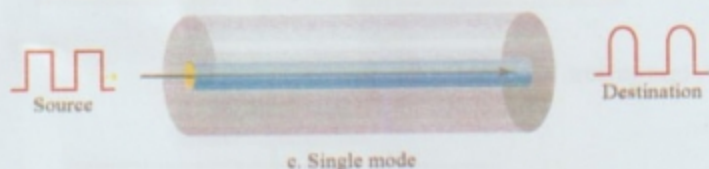
A fiber-optic cable is made of glass or plastic and transmits signals in the form of light.

To understand optical fiber, we first need to explore several aspects of the nature of light. Light travels in a straight line as long as it is moving through a single uniform substance. If a ray of light traveling through one substance suddenly enters another substance (of a different density), the ray changes direction.

While **multimode graded-index fiber**, has varying densities. Density is highest at the center of the core and decreases gradually to its lowest at the edge.



Single-Mode: Single mode uses step index fiber and a highly focused source of light that limits beams to a small range of angles, all close to the horizontal.

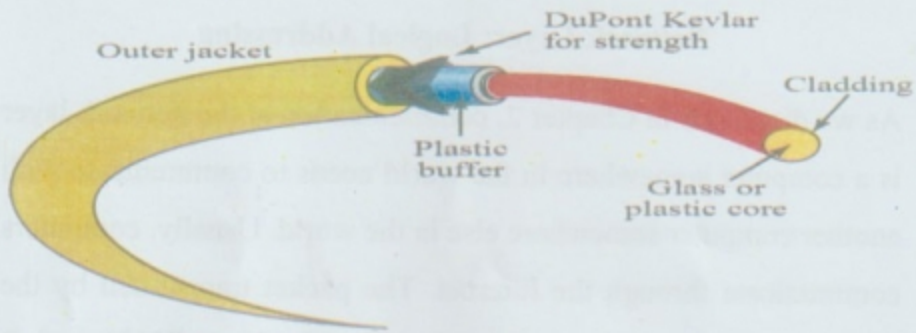


- **Fiber Sizes:**

Optical fibers are defined by the ratio of the diameter of their core to the diameter of their cladding, both expressed in micrometers.

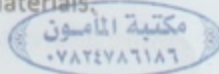
- **Cable Composition:**

The outer jacket is made of either PVC or Teflon. Inside the jacket are Kevlar strands (strong material) to strengthen the cable. Below the Kevlar is another plastic coating to cushion the fiber. The fiber is at the center of the cable, and it consists of cladding and core.



Advantages of Optical fiber:

- Higher bandwidth and hence data rates.
- Less signal attenuation. A signal can run 50 km without regeneration, while we need repeaters every 5 km for other cables.
- Immunity to electromagnetic interference.
- Resistance to corrosive materials.
- Light weight.
- Greater immunity to tapping.



Disadvantages:

- Installation and maintenance.
- Unidirectional light propagation. If we need bidirectional, two fibers are needed.
- More expensive cost.

Network Layer: Logical Addressing

As we discussed in Chapter 2, communication at the network layer is a computer somewhere in the world needs to communicate with another computer somewhere else in the world. Usually, computers communicate through the Internet. The packet transmitted by the sending computer may pass through several LANs or WANs before reaching the destination computer.

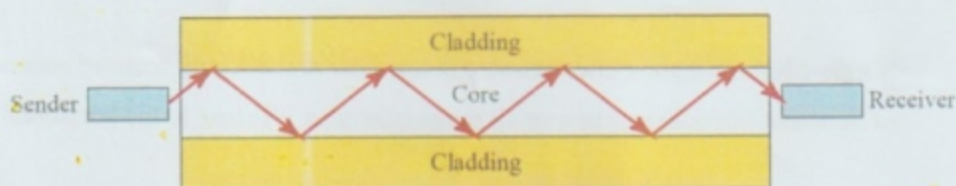
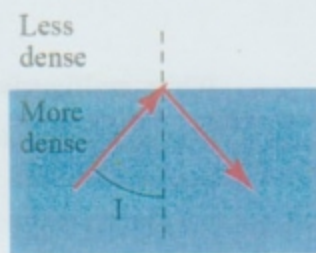
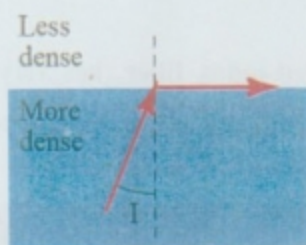
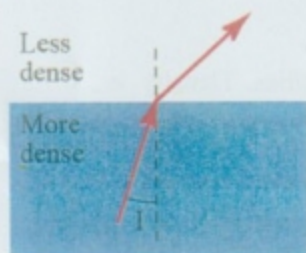
For this level of communication, we need a global addressing scheme; we called this logical addressing. Today, we use the term IP address to mean a logical address in the network layer of the TCP/IP protocol suite.

The Internet addresses are 32 bits in length; this gives us a maximum of 2^{32} addresses. These addresses are referred to as IPv4 (IP version 4) addresses or simply IP addresses if there is no confusion.

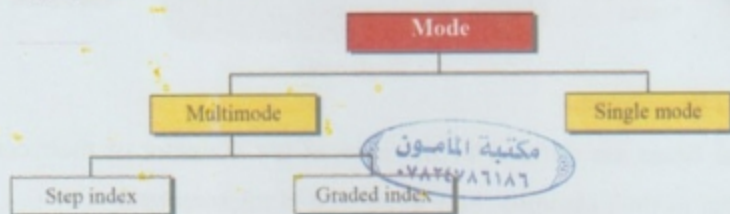
The need for more addresses, in addition to other concerns about the IP layer, motivated a new design of the IP layer called the new generation of IP or IPv6 (IP version 6).

In this version, the Internet uses 128-bit addresses that give much greater flexibility in address allocation. These addresses are referred to as IPv6 (IP version 6) addresses.

In this chapter, we discuss IPv4 addresses, which are currently being used in the Internet.

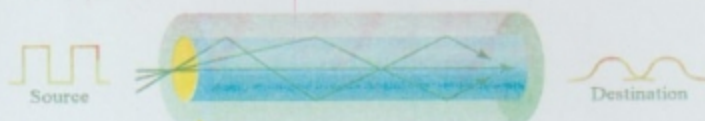


• Propagation Modes:



Multimode: multiple beams from a light source move through the core in different paths.

In **multimode step-index fiber**: the density of the core remains constant from the center to the edges. A beam of light moves through this constant density in a straight line until it reaches the interface of the core and the cladding. At the interface, there is an abrupt change due to a lower density; this alters the angle of the beam's motion.



IPv4 ADDRESSES

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. We will see later that, by using some strategies, an address may be assigned to a device for a time period and then taken away and assigned to another device.

The IPv4 addresses are universal in the sense that the addressing system must be accepted by any host that wants to be connected to the Internet.

The IPv4 addresses are unique and universal

Address Space

A protocol such as IPv4 that defines addresses has an address space. An address space is the total number of addresses used by the protocol. If a protocol uses N bits to define an address, the address

space is 2^N because each bit can have two different values (0 or 1) and N bits can have 2^N values.

IPv4 uses 32-bit addresses, which means that the address space is 2^{32} or 4,294,967,296 (more than 4 billion). This means that, theoretically, if there were no restrictions, more than 4 billion devices could be connected to the Internet. We will see shortly that the actual number is much less because of the restrictions imposed on the addresses.

Notations

There are two prevalent notations to show an IPv4 address: binary notation and dotted-decimal notation.

Binary Notation

In binary notation, the IPv4 address is displayed as 32 bits. Each octet is often referred to as a byte. So it is common to hear an IPv4 address referred to as a 32-bit address or a 4-byte address. The following is an example of an IPv4 address in binary notation:

01110101 10010101 00011101 00000010

Dotted-Decimal Notation

To make the IPv4 address more compact and easier to read, Internet addresses are usually written in decimal form with a decimal point (dot) separating the bytes. The following is the dotted-decimal notation of the above address:

117.149.29.2

Figure 1 shows an IPv4 address in both binary and dotted-decimal notation. Note that because each byte (octet) is 8 bits, each number in dotted-decimal notation is a value ranging from 0 to 255.

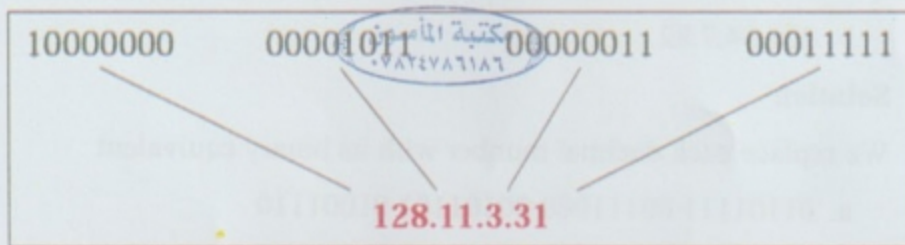


Figure 1

Example 1

Change the following IPv4 addresses from binary notation to dotted-decimal notation.

a. 10000001 00001011 00001011 11101111

b. 11000001 10000011 00011011 11111111

Solution

We replace each group of 8 bits with its equivalent decimal number and add dots for separation.

a. 129.11.11.239

b. 193.131.27.255

Example 2

Change the following IPv4 addresses from dotted-decimal notation to binary notation.

a. 111.56.45.78

b. 221.34.7.82

Solution

We replace each decimal number with its binary equivalent

a. 01101111 00111000 00101101 01001110

b. 11011101 00100010 00000111 01010010

Example 3

Find the error, if any, in the following IPv4 addresses.

- a. 111.56.045.78
- b. 221.34.7.8.20
- c. 75.45.301.14
- d. 11100010.23.14.67

Solution

- a. There must be no leading zero (045).
- b. There can be no more than four numbers in an IPv4 address.
- c. Each number needs to be less than or equal to 255 (301 is outside this range).
- d. A mixture of binary notation and dotted-decimal notation is not allowed

Classful Addressing

IPv4 addressing, at its inception, used the concept of classes. This architecture is called classful addressing. Although this scheme is becoming obsolete, we briefly discuss it here to show the rationale behind classless addressing. In classful addressing, the address space is divided into five classes: **A**, **B**, **C**, **D**, and **E**. Each class occupies some part of the address space.

We can find the class of an address when given the address in binary notation or dotted-decimal notation. If the address is given in binary notation, the first few bits can immediately tell us the class of the

address. If the address is given in decimal-dotted notation, the first byte defines the class. Both methods are shown in Figure

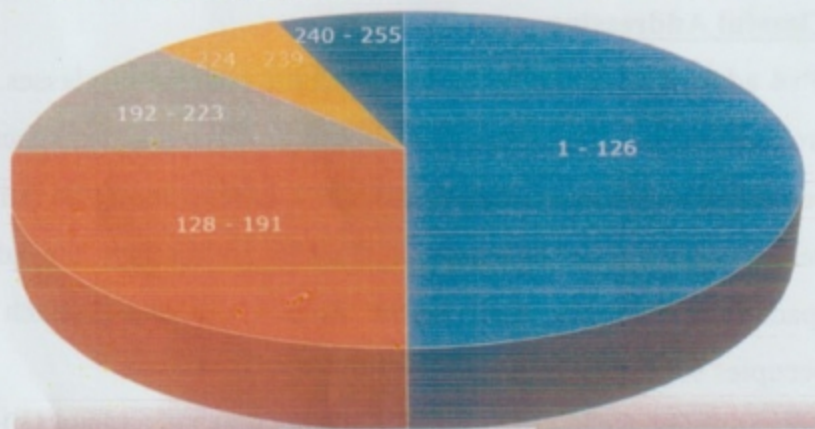
	First byte	Second byte	Third byte	Fourth byte
Class A	0			
Class B	10			
Class C	110			
Class D	1110			
Class E	1111			

a. Binary notation

	First byte	Second byte	Third byte	Fourth byte
Class A	0-127			
Class B	128-191			
Class C	192-223			
Class D	224-239			
Class E	240-255			

b. Dotted-decimal notation

IPv4 Classes Identified by the first octet 22:xx:xx:xx



Class	Range
Class A	1 - 126
Class B	128 - 191
Class C	192 - 223

Example 4

Find the class of each address.

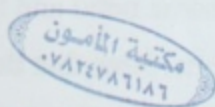
- a. 00000001 00001011 00001011 11101111
- b. 11000001 10000011 00011011 11111111
- c. 14.23.120.8
- d. 252.5.15.111

Solution

- a. The first bit is 0. This is a class A address.
- b. The first 2 bits are 1; the third bit is 0. 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

Classes and Blocks

One problem with classful addressing is that each class is divided into a fixed number of blocks with each block having a fixed size as shown in Table



<i>Class</i>	<i>Number of Blocks</i>	<i>Block Size</i>	<i>Application</i>
A	128	16,777,216	Unicast
B	16,384	65,536	Unicast
C	2,097,152	256	Unicast
D	1	268,435,456	Multicast
E	1	268,435,456	Reserved

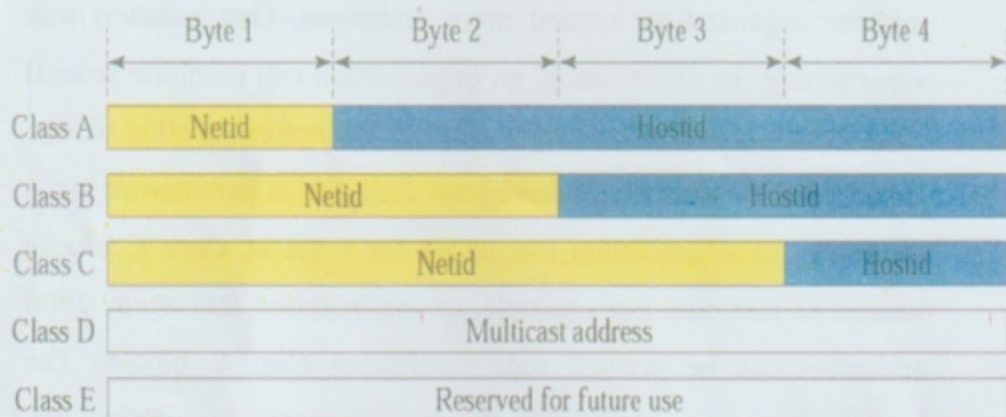
In classful addressing, a large part of the available addresses were wasted

Netid and Hostid

In classful addressing, an IP address in class A, B, or C is divided into netid and hostid. These parts are of varying lengths, depending on the class of the address. Figure 19.2

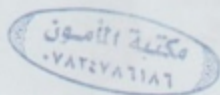
shows some netid and hostid bytes. The netid is in color, the hostid is in white. Note that the concept does not apply to classes D and E.

In class A, one byte defines the netid and three bytes define the hostid. In class B, two bytes define the netid and two bytes define the hostid. In class C, three bytes define the netid and one byte defines the hostid.



Subnetting

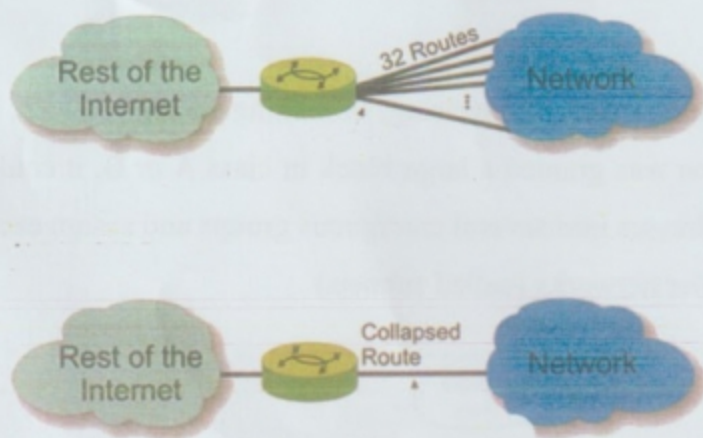
During the era of classful addressing, subnetting was introduced. If an organization was granted a large block in class A or B, it could divide the addresses into several contiguous groups and assign each group to smaller networks (called subnets)



Supernetting

The time came when most of the class A and class B addresses were depleted; however, there was still a huge demand for midsize blocks. The size of a class C block with a maximum number of 256

addresses did not satisfy the needs of most organizations. Even a midsize organization needed more addresses. One solution was supernetting. In supernetting, an organization can combine several class C blocks to create a larger range of addresses. In other words, several networks are combined to create a supernetwork or a supemet. An organization can apply for a set of class C blocks instead of just one. For example, an organization that needs 1000 addresses can be granted four contiguous class C blocks. The organization can then use these addresses to create one supernetwork. Supernetting decreases the number of 1s in the mask.



Dynamic Host Configuration Protocol (DHCP)

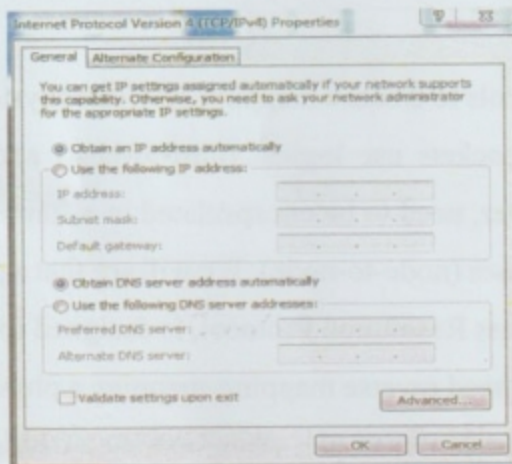
Last Chapter we discussed the Internet Protocol (IP) as the main protocol at the network layer. IP was designed as a best-effort delivery protocol, but it lacks some features such as flow control and error control. It is a host-to-host protocol using logical addressing. To make IP more responsive to some requirements in today's internetworking, we need the help of other protocols

We need protocols to create a mapping between physical and logical addresses. IP packets use logical (host-to-host) addresses. These packets, however, need to be encapsulated in a frame, which needs physical addresses (node-to-node). We will see that a protocol called ARP, the Address Resolution Protocol, is designed for this purpose. We sometimes need reverse mapping-mapping a physical address to a logical address. For example, when booting a diskless network or leasing an IP address to a host. Three protocols are designed for this purpose: RARP, BOOTP, and DHCP.

So now we will discuss Dynamic Host Configuration Protocol (DHCP)

DHCP (Dynamic Host Configuration Protocol) is a network protocol used for assigning IP address to network clients

dynamically from pre-defined IP pool. It is useful for LAN network, but not generally used for production servers. DHCP works on concept of a 'lease' (amount) of time that a given IP address will be valid for a network computer. The lease time can vary depending on how long a user is likely to require the Internet connection or DHCP configuration.



DHCP supports three mechanisms for IP address allocation:

- Automatic allocation—DHCP assigns a permanent IP address to a client.
- Dynamic allocation—DHCP assigns an IP address to a client for a limited period of time (or until the client explicitly relinquishes the address).

- Manual allocation—The network administrator assigns an IP address to a client and DHCP is used simply to convey the assigned address to the client

DHCP Server Overview

There are six messages must DHCP deal with it

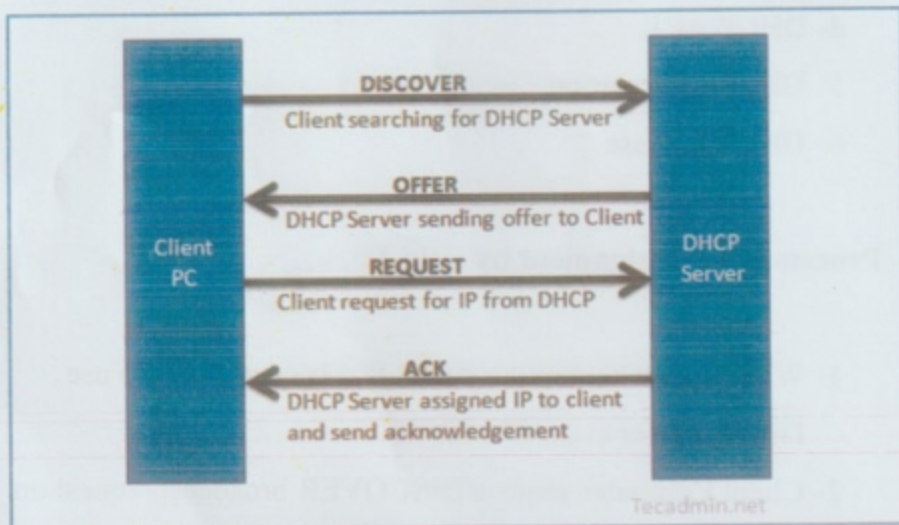
- 1- DHCP Discovery
- 2- DHCP Offer
- 3- DHCP Request
- 4- DHCP ACK
- 5- DHCP Information
- 6- DHCP Release

Process of IP Assignment by DHCP

- 1- When a client computer starts, If it is configured to use DHCP Server to get IP address.
- 2- Client Computer sends a DISCOVER broadcast request on network.
- 3- DHCP server received this request and sends an OFFER packet back to client.

4- After getting offer packet, Client sends a REQUEST packet to DHCP server to assign his an IP.

5- DHCP Server assigned the IP to client computer and sends a ACK packet to client.



A DHCP client may receive offers from multiple DHCP Servers and can accept any one of the offers; however, the client usually accepts the first offer it receives. Additionally, the offer from the DHCP

Server is not a guarantee that the IP address will be allocated to the client; however, the server usually reserves the address until the client has had a chance to formally request the address.

The client returns a formal request for the offered IP address to the DHCP Server in a DHCPREQUEST broadcast message. The DHCP Server confirms that the IP address has been allocated to the client by returning a DHCPACK unicast message to the client.

The formal request for the offered IP address (the DHCPREQUEST message) that is sent by the client is broadcast so that all other DHCP Servers that received the DHCPDISCOVER broadcast message from the client can reclaim the IP addresses that they offered to the client.

If the configuration parameters sent to the client in the DHCP OFFER unicast message by the DHCP Server are invalid (a misconfiguration error exists) the client returns a DHCPDECLINE broadcast message to the DHCP Server.

The DHCP Server will send to the client a DHCPNAK denial broadcast message, which means the offered configuration parameters have not been assigned, if an error has occurred during

the negotiation of the parameters or the client has been slow in responding to the DHCPOFFER message (the DHCP Server assigned the parameters to another client) of the DHCP Server.

DHCP Server feature offers the following benefits

1- Reduced Internet access costs

Using automatic IP address assignment at each remote site substantially reduces Internet access costs. Static IP addresses are considerably more expensive to purchase than are automatically allocated IP addresses.

2- Reduced client configuration tasks and costs

Because DHCP is easy to configure, it minimizes operational overhead and costs associated with device configuration tasks and eases deployment by nontechnical users.

3- Centralized management

Because the DHCP Server maintains configurations for several subnets, an administrator only needs to update a single, central server when configuration parameters change.

4- Identify an external File Transport Protocol (FTP), Trivial File Transfer Protocol (TFTP), or remote copy protocol (rcp) server that you will use to store the DHCP bindings database.

5- Identify the IP addresses that you will enable the DHCP Server to assign, and the IP addresses that you will exclude.

6- Identify DHCP options for devices where necessary, including the following:

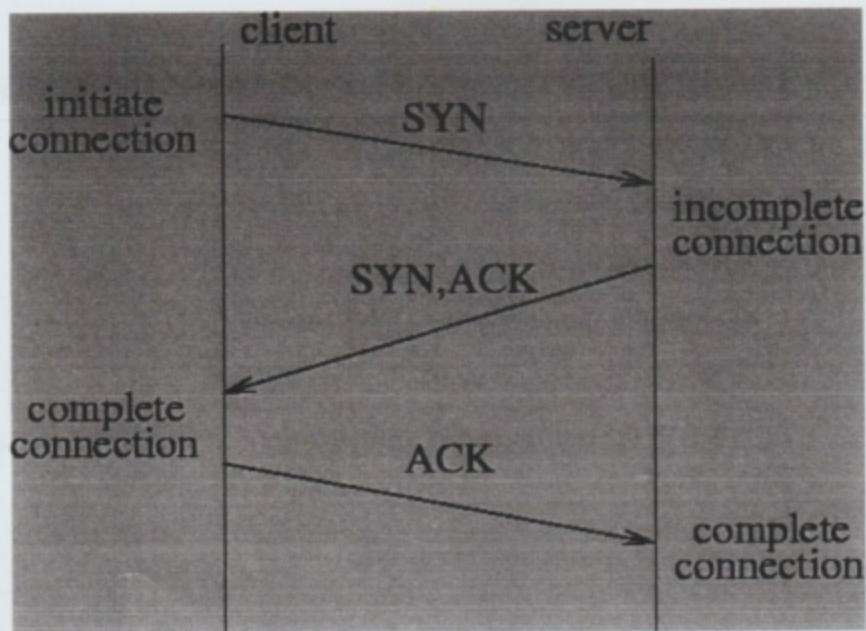
- Default boot image name
- Default routers
- Domain Name System (DNS) servers
- NetBIOS name server

TCP Transmission (Connection)

TCP Connection Establishment Process:

1- The "Three-Way Handshake"

To establish a connection, each device must send a SYN and receive an ACK for it from the other device. Thus, conceptually, we need to have four control messages pass between the devices. However, it's inefficient to send a SYN and an ACK in separate messages when one could communicate both simultaneously. Thus, in the normal sequence of events in connection establishment, one of the SYNs and one of the ACKs is sent together by setting both of the relevant bits (a message sometimes called a SYN+ACK). This makes a total of three messages, and for this reason the connection procedure is called a three-way handshake.



SYN: Synchronize sequence numbers Sent in the first packet when initiating a connection

ACK: Acknowledgement

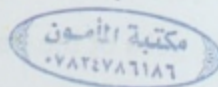
2- Window Size Adjustment and Flow Control (**windowing**)

The importance of the concept of window size to TCP's sliding window mechanism. In a connection between a client and a server, the client tells the server the number of bytes it is willing to receive at one time from the server; this is the client's receive window,

which becomes the server's send window. Likewise, the server tells the client how many bytes of data it is willing to take from the client at one time; this is the server's receive window and the client's send window.

Impact of Buffer Management on TCP Window Size

To understand why the window size may fluctuate, we need to understand what it represents. The simplest way of considering the window size is that it indicates the size of the device's receive buffer for the particular connection. That is, window size represents how much data a device can handle from its peer at one time before it is passed to the application process. Let's consider the aforementioned example. I said that the server's window size was 360. This means the server is willing to take no more than 360 bytes at a time from the client.



When the server receives data from the client, it places it into this buffer. The server must then do two distinct things with this data:

- **Acknowledgment:** The server must send an acknowledgment back to the client to indicate that the data was received.

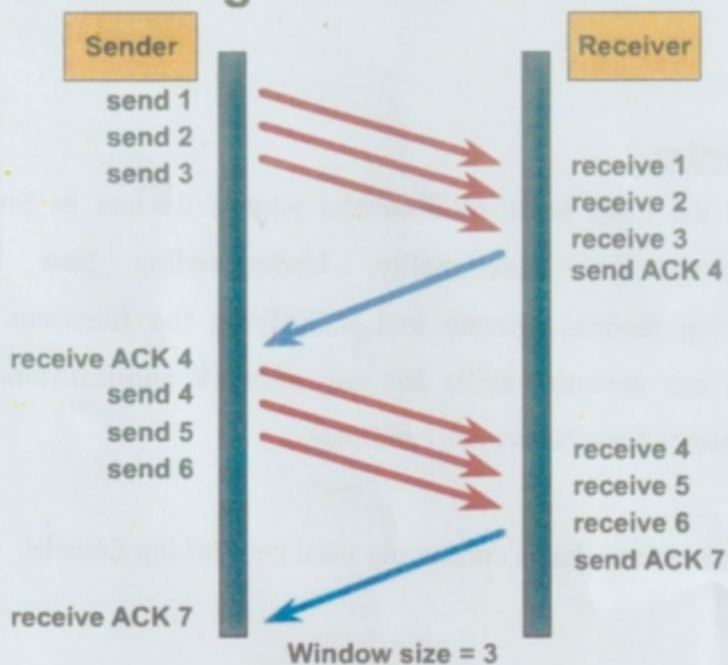
- Transfer: The server must process the data, transferring it to the destination application process.

It is critically important that we differentiate between these two activities. Unfortunately, the TCP standards don't do a great job in this regard, which makes them very difficult to understand. The key point is that in the basic sliding windows system, data is acknowledged when received, but not necessarily immediately transferred out of the buffer. This means that it is possible for the buffer to fill up with received data faster than the receiving TCP can empty it. When this occurs, the receiving device may need to adjust window size to prevent the buffer from being overloaded.

Since the window size can be used in this manner to manage the rate at which data flows between the devices at the ends of the connection, it is the method by which TCP implements flow control, one of the jobs of the transport layer.

Flow control is vitally important to TCP, as it is the method by which devices communicate their status to each other. By reducing or increasing window size, the server and client each ensure that the other device sends data just as fast as the recipient can deal with it.

TCP Sliding Window



Networking Devices

Introduction

All and the most basic of networks require devices to provide connectivity and functionality. Understanding how these networking devices operate and identifying the functions they perform are essential skills for any network administrator and requirements for a Network+ candidate.

This chapter introduces commonly used networking devices

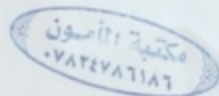
Hubs

Hubs are used in networks that use twisted-pair cabling to connect devices. Hubs can also be joined together to create larger networks. Hubs are simple devices that direct data packets to all devices connected to the hub, regardless of whether the data package is destined for the device. This makes them inefficient devices and can create a performance bottleneck on busy networks. In its most basic form, a hub does nothing except provide a pathway for the electrical signals to travel along. Such a device is called a passive hub.

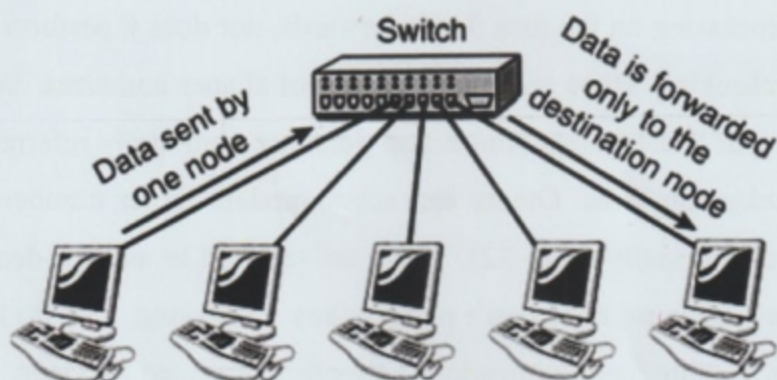
Far more common nowadays is an active hub, which, as well as providing a path for the data signals, regenerates the signal before it

forwards it to all of the connected devices. A hub does not perform any processing on the data that it forwards, nor does it perform any error checking. Hubs come in a variety of shapes and sizes. Small hubs with five or eight connection ports are commonly referred to as workgroup hubs. Others can accommodate larger numbers of devices (normally up to 32). These are referred to as high-density devices. Because hubs don't perform any processing, they do little except enable communication between connected devices. For today's high-demand network applications, something with a little more intelligence is required. That's where switches come in.

Switches



Like hubs, switches are the connectivity points of an Ethernet network. Devices connect to switches via twisted-pair cabling, one cable for each device. The difference between hubs and switches is in how the devices deal with the data that they receive. Whereas a hub forwards the data it receives to all of the ports on the device, a switch forwards it only to the port that connects to the destination device. It does this by learning the MAC address of the devices attached to it, and then by matching the destination MAC address in the data it receives. Figure 3.1 shows how a switch works.



By forwarding data only to the connection that should receive it, the switch can improve network performance in two ways. First, by creating a direct path between two devices and controlling their communication, it can greatly reduce the number of collisions on the network. As you might recall, collisions occur on Ethernet networks when two devices attempt to transmit at exactly the same time. In addition, the lack of collisions enables switches to communicate with devices in full-duplex mode. In a full-duplex configuration, devices can send and receive data from the switch at the same time. Contrast this with half-duplex communication, in which communication can occur in only one direction at a time. Full-

duplex transmission speeds are double that of a standard, half-duplex, connection. So, a 10Mbps connection becomes 20Mbps, and a 100Mbps connection becomes 200Mbps.

The net result of these measures is that switches can offer significant performance improvements over hub-based networks, particularly when network use is high.

Irrespective of whether a connection is at full or half duplex, the method of switching dictates how the switch deals with the data it receives. The following is a brief explanation of each method:

- **Cut-through**—In a cut-through switching environment, the packet begins to be forwarded as soon as it is received. This method is very fast, but creates the possibility of errors being propagated through the network, as there is no error checking.
- **Store-and-forward**—Unlike cut-through, in a store-and-forward switching environment, the entire packet is received and error checked before being forwarded. The upside of this method is that errors are not propagated through the network. The downside is that the error checking process takes a relatively long time, and store-and-forward switching is considerably slower as a result.
- **FragmentFree**—To take advantage of the error checking of store-and forward switching, but still offer performance levels

nearing that of cutthrough switching, FragmentFree switching can be used. In a FragmentFree-switching environment, enough of the packet is read so that the switch can determine whether the packet has been involved in a collision. As soon as the collision status has been determined, the packet is forwarded.

Bridges

Bridges are used to divide larger networks into smaller sections. They do this by sitting between two physical network segments and managing the flow of data between the two. By looking at the MAC address of the devices connected to each segment, bridges can elect to forward the data (if they believe that the destination address is on another interface), or block it from crossing (if they can verify that it is on the interface from which it came). Figure 3.4 shows how a bridge can be used to segregate a network.

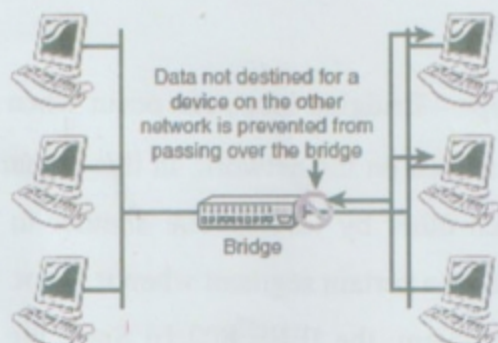


Figure 3.4 How a bridge is used to segregate networks.

When bridges were introduced, the MAC addresses of the devices on the connected networks had to be entered manually, a time-consuming process that had plenty of opportunity for error. Today, almost all bridges can build a list of the MAC addresses on an interface by watching the traffic on the network. Such devices are called learning bridges because of this functionality.

Bridge Placement and Bridging Loops

There are two issues that you must consider when using bridges. The first is the bridge placement, and the other is the elimination of bridging loops:

➤ **Placement**—Bridges should be positioned in the network using the 80/20 rule. This rule dictates that 80% of the data should be local

and that the other 20% should be destined for devices on the other side of the bridge.

➤ **Bridging loops**—Bridging loops can occur when more than one bridge is implemented on the network. In this scenario, the bridges can confuse each other by leading one another to believe that a device is located on a certain segment when it is not. To combat the bridging loop problem, the IEEE 802.1d Spanning Tree protocol enables bridge interfaces to be assigned a value that is then used to control the bridge-learning process.

Types of Bridges

Three types of bridges are used in networks:

➤ **Transparent bridge**—Derives its name from the fact that the devices on the network are unaware of its existence. A transparent bridge does nothing except block or forward data based on the MAC address.

➤ **Source route bridge**—Used in Token Ring networks. The source route bridge derives its name from the fact that the entire path that the packet is to take through the network is embedded within the packet.

➤ **Translational bridge**—Used to convert one networking data format to another; for example, from Token Ring to Ethernet and vice versa.

Today, bridges are slowly but surely falling out of favor. Ethernet switches offer similar functionality; they can provide logical divisions, or segments, in the network. In fact, switches are sometimes referred to as multiport bridges because of the way they operate.

Routers

In a common configuration, routers are used to create larger networks by joining two network segments. Such as a SOHO router used to connect a user to the Internet. A router can be a dedicated hardware device or a computer system with more than one network interface and the appropriate routing software. All modern network operating systems include the functionality to act as a router.

A router derives its name from the fact that it can route data it receives from one network onto another. When a router receives a packet of data, it reads the header of the packet to determine the destination address. Once it has determined the address, it looks in its routing table to determine whether it knows how to reach the destination and, if it does, it forwards the packet to the next hop on

the route. The next hop might be the final destination, or it might be another router. Figure 3.5 shows, in basic terms, how a router works.

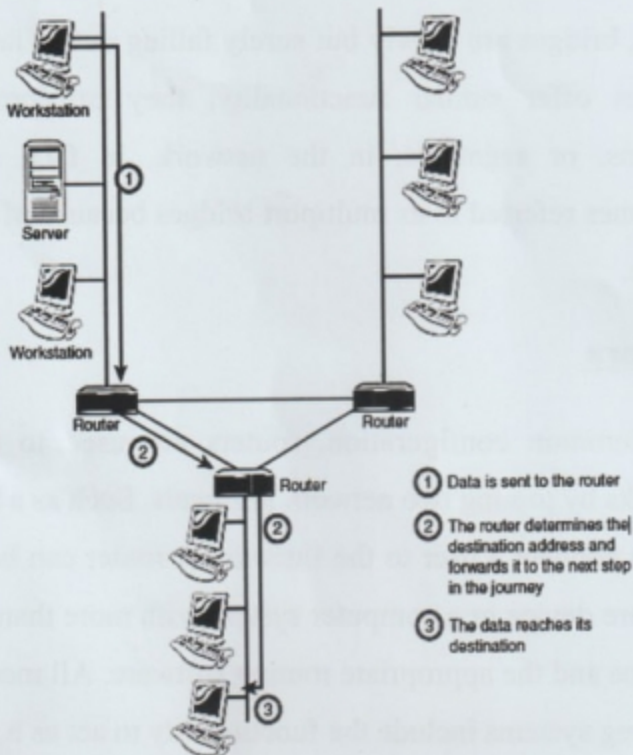


Figure 3.5 How a router works.

As you can see from this example, routing tables play a very important role in the routing process. They are the means by which

the router makes its decisions. For this reason, a routing table needs to be two things. It must be up-to-date, and it must be complete. There are two ways that the router can get the information for the routing table—through static routing or dynamic routing.

Static Routing

In environments that use static routing, routes and route information are entered into the routing tables manually. Not only can this be a time-consuming task, but also errors are more common. Additionally, when there is a change in the layout, or topology, of the network, statically configured routers must be manually updated with the changes. Again, this is a time consuming and potentially error-laden task. For these reasons, static routing is suited to only the smallest environments with perhaps just one or two routers. A far more practical solution, particularly in larger environments, is to use dynamic routing.

Dynamic Routing

In a dynamic routing environment, routers use special routing protocols to communicate. The purpose of these protocols is simple; they enable routers to pass on information about themselves to other routers so that other routers can build routing tables. There are two types of routing protocols used—the older **distance vector protocols** and the newer **link state protocols**.

Distance Vector Routing

The two most commonly used distance vector routing protocols are both called Routing Information Protocol (RIP). One version is used on networks running TCP/IP. The other, sometimes referred to as IPX RIP, is designed for use on networks running the IPX/SPX protocol.

RIP works on the basis of hop counts. A hop is defined as one step on the journey to the data's destination. Each router that the data has to cross to reach its destination constitutes a hop. The maximum number of hops that RIP can accommodate is 15. That is to say that in a network that uses RIP, all routers must be within 15 hops of each other to communicate. Any hop count that is in excess of 15 is considered unreachable.

Distance vector routing protocols operate by having each router send updates about all the other routers it knows about to the routers directly connected to it. These updates are used by the routers to compile their routing tables. The updates are sent out automatically every 30 or 60 seconds. The actual interval depends on the routing protocol being used. Apart from the periodic updates, routers can also be configured to send a triggered update if a change in the network topology is detected. The process by which routers learn of a change in the network topology is known as convergence.

Although distance vector protocols are capable of maintaining routing tables, they have three problems. The first is that the periodic update system can make the update process very slow. The second problem is that the periodic updates can create large amounts of network traffic—much of the time unnecessarily as the topology of the network should rarely change. The last, and perhaps more significant, problem is that because the routers only know about the next hop in the journey, incorrect information can be propagated between routers, creating routing loops.

Two strategies are used to combat this last problem. One, split horizon, works by preventing the router from advertising a route back to the other router from which it was learned. The other, poison reverse (also called split horizon with poison reverse), dictates that the route is advertised back on the interface from which it was learned, but that it has a metric of 16. Recall that a metric of 16 is considered an unreachable destination.

Link State Routing

Link state routing works quite differently from distance vector based routing. Rather than each router telling each other connected router about the routes it is aware of, routers in a link state environment send out special packets, called link state advertisements (LSA), which contain information only about that router. These LSAs are

forwarded to all the routers on the network, which enables them to build a map of the entire network. The advertisements are sent when the router is first brought onto the network and when a change in the topology is detected.

Of the two (distance vector and link state), distance vector routing is better suited to small networks and link state routing to larger ones. Link state protocols do not suffer from the constant updates and limited hop count, and they are also quicker to correct themselves (to converge) when the network topology changes.

On TCP/IP networks, the most commonly used link state routing protocol is the Open Shortest Path First (OSPF). On IPX networks, the NetWare Link State Protocol (NLSP) is used. Table 3.1 summarizes the distance vector and link state protocols used with each network protocol.

Device	Function/Purpose	Key Points
Hub	Connects devices on a twisted-pair network.	A hub does not perform any tasks besides signal regeneration.
Switch	Connects devices on a twisted-pair network.	A switch forwards data to its destination by using the MAC address embedded in each packet.
Bridge	Divides networks to reduce overall network traffic.	A bridge allows or prevents data from passing through it by reading the MAC address.
Router	Connects networks together.	A router uses the software-configured network address to make forwarding decisions.