KARPAGAM ACADEMY OF HIGHER EDUCATION
(Deemed to be University)
(Established Under Section 3 of UGC Act ,1956)
COIMBATORE - 641 021

Syllabus

YEAR: 2019 -2021 (LATERAL ENTRY)

Semester-IV

18CAP405N	**TCP/IP**	**4H- 4C**

Instruction Hours / week: L: 4 T: 0 P: 0 C: 4 **Marks: Internal: 40 External: 60 Total: 100**

End Semester Exam: 3Hours

COURSE OBJECTIVE

- To learn about IPv4 forwarding and routing.
- To learn about host name resolution and the Domain Name System (DNS).
- To learn about IPv6 addresses.
- To know the Architectural Overview of the TCP/IP Protocol Suite

COURSE OUTCOMES (COs)

- Configure subnets using IP classes B and C
- Explain TCP/IP protocols, ports, sockets, and data encapsulation
- Describe the process of packet fragmentation and reassembly
- Explain the key features and functions of TCP and UDP
- Use Wireshark to identify ICMP request and reply packets
- Describe the DHCP discovery process
- Explain DNS queries, name resolution, zone data transfers and reverse DNS queries
- Describe how basic routing works including the use of routing protocols

UNIT I

Introduction: WAN, WAN technologies - Internetworking concepts - Protocols and Standards - TCP/IP protocol suite - Internetworking Devices – Routing Concept - Classful IP Addressing – Subnetting – Supernetting – Classless Addressing

UNIT II

ARP & RARP – Proxy ARP – ARP over ATM – ARP and RARP Protocol Format. IP Datagram – Fragmentation – Options – IP Datagram Format – Routing IP Datagrams – Checksum. ICMP : Types of Messages - Message Format – Error Reporting – Query – Checksum - ICMP Package

UNIT III

Routing and Routed Protocols - Autonomous Systems – Routing Table - Interior Gateway Protocols – Exterior Gateway Protocols – Routing in Internet. Group Management – IGMP Message – IGMP Operation – Process to Process Communication.

UNIT IV

UDP Operation – TCP Services - Flow Control – Multicast Routing – Multicast Routing Protocols. BOOTP - DHCP – Address Discovery and Binding. DNS – Name Space – DNS in Internet – Resolution – Resource Records.

UNIT V

Remote Login - FTP – SMTP – SNMP. IP over ATM Wan – Cells – Routing the Cells. Mobile IP : Addressing – Agents – Agent discovery – Registration – Data Transfer – VPN.

SUGGESTED READINGS

1. Behrouz A. Forouzan. (2010), TCP/IP Protocol Suite, 4th Edition. New Delhi: Tata McGraw Hill Publication.
2. Douglas E. Comer (2000), Internetworking With TCP/IP, Vol 1: Principles Protocols and Architecture. 4th Edition. New Delhi: Pearson Education.
3. William Stallings (1997), Data and Computer Communication. 5th Edition. New Delhi: Prentice Hall of India.

WEB SITES

1. en.wikipedia.org/wiki/Internet_protocol_suite
2. http://docwiki.cisco.com/wiki/Introduction_to_WAN_Technologies
3. www.yale.edu/pclt/COMM/TCPIP.HTM
4. www.w3schools.com/tcpip/default.asp

Question Paper Pattern:

CIA	**Max.Marks : 50**
Part – A	Objective type (20 x 1=20)
Part- B	Short Answer Type (3 x 2 =6)
Part- C	3 Eight marks Questions 'either – or' Type Choice (3 x 8 =24 Marks)

ESE	**Max.Marks : 60**
Part – A	20 Questions (20 x 1 = 20 Marks) Question No. 1 to 20 Online Multiple Choice Questions
Part- B	5 six mark Questions (5 x 6 = 30 Marks.) Question No. 21 to 25 will be 'either-or' type, covering all five units of the syllabus; i.e.,Question No. 21: Unit - I, either 21 (a) or 21 (b), Question No. 22: Unit - II, either 22 (a) or 22 (b), Question No. 23: Unit - III, either 23 (a) or 23 (b), Question No. 24: Unit - IV, either 24 (a) or 24 (b), Question No. 25: Unit - V, either 25 (a) or 25 (b)
Part - C	One Ten mark Question (1 x 10 = 10 Marks) Question No.26.

KARPAGAM ACADEMY OF HIGHER EDUCATION
(Deemed to be University)
(Established Under Section 3 of UGC Act, 1956)
Coimbatore-21

DEPARTMENT OF CS, CA & IT

TCP/ IP 18CAP405N

Lesson Plan

UNIT-I			
S.No	**Lecture Duration Period**	**Topics to be Covered**	**Support Material/Page Nos**
1.	1	Introduction : WAN , WAN Technologies	T1:2-6,W1
2.	1	Internetworking Concepts, Protocols and Standards	T1:6-11,23-27, J1
3.	1	TCP/IP Protocol Suite	T1:30 – 32
4.	1	Internetworking Devices, Routing Concept	T1:69 – 87
5.	1	Classful IP Addressing	T1:148 – 149
6.	1	Subnetting, Supernetting	T1:102-109
7.	1	Classless IP Addressing	T1:115-124
8.	1	Recapitulation and Discussion of important questions	T1: 71-74, R1:56-62

SUGGESTED READINGS:

T1 : Behrouz A. Forouzan. 2010. TCP/IP Protocol Suite. 4th Edition. New Delhi: Tata McGraw Hill Publication.

R1 : Douglas E. Comer.2000. Internetworking With TCP/IP, Vol 1: Principles Protocols and Architecture. 4th Edition. New Delhi: Pearson Education.

WEB SITE :

W1 : en.wikipedia.org/wiki/Internet_protocol_suite

JOURNAL:

J1 :olabenjo Babatunde, Omar AI-Debagy, "A Comparative Review of Internet Protocols (IPV4) and Internat Protocol version6(IPV6), IJCIT, Vol.13 No.1, July 2014.

UNIT-II			
S.No	**Lecture Duration Period**	**Topics to be Covered**	**Support Material/Page Nos**
1.	1	ARP and RARP, Proxy ARP, ARP over ATM WAN	T1:160-171
2.	1	ARP and RARP Protocol Format	T1:173-175
3.	1	IP Datagram, Fragmentation , Options	T1:180-200
4.	1	IP datagram Format, Routing IP Datagram, Checksum	T1:200-203,J1
5.	1	ICMP, Types of Messages, Message Format	T1:212-213 R2:129-132
6.	1	Error Reporting, Query Checksum	T1:213-227, W2
7.	1	ICMP Package	T1:232-234
8.	1	Recapitulation and Discussion of Important Questions	

SUGGESTED READINGS:

T1 : Behrouz A. Forouzan. 2010. TCP/IP Protocol Suite. 4th Edition. New Delhi: Tata McGraw Hill Publication.

R2 : William Stallings.1997.Data and Computer Communication. 5th Edition. New Delhi: Prentice Hall of India.

WEB SITE :

W2 : http://docwiki.cisco.com/wiki/Introduction_to_WAN_Technologies

JOURNAL:

J1 :olabenjo Babatunde, Omar AI-Debagy, "A Comparative Review of Internet Protocols (IPV4) and Internat Protocol version6(IPV6), IJCIT, Vol.13 No.1, July 2014.

UNIT-III			
S.No	**Lecture Duration Period**	**Topics to be Covered**	**Support Material/Page Nos**
1.	1	Routing and Routed Protocols	T1:131 - 148 , R2:350-373
2.	1	Autonomous Systems, Routing Table	T1:307- 386
3.	1	Interior Gateway Protocols, Exterior gateway Protocols	R2:454-458
4.	1	Routing in Internet,	T1:418-427
5.	1	Group Management, IGMP Messages	T1:237-239,W4
6.	1	IGMP Operation	T1:239 - 245
7.	1	Process to Process Communication	R2:256 - 260
8.	1	Recapitulation and Discussion of Important Questions	

SUGGESTED READINGS:

T1 : Behrouz A. Forouzan. 2010. TCP/IP Protocol Suite. 4th Edition. New Delhi: Tata McGraw Hill Publication.

R2 : William Stallings.1997.Data and Computer Communication. 5th Edition. New Delhi: Prentice Hall of India.

WEB SITE :

W4 : www.w3schools.com/tcpip/default.asp

UNIT-IV			
S.No	**Lecture Duration Period**	**Topics to be Covered**	**Support Material/Page Nos**
1.	1	UDP Operation, TCP Services	T1:201-2013, 264-279,w1
2.	1	Flow control	T1:299-304 R1:216 – 220
3.	1	Multicasting Routing, Multicast Routing Protocols	T1:441-446
4.	1	BOOTP, DHCP	T1:457 –467 R1:450-452
5.	1	Address Discovery and Binding	W2
6.	1	Domain Name System (DNS), Name Space	T1:471-476,w1
7.	1	DNS in Internet, Resolution, Resource Records	T1:477-487
8.	1	Recapitulation and Discussion of Important Questions	

SUGGESTED READINGS:

T1 : Behrouz A. Forouzan. 2010. TCP/IP Protocol Suite. 4th Edition. New Delhi: Tata McGraw Hill Publication.

R1 : Douglas E. Comer.2000. Internetworking With TCP/IP, Vol 1: Principles Protocols and Architecture. 4th Edition. New Delhi: Pearson Education.

WEB SITE :

W1 : en.wikipedia.org/wiki/Internet_protocol_suite

W2 : http://docwiki.cisco.com/wiki/Introduction_to_WAN_Technologies

UNIT- V			
S.No	**Lecture Duration Period**	**Topics to be Covered**	**Support Material/Page Nos**
1.	1	Remote Login , File Transfer Protocol (FTP), SMTP, SNMP	T1:499-532, W1:547-580 R1:553-569
2.	1	IP over ATM WAN, Cells, Routing the Cells	T1:621-625, T1:626-632
3.	1	Mobile IP: Addressing, Agents ,Agent Discovery	T1:637-638 T1:639-641,W2
4.	1	Registration, Data Transfer, Virtual Private Networks (VPN)	T1:642-645 T1:680-685, J1
5.	1	Recapitulation and Discussion of Important Questions	
6.	1	Discussion of Previous ESE Question Papers	
7.	1	Discussion of Previous ESE Question Papers	
8.	1	Discussion of Previous ESE Question Papers	

SUGGESTED READINGS:

T1 : Behrouz A. Forouzan. 2010. TCP/IP Protocol Suite. 4th Edition. New Delhi: Tata McGraw Hill Publication.

R1 : Douglas E. Comer.2000. Internetworking With TCP/IP, Vol 1: Principles Protocols and Architecture. 4th Edition. New Delhi: Pearson Education.

R2 : William Stallings.1997.Data and Computer Communication. 5th Edition. New Delhi: Prentice Hall of India.

WEB SITE :

W1 : en.wikipedia.org/wiki/Internet_protocol_suite

W2 : http://docwiki.cisco.com/wiki/Introduction_to_WAN_Technologies

W3 : www.yale.edu/pclt/COMM/TCPIP.HTM

W4 : www.w3schools.com/tcpip/default.asp

JOURNAL:

J1 :olabenjo Babatunde, Omar AI-Debagy, "A Comparative Review of Internet Protocols (IPV4) and Internat Protocol version6(IPV6), IJCIT, Vol.13 No.1, July 2014.

UNIT-I

SYLLABUS

Introduction: WAN, WAN technologies - Internetworking concepts - Protocols and Standards - TCP/IP protocol suite- Internetworking Devices – Routing Concept - Classful IP Addressing – Subnetting – Supernetting – Classless Addressing

1. NETWORKS

Network can be defined as interconnected connection of Computers. A computer network is simply two or more computers connected together so they can exchange information. A small network can be as simple as two computers linked together by a single cable. Most networks use hubs to connect computers together. A large network may connect thousands of computers and other devices together.

Figure 1 a) Wired Network b) Wireless Network

A wireless network connects computers without a hub or network cables. Computers use radio communications to send data between each other. Without a network, you can access resources only on your own computer. These resources may be devices in your computer, such as a folder or disk drive, or they may be connected to your computer, such as a printer or CDROM drive. These devices, accessible only to you, are local resources. Networking allows you to share resources among a group of computer users. Resources among a group of computer users. Each computer on your network can share folders, entire disk drives, or a CD-ROM drive. Then other computers on your network can access documents and other files stored in the folders and on the drives If you have a printer connected to your computer,

you can share the printer with other computers on the network. Then instead of buying a printer for every computer, all the computers can print across the network to the printer.

1. 1 USES OF NETWORKS

File sharing - Network file sharing between computers gives you more flexibility than using floppy drives or Zip drives. Not only can you share photos, music files, and documents, you can also use a home network to save copies of all of your important data on a different computer. *Backups* are one of the most critical yet overlooked tasks in home networking.

Printer / Peripheral sharing - Once a home network is in place, it's easy to then set up all of the computers to share a single printer. No longer will you need to bounce from one system or another just to print out an email message. Other computer peripherals can be shared similarly such as network scanners, Web cams, and CD burners.

Internet connection sharing - Using a home network, multiple family members can access the Internet simultaneously without having to pay an ISP for multiple accounts. You will notice the Internet connection slows down when several people share it, but broadband Internet can handle the extra load with little trouble. Sharing dial-up Internet connections works, too. Painfully slow sometimes, you will still appreciate having shared dial-up on those occasions you really need it.

Multi-player games - Many popular home computer games support *LAN mode* where friends and family can play together, if they have their computers networked.

Internet telephone service - So-called Voice over IP (VoIP1) services allow you to make and receive phone calls through your home network across the Internet, saving you money.

Home entertainment - Newer home entertainment products such as digital video recorders (DVDs) and video game consoles now support either wired or wireless home networking. Having these products integrated into your network enables online Internet gaming, video sharing and other advanced features.

Other Applications of Networks

1. Mobile Technology
2. Banking Sectors
3. Railway and Airline Reservation
4. E Learning
5. Research.

WAN TECHNOLOGY

A *WAN* is a data communications network that covers a relatively broad geographic area and that often uses transmission facilities provided by common carriers, such as telephone companies. WANs are used to connect LANs and other types of networks together, so that users and computers in one location can communicate with users and computers in other locations. Many WANs are built for one particular organization and are private. Others, built by Internet service providers, provide connections from an organization's LAN to the Internet. WAN technologies generally function at the lower three layers of the OSI reference model: the physical layer, the data link layer, and the network layer. WANs are often built using leased lines. At each end of the leased line, a router connects to the LAN on one side and a hub within the WAN on the other. Leased lines can be very expensive. Instead of using leased lines, WANs can also be built using less costly circuit switching or packet switching methods. Network protocols including TCP/IP deliver transport and addressing functions

S.No	Option	Description
1	Leased Line	Point-to-Point connection between two computers or Local Area Networks (LANs)
2	Circuit Switching	A dedicated circuit path is created between end points. Best example is dialup connections
3	Packet Switching	Devices transport packets via a shared single point-to-point or point-to-multipoint link across a carrier internetwork.
4	Cell Replay	Similar to packet switching, but uses fixed length cells instead of variable length packets. Data is divided into fixed-length cells and then transported across virtual circuits

WAN DEVICES:

WANs use numerous types of devices that are specific to WAN environments such as WAN switches, access servers, modems and ISDN terminal adapters. A WAN switch is a multiport internetworking device used in carrier networks. An access server acts as a concentration point for dial-in and dial-out connections. An ISDN terminal adapter is a device used to connect ISDN Basic Rate Interface (BRI) connections to other interfaces

2 INTERNET

An interconnection of Network is called internet. The Internet is a global system of interconnected computer networks that use the standardized Internet Protocol Suite (TCP/IP) to serve billions of users worldwide. It is a *network of networks* that consists of millions of private and public, academic, business, and government networks of local to global scope that are linked by copper wires, fiber-optic cables, wireless connections, and other technologies. The Internet carries a vast array of information resources and services, most notably the inter-linked hypertext documents of the World Wide Web (WWW) and the infrastructure to support electronic mail. In addition it supports popular services such as online chat, file transfer and file sharing, gaming, commerce, social networking, publishing, video on demand, and teleconferencing and telecommunications. Voice over Internet Protocol (VoIP) applications allow person-to-person communication via voice and video.

2.1. HISTORY OF INTERNET

The origins of the Internet reach back to the 1960s. In the mid 1960's mainframe organizations were stand alone devices. Computers from different manufacturers were unable to communicate with one another. Hardware devices are incompatible with other devices and even software will not be able to install in the system. The Advanced Research Project Agency (ARPA) in the department of Defense DOD) was interested in finding a solution to connect computers together so that the information can be shared among others such as researchers can share their findings with others. The ARPANET team proposed an idea by which each host would be attached to a specialized computer called an Interface Message Processor. (IMP). The IMP in turn connected to each machine.. Each IMP can communicate with other IMP and also with their own host. Network Control Protocol (NCP) is software that is used to provide communication between hosts.

In 1983 TCP Protocol were introduced. It includes the concept of encapsulation, datagram, and the function of gateway. After a while TCP split into protocols namely TCP (Transmission Control Protocol and IP (Internetworking Protocol. IP would handle datagram routing and TCP responsible for higher level Functions such as segmentation, reassembly and error detection.

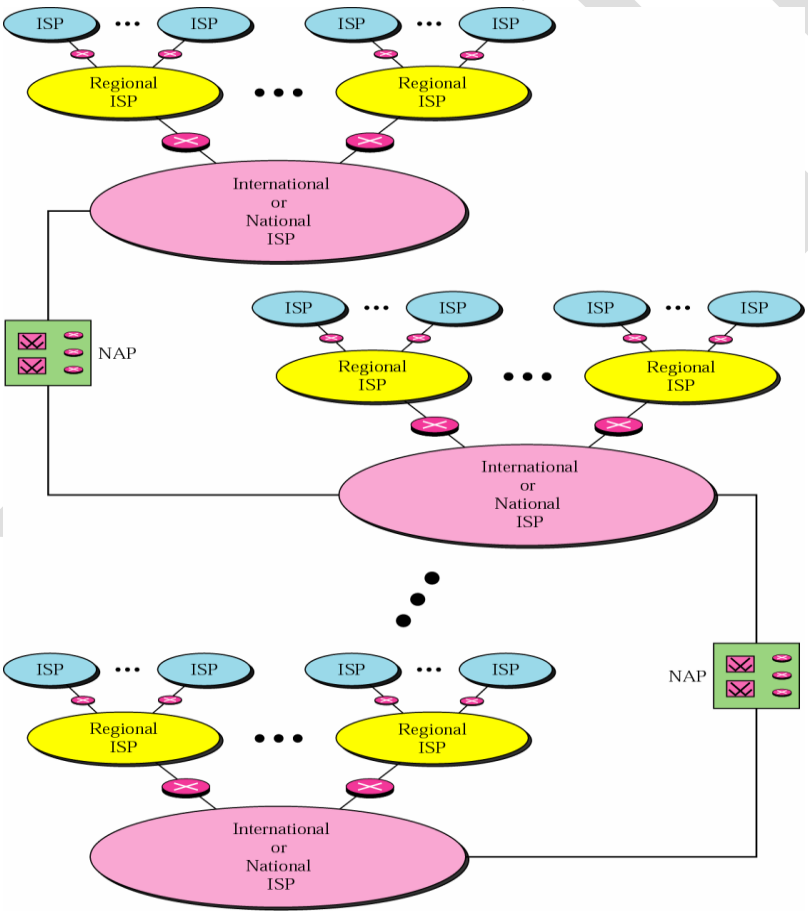
In 1970, ARPANET hosts started to use Network Control Protocol (NCP), a preliminary form of what would become the Transmission Control Protocol (TCP).

In 1972, the Telnet protocol was introduced. Telnet is used for terminal emulation to connect dissimilar systems. In the early 1970s, these systems were different types of mainframe computers. In 1973, the File Transfer Protocol (FTP) was introduced. FTP is used to exchange files between dissimilar systems. In 1974, the Transmission Control Protocol (TCP) was specified in detail. TCP replaced NCP and

provided enhanced reliable communication services. · In 1981, the Internet Protocol (IP) (also known as IP version 4 [IPv4]) was specified in detail. IP provides addressing and routing functions for end-to-end delivery.

In 1982, the Defense Communications Agency (DCA) and ARPA established the Transmission Control Protocol (TCP) and Internet Protocol (IP) as the TCP/IP protocol suite. · In 1983, ARPANET switched from NCP to TCP/IP. · In 1984, the Domain Name System (DNS) was introduced. DNS resolves domain names (such as www.example.com) to IP addresses (such as 192.168.5.18). In 1995, Internet service providers (ISPs) began to offer Internet access to businesses and individuals. · In 1996, the Hypertext Transfer Protocol (HTTP) was introduced. The World Wide Web uses HTTP.

In 1996, the first set of IP version 6 (IPv6) standards were published.



2.2 PROTOCOLS AND STANDARDS

A Protocol is a set of rules that govern data communication. The protocol defines what is communicated, how it is communicated, and when it is communicated.. Protocols are rules

governing communication between devices or applications, and the creation or manipulation of any logical or communicative artifacts concomitant with such communication. The Internet and web are built on a slew of such protocols, including:
hyper-text transfer protocol (HTTP)
file transfer protocol (FTP)
transmission control protocol / internet protocol (TCP/IP)
secure sockets layer

Protocols like these provide the backbone instructions for moving information around Internet and web. The elements of a protocol are Syntax, semantics and timing..

2.3 STANDARD ORGANIZATION

The Internet Society (ISOC) was created in 1992 and is a global organization responsible for the internetworking technologies and applications of the Internet. Although the society's principal purpose is to encourage the development and availability of the Internet, it is also responsible for the further development of the standards and protocols that allow the Internet to function.

The ISOC sponsors the Internet Architecture Board (IAB), a technical advisory group that sets Internet standards, publishes RFCs, and oversees the Internet standards process. The IAB governs the following bodies:

1 The Internet Assigned Number Authority (IANA) oversees and coordinates the assignment of protocol identifiers used on the Internet.

2 The Internet Research Task Force (IRTF) coordinates all TCP/IP-related research projects.

3 The Internet Engineering Task Force (IETF) solves technical problems and needs as they arise on theInternet and develops Internet standards and protocols. IETF working groups define standards known as RFCs.

3.1 OSI REFERENCE MODEL

Open Systems Interconnection (OSI) is a standard reference model for communication between two end users in a network. The model is used in developing products and understanding networks. OSI divides telecommunication into seven layers. The layers are in two groups. The upper four layers are used whenever a message passes from or to a user. The lower three layers are used when any message passes through the host computer. Messages intended for this computer pass to the upper layers. Messages destined for some other host are not passed up to the upper layers but are forwarded to another host. The seven layers are:

Layer 7: The application layer ...This is the layer at which communication partners are identified, quality of service is identified, user authentication and privacy are considered, and any constraints on data syntax are identified. (This layer is *not* the application itself, although some applications may perform application layer functions.)

Layer 6: The presentation layer ...This is a layer, usually part of an operating system, that converts incoming and outgoing data from one presentation format to another (for example, from a text stream into a popup window with the newly arrived text). Sometimes called the syntax layer.

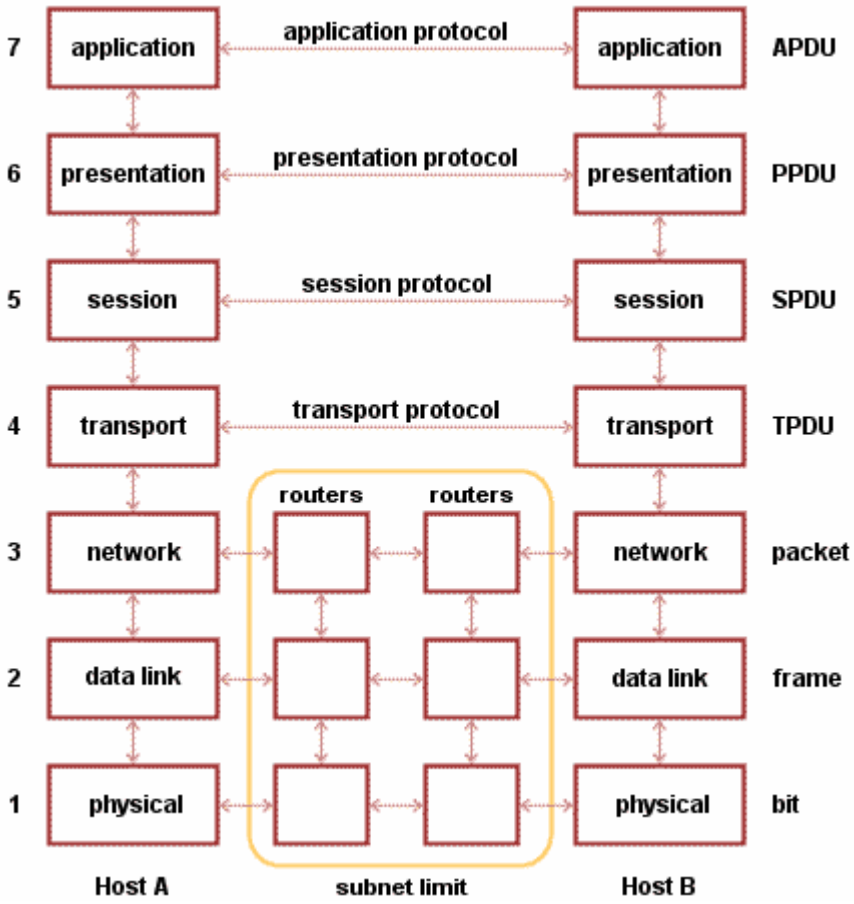
Layer 5: The session layer ...This layer sets up, coordinates, and terminates conversations, exchanges, and dialogs between the applications at each end. It deals with session and connection coordination.

Layer 4: The transport layer ...This layer manages the end-to-end control (for example, determining whether all packets have arrived) and error-checking. It ensures complete data transfer.

Layer 3: The network layer ...This layer handles the routing of the data (sending it in the right direction to the right destination on outgoing transmissions and receiving incoming transmissions at the packet level). The network layer does routing and forwarding.

Layer 2: The data-link layer ...This layer provides synchronization for the physical level and does bit-stuffing for strings of 1's in excess of 5. It furnishes transmission protocol knowledge and management.

Layer 1: The physical layer ...This layer conveys the bit stream through the network at the electrical and mechanical level. It provides the hardware means of sending and receiving data on a carrier.



TCP/IP Protocol Suite

The TCP/IP protocol suite is made of five layers: physical, data link, network, transport, and application. The first four layers provide physical standards, network interface, internetworking, and transport functions that correspond to the first four layers of the OSI model. The three topmost layers in the OSI model, however, are represented in TCP/IP by a single layer called the application layer.

TCP/IP model define 4 layers that are as follows:

1. Internet Layer.

Packet switching network depends upon a connectionless internet work layer. This layer is known as internet layer, is the linchpin that holds the whole design together. Its job is to allow hosts to insert packets into any network and have them to deliver independently to the destination. They may appear in a different order than they were sent in each case it is job of higher layers to rearrange them in order to deliver them to proper destination.

The internet layer specifies an official packet format and protocol known as internet protocol. The job of internet layer is to transport IP packets to appropriate destination. Packet routing is very essential task

in order to avoid congestion. For these reason it is say that TCP/IP internet layer perform same function as that of OSI network layer.

2) Transport layer :

In the TCP/IP model, the layer above the internet layer is known as transport layer. It is developed to permit entities on the source and destination hosts to carry on a conversation. It specifies 2 end-to-end protocols.

a)TCP(Transmission Control Protocol)
b)UDP (User Datagram Protocol)

a**) TCP**

It is a reliable connection-oriented protocol that permits a byte stream originating on one machine to be transported without error on any machine in the internet. It divides the incoming byte stream into discrete message and passes each one onto the internet layer. At the destination, the receiving TCP process collects the received message into the output stream. TCP deals with flow control to make sure a fast sender cannot swamp a slow receiver with more message.

b**)UDP**

It is an unreliable, connectionless protocol for applications that do not want TCP's sequencing on flow control and wish to offer their own. It is also used for client-server type request-reply queries and applications in which prompt delivery is more important than accurate delivery such as transmitting speech or video.

3. Application Layer :

In TCP/IP model, session or presentation layer are not present. Application layer is present on the top of the Transport layer. It includes all the higher-level protocols which are virtual terminal (TELNET), file transfer (FTP) and electronic mail (SMTP).The virtual terminal protocol permits a user on one machine to log into a distant machine and work there. The file transfer protocol offers a way to move data efficiently from one machine to another.

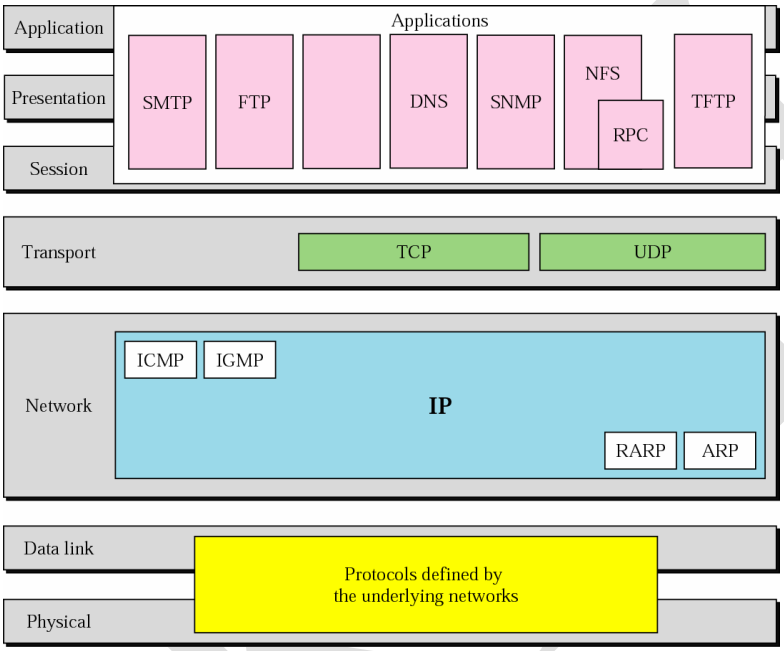
File Transfer Protocol (FTP)

It was designed to permit reliable transfer of files over different platforms. At the transport layer to ensure reliability, FTP uses TCP. FTP offers simple commands and makes the differences in storage methods across networks transparent to the user. The FTP client is able to interact with any FTP server; therefore the FTP server must also be able to interact with any FTP client. FTP does not offer a user

interface, but it does offer an application program interface for file transfer. The client part of the protocol is called as FTP and the server part of the protocol is known as FTPd. The suffix "d" means Daemon this is a legacy from Unix computing where a daemon is a piece of software running on a server that offers a service.

Hyper Text Transfer Protocol

HTTP permits applications such as browsers to upload and download web pages. It makes use of TCP at the transport layer again to check reliability. HTTP is a connectionless protocol that sends a request, receives a response and then disconnects the connection. HTTP delivers HTML documents plus all of the other components supported within HTML such as JavaScript, Visual script and applets.



Simple Mail Transfer Protocol

By using TCP, SMTP sends email to other computers that support the TCP/IP protocol suite. SMTP provides extension to the local mail services that existed in the early years of LANs. It supervises the email sending from the local mail host to a remote mail host. It is not reliable for accepting mail from local users or distributing received mail to recipients this is the responsibility of the local mail system.

SMTP makes use of TCP to establish a connection to the remote mail host, the mail is sent, any waiting mail is requested and then the connection is disconnected. It can also return a forwarding address if the intended recipient no longer receives email at that destination. To enable mail to be delivered across

differing systems, a mail gateway is used.

Simple Network Management Protocol

For the transport of network management information, SNMP is used as standardized protocol. Managed network devices can be cross examined by a computer running to return details about their status and level of activity. Observing software can also trigger alarms if certain performance criteria drop below acceptable restrictions. At the transport layer SNMP protocol uses UDP. The use of UDP results in decreasing network traffic overheads.

4) The Host to Network Layer:

Below the internet layer is great void. The TCP/IP reference model does not really say such about what happen here, except to point out that the host has connect to the network using some protocol so it can transmit IP packets over it. This protocol is not specified and varies from host to host and network to network.

Services	OSI Model Reference	TCP/IP Model Reference
Service, interface and protocol	Service, interface and protocol are not clearly defined. For example, the only real services offered by the Internet layer are - Send IP Packet - Receive IP Packet	Protocols in the OSI model are better hidden and can be replaced relatively easily as the technology changes, which is one of the main objective of layered protocols.
Functionalities	Because models were invented before protocols, functionalities put in each layer are not very optimized.	In this case, the protocols have been invented before models, so the functionalities are perfectly described.
Numbers of layers	Seven layers, Network (Internet), Transport and Application layers being similar to TCP/IP	Only four layers.
Connectionless/ Connection-oriented	Both connectionless and connection-oriented	Only one mode in the network layer (connectionless) but both modes in

communication	communication are supported in the network layer, but only connection-oriented communication in the transport layer.	the transport layer are supported, giving the users a choice.

These are the differences between OSI reference model and TCP/IP Model.

4. ADDRESSING AND INTERCONNECTING DEVICES

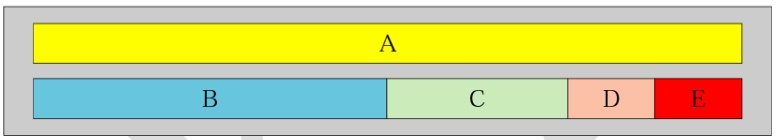
IP ADDRESING

The identifier used in the IP layer of the TCP/IP protocol suite to identify each device connected to the Internet is called the Internet address or IP address. An IP address is a 32-bit address that uniquely and universally defines the connection of a host or a router to the Internet. IP 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.

Address Space:

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 and 1) and *N* bits can have 2*N* values

Address space



IPV4 uses 32 bit addresses which mean that the address space is 2^{32}.The address space of IPv4 is 232 or 4,294,967,296.

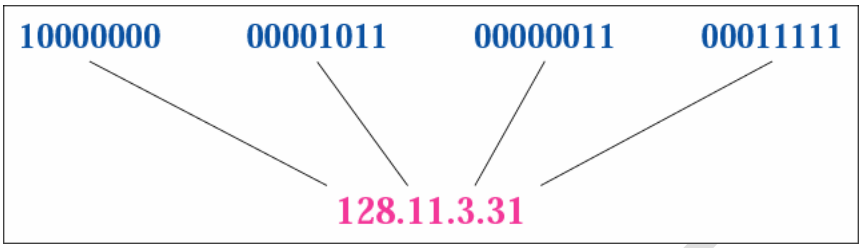
Notation of IP Address:

There are three common notations to show an IP address such as Binary notation, Dotted decimal notation and Hexa decimal notation.

1. Binary Notation

01110101 10010101 00011101 11101010

2. Dotted Decimal Notation



3. Hexa Decimal Notation:

0111 0101 1001 0101 0001 1101 1110 1010

75 95 1D EA

0x75951DEA

Example : 1

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

a.10000001	00001011	00001011	11101111
b.11000001	10000011	00011011	11111111
c.11100111	11011011	10001011	01101111
d.11111001	10011011	11111011	00001111

We replace each group of 8 bits with its equivalent decimal number (see Appendix B) and add dots for separation:

a. 129.11.11.239 b. 193.131.27.255
c. 231.219.139.111 d. 249.155.251.15

Example : 2

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

a. 111.56.45.78 b. 221.34.7.82
c. 241.8.56.12 d. 75.45.34.78

We replace each decimal number with its binary equivalent:

a. 01101111 00111000 00101101 01001110
b. 11011101 00100010 00000111 01010010
c. 11110001 00001000 00111000 00001100
d. 01001011 00101101 00100010 01001110

4.3. INTERNETWORKING DEVICES

Some of the interconnecting devices available for connections are

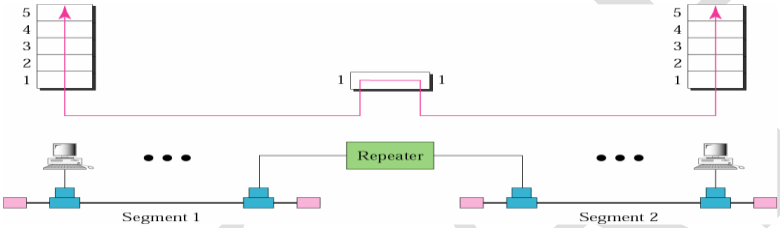
1) Repeaters

2) Bridges

3) Routers

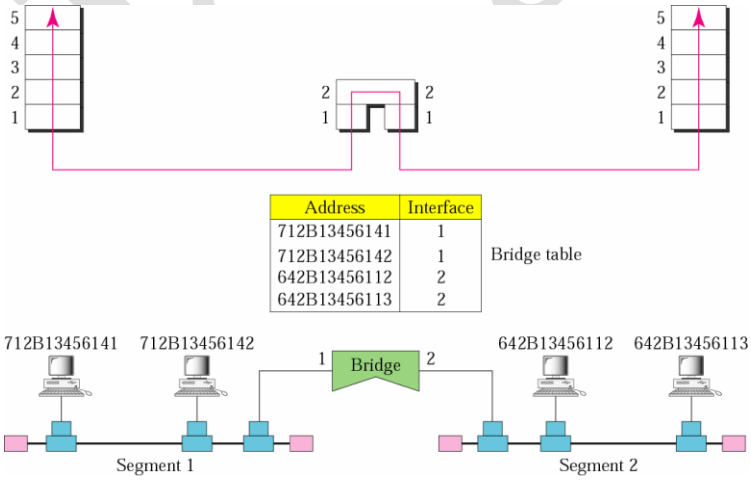
1) Repeater:

A Repeater Connects segments of LANS Together A repeater just forward every Packet. It has no filtering capacity.



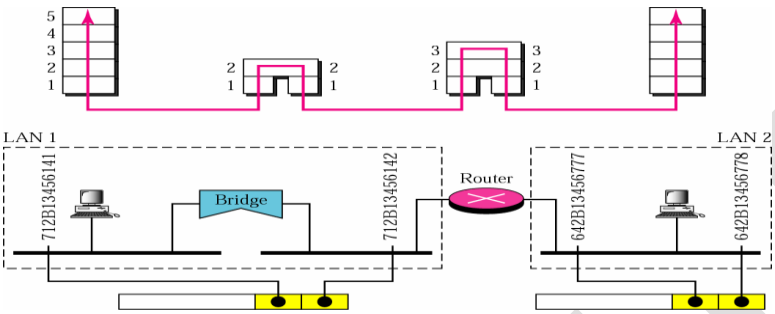
2. Bridges :

A Bridges Connects segments of LANS Together It has a table used for filtering of packets.



3. Router:

Router connects two independent LANs Together. It is a device that forwards data packets along networks. It uses Headers and Forwarding Tables to determine the best path for forwarding the packets. It uses ICMP Protocol for Communication Very little Filtering is done.



5. IP ADDRESSING

Every network interface on a TCP/IP device is identified by a globally unique IP address. Host devices, for example, PCs, typically have a single IP address.Routers typically have two or more IP addresses, depending on the number of interfaces they have. Each IP address is 32 bits long and is composed of four 8-bit fields, called octets. The address is normally represented in 'dotted decimal notation. Two types of IP addressing are

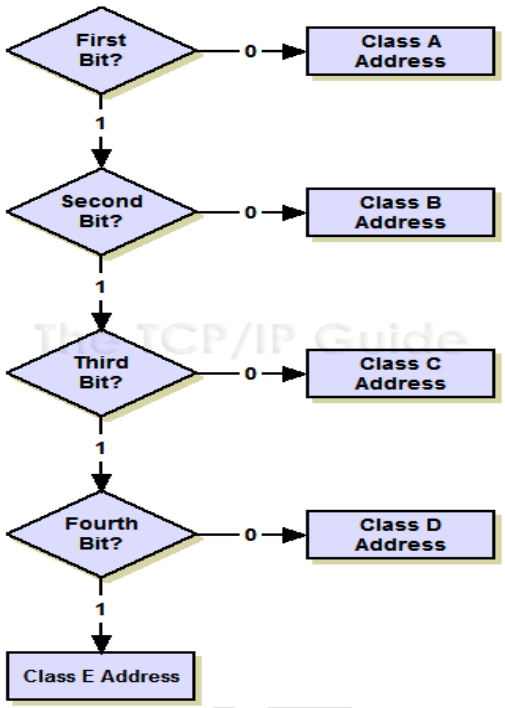
1. **Classful IP Addressing**
2. **Classless IP Addressing**

5.1 CLASSFUL ADDRESSING

When TCP/IP was first created computer technology was still in its infancy, compared to its current state. Routers needed to be able to quickly make decisions about how to move IP

Datagrams around. The IP address space was split into classes in a way that looking at only the first few bits of any IP address would tell the router where to "draw the line" between the network ID and host ID, and thus what to do with the datagram.

The number of bits the router needs to look at may be as few as one or as many as four, depending on what it finds when it starts looking. The algorithm used corresponds to the system used to divide the address space; it involves four very basic steps



1. If the first bit is a “0”, it's a class A address and we're done. (Half the address space has a “0” for the first bit, so this is why class A takes up half the address space.) If it's a “1”, continue to step two.

2. If the second bit is a “0”, it's a class B address and we're done. (Half of the remaining non-class-A addresses, or one quarter of the total.) If it's a “1”, continue to step three.

3. If the third bit is a “0”, it's a class C address and we're done. (Half again of what's left, or one eighth of the total.) If it's a “1”, continue to step four.

4. If the fourth bit is a “0”, it's a class D address. (Half the remainder, or one sixteenth of the address space.) If it's a “1”, it's a class E address. (The other half, one sixteenth.)

Class A Network -- binary address start with 0, therefore the decimal number can be anywhere from 1 to 126. The first 8 bits (the first octet) identify the network and the remaining 24 bits indicate the host within the network. An example of a Class A IP address is 102.168.212.226, where "102" identifies the network and "168.212.226" identifies the host on that network.

Class B Network -- binary addresses start with 10, therefore the decimal number can be anywhere from 128 to 191. (The number 127 is reserved for loopback and is used for internal testing on the local machine.) The first 16 bits (the first two octets) identify the network and the remaining 16 bits indicate the host within the network. An example of a Class B IP address is 168.212.226.204 where "168.212" identifies the network and "226.204" identifies the host on that network.

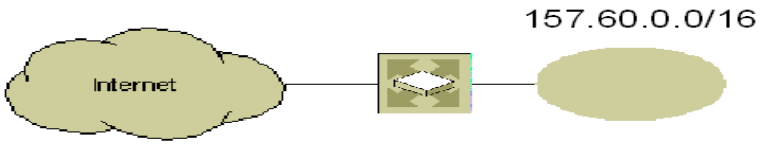
Class C Network -- binary addresses start with 110, therefore the decimal number can be anywhere from 192 to 223. The first 24 bits (the first three octets) identify the network and the remaining 8 bits indicate the host within the network. An example of a Class C IP address is 200.168.212.226 where "200.168.212" identifies the network and "226" identifies the host on that network

Class D Network -- binary addresses start with 1110, therefore the decimal number can be anywhere from 224 to 239. Class D networks are used to support multicasting.

Class E Network -- binary addresses start with 1111, therefore the decimal number can be anywhere from 240 to 255. Class E networks are used for experimentation. They have never been documented or utilized in a standard way

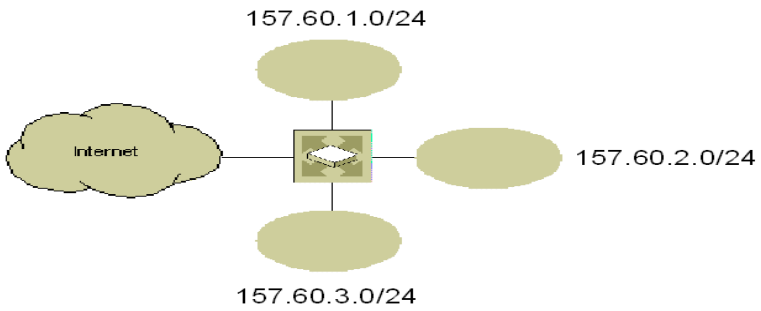
5. SUBNETTING

Subnetting is a set of techniques that you can use to efficiently divide the address space of a unicast address prefix for allocation among the subnets of an organization network. The fixed portion of a unicast address prefix includes the bits up to and including the prefix length that have a defined value. The variable portion of a unicast address prefix includes the bits beyond the prefix length that are set to 0. Subnetting is the use of the variable portion of a unicast address prefix to create address prefixes that are more efficient (that waste fewer possible addresses) for assignment to the subnets of an organization network. Subnetting for IPv4 was originally defined to make better use of the host bits for Class A and Class B IPv4 public address prefixes. Consider the example network



The subnet using the class B address prefix of 157.60.0.0/16 can support up to 65,534 nodes, which is far too many nodes to have on the same subnet. You want to better use the address space of 157.60.0.0/16 through subnetting. However, subnetting 157.60.0.0/16 should not require the reconfiguration of the routers of the Internet.

In a simple example of subnetting, you can subnet 157.60.0.0/16 by using the first 8 host bits (the third octet) for the new Subnetted address prefix. If you subnetted 157.60.0.0/16 as shown in Figure you would create separate subnets with their own subnetted address prefixes (157.60.1.0/24, 157.60.2.0/24, 157.60.3.0/24), with up to 254 host IDs on each subnet. The router would become aware of the separate subnetted address prefixes and route IPv4 packets to the appropriate subnet.



The routers of the Internet would still regard all the nodes on the three subnets as being located on the address prefix 157.60.0.0/16. The Internet routers would be unaware of the subnetting being done to 157.60.0.0/16 and therefore require no reconfiguration.

The subnetting of an address prefix is not visible to the routers outside the network being subnetted. When you assign IPv4 address prefixes in the form of subnet prefixes to the subnets of your organization, you should begin with one or more public address prefixes assigned by the Internet Corporation for Assigned Names and Numbers (ICANN) or an Internet service provider (ISP), the private address space (10.0.0.0/8, 172.16.0.0/12, and 192.168.0.0/16), or both. The set of starting address prefixes represent a fixed address space.

You can divide the variable portion of an IPv4 address prefix to represent additional subnets and the host IDs on each subnet. For example, the IPv4 address prefix 131.107.192.0/18 has 18 fixed bits (as the prefix length shows) and 14 variable bits (the bits in the host ID portion of the address prefix). You might determine that your organization needs up to 50 subnets.

Therefore, you divide the 14 variable bits into 6 bits, which you will use to identify subnets (you can express up to 64 subnets with 6 bits) and 8 bits, which you will use to identify up to 254 host IDs on each subnet. The resulting address prefix for each subnetted address prefix has a 24-bit prefix length (the original 18 bits plus 6 bits used for subnetting). Subnetting for IPv4 produces a set of subnetted address prefixes and their corresponding ranges of valid IPv4 addresses, By assigning subnetted address prefixes that contain an appropriate number of host IDs to the physical and logical subnets of an organization's IPv4 network, network administrators can use the available address space in the most efficient manner possible.

You must determine your organization's current requirements and plan for future requirements. Follow these guidelines:

· Determine how many subnets your network requires. Subnets include physical or logical subnets to which hosts connect and possibly private wide area network (WAN) links between sites.

· Determine how many host IDs each subnet requires. Each host and router interface running IPv4 requires at least one IPv4 address.

Based on those requirements, you will define a set of subnetted address prefixes with a range of valid IPv4 addresses for each subnetted address prefix. Your subnets do not all need to have the same number of hosts; most IPv4 networks include subnets of various sizes. Although the concept of subnetting by using host ID bits is straightforward, the actual mechanics of

subnetting are a bit more complicated. Subnetting requires a three-step procedure:

1. Determine how many host bits to use for the subnetting.

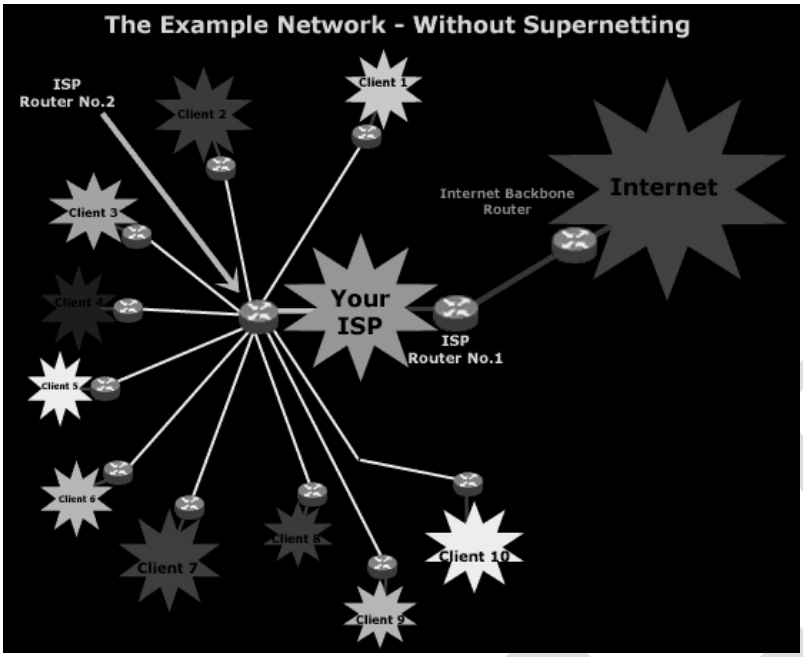
2. Enumerate the new subnetted address prefixes.

3. Enumerate the range of IPv4 addresses for each new subnetted address prefix.

Supernetting

Supernetting, also known as Classless InterDomain Routing (CIDR), is another awesome subject. It exists thanks to the wide adoption of the Internet, which lead to the exhaustion of the available IP Addresses. More specifically, supernetting was invented in 1993 with the purpose of extending the 32 bit IP address lifetime until the adoption of IPv6 was complete. Putting it as simply as possible, supernets are used to combine multiple Class C networks into groups, which the router, in turn, treats as one big network. It might not seem like a smart thing to do, but if you look at the picture on a larger scale you will notice some of the really awesome advantages this offers. The creation of Supernets is also known as *Address Aggregation*.

Consider this realistic example: You work for a large ISP with a few hundred networks to which it provides services like Internet access, e-mail etc. These networks, which basically are your ISP's clients, consist of 254 host IPs each (One full Class C network for each client), and they each have a permanent connection to your headquarters via ISDN (represented by the yellow lines) and from there your ISP has a direct connection to the Internet Backbone.

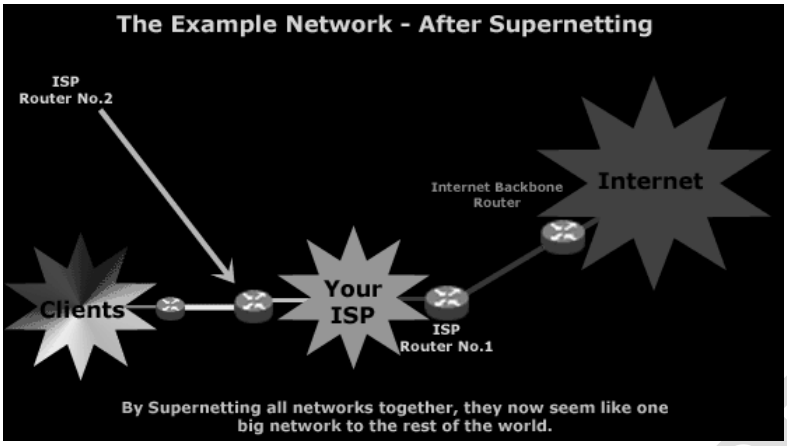


This diagram shows the example network we're talking about. Our main focus is the two routers the ISP has, Router No.1 and Router No.2, because these will be affected when we supernet the networks.

Routers No.1 & No.2 exchange information with each other and update their tables, which contain the networks they know about. Router 2 connects directly to 10 networks and needs to let Router 1 know about each one of them. Router 1 in turn will also advertise these networks to the Internet Backbone Router so it too will know about these networks.

The above setup requires that Router No.1 and the Internet Backbone Router each have more than 13 separate entries in their routing tables to make sure that each network is accessible from them. This is not so bad for this example, but try to imagine the problems and the complexity of a similar setup where you have thousands of networks, where the routing tables would be enormous ! Also, you should keep in mind that the larger the routing table, the more work the router needs to do because it has a huge table of routes to maintain and look through all the time.

By using Supernetting, we could supernet the whole network so it appears to the Internet as follows:



You can clearly see that all the clients' networks have been combined into one big network. Even though Router No.1 and the Internet Backbone router see only one big network, Router No.2 knows all about the smaller Class C networks since it is the one "hiding" them from the rest of the world and makes sure it sends the correct data to each network.

There are some limitations with Supernetting - this is why there is a rule which we must follow so we don't bump into big routing problems and upset the network. We will have a closer look at the rule on the next page.

The reason for evolution

Supernetting has become very popular and there are a lot of reasons why:

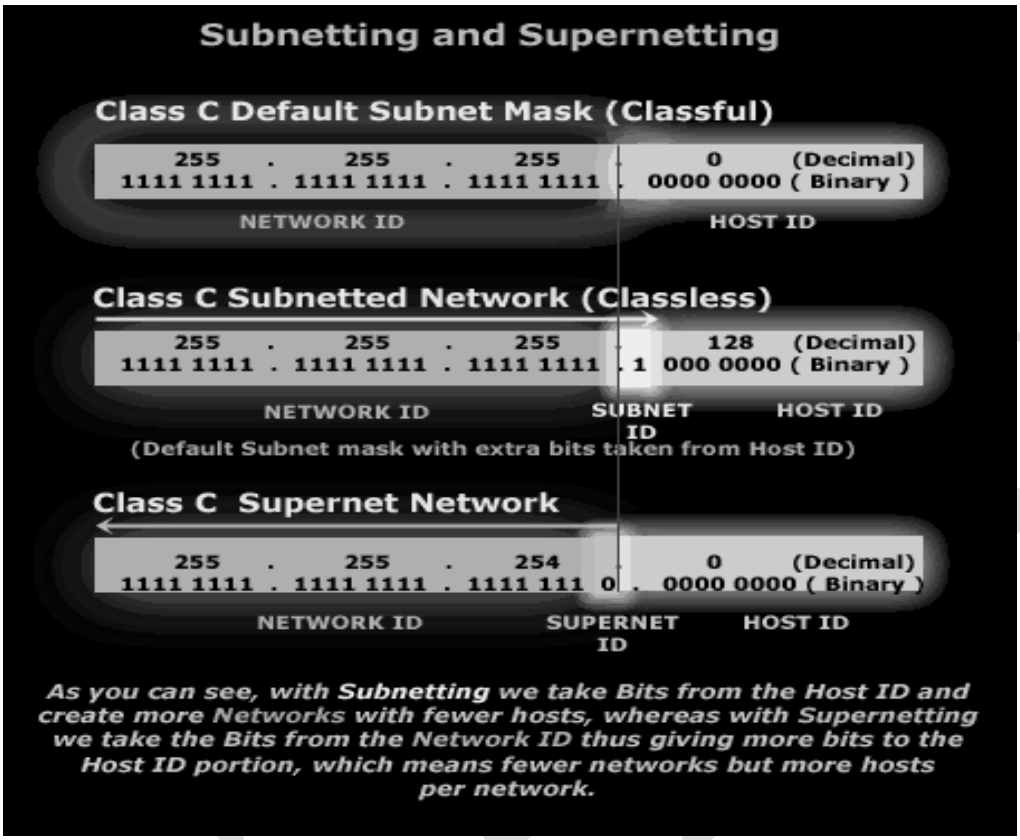
- Class B network address space has nearly been exhausted
- A small percentage of class C network addresses have been assigned to networks
- Routing tables in Internet routers have grown to a size beyond the ability of software and people to effectively manage
- The 32-bit IP address space will eventually be exhausted

How Supernets work

If you understand how Subnetting works, then you will surely understand Supernetting.

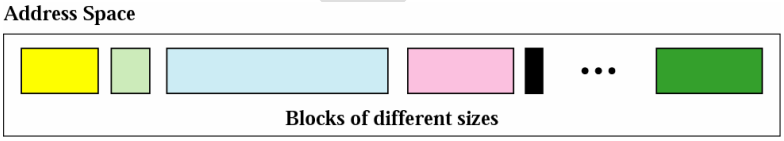
Supernets are the opposite of Subnets in that they combine multiple Class C networks into blocks rather than dividing them into segments.

When Subnetting, we borrow bits from the Host ID portion, which increases the number of bits used for the Network ID portion. With Supernetting we do exactly the opposite, meaning we take the bits from the Network ID portion and give them to the Host ID portion, as illustrated in the picture below:



VARIABLE LENGTH BLOCKS

In classless addressing, IP address space is not divided into classes. So in this case the wastage of address space is eliminated.Variable-length blocks are assigned that belong to no class. In this architecture, the entire address space (232 addresses) is divided into blocks of different sizes.



In this architecture the enire address space is divided into blocks of different sizes. An organization is granted a block suitable for its purpose.

Number of Addresses in a Block

There is only one condition on the number of addresses in a block; it must be a power of 2 (2, 4, 8, . . .). A household may be given a block of 2 addresses. A small business may be given 16 addresses. A large organization may be given 1024 addresses

Beginning Address

The beginning address must be evenly divisible by the number of addresses. For example, if a block contains 4 addresses, the beginning address must be divisible by 4. If the block has less than 256 addresses, we need to check only the rightmost byte. If it has less than 65,536 addresses, we need to check only the two rightmost bytes, and so on.

Problem :

Which of the following can be the beginning address of a block that contains 16 addresses?

205.16.37.32
190.16.42.44
17.17.33.80
123.45.24.52

Solution:

To be divisible by 1024, the rightmost byte of an address should be 0 and the second rightmost byte must be divisible by 4. Only the address 17.17.32.0 meets this condition

Mask:

In classless addressing, the mask for each block is implicit. The mask for class A Block is 255.0.0.0(/8). The mask for Class B is 255.255.0.0(/16), The mask for class C is 255.255.255.0(/24). When an address is given we can find the class of the address. By applying mask we can find the beginning address and the range of address in the block.

CIDR Notation:

Classless Inter Domain Routing (CIDR) is a method for assigning IP addresses without using the standard IP address classes like Class A, Class B or Class C.

In CIDR notation, an IP address is represented as A.B.C.D /n, where "/n" is called the IP prefix or network prefix. The IP prefix identifies the number of significant bits used to identify a network. For example, 192.9.205.22 /18 means, the first 18 bits are used to represent the network and the remaining 14 bits are used to identify hosts. Common prefixes are 8, 16, 24, and 32.

Format of classless addressing address is ***x.y.z.t/n***

/n	Mask	/n	Mask	/n	Mask	/n	Mask
/1	128.0.0.0	/9	255.128.0.0	/17	255.255.128.0	/25	255.255.255.128
/2	192.0.0.0	/10	255.192.0.0	/18	255.255.192.0	/26	255.255.255.192
/3	224.0.0.0	/11	255.224.0.0	/19	255.255.224.0	/27	255.255.255.224
/4	240.0.0.0	/12	255.240.0.0	/20	255.255.240.0	/28	255.255.255.240
/5	248.0.0.0	/13	255.248.0.0	/21	255.255.248.0	/29	255.255.255.248
/6	252.0.0.0	/14	255.252.0.0	/22	255.255.252.0	/30	255.255.255.252
/7	254.0.0.0	/15	255.254.0.0	/23	255.255.254.0	/31	255.255.255.254
/8	255.0.0.0	/16	255.255.0.0	/24	255.255.255.0	/32	255.255.255.255

Finding the First address

What is the first address in the block if one of the addresses is 167.199.170.82/27?

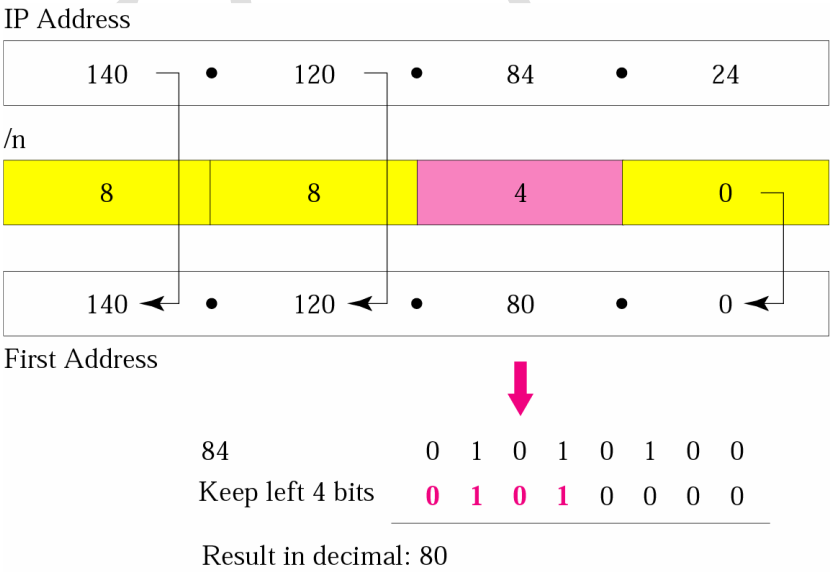
The prefix length is 27, which means that we must keep the first 27 bits as is and change the remaining bits (5) to 0s. The following shows the process:

Address in binary:10100111 11000111 10101010 01010010

Keep the left 27 bits: 10100111 11000111 10101010 01000000
Result in CIDR notation: 167.199.170.64/27

What is the first address in the block if one of the addresses is 140.120.84.24/20?

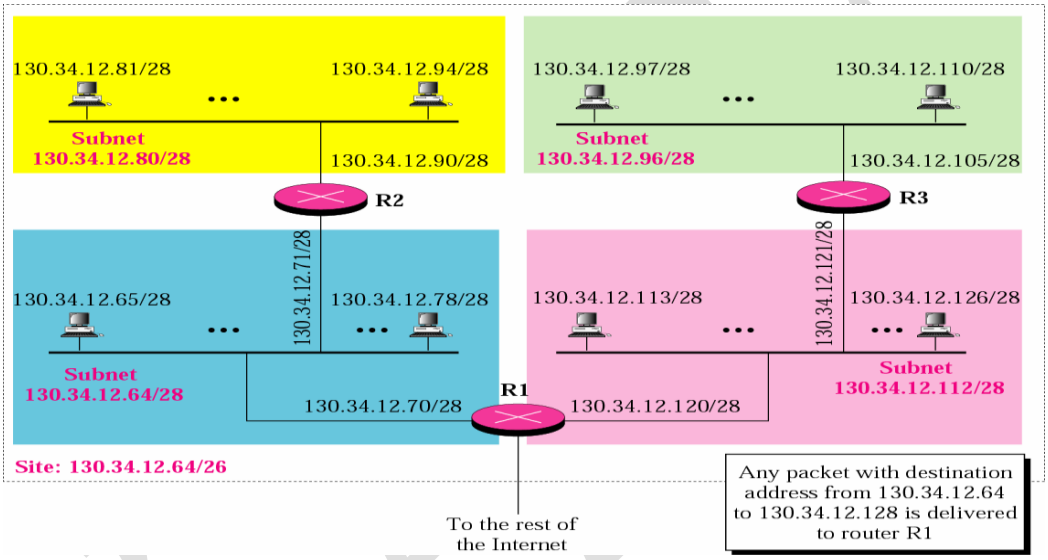
Figure shows the solution. The first, second, and fourth bytes are easy; for the third byte we keep the bits corresponding to the number of 1s in that group. The first address is 140.120.80.0/20.



Subnetting

When an organization is granted a block of addresses, it can create subnets to meet its needs. The prefix length increases to define the subnet prefix length. In fixed-length subnetting, the number of subnets is a power of 2. An organization is granted the block 130.34.12.64/26. The organization needs 4 subnets. What is the subnet prefix length? We need 4 subnets, which means we need to add two more 1s (log2 4 = 2) to the site prefix. The subnet prefix is then /28.

What are the subnet addresses and the range of addresses for each subnet in the previous example?



Problem :

The site has 232−26 = 64 addresses. Each subnet has 232–28 = 16 addresses. Now let us find the first and last address in each subnet.

Solution :

1) The first address in the first subnet is 130.34.12.64/28, using the procedure we showed in the previous examples. Note that the first address of the first subnet is the first address of the block. The last address of the subnet can be found by adding 15 (16 −1) to the first address. The last address is 130.34.12.79/28.
2) .The first address in the second subnet is 130.34.12.80/28; it is found by adding 1 to the last address of the previous subnet. Again adding 15 to the first address, we obtain the last address, 130.34.12.95/28.
3) Similarly, we find the first address of the third subnet to be 130.34.12.96/28 and the last to be 130.34.12.111/28.
4) Similarly, we find the first address of the fourth subnet to be 130.34.12.112/28 and the last to be 130.34.12.127/28.

Variable Length Subnets

An organization is granted a block of addresses with the beginning address 14.24.74.0/24. There are 232−24= 256 addresses in this block. The organization needs to have 11 subnets as shown below:

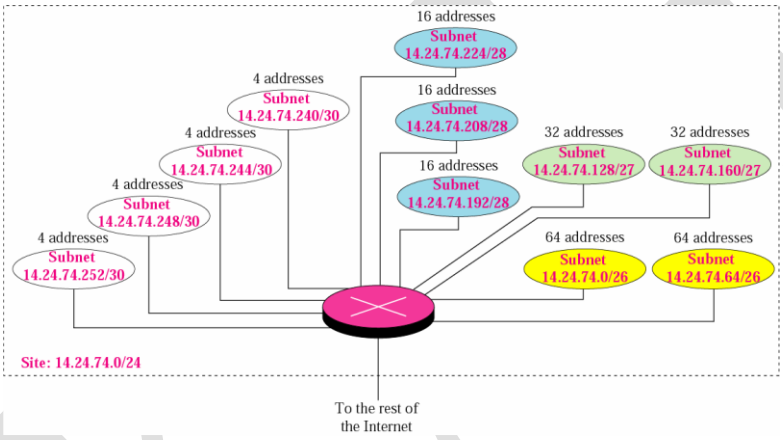
a. two subnets, each with 64 addresses.

b. two subnets, each with 32 addresses.

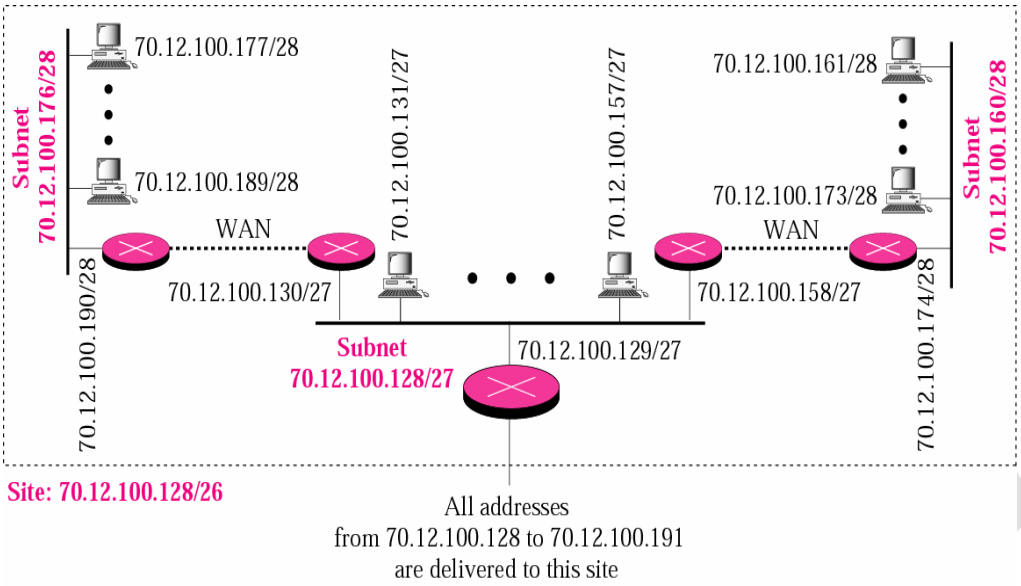
c. three subnets, each with 16 addresses.

d. four subnets, each with 4 addresses.

Design the subnets.



1) We use the first 128 addresses for the first two subnets, each with 64 addresses. Note that the mask for each network is /26. The subnet address for each subnet is given in the figure.
2) We use the next 64 addresses for the next two subnets, each with 32 addresses. Note that the mask for each network is /27. The subnet address for each subnet is given in the figure.
3) We use the next 48 addresses for the next three subnets, each with 16 addresses. Note that the mask for each network is /28. The subnet address for each subnet is given in the figure.
4) We use the last 16 addresses for the last four subnets, each with 4 addresses. Note that the mask for each network is /30. The subnet address for each subnet is given in the figure.
5) As another example, assume a company has three offices: Central, East, and West. The Central office is connected to the East and West offices via private, point-to-point WAN lines. The company is granted a block of 64 addresses with the beginning address 70.12.100.128/26. The management has decided to allocate 32 addresses for the Central office and divides the rest of addresses between the two offices. Figure 5.8 shows the configuration designed by the management.



Address Allocation

Address allocation is the responsibility of a global authority called the Internet Corporation for Assigned Names and Addresses (ICANN). It usually assigns a large block of addresses to an ISP to be distributed to its Internet users.

Example

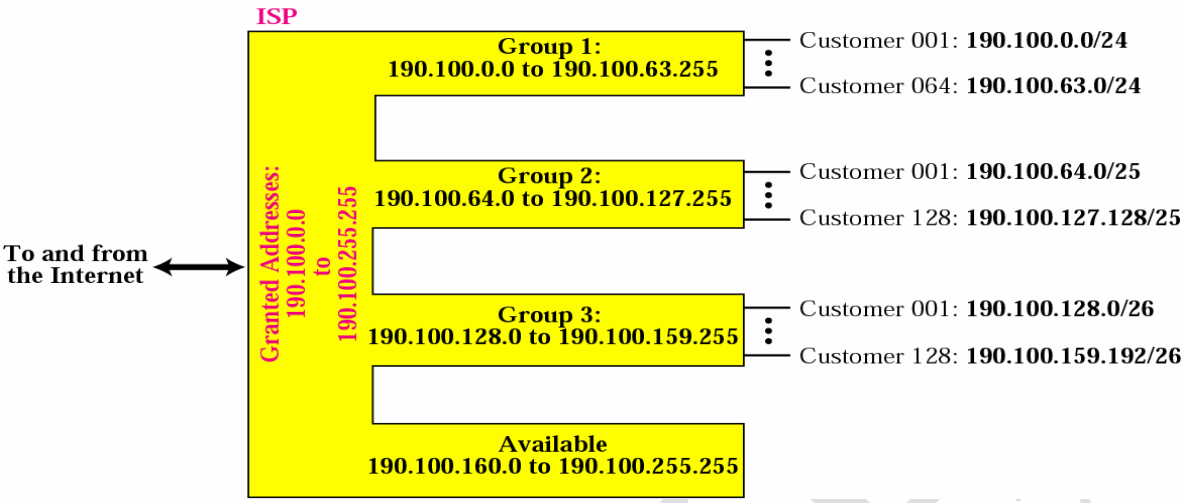
An ISP is granted a block of addresses starting with 190.100.0.0/16 (65,536 addresses). The ISP needs to distribute these addresses to three groups of customers as follows:

The first group has 64 customers; each needs 256 addresses.

The second group has 128 customers; each needs 128 addresses

The third group has 128 customers; each needs 64 addresses.

Design the subblocks and find out how many addresses are still available after these allocations.



Group 1

For this group, ea

ch customer needs 256 addresses. This means the suffix length is 8 (28 =256). The prefix length is then 32 − 8 = 24. The addresses are:

1st Customer	190.100.0.0/24	190.100.0.255/24
2nd Customer	190.100.1.0/24	190.100.1.255/24
. . .		
64th Customer	190.100.63.0/24	190.100.63.255/24
Total = 64 × 256 = 16,384		

Group 2

For this group, each customer needs 128 addresses. This means the suffix length is 7 (27 =128). The prefix length is then 32 − 7 = 25. The addresses are:

1st Customer	190.100.64.0/25	190.100.64.127/25
2nd Customer	190.100.64.128/25	190.100.64.255/25
. . .		
128th Customer	190.100.127.128/25	190.100.127.255/25
Total = 128 × 128 = 16,384		

Group 3

For this group, each customer needs 64 addresses. This means the suffix length is 6 (26 = 64). The prefix length is then 32 − 6 = 26. The addresses are:

1st Customer	**190.100.128.0/26**	**190.100.128.63/26**
2nd Customer	**190.100.128.64/26**	**190.100.128.127/26**
. . .		
128th Customer	**190.100.159.192/26**	**190.100.159.255/26**
Total = 128 × 64 = 8,192		

Number of granted addresses to the ISP: 65,536

Number of allocated addresses by the ISP: 40,960

Number of available addresses: 24,576

6. ROUTING

Routing refers to the way routing tables are created to help the forwarding. Two types of routing are Statci and dynamic. Delivery refers to the way a packet is handled by the underlying networks under the control of network layer. Forwarding refers to the way the packet is delivered to the next station or stations.

DELIVERY

The network layer supervises delivery, the handling of the packets by the underlying physical networks. Two important concepts are the type of connection and direct versus indirect delivery

Connection-Oriented Services

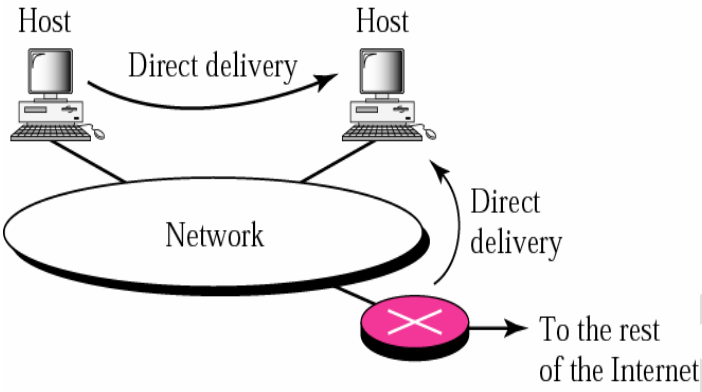
- The network layer establishes a connection between a source and a destination
- Packets are sent along the connection.
- The decision about the route is made *once* at connection establishment
- Routers/switches in connection-oriented networks are stateful

Connectionless Services

- The network layer treats each packet independently
- Route lookup for each packet (routing table)
- IP is connectionless
- IP routers are stateless

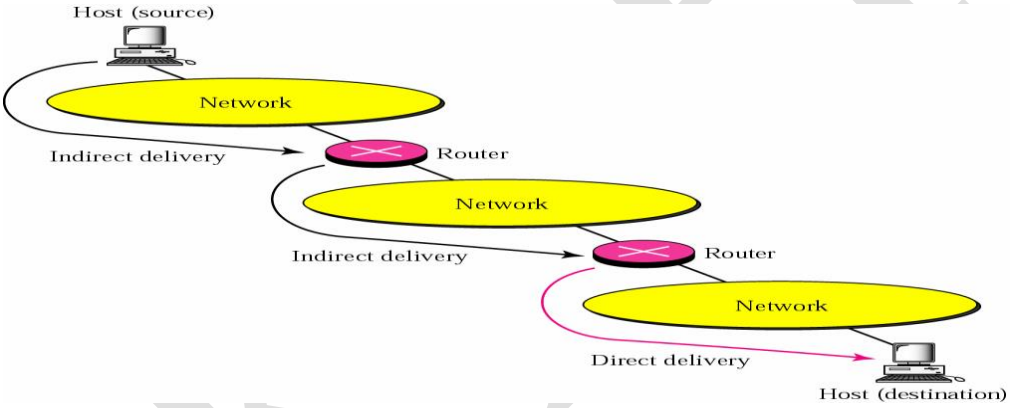
Direct delivery

The final destination is connected to the same pysical network as the sender. IP destination address and local interface has same netmask. Map IP address to physical address: ARP



Indirect delivery

If the destination host is not in the same network as the deliverer, the packet is delivered indirectly. In an indirect delivery, the packet goes from router to router until it reaches the destination.



FORWARDING

Forwarding means to place the packet in its route to its destination. Forwarding requires a host or a router to have a routing table. .

Forwarding Techniques

Several techniques can make the size of the routing table manageable and also handle issues such as security. Security is the main concern for routing table. Because the intruder can easily find the destination address of packet and gather complete information about the IP Packet.

1) Next Hop Method.

Routing table for host A

Destination	Route
Host B	R1, R2, Host B

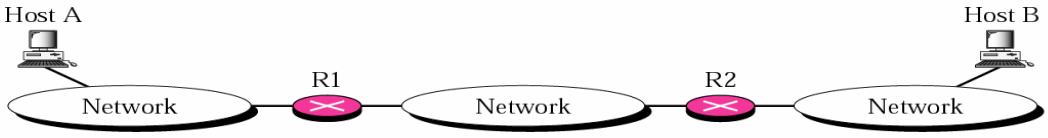
Routing table for R1

Destination	Route
Host B	R2, Host B

Routing table for R2

Destination	Route
Host B	Host B

a. Routing tables based on route



Routing table for host A

Destination	Next Hop
Host B	R1

Routing table for R1

Destination	Next Hop
Host B	R2

Routing table for R2

Destination	Next Hop
Host B	Ñ

b. Routing tables based on next hop

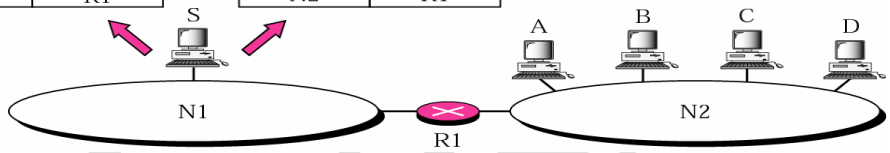
2) Network Specific Method

Routing table for host S based on host-specific method

Destination	Next Hop
A	R1
B	R1
C	R1
D	R1

Routing table for host S based on network-specific method

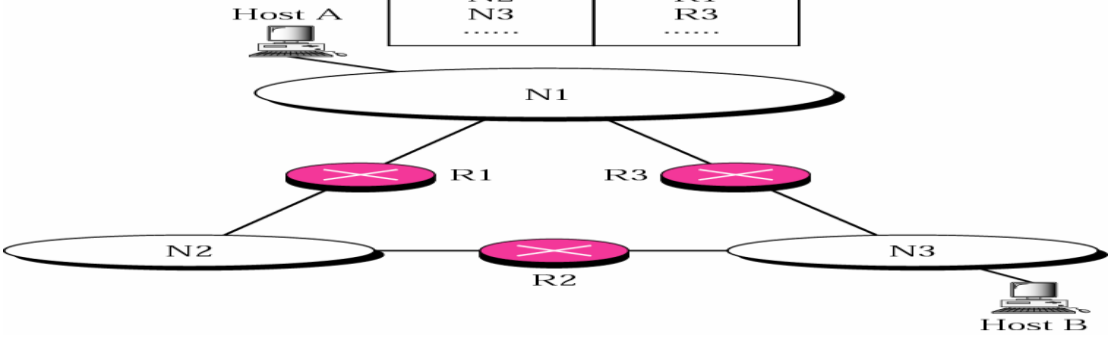
Destination	Next Hop
N2	R1



3) Host Specific Method

Routing table for host A

Destination	Next Hop
Host B	R3
N2	R1
N3	R3
.....	.....



6. ROUTING

Routing is the act of moving information across an internetwork from a source to a destination. Along the way, at least one intermediate node typically is encountered. Routing is often contrasted with bridging, which might seem to accomplish precisely the same thing to the casual observer. The primary difference between the two is that bridging occurs at Layer 2 (the link layer) of the OSI reference model, whereas routing occurs at Layer 3 (the network layer). This distinction provides routing and bridging with different information to use in the process of moving information from source to destination, so the two functions accomplish their tasks in different ways. Routing deals with the issues of creating and maintaining routing tables.

Static Vs Dynamic Routing Tables

A static routing table contains information entered anually. A dynamic routing table is updated periodically using one of the dynamic routing protocols such as RIP, OSPF, or BGP. In classful addressing, each address has self-contained information that facilitates routing table searching. In classless addressing, there is no self-contained information
in the destination address to facilitate routing table searching**.**

Routing Table

A routing table for classless addressing has a minimum of four fields. But depending on the situation and vendors there may be be change. The common fields in modern routing table has fields like shown below.

Mask	Network address	Next-hop address	Interface	Flags	Reference count	Use
.....	.....	.....	.....	.....	.....	.....

Mask : It defines the mask applied for the entry.

Network Address : It defines the address of the network to which the packet is finally delivered.

Next hop address : It defines address of next hop router to which the packet is delivered.

Interface: It gives the name of interface.

Flags :

U: It denotes that router is up and running.

G: It denotes that the destination is in another network

H : It denotes that the network address present is host specific.

D : It denotes that the routing information for this destination has been added to the host routing table by the redirection and the ICMP Protocol.

M : It denotes that the routing information for this destination has been modified by a redirection message from ICMP Protocol.

Reference count : It denotes number of users using the route at that time.

Use : It shows number of packets transmitted through this router for the corresponding destination.

STRUCTURE OF A ROUTER:

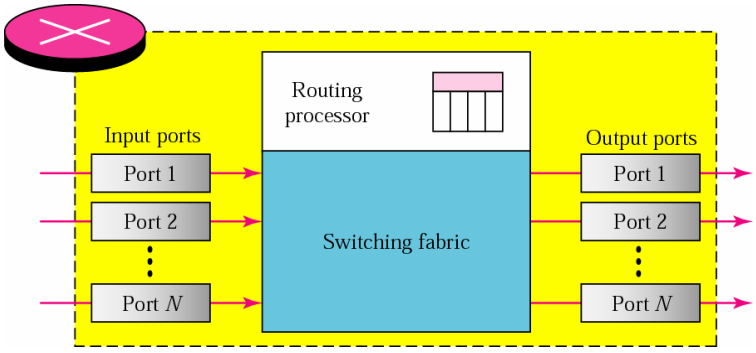
Router is a hardware device that accepts incoming packets from one of the input ports(interfaces), uses a routing table to find the output port from which the packets departs and sends the packet from this output port. Two key router functions:

❒ run routing algorithms/protocol (RIP, OSPF, BGP)

❒ switching datagrams from incoming to outgoing link

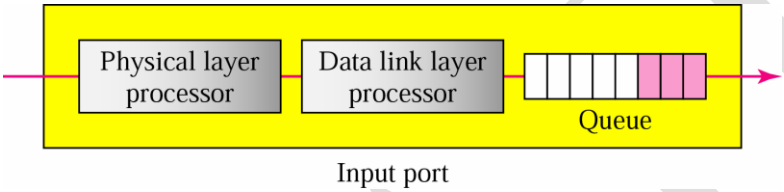
Components of a Router:

Router has four components such as Input Ports, Output Ports, Routing Processor, Switching fabric.



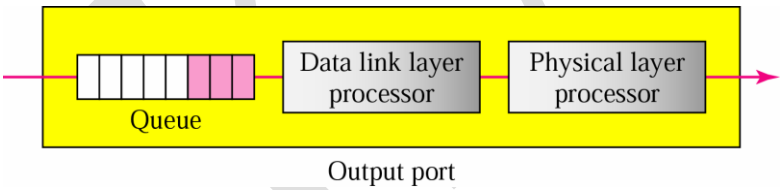
Input Ports:

An input port performs the physical and data link layer functions of the router. The bits are constructed from the received end. The packet is decapsulated from the frame. Errors are detected and corrected. The packet is ready to be forwarding by the network layer**.**



Output Ports:

Output port performs the same function as input port but in reverse order. Buffering required when datagrams arrive from fabric faster than the transmission rate Scheduling discipline chooses among queued datagrams for transmission buffering when arrival rate via switch exceeds ouput line speed queueing (delay) and loss due to output port buffer overflow!

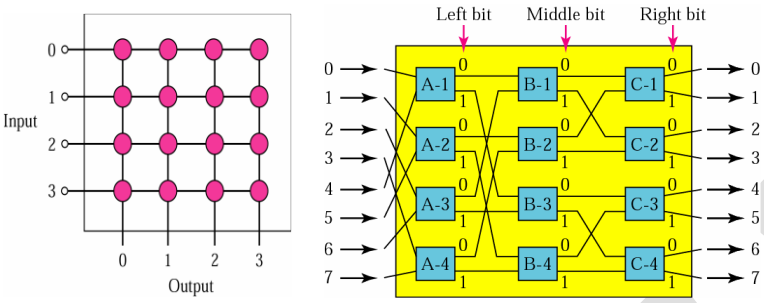


Router Processor

The router processor performs the functions of the network layer. The destination address is used to find the address of the next hop and at the same time the output port number from which the packet is send out.. This process is some times known as Table lookup.

Switching Fabrics:

The most difficult task in a router is to move packets from the input queue to the output queue. The spped affects the size of input /output queue.So routers are specialized mechanisms that use a variety of switching fabrics. Such as Crossbar switch, banayan switch etc.



a) Crossbar Switch b) Banyan Switch

7. POINTS TO REMEMBER

- Internet is composed of hundreds of thousands of interconnected networks.
- ARPA Stands for Advanced Research Project Agency
- Two types of ARPANET are ARPANET & MILNET
- In ARPANET each host is attached to a specialized computer called IMP
- TCP is responsible for higher level function such as segmentation, reassembling and error detection
- IP handles Datagram Routing
- Protocols is a set of rule that governs data communication
- Key elements of Protocols are Syntax & Semantic & timing
- Defacto is a Standard that have not been approved by the Organized body
- OSI Stands for Open System Interconnection The Process on each machine that communicate at a given layer is Point – to - Point
- Transport Layer is responsible for delivery of a message from one process to another
- Application Layer provides services to the user
- An IP Address is a 4 byte address

- In Binary notation one or more spaces is inserted between each octet.
- Each Octet is referred to as a Byte
- In Class full addressing IP address is divided into 5 Classes
- If the first two its are zero, then the IP address is of A
- The range of Class D address is 224 to 239
- In Transparent Bridge the station are unaware of the existence of bridge .

Possible Questions

Part B
(Each Question carries 6 Marks)

1. Explain TCP/IP protocol suite.
2. Give a brief note on subnetting in classful addressing.
3. Discuss in detail addressing modes with suitable examples.
4. Give a brief note on inter - networking devices in detail.
5. Explain about any two connecting devices.
6. Write about classless addressing with example
7. Write about classes and blocks in Class A, B, C, D and E
8. Write the procedures to find first address and last address in a block in classless addressing
9. Discuss the importance of mask with an example in classful addressing
10. Explain components of a router with neat diagram

Part C
(Each Question carries 10 Marks)

1. An organization is granted a block 130.34.12.64/26. The organization needs 4 subnets each with an equal number of hosts. Design the subnet and find information about each network

Karpagam Academy of Higher Education
Coimbatore - 21

DEPARTMENT OF COMPUTER APPLICATIONS
CLASS : II MCA

TCP/IP (18CAP405N)

Unit I

S.NO	Question	Choice1	Choice2	Choice3	Choice4	Choice 5	Choice 6	Answer
1	_____ is composed of hundreds of thousands of interconnected networks.	Internet	Intranet	Extranet	Arpanet			Internet
2	ARPA Stands for _____	Advanced Research Protocol Agency	Automated Research Provider Agency	Advanced Research Project Agency	Advanced Research Program			Advanced Research Project Agency
3	Two types of ARPANET are _____	NSFNET & CSNET	ANSNET & MILNET	INTERNET & INTRANET	ARPANET & MILNET			ARPANET & MILNET
4	In ARPANET each host is attached to a specialized computer called _____	MIP	IMP	PIM	IGMP			IMP
5	___ is responsible for higher level function such as error detection	IP	TCP/IP	NCP	TCP			TCP
6	IP handles _____	Segmentation	Error Detection	Datagram Routing	Reassembly			Datagram Routing
7	_____ is a set of rule that governs data communication	Standards	Protocols	Organization	Routing			Protocols
8	Key elements of Protocols are _____	Defacto & Dejure	Requirement & Packet size	Interface & Timing & Packets	Syntax & Semantic & timing.			Syntax & Semantic & timing.
9	_____ Standard that have not been approved by the Organized body	Dejure	Defacto	Dejery	Decimal			Defacto
10	_____ is the Organization and _____ is the Model	ISO , OSI	OSI , ISI	ISA , OSI	ISA , ISI			ISO , OSI
11	OSI Stands for _____	Open System Interconnection	Open Standard Interconnection	Organizational Standard interface	Open Source interconnection			Open System Interconnection
12	The Process on each machine that communicate at a given layer is _____	Interface	Point – to - Point	Routing	Peer to Peer			Point – to - Point

13	_____ Layer is responsible for delivery of a message from one process to another	Transport layer	Data Link layer	Physical layer	Network layer			Transport layer
14	_____ provides services to the user	Application Layer	Transport Layer	Session Layer	Presentation Layer			Application Layer
15	In TCP/IP application layer is the combination of ____,_____ and ___	Application , Network, Data Link	Application , Network, Physical	Application , Network, Session	Application , Session , Presentation			Application , Session , Presentation
16	Which of the following is not a connection device _____	Hub	Router	Amplifier	Bridge			Amplifier
17	Repeater is a _____ not a _____	Amplifier, Regenerator	Regenerator , Amplifier	Connector, segmented	Hub, Bridge			Regenerator , Amplifier
18	_____ is a multi port repeater	Bridge	Router	Hub	Amplifier			Hub
19	_____ has filtering capacity	Router	Hub	Bridge	Generator			Bridge
20	Router is a _____ device.	One layer	Two layer	Tree layer	Four Layer.			Tree layer
21	An IP Address is a _____address	4 byte	8 byte	34byte	1 byte			4 byte
22	In_____ notation one or more spaces is inserted between each octet.	Hexadecimal	Binary	ASCII	Decimal			Binary
23	Each Octet is referred to as a _____	Bit	Byte	Word	Pixel			Byte
24	In Class full addressing IP address is divided into _____ Classes	3	5	4	6			5
25	if the first two its are zero, then the IP address is of _____ Class	A	B	C	D			A
26	The range of Class D address is _____	223 to 240	224 to 239	225 to 237	221 to 234			224 to 239
27	In _____ the station are unaware of the existence of bridge	Transformer	Transparent Bridge	Bridge	Router			Transparent Bridge
28	ARP stands for_____	Address Reverse Protocol	Address Resolution Protocol	Advanced Research Project	Advanced Resolution Protocol			Address Resolution Protocol
29	The protocol used to associate an IP address with physical address is ___	ARP	RARP	PROXY ARP	ICMP			ARP

30	_____is an internet work address	physical address	Logical address	IP address	Network address.			Logical address
31	The logical address in the TCP/IP protocol suit are called_____	logical address	Network address	IP address	Physical address			IP address
32	In _____each time a machine knows one of the 2 addresses.	Dynamic mapping	Static mapping	Temporary mapping	Fully Mapping			Dynamic mapping
33	RARP Stand for_____	Resolution Address Reverse Protocol	Routing Address Resolution Protocol	Routing Address Reverse protocol	Reverse Address Resolution			Reverse Address Resolution Protocol
34	____ allows a host to discover its internet address when it knows only its physical address.	ARP	PROXY ARP	RARP	ICMP			RARP
35	_____ means creating a table that associates a logical address with the physical address.	Dynamic mapping	Static mapping	Temporary mapping	Physical mapping			Static mapping
36	__ is a 16-bit field defining the type of the network on which ARP is running.	Protocol type	Network type	Software type	Hardware type.			Hardware type.
37	_____is a m16-bit field defining the protocol.	Hardware type	Software type	Protocol type	Network type.			Protocol type
38	An ARP request is_____	Unicast	Broadcast	Telecast	Supercast			Broadcast
39	An ARP reply is_____	Broadcast	Telecast	Unicast	Specialcast			Unicast
40	_____is used to create a subnetting effect.	PROXY ARP	ARP	RARP	ICMP			PROXY ARP
41	An ARP package involves_____components.	2	8	5	7			5
42	_____ Waits until an ARP Packet arrives.	Output module	Input module	Control module	speed module			Input module
43	Packets in the IP Layer are called_____	Components	Interface	Tokens	Datagram.			Datagram.
44	The process of dividing the datagram to pass through the tworks_____	Segmentation	Fragmentation	Splitting	Encapsulation			Fragmentation
45	The two-bit subfield defines the general purpose of the option is _____	Copy	Class	Number	object			Class
46	The error detection method used by most TCP/IP protocol is called the _____	checksum	parity	checkerror	Debugg			checksum

47	The maximum length of the diagram is _____ bytes	65,534	65,535	66,334	65,432			65535
48	A datagram consist of _____ and _____	Title and data	Header and information	Content and header	Header and data			Header and data
49	ICMP stands for _____	Internet Control Message Protocol	Intranet Control Message Protocol	Internet Content Message Protocol	Intranet Content Message Protocol			Internet Control Message Protocol
50	ICMP is a _____ layer protocol	Application	Transport	Network	Physical			Network
51	ICMP messages are divided into _____ categories.	5	2	3	4			2
52	One of the main responsibilities of ICMP is to _____ errors.	Detect	Correct	Check	Report			Report
53	An ICMP message has _____ byte header.	8	6	4	2			8
54	There is no _____ mechanism in the IP protocol.	Data Control	Source Control	Flow Control	Error Control			Flow Control
55	A _____ message is send from router to a host on the same local network.	Redirection	Bidirection	Unidirection	Multidirection			Redirection
56	_____ program is used to find if a host is alive and responding.	Traceroute	Ping	Tracert	Trace			Ping
57	A parameter problem message can be created by a _____	Router	hub	Bridge	Switch			Router
58	_____table is used by the reassembly module.	Fragmentation table	Bridge table	Reassembly table	Routing table			Reassembly table
59	A _____ option is used to record the time of datagram processing.	Loose source Route	Time Stamp	Time Slicing	check sum			Time Stamp
60	An end of option is a _____ byte option.	4	6	8	1			1

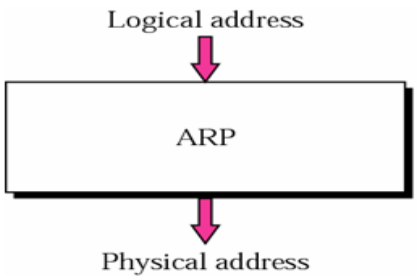
UNIT-II

SYLLABUS

ARP & RARP – Proxy ARP – ARP over ATM – ARP and RARP Protocol Format. IP Datagram – Fragmentation– Options – IP Datagram Format – Routing IP Datagrams – Checksum. ICMP : Types of Messages - Message Format – Error Reporting – Query – Checksum - ICMP Package

1. ARP

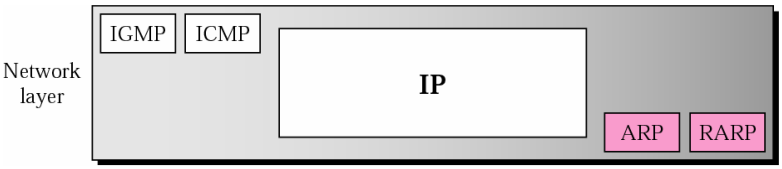
The address resolution protocol (arp) is a protocol used by the Internet Protocol (IP) [RFC826], specifically IPv4, to map IP network addresses to the hardware addresses used by a data link protocol. The protocol operates below the network layer as a part of the interface between the OSI network and OSI link layer. It is used when IPv4 is used over Ethernet.



The term address resolution refers to the process of finding an address of a computer in a network. The address is "resolved" using a protocol in which a piece of information is sent by a client process executing on the local computer to a server process executing on a remote computer. The information received by the server allows the server to uniquely identify the network system for which the address was required and therefore to provide the required address. The address resolution procedure is completed when the client receives a response from the server containing the required address.

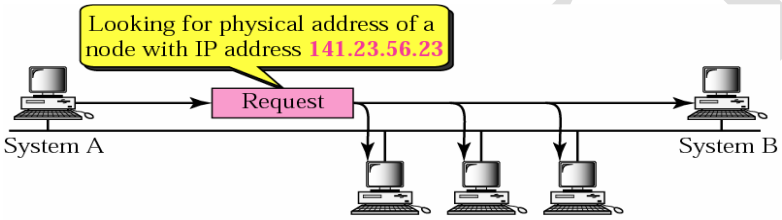
Position of ARP in TCP/IP Protocol Suite

ARP associates an IP address with its physical address. On a typical physical network, such as a LAN, each device on a link is identified by a physical or station address that is usually imprinted on the NIC.

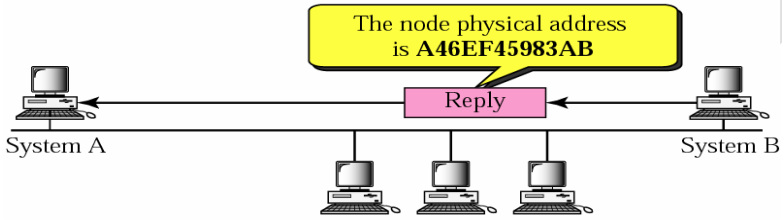


1.1 ARP OPERATION

The ARP request packets are broadcast; the RARP reply packets are unicast.



a. ARP request is broadcast



b. ARP reply is unicast

1.2 ARP PACKET FORMAT

The Packet format for ARP is given below.

Hardware Type		Protocol Type
Hardware length	Protocol length	Operation Request 1, Reply 2
Sender hardware address (For example, 6 bytes for Ethernet)		
Sender protocol address (For example, 4 bytes for IP)		
Target hardware address (For example, 6 bytes for Ethernet) (It is not filled in a request)		
Target protocol address (For example, 4 bytes for IP)		

The fields are as follows

Hardware type (HTYPE): This field specifies the Link Layer protocol type. Example: Ethernet is 1.

Protocol type (PTYPE): This field specifies the upper layer protocol for which the ARP request is intended. For example, Internet Protocol (IPv4) is encoded as 0x0800.

Hardware length (HLEN): Length (in octets) of a hardware address. Ethernet addresses size is 6.

Protocol length (PLEN): Length (in octets) of a logical address of the specified protocol (cf. PTYPE). IPv4 address size is 4.

Operation: Specifies the operation that the sender is performing: 1 for request, 2 for reply.

Sender hardware address (SHA): Hardware (MAC) address of the sender.

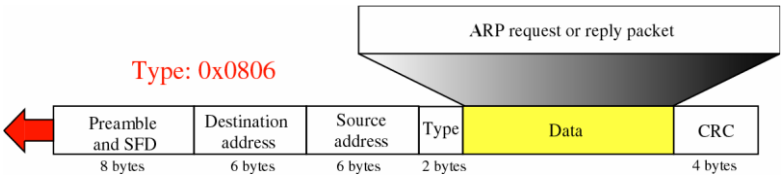
Sender protocols address (SPA): Upper layer protocol address of the sender.

Target hardware address (THA) : Hardware address of the intended receiver. This field is ignored in requests.

Target protocol address (TPA): Upper layer protocol address of the intended receiver.

1.3 ENCAPSULATION

ARP packet is encapsulated directly into a data link frame ARP packet encapsulated in an Ethernet frame

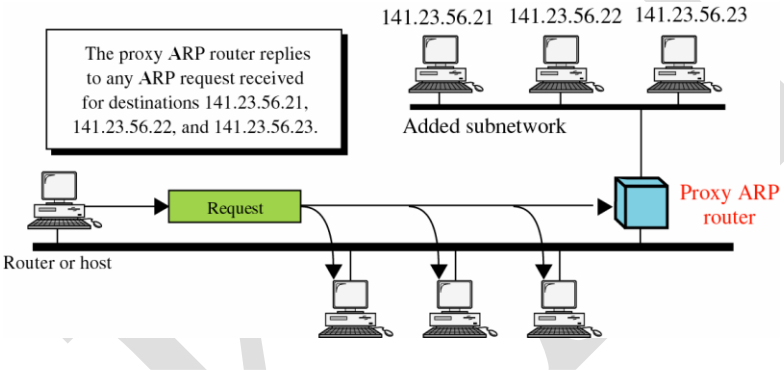


1.4 ARP OVER ATM

ARP is also used when an IP Packet wants to pass over an ATM Network

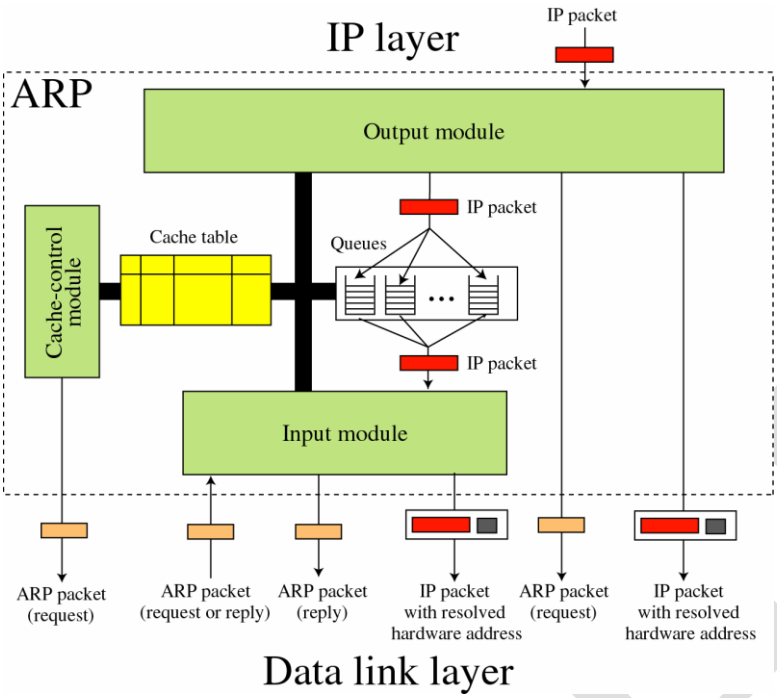
PROXY ARP:

ARP that acts on behalf of a set of hosts. Whenever the router running a proxy ARP receives an ARP request looking for the IP address of one of these hosts, router sends an ARP reply announcing its own hardware (physical) address later, when the router receives the actual IP packet, it will send the packet to the appropriate host or router



1.5 ARP PACKAGE

ARP Package includes five components namely Input module, Output module, Cache table, Query and cache control module.



1. Cache table:

When a host or router receives the corresponding physical address for an IP datagram, the address can be saved in the cache table. This address can be used for the datagram's destined for the same receiver within the next few minutes

2. Input Module:

Waiting until an ARP packet (request or reply) arrives. It role is checking the cache table to find an entry corresponding to this ARP packet

Input Module

1. **Sleep until an ARP packet (request or reply) arrives**
2. **Check the cache to find an entry corresponding to the this ARP packet**
3. **If (found)**
    1. **If (the state is PENDING)**
        1. **Update the entry**
        2. **While the queue is not empty**
            1. **Dequeue one packet**
            2. **Send the packet and the hardware address to date link**
    2. **If (the state is RESOLVED)**
        1. **Update the entry**
4. **If (not found)**
    1. **Create an entry**
    2. **Add the entry to the table**

5. **If (the packet is a request)**
    1. **Send an ARP reply**
6. **Return**

3. **Output Module:**

Waiting for an IP packet from the IP software. Its main role is checking the cache table to find an entry corresponding to the destination IP address of this packet

Output module

1. **Sleep until an IP packet is received from IP software**
2. **Check the cache table to find an entry corresponding to this IP packet**
3. **If (found)**
    1. **If (the state js Resolved)**
        1. **Extract the value of the hardware address from the entry**
        2. **Send the packet and the hardware address to data link layer**
        3. **Return**
    2. **If (the state is PENDING)**
        1. **Enqueue the packet to the corresponding queue**
        2. **Return**
4. **If (not found)**
    1. **Create a queue**
    2. **Enqueue the packet**
    3. **Create a cache entry with state set to PENDING and ATTEMPS set to 1**
    4. **Send an ARP request**

Return

4. Queues:

Holding the IP address while ARP tries to resolve the hardware address

5. Cache-control module

This nodule is responsible for maintaining the cache table. It periodically checking the cache table, entry by entry

Cache-control module

1. **Sleep until the periodic timer matures.**
2. **For every entry in the cache table**
    1. **If(the state is Free)**
        1. **Continue.**
    2. **If(the state is PENDING)**
        1. **Increment the value of attempts by 1.**

2. If(attempts greater than maximum)
    1. Change the state to FREE
    2. Destroy the corresponding queue.
3. 3. If(not)
    1. Send and ARP request.
4. Continue.

3. If(the state is RESOLVED)
    1. Decrement the value of time-out by the value of elapsed time.
    2. If(time-out less than or equal to zero)
        1. Change the state to FREE.
        2. Destroy the corresponding queue

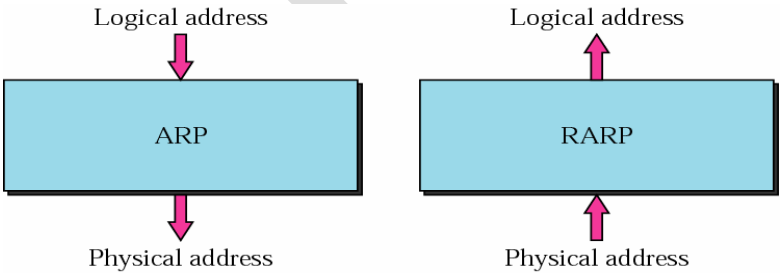
3. Return.

Example:

State	*Queue*	*Attempt*	*Time-out*	*Protocol Addr.*	*Hardware Addr.*
R	5		900	180.3.6.1	ACAE32457342
P	2	2		129.34.4.8	
P	14	5		201.11.56.7	
R	8		450	114.5.7.89	457342ACAE32
P	12	1		220.55.5.7	
P	23	1		116.1.7.22	
R	9		60	19.1.7.82	4573E3242ACA
R	**18**		**900**	**188.11.8.71**	**E34573242ACA**

2.0 RARP

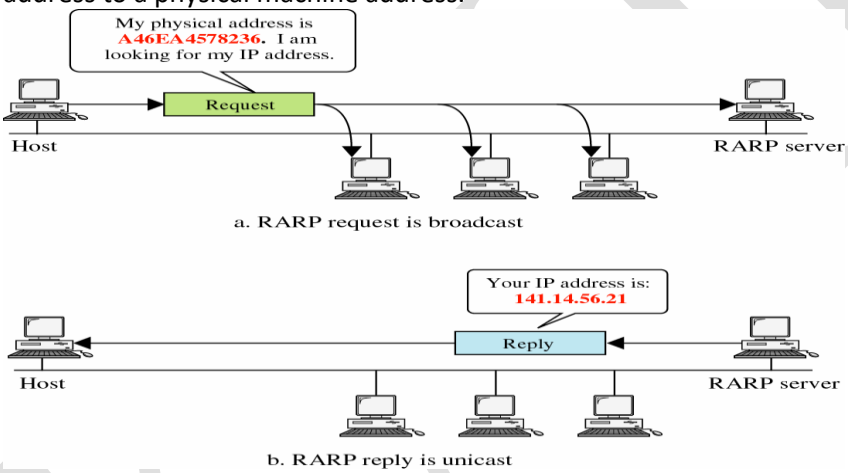
RARP (Reverse Address Resolution Protocol) allows a physical machine in a local area network to request its IP address from a gateway server's Address Resolution Protocol (ARP) table or cache.



To create an IP datagram a host or a router needs to know its own IP address or addresses. The IP address of a machine is usually read from its configuration file stored on a disk file

Advantages of RARP

1) RARP (Reverse Address Resolution Protocol) allows a physical machine in a local area network to request its IP address from a gateway server's Address Resolution Protocol (ARP) cache or table.
2) A network administrator creates a table in a local area network's gateway router that maps the physical machine (or Media Access Control - MAC address) addresses to corresponding to the Internet Protocol addresses (IP address).
3) When a new machine is set up, its RARP client program requests from RARP server on the router to be sent its IP address.
4) Assuming that an entry has been set up in the router table and the RARP server will return the IP address to the machine which can store it for future use.
5) RARP is available for Fiber Distributed-Data Interface, Ethernet, and Token Ring LANs and ARP (Address Resolution Protocol) performs the opposite function as the RARP: mapping of an IP address to a physical machine address.



2.1 PACKET FORMAT

RARP functionality supports multiple physical network types. All machines receive RARP requests but only those authorized to supply RARP services can respond. Machines supplying RARP services are called RARP servers.. RARP is only used on LANs with a low probability of failure since bootstrapping requires quick responses.

Hardware type		Protocol type
Hardware length	Protocol length	Operation Request 3, Reply 4
Sender hardware address (For example, 6 bytes for Ethernet)		
Sender protocol address (For example, 4 bytes for IP) (It is not filled for request)		
Target hardware address (For example, 6 bytes for Ethernet) (It is not filled for request)		
Target protocol address (For example, 4 bytes for IP) (It is not filled for request)		

Hardware type - which specifies hardware interface type for which the sender requires a response.

Protocol type -which specifies the type of the high-level protocol address the sender has supplied.

Hlen - Hardware address length.

Plen - Protocol address length.

Sender hardware address -HLen bytes in length.

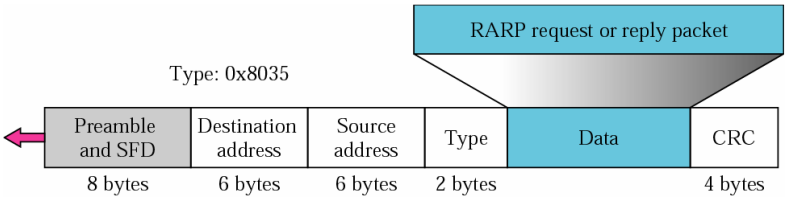
Sender protocol address - PLen bytes in length.

Target hardware address - HLen bytes in length.

Target protocol address - PLen bytes in length.

2.2 ENCAPSULATION

A RARP packet is encapsulated directly in to a data link frame. For example, in above fig. shows a RARP packet encapsulated in an Ethernet frame. Note that the type of fields shows that the data carried by a frame is a RARP packet.



2.3 RARP SERVER:

All the mappings between the hardware MAC addresses and the IP addresses of the hosts are stored in a configuration file in a host in the network. This host is called the RARP server. This host responds to all the RARP requests. The mapping between MAC addresses and IP addresses is usually stored in a configuration file in the local hard disk in the RARP server. When a RARP server receives a RARP request packet it performs the following steps:

- The MAC address in the request packet is looked up in the configuration file and mapped to the corresponding IP address.
- If the mapping is not found, the packet is discarded.
- If the mapping is found, a RARP reply packet is generated with the MAC and IP address. This packet is sent to the host, which originated the RARP request.

Alternative Solutions to RARP

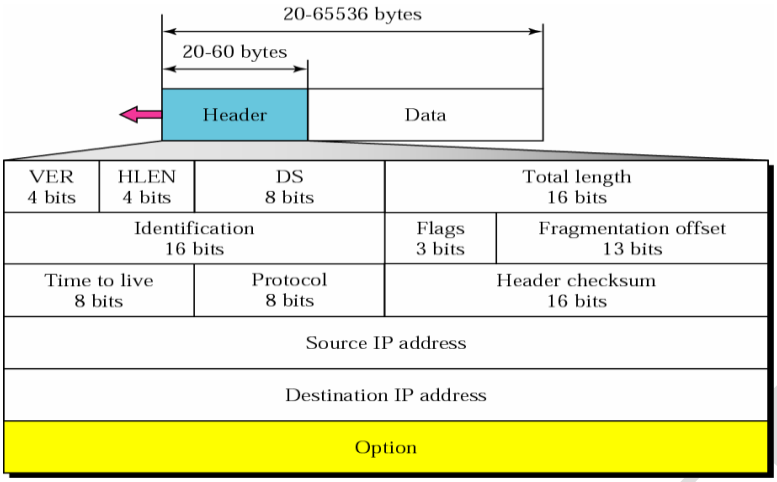
When a diskless computer is booted, it needs more information in addition to its IP address. It needs to know its subnet mask, the IP address of a router, and the IP address of a name server. RARP cannot provide this extra information. New protocols have been developed to provide this information. The two protocols, BOOTP and DHCP, that can be used instead of RARP.

3 INTERNET PROTOCOL

The Internet Protocol (IP) is a network-layer (Layer 3) protocol that contains addressing information and some control information that enables packets to be routed. IP is documented in RFC 791 and is the primary network-layer protocol in the Internet protocol suite. Along with the Transmission Control Protocol (TCP), IP represents the heart of the Internet protocols. IP has two primary responsibilities: providing connectionless, best-effort delivery of datagram's through an internetwork; and providing fragmentation and reassembly of datagrams to support data links with different maximum-transmission unit (MTU) size

3.1 IP DATAGRAM

A packet in the IP layer is called a datagram, a variable-length packet consisting of two parts: header and data. The header is 20 to 60 bytes in length and contains information essential to routing and delivery.



The following are the fields in IP Datagram

Version— Indicates the version of IP currently used.

***IP Header Length (*IHL)**—indicates the datagram header length in 32-bit words.

Type-of-Service—Specifies how an upper-layer protocol would like a current datagram to be handled, and assigns datagrams various levels of importance.

Identification—Contains an integer that identifies the current datagram. This field is used to help piece together datagram fragments.

Flags—Consists of a 3-bit field of which the two low-order (least-significant) bits control

Fragmentation. The low-order bit specifies whether the packet can be fragmented. The middle bit specifies whether the packet is the last fragment in a series of fragmented packets. The third or high-order bit is not used.

Fragment Offset—Indicates the position of the fragment's data relative to the beginning of the data in the original datagram, which allows the destination IP process to properly reconstruct the original datagram.

Time-to-Live—Maintains a counter that gradually decrements down to zero, at which point the datagram is discarded. This keeps packets from looping endlessly.

Protocol—Indicates which upper-layer protocol receives incoming packets after IP processing is complete.

Header Checksum—helps ensure IP header integrity.

Source Address—Specifies the sending node.

Destination Address—Specifies the receiving node..

Options—Allows IP to support various options, such as security.

Data—Contains upper-layer information.

3.2 ROUTING IP DATAGRAM

If all computers were directly connected on the same physical network, there would be little need for the IP protocol. After all, so far in this description of the protocol, the only job IP has performed has been wrapping the transport layer packet into an IP datagram for transmission by the network level. In reality, an

IP datagram sent between two computers on the public network typically passes through many different IP network devices along the way. It is the ability to route IP packets across different physical networks that is the heart of the Internet.

The public network, or Internet, is actually a collection of thousands of individual networks, interconnected together. These interconnections form a mesh network, creating millions of paths between the individual computers on the Internet. Routers are dedicated devices that are the interconnection point for the networks of the world. Routers are responsible for passing IP packets along from the source to the destination, across the various network interconnection points.

Each router that an IP packet passes through is referred to as a hop. In general, as the packet traverses the network, a router is only responsible for getting a packet to the next hop along its path. Routers use the Internet and network layer. Routers need access to the network layer so they can physically receive packets. The network layer then passes the IP datagram up to the router IP layer. The router processes the destination address contained in the IP header and determines which device the send the IP packet on to, typically another router. The transport and user level data is not needed and is not unpacked from the IP datagram. This allows routers to function very quickly, as they are able to unpack the necessary information from the IP packet using specially designed hardware.

Routing Protocols

Routers are responsible for routing IP packets between a source and destination address. Typically, each router is responsible for only getting a packet to the next router along the path. As such, a router only needs to know the addresses of the routers to which it is directly connected. It also needs to know which connected router should be used for forwarding a packet. When the router examines the IP address of an incoming datagram, it accesses a database or table to determine which router should form the next hop in the path. Routers use various protocols to communicate with each other in order to set up the tables used to route packets.

Some common routing protocols include:

- • Router Information Protocol (RIP)
- • Open Shortest Path First (OSPF)
- • Interior Gateway Routing Protocol (IGRP)
- • Enhanced IGRP (EIGRP)
- • Border Gateway Protocol (BGP)
- • Intermediate System to Intermediate System (IS-IS)
- • Constrained Shortest Path First (CSPF)

While routers operate on packets at the Internet layer, they also use transport layer services such as UDP and TCP to communicate with each other to build routing tables.

3.3 FRAGMENTATION

If a router receives an IP packet that is too large for the network onto which the packet is being forwarded

IP will fragment the original packet into smaller packets that will fit on the downstream network. When the packets arrive at their final destination, IP at the destination host reassembles the fragments into the original payload. This process is referred to as fragmentation and reassembly. Fragmentation can occur in environments that have a mix of networking technologies, such as Ethernet and Token Ring.

The fragmentation and reassembly works as follows:

1. When an IP packet is sent by the source, it places a unique value in the Identification field.

2. The IP packet is received at the router. The IP router notes that the maximum transmission unit (MTU) of the network onto which the packet is to be forwarded is smaller than the size of the IP packet.

3. IP fragments the original IP payload into fragments that will fit on the next network. Each fragment is

Sent with its own IP header which contains

IP Header Field	Function
Source IP Address	The IP address of the original source of the IP datagram.
Destination IP Address	The IP address of the final destination of the IP datagram.
Identification	Used to identify a specific IP datagram and to identify all fragments of a specific IP datagram if fragmentation occurs.
Protocol	Informs IP at the destination host whether to pass the packet up to TCP, UDP, ICMP, or other protocols.
Checksum	A simple mathematical computation used to verify the integrity of the IP header.
Time to Live (TTL)	Designates the number of networks on which the datagram is allowed to travel before being discarded by a router. The TTL is set by the sending host and is used to prevent packets from endlessly circulating on an IP internetwork. When forwarding an IP packet, routers are required to decrease the TTL by at least one.

The original Identification field identifies all fragments that belong together. The More Fragments Flag indicates that other fragments follow. The More Fragments Flag is not set

on the last fragment, because no other fragments follow it.

The Fragment Offset field indicates the position of the fragment relative to the original IP payload. When the fragments are received by IP at the remote host, they are identified by the Identification field as belonging together. The Fragment Offset is then used to reassemble the fragments into the original IP payload.

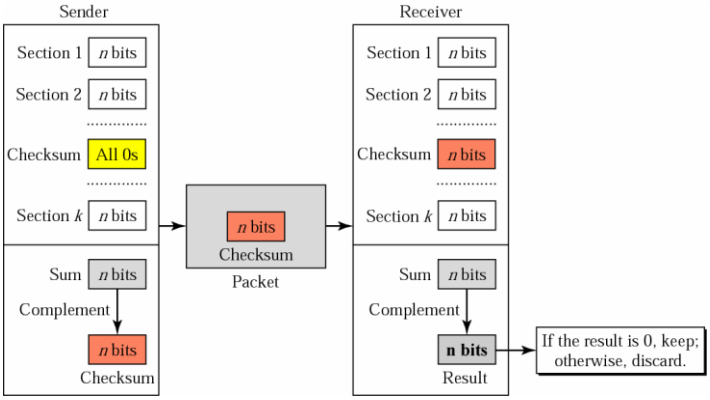
3.4 CHECKSUM

The error detection method used by most TCP/IP protocols is called the checksum. The checksum protects against the corruption that may occur during the transmission of a packet. It is redundant information added to the packet.

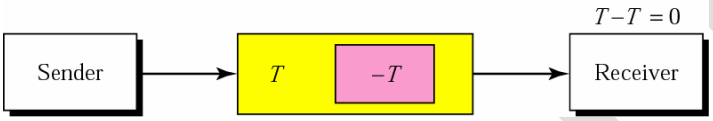
To create the checksum the sender does the following:

1) The packet is divided into k sections, each of n bits.
2) All sections are added together using 1's complement arithmetic.
3) The final result is complemented to make the checksum.

Checksum Concept:

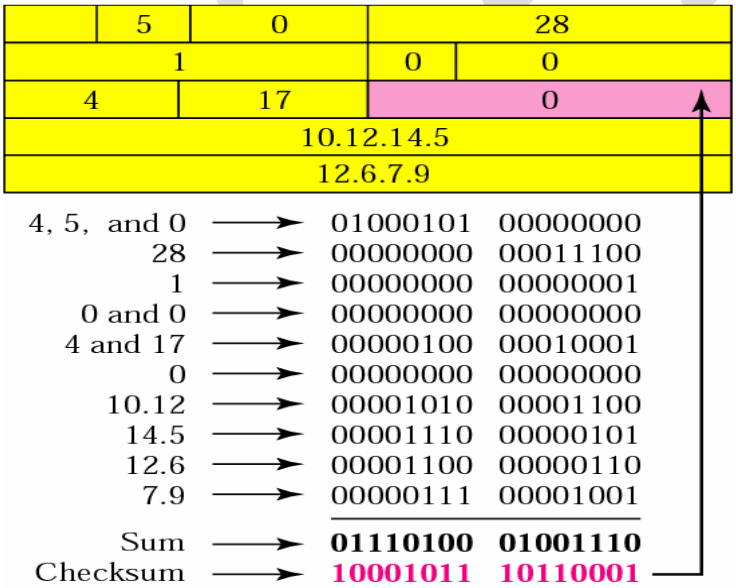


Checksum in one's complement arithmetic



Example:

The following shows an example of a checksum calculation for an IP header without options. The header is divided into 16-bit sections. All the sections are added and the sum is complemented. The result is inserted in the checksum field.



4. INTERNET CONTROL MESSAGE PROTOCOL

The Internet Control Message Protocol (ICMP) protocol is classic example of a client server application. The ICMP server executes on all IP end system computers and all IP intermediate systems (i.e routers). The protocol is used to report problems with delivery of IP datagrams within an IP network. It can be sued to show when a particular End System (ES) is not responding, when an IP network is not reachable, when a node is overloaded, when an error occurs in the IP header information, etc. The protocol is also frequently used by Internet managers to verify correct operations of End Systems (ES) and to check that routers are correctly routing packets to the specified destination address.

4.1 ICMP

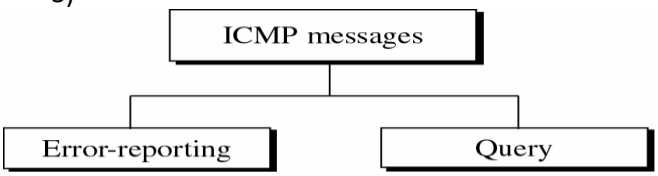
ICMP messages generated by router R1, in response to message sent by H0 to H1 and forwarded by R0. This message could, for instance be generated if the MTU of the link between R0 and R1 was smaller than size of the IP

packet, and the packet had the Don't Fragment (DF) bit set in the IP packet header.The ICMP message is returned to Ho, since this is the source address specified in the IP packet that suffered the problem.

4.2 ICMP MESSAGES

ICMP Messages are used by IP to send error and control messages ICMP uses IP to send messages. It does not report errors on ICMP messages. ICMP message are not required on datagram checksum errors. There are two types of messages namely

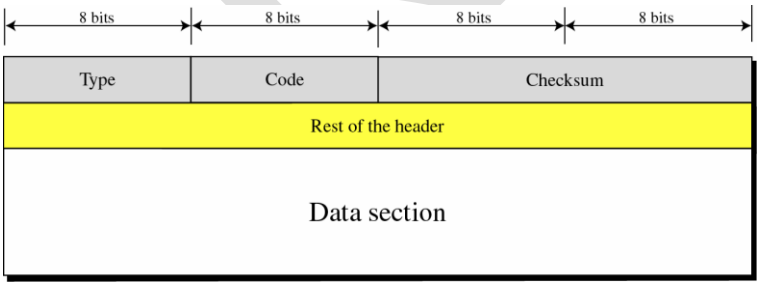
1) Error reporting
2) Query messages.
3)



Category	*Type*	*Message*
Error-reporting messages	3	Destination unreachable
	4	Source quench
	11	Time exceeded
	12	Parameter problem
	5	Redirection
Query messages	8 or 0	Echo request or reply
	13 or 14	Timestamp request or reply
	17 or 18	Address mask request or reply
	10 or 9	Router solicitation or advertisement

General format of ICMP messages

An ICMP message has an 8-byte header and a variable-size data section. Although the general format of the header is different for each message type, the first 4 bytes are common to all.



Type (8): specifies the type of ICMP message
Code (8): used to specify parameters of the message that can be encoded in a few bits

Checksum (16): hecksum of the entire ICMP message

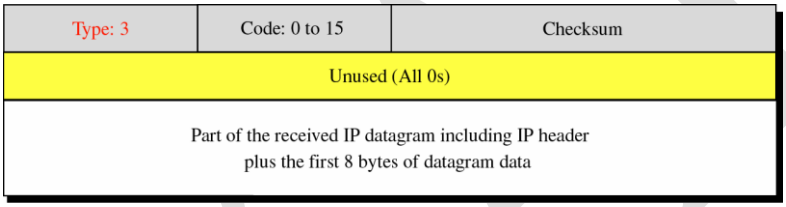
ERROR REPORTING

ICMP always reports error messages to the original source

ICMP error messages report error conditions Typically sent when a datagram is discarded Error message is often passed from ICMP to the application program ICMP error essages include the complete IP header and the first 8 bytes of the payload (typically: UDP, TCP)

a) Destination-unreachable

Destination-unreachable messages with codes 2 or 3 can be created only by the destination host. Other destination-unreachable messages can be created only by routers. A router cannot detect all problems that prevent the delivery of a packet



Code 0 Net Unreachable

Code 1 Host Unreachable

Code 2 Protocol Unreachable

Code 3 Port Unreachable

Code 4 Fragmentation needed & Don't Fragment was set

Code 5 Source Route failed

Code 6 Destination Network Unknown

Code 7 Destination Host Unknown

Code 8 Source Host Isolated

Code 9 Communication Destination Network is Administratively Prohibited

Code 10 Communication Destination Host is Administratively Prohibited

Code 11 Destination Network Unreachable for Type of Service

Code 12 Destination Host Unreachable for Type of Service

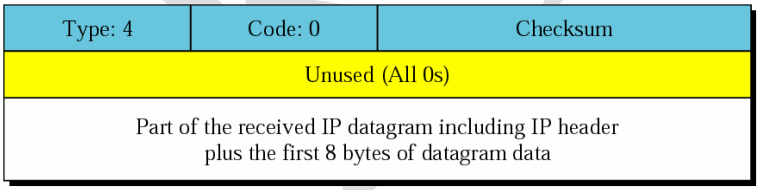
Code 13 Communication Administratively Prohibited

Code 14 Host Precedence Violation

Code 15 Precedence Cutoff Violation

b) Source-quench

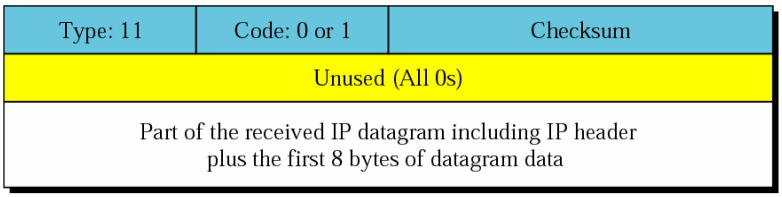
A source-quench message informs the source that a atagram has been discarded due to congestion in a router or the destination host. The source must slow down the sending of datagrams until the congestion is relieved. One source-quench message should be sent for each datagram that is discarded due to congestion.



c) Time Exceed

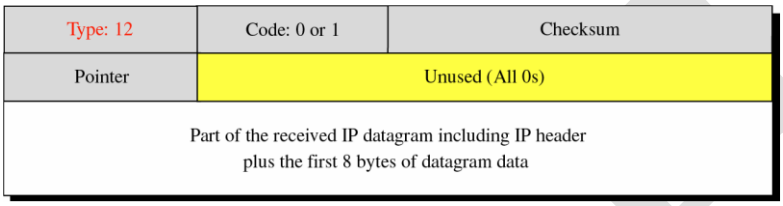
Whenever a router decrements a datagram with a time-to-live value to zero, it discards the datagram and sends a time-exceeded message to the original source. When the final destination does not receive all of the fragments in a set time, it discards the received fragments and sends a time-exceeded message to the original source. In a time-exceeded message, code 0 is used only by routers to show that the

value of the time-to-live field is zero. Code 1 is used only by the destination host to show that not all of the fragments have arrived within a set time.



d) Parameter Problem

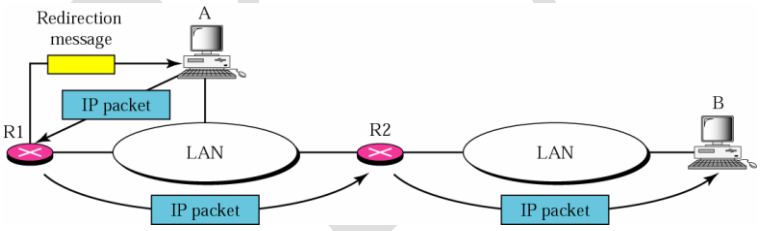
A parameter-problem message can be created by a router or the destination host.

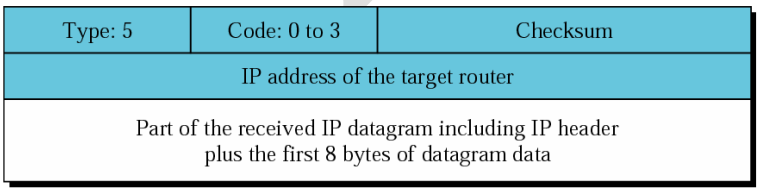


Code 0: Main header problem
Code 1: Problem in the option field

e) Redirection

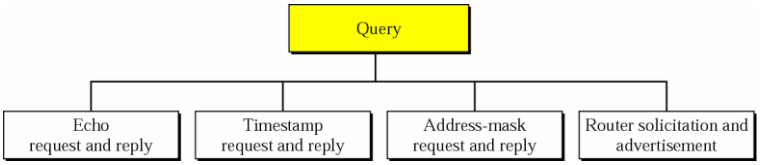
A host usually starts with a small routing table that is gradually augmented and updated. One of the tools to accomplish this is the redirection message. A redirection message is sent from a router to a host on the same local network.





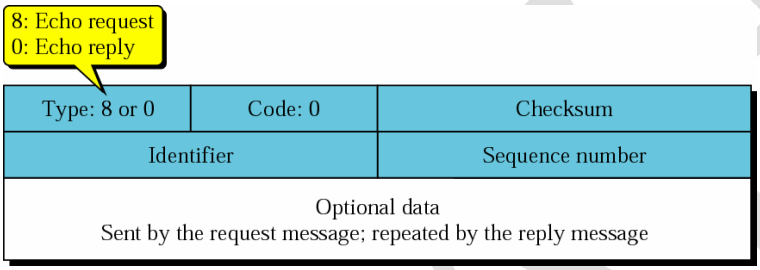
QUERY MESSAGES

ICMP can also diagnose some network problems through the query messages, a group of four different pairs of messages. In this type of ICMP message, a node sends a message that is answered in a specific format by the destination node.



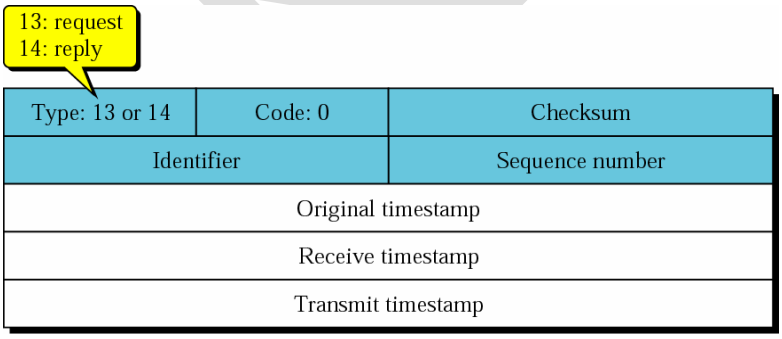
a) Echo Request and Reply:

An echo-request message can be sent by a host or router. An echo-reply message is sent by the host or router which receives an echo-request message. Echo-request and echo-reply messages can be used by network managers to check the operation of the IP protocol. Echo-request and echo-reply messages can test the reachability of a host. This is usually done by invoking the ping command.



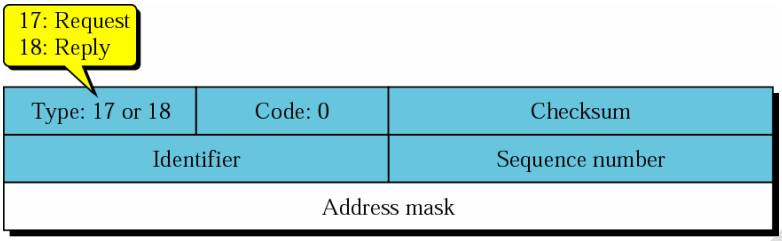
b) Timestamp Request and Reply

Timestamp-request and timestamp-reply messages can be used to calculate the round-trip time between a source and a destination machine even if their clocks are not synchronized. The timestamp-request and timestamp-reply messages can be used to synchronize two clocks in two machines if the exact one-way time duration is known.



c) Address Mask Request & Address Mask Reply

A booting computer to determine the subnet mask in use on the local network uses the Address Mask Request ICMP Type 17. An intermediary device or computer acting as an intermediary device will reply with a Type 18 ICMP Address Mask Reply ICMP.

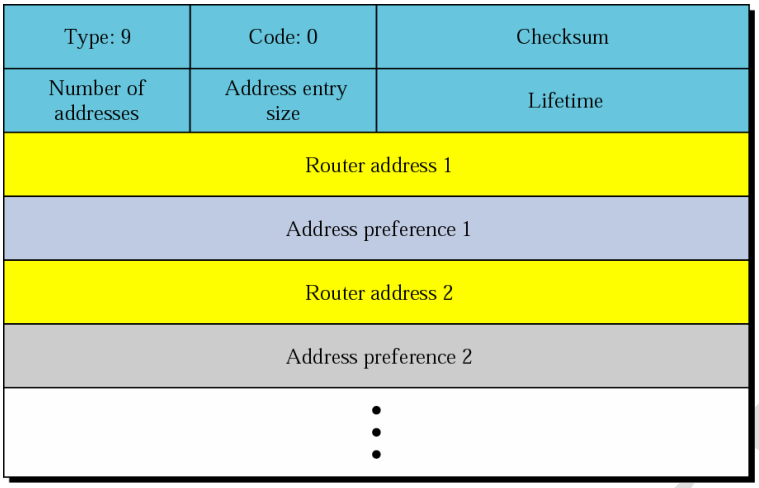


d) **Router-solicitation message**

Router discovery uses Internet Control Message Protocol (ICMP) router advertisements and router solicitation messages to allow a host to discover the addresses of operational routers on the subnet. Hosts must discover routers before they can send IP datagrams outside their subnet. Router discovery allows a host to discover the addresses of operational routers on the subnet. Each router periodically multicasts a router advertisement from each of its multicast interfaces, announcing the IP address of that interface. Hosts listen for advertisements to discover the addresses of their neighboring routers. When a host starts, it can send a multicast router solicitation to ask for immediate advertisements.

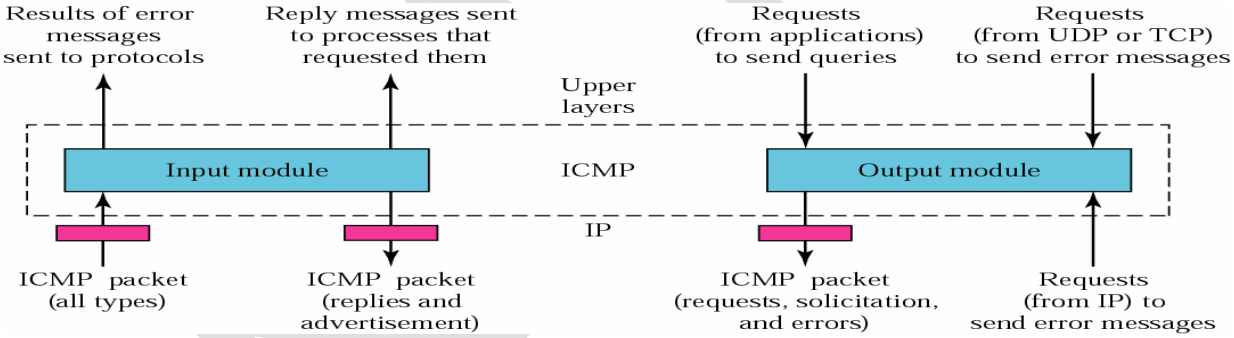
e) Router advertisement message

Router advertisement messages include a preference level and a lifetime field for each advertised router address. The preference level specifies the router's preference to become the default router. When a host chooses a default router address, it chooses the address with the highest preference. You can configure the preference level with the priority statement. The lifetime field indicates the maximum length of time that the advertised addresses are to be considered valid by hosts in the absence of further advertisements..



4.3 ICMP PACKAGE

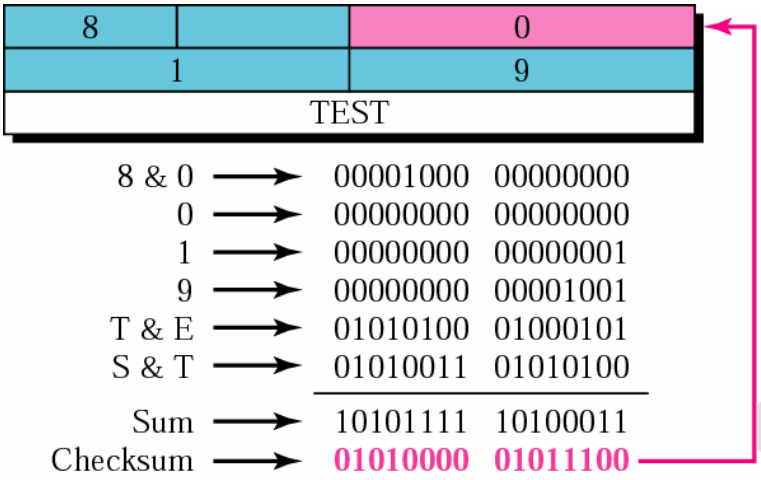
To give an idea of how ICMP can handle the sending and receiving of ICMP messages, we present our version of an ICMP package made of two modules: an input module and an output module



4.4 CHECKSUM

In ICMP the checksum is calculated over the entire message (header and data). The below figure shows an example of checksum calculation for a simple echo-request message.

We randomly chose the identifier to be 1 and the sequence number to be 9. The message is divided into 16-bit (2-byte) words. The words are added together and the sum is complemented. Now the sender can put this value in the checksum field.

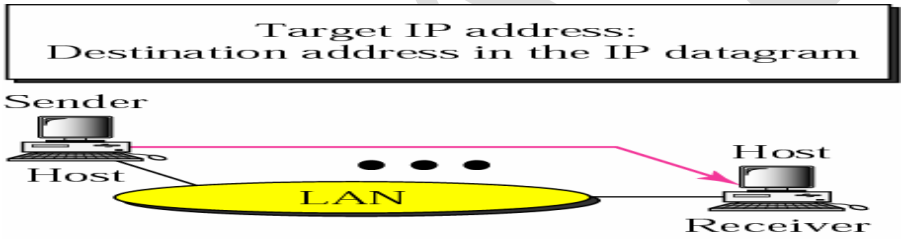


5. EXAMPLES

Example 1:

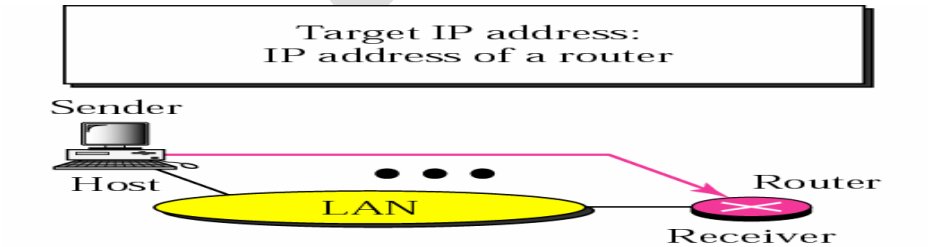
Four cases of ARP:

Case 1:



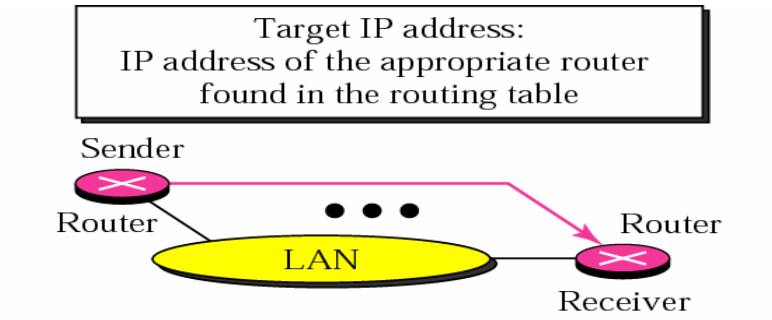
Case 1. A host has a packet to send to another host on the same network.

Case 2:



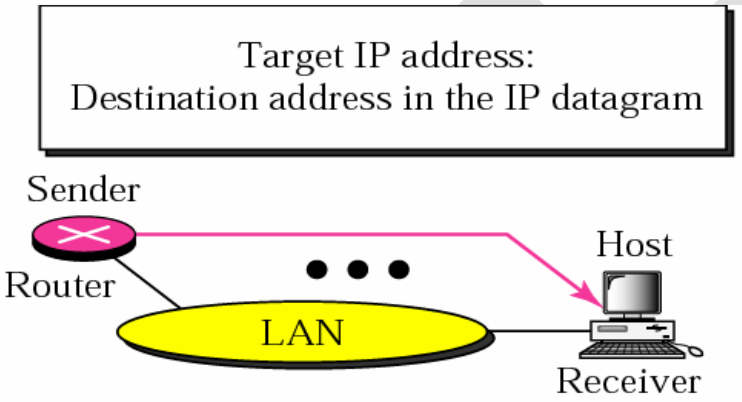
Case 2. A host wants to send a packet to another host on another network.
It must first be delivered to a router.

Case 3 :



Case 3. A router receives a packet to be sent to a host on another network. It must first be delivered to the appropriate router.

Case 4:

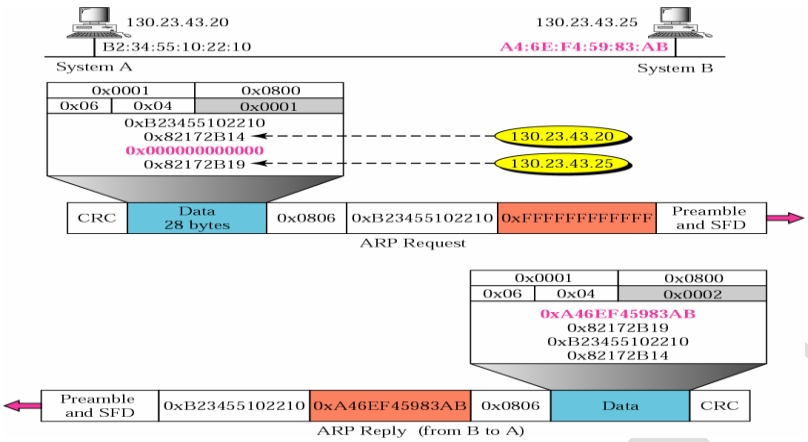


Case 4. A router receives a packet to be sent to a host on the same network.

Example 2:

Problem : A host with IP address 130.23.43.20 and physical address B2:34:55:10:22:10 has a packet to send to another host with IP address 130.23.43.25 and physical address A4:6E:F4:59:83:AB (which is unknown to the first host). The two hosts are on the same Ethernet network. Show the ARP request and reply packets encapsulated in Ethernet frames.

Solution :



The above figure shows the ARP request and reply packets. Note that the ARP data field in this case is 28 bytes, and that the individual addresses do not fit in the 4-byte boundary. That is why we do not show the regular 4-byte boundaries for these addresses. Also note that the IP addresses are shown in hexadecimal.

Example 3:

Timestamp-request and timestamp-reply messages can be used to calculate the round-trip time between a source and a destination machine even if their clocks are not synchronized

Sending time = value of receive timestamp - value of original

Timestamp

Receiving time = time the packet returned - value of transmit

Timestamp

Round-trip time = sending time + receiving time

Given the following information:

Value of original timestamp: 46

Value of receive timestamp: 59

Value of transmit timestamp: 60

Time the packet arrived: 67

Sending time = 59 - 46 = 13 milliseconds

Receiving time = 67 - 60 = 7 milliseconds

Round-trip time = 13 + 7 = 20 milliseconds

Time diff= receive timestamp -(original timestamp Field
+ one-way time duration)

5.1. GLOSSARY

Broadcast

Broadcast is the term used to describe communication where a piece of information is sent from one point to all other points. Broadcasting is a useful feature in e-mail systems. It is also supported by some fax systems.

Ethernet

A local-area network (LAN) architecture developed by Xerox Corporation in cooperation with DEC and Intel in 1976. Ethernet uses a bus or star topology and supports data transfer rates of 10 Mbps. The Ethernet specification served as the basis for the IEEE 802.3 standard, which specifies the physical and lower software layers. Ethernet uses the CSMA/CD access method to handle simultaneous demands. It is one of the most widely implemented LAN standards.

IP

The IP (Internet Protocol) is a protocol which uses datagram's to communicate over a packet-switched network. IP specifies the format of packets, also called datagram's, and the addressing scheme. Most networks combine IP with a higher-level protocol called Transmission Control Protocol (TCP), which establishes a virtual connection between a destination and a source.

IP network addresses

An address is a data structure understood by a network which uniquely identifies the recipient within the network.

MAC

MAC (Medium Access Control) is a hardware address that uniquely identifies each node of a network. In IEEE 802 networks, the Data Link Control (DLC) layer of the OSI Reference Model is divided into two sub layers: the Logical Link Control (LLC) layer and the Media Access Control (MAC) layer. The MAC layer interfaces directly with the network medium. Consequently, each different type of network medium requires a different MAC layer.

NIC

NIC (Network interface card) is an expansion board you insert into a computer so the computer can be connected to a network. Most NICs are designed for a particular type of network, protocol, and media, although some can serve multiple networks.

5.3. POINTS TO REMEMBER

- The protocol used to associate an IP address with physical address is ARP
- Logical address is an internet work address
- The logical address in the TCP/IP protocol suit are called a logical address IP address
- In Dynamic mapping each time a machine knows one of the 2 addresses.
- RARP Stand for Reverse Address Resolution Protocol
- RARP allows a host to discover its internet address when it knows only its physical address.
- Static mapping means creating a table that associates a logical address with the physical address.
- Hardware type. is a 16-bit field defining the type of the network on which ARP is running.
- Protocol type is a m16-bit field defining the protocol.
- An ARP request is Broadcast
- An ARP reply is Unicast
- PROXY ARP is used to create a subletting effect.
- An ARP package involves 5 Components
- A Logical address is an internet work address with universal jurisdiction
- The logical addresses in the TCP/IP protocol suite are called IP addresses.
- Physical is a local address. Its jurisdiction is over a local network.
- The sender is a router that has received a datagram destined for a host on another network. The logical address that must be mapped to a physical address is the destination IP address in the datagram header
- If the sender is a host and wants to send a packet to another host on the same network, the logical address that must be mapped to a physical address is the destination IP address in the datagram header

- If the sender is a host and wants to send a packet to another host on another network, the logical address that must be mapped to a physical address is the destination IP address in the datagram header
- The sender is a router that has received a datagram destined for a host in the same network. The logical address that must be mapped to a physical address is the IP address of the router found in the routing table
- An ICMP message has 8 byte header and a variable-size data section.
- If a host needs to synchronize its clock with another host, it sends a Timestamp message.
- The purpose of echo request and echo reply is to check node to node communication

Possible Questions

Part B
(Each Question carries 6 Marks)

1. Discuss on packet format and operation of ARP.
2. Discuss on check sum in IP datagram.
3. Discuss about RARP operation
4. Give a brief description about ICMP protocol.
5. Explain IP datagram format and give a brief description of each field in it in order
6. Discuss about message format and error reporting in ICMP
7. Explain about ICMP Messages in detail.
8. Describe internet message control protocol in detail.
9. Write notes the option field in IP datagram
10. Discuss about the process of fragmentation with an example

Part C

(Each Question carries 10 Marks)

A host with IP address 130.23.43.20 and physical address B2:34:55:10:22:10 has a packet to send to another host with IP address 130.23.43.25 and physical address A4:6E:F4:59:83:AB (which is unknown to the first host). The two hosts are on the same Ethernet network. Draw and explain the ARP request and reply packets encapsulated in Ethernet frames.

Karpagam Academy of Higher Education
Coimbatore - 21

DEPARTMENT OF COMPUTER APPLICATIONS
CLASS : II MCA

TCP/IP (18CAP405N)

Unit II

S.NO	Question	Choice1	Choice2	Choice3	Choice4	Choice 5	Choice 6	Answer
1	MTU Stands for _____	Minimum Transfer Unit	Maximum Transfer Unit	Maximum Transport Unit	Minimum Transceiver Unit			Maximum Transfer Unit
2	A loose source route option is similar to _____ option.	Time Stamp option	Strict Source route option	End of option	rubber stamp			Strict Source route option
3	The _____ contains the data that specific option requires.	Length field	data field	Width field	No field			data field
4	HLEN stands for _____	Header Length	Heading Length	Highlight Length Enable Network	High length			Header Length
5	_____ is a 3 bit sub field ranging from 0 to 7	Precedence	TOS Bits	Code point	protocol			Precedence
6	_____ is a local address.	physical	logical	network	IP			physical
7	_____ attaches to two or more physical networks and forwards IP datagrams	Router	Hub	Bridge	Amplifier			Router
8	Internet routers could be portioned into _____ and _____	MILNET and ARPANET	Core and Non core	Individual and Non individual	MANET			Core and Non core
9	_____ routers are controlled by individual routers	Core	Non core	coredeo	dicode			Non core
10	_____ travels from routers to routers	Text	Numbers	Datagrams	Characters.			Datagrams
11	_____ is used to measure distance	Core	Vector	Dynamic	Hops			Hops
12	___is a group of network controlled by single administrative autthority	Non autonomous system	Autonomous system	Single system	double system			Autonomous system

13	_____ forms the central interconnect for an internet	Base header	Application Gateway	Backbone network	Local Area Network			Backbone network
14	_____ is used to pass information between two autonomous system	EGP	IGP	DGP	CGP			EGP
15	BGP evolves _____ versions	3	2	4	1			4
16	__ protocol is used to propagate routing information inside a autonomous system	IGP	RIP	RFC	bbb			RIP
17	RIP messages can be broadly classified into _____ types	3	2	4	5			2
18	_____ reference to processor choosing the path to send packets	Router	Hub	Bridge	Switch			Router
19	Routers exchange _____ to initialize their network	OSFP	OSPF	OPSF	OFSP			OSPF
20	OSPF sends _____ message to test neighbour reach ability	Hello	Check	Transmit	Receive			Hello
21	Command1 terns on the _____ mode	update	Request	Trace	Report			Trace
22	Final technique for solving the slow convengaice problem is called _____	Triggered updates	Poison reverce	Poison Receive	Trigger			Poison reverce
23	Path information in BGP allows the receiver to check for _____	Routing loops	Routing algorithms	Distance Routing	Routing table			Routing loops
24	BGP periodically exchange ____message to text network connectivity	OPEN	EXCHANGE	CHECK	KEEPALIVE			KEEPALIVE
25	Command 10 in RIP is _____	Update Request	Update Response	Update acknowledgement	Update acknowledge			Update Response
26	_____ used by routing protocols to combine multiple destination same hope	Route aggregation	Route alert	Route integration	Route aggregated			Route aggregation
27	The _____is one of the necessary protocols that is involved In multicasting	ICMP	IGMP	TCP	IP			IGMP
28	_____ defines the amount of time in which a query must be answered	Maximum response time	Minimum response time	Response time	Check sum			Maximum response time

29	To prevent unnecessary traffic , IGMP uses a _____ strategy	Maximum response time	Minimum response time	Delayed response	response			Delayed response
30	Well known port numbers are less than_____	255	1024	125	206			1024
31	_____are the ports ranging from 0 to 1023, are assigned and controlled by LCANN	Registered ports	Dynamic ports	Static ports	Well known ports			Well known ports
32	The combination of _____ and _____, called socket address	IP Address, Port Number	IP Header, Port Number	Socket Number, Port Number	IP Header, Socket Number			IP Address, Port Number
33	TCP offers _____ services	Full duplex	Half Duplex	Both a and b	Multi duplex			Full duplex
34	A packet in a TCP is called _____	Frame	Segment	Data	Bits			Segment
35	_____ is a machine that goes through a limited number of states	State machine	Finite State machine	Limited state machine	Stateless Machine			Finite State machine
36	To accomplish flow control, TCP uses a _____ protocol	ICMP	IGMP	Sliding window	SNMP			Sliding window
37	A _____ is packet sent by a router to the source to inform it of congestion	Choke point	Back pressure	Implicit Signaling	Explicit signaling			Choke point
38	In _____ routing router needs to construct a shortest path tree for each group	Unicast routing	Multicast routing	Multicast link state routing	Multicast distance vector routing			Multicast routing
39	_____ protocol is a group shared protocol that uses a core as the root of the tree	ICMP	DVMRP	CBT protocol	IGMP			CBT protocol
40	PIM-SM is used in a sparse multicast environment such as _____	WAN	LAN	MAN	WIFI			WAN
41	The value of group address is _____ for a general query message	2	3	0	4			0
42	The port ranging from 1024 to 49151, assigned or controlled by ICANN, known as	Dynamic ports	Static ports	Well known ports	Registered ports			Registered ports
43	The UDP packets, called used data grams have a fixed size header of _____ bytes	6	8	32	64			8
44	The connection establishing in TCP is called _____	3-way hand shaking	4-way hand shaking	2-way hand shaking	1-way hand shaking			3-way hand shaking

45	SYN flooding attack belongs to a group of security attacks known as _____ attack	Denial of service	Mosquerade	Replay	Receive			Denial of service
46	One of the algorithm used in TCP congestion control is _____	Fast start	Slow start	RFC algorithm	Medium Start			Slow start
47	_____is a client/server protocol designed to provide information for a disk less computer	BOOTP	RARP	ICMP	ARP			BOOTP
48	Bootstrap protocol is an _____ layer program.	network	transport	application	physical			application
49	The host that can be used as a relay to operate at the application layer is _____	BOOTP server	BOOT client	relay agent	Local agent			relay agent
50	The relay agent knows the _____address of a BOOT server	multicast	unicast	broadcast	network address			unicast
51	The ID which is randomly choosen for each connection involving BOOTP is_____	transaction ID	net ID	host id	BOOTP id			transaction ID
52	BOOTP uses _____which does not provide error control	TCP	UDP	TFTP	ARP			UDP
53	_____ is a 8 bit field defining the maximum number of hops the packet can travel	operation code	hardware type	hardware length	hop count			hop count
54	_____ is a 4 bit field containing the IP address of a router	gate way IP address	client IP address	server IP address	global IP address			gate way IP address
55	_____ provides static and dynamic address allocation that can be manual or automatic.	BOOTP	DHCP	TFTP	RARP			DHCP
56	_____ is backward compatible with BOOTP	DHCP	BOOTP	UDP	ARP			DHCP
57	DHCP server issues a_____for a specific period of time	time stamp	time slice	lease	least			lease
58	A server reply can be _____	broadcast	unicast	multicast	broadcast and imocast			Broadcast and unicast
59	_____is a static configuration protocol	BOOTP	TFPT	UDP	RARP			BOOTP
60	_____ is a client server application on the internet with the unique user friendly name	DNS	DDNS	FQDN	PQDN			DNS

UNIT-III

SYLLABUS

Routing and Routed Protocols - Autonomous Systems – Routing Table - Interior Gateway Protocols – Exterior Gateway Protocols – Routing in Internet. Group Management – IGMP Message – IGMP Operation – Process to Process Communication

Routing and Routed Protocols:

Unicast Routing

Unicast is the process of forwarding unicasted traffic from a source to a destination on network. Unicasted traffic is destined for a unique address. In this case there is just one sender, and one receiver. The term exists in contradistinction to multicast, communication between a single sender and a group of selected receivers, and anycast, communication between any sender and a group of receivers near the sender in a network. An earlier term, *point-to-point* communication, is similar in meaning to unicast.

Unicast transmission is the predominant form of transmission on LANs and within the Internet. All LANs (e.g. Ethernet) and IP networks support the unicast transfer mode, and most users are familiar with the standard unicast applications such as http, smtp, ftp and telnet, which employ the TCP transport protocol. The new Internet Protocol version 6 (IPv6) supports unicast as well as anycast and multicast. Many routing protocols such as Routing Information Protocol (RIP) and Open Shortest Path First (OSPF), are designed to provide unicast routing on Internet effectively. IPv6 has unicast, multicast, and anycast. Broadcast has disappeared as a term, but is considered one form of multicast.

Distance Vector Routing

Modern computer networks generally use dynamic routing algorithms rather than the static ones described above because static algorithms do not take the current network load into account. Two dynamic algorithms in particular, distance vector routing and link state routing, are the most popular. In this section we will look at the former algorithm. In the following section we will study the latter algorithm.

Distance vector routing algorithms operate by having each router maintain a table (i.e, a vector) giving the best known distance to each destination and which line to use to get there. These tables are updated by exchanging information with the neighbors.

The distance vector routing algorithm is sometimes called by other names, most commonly the distributed Bellman-Ford routing algorithm and the Ford-Fulkerson algorithm, after the researchers who developed it (Bellman, 1957; and Ford and Fulkerson, 1962). It was the original ARPANET routing algorithm and was also used in the Internet under the name RIP.

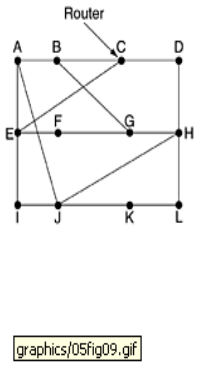
In distance vector routing, each router maintains a routing table indexed by, and containing one entry for, each router in the subnet. This entry contains two parts: the preferred outgoing line to use for that destination and an estimate of the time or distance to that destination. The metric used might be number of hops, time delay in milliseconds, total number of packets queued along the path, or something similar.

The router is assumed to know the "distance" to each of its neighbors. If the metric is hops, the distance is just one hop. If the metric is queue length, the router simply examines each queue. If the metric is delay, the router can measure it directly with special ECHO packets that the receiver just timestamps and sends back as fast as it can.

As an example, assume that delay is used as a metric and that the router knows the delay to each of its neighbors. Once every T msec each router sends to each neighbor a list of its estimated delays to each destination. It also receives a similar list from each neighbor. Imagine that one of these tables has just come in from neighbor X, with X_i being X's estimate of how long it takes to get to router i. If the router knows that the delay to X is m msec, it also knows that it can reach

router i via X in X_i + m msec. By performing this calculation for each neighbor, a router can find out which estimate seems the best and use that estimate and the corresponding line in its new routing table. Note that the old routing table is not used in the calculation.

This updating process is illustrated in figure below.. Part (a) shows a subnet. The first four columns of part (b) show the delay vectors received from the neighbors of router J. A claims to have a 12-msec delay to B, a 25-msec delay to C, a 40-msec delay to D, etc. Suppose that J has measured or estimated its delay to its neighbors, A, I, H, and K as 8, 10, 12, and 6 msec, respectively.



(a)

To	A	I	H	K	New estimated delay from J	Line
A	0	24	20	21	8	A
B	12	36	31	28	20	A
C	25	18	19	36	28	I
D	40	27	8	24	20	H
E	14	7	30	22	17	I
F	23	20	19	40	30	I
G	18	31	6	31	18	H
H	17	20	0	19	12	H
I	21	0	14	22	10	I
J	9	11	7	10	0	–
K	24	22	22	0	6	K
L	29	33	9	9	15	K
	JA delay is 8	JI delay is 10	JH delay is 12	JK delay is 6		

Vectors received from J's four neighbors

New routing table for J

(b)

msec, and A claims to be able to get to G in 18 msec, so J knows it can count on a delay of 26 msec to G if it forwards packets bound for G to A. Similarly, it computes the delay to G via I, H, and K as 41 (31 + 10), 18 (6 + 12), and 37 (31 + 6) msec, respectively. The best of these values is 18, so it makes an entry in its routing table that the delay to G is 18 msec and that the route to use is via H. The same calculation is performed for all the other destinations, with the new routing table shown in the last column of the figure.

Distance vector routing was used in the ARPANET until 1979, when it was replaced by link state routing. Two primary problems caused its demise. First, since the delay metric was queue length, it did not take line bandwidth into account when choosing routes. Initially, all the lines were 56 kbps, so line bandwidth was not an issue, but after some lines had been upgraded to 230 kbps and others to 1.544 Mbps, not taking bandwidth into account was a major problem. Of course, it would have been possible to change the delay metric to factor in line bandwidth, but a second problem also existed, namely, the algorithm often took too long to converge (the count-to-infinity problem). For these reasons, it was replaced by an entirely new algorithm, now called link state routing. Variants of link state routing are now widely used.

The idea behind link state routing is simple and can be stated as five parts. Each router must do the following:

Discover its neighbors and learn their network addresses.

Measure the delay or cost to each of its neighbors.

Construct a packet telling all it has just learned.

Send this packet to all other routers.

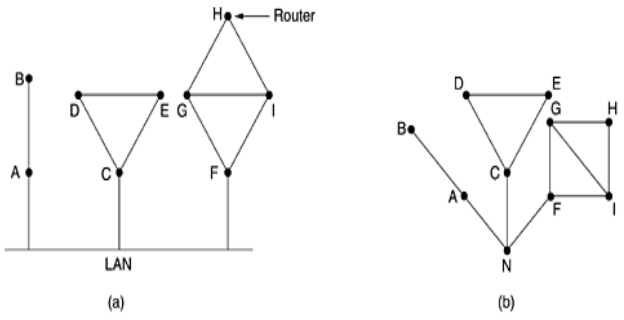
Compute the shortest path to every other router.

In effect, the complete topology and all delays are experimentally measured and distributed to every router. Then Dijkstra's algorithm can be run to find the shortest path to every other router. Below we will consider each of these five steps in more detail.

Learning about the Neighbors

When a router is booted, its first task is to learn who its neighbors are. It accomplishes this goal by sending a special HELLO packet on each point-to-point line. The router on the other end is expected to send back a reply telling who it is. These names must be globally unique because when a distant router later hears that three routers are all connected to F, it is essential that it can determine whether all three mean the same F.

When two or more routers are connected by a LAN, the situation is slightly more complicated below figure illustrates a LAN to which three routers, A, C, and F, are directly connected. Each of these routers is connected to one or more additional routers, as shown.



One way to model the LAN is to consider it as a node itself, as shown in Figure Here we have introduced a new, artificial node, N, to which A, C, and F are connected. The fact that it is possible to go from A to C on the LAN is represented by the path ANC here.

Measuring Line Cost

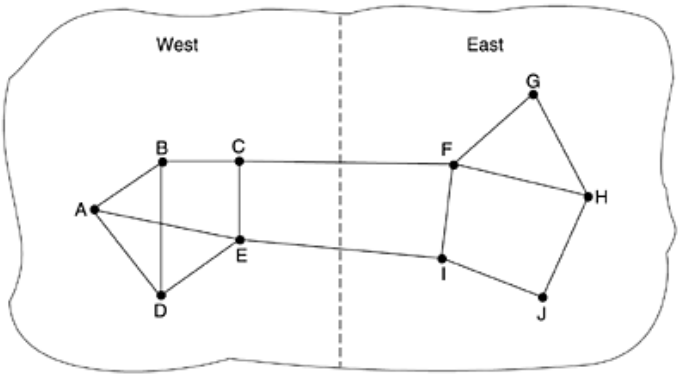
The link state routing algorithm requires each router to know, or at least have a reasonable estimate of, the delay to each of its neighbors. The most direct way to determine this delay is to

send over the line a special ECHO packet that the other side is required to send back immediately. By measuring the round-trip time and dividing it by two, the sending router can get a reasonable estimate of the delay. For even better results, the test can be conducted several times, and the average used. Of course, this method implicitly assumes the delays are symmetric, which may not always be the case.

An interesting issue is whether to take the load into account when measuring the delay. To factor the load in, the round-trip timer must be started when the ECHO packet is queued. To ignore the load, the timer should be started when the ECHO packet reaches the front of the queue.

Arguments can be made both ways. Including traffic-induced delays in the measurements means that when a router has a choice between two lines with the same bandwidth, one of which is heavily loaded all the time and one of which is not, the router will regard the route over the unloaded line as a shorter path. This choice will result in better performance.

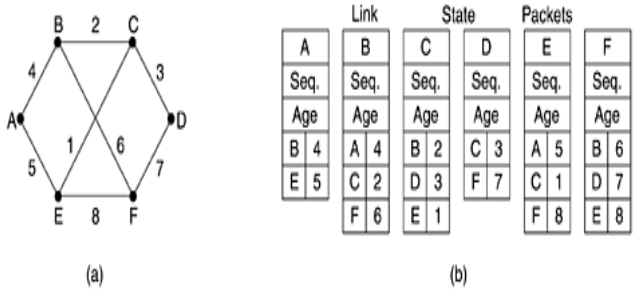
Unfortunately, there is also an argument against including the load in the delay calculation. Consider the subnet of figure which is divided into two parts, East and West, connected by two lines, CF and EI.



Suppose that most of the traffic between East and West is using line CF, and as a result, this line is heavily loaded with long delays. Including queueing delay in the shortest path calculation will make EI more attractive. After the new routing tables have been installed, most of the East-West traffic will now go over EI, overloading this line. Consequently, in the next update, CF will appear to be the shortest path. As a result, the routing tables may oscillate wildly, leading to erratic routing and many potential problems. If load is ignored and only bandwidth is considered, this problem does not occur. Alternatively, the load can be spread over both lines, but this solution does not fully utilize the best path. Nevertheless, to avoid oscillations in the choice of best path, it may be wise to distribute the load over multiple lines, with some known fraction going over each line.

Building Link State Packets

Once the information needed for the exchange has been collected, the next step is for each router to build a packet containing all the data. The packet starts with the identity of the sender, followed by a sequence number and age (to be described later), and a list of neighbors. For each neighbor, the delay to that neighbor is given. An example subnet is given in fig a) with delays shown as labels on the lines. The corresponding link state packets for all six routers are shown in fig b



Building the link state packets is easy. The hard part is determining when to build them. One possibility is to build them periodically, that is, at regular intervals. Another possibility is to build them when some significant event occurs, such as a line or neighbor going down or coming back up again or changing its properties appreciably.

Distributing the Link State Packets

The trickiest part of the algorithm is distributing the link state packets reliably. As the packets are distributed and installed, the routers getting the first ones will change their routes. Consequently, the different routers may be using different versions of the topology, which can lead to inconsistencies, loops, unreachable machines, and other problems.

First we will describe the basic distribution algorithm. Later we will give some refinements. The fundamental idea is to use flooding to distribute the link state packets. To keep the flood in check, each packet contains a sequence number that is incremented for each new packet sent. Routers keep track of all the (source router, sequence) pairs they see.

When a new link state packet comes in, it is checked against the list of packets already seen. If it is new, it is forwarded on all lines except the one it arrived on. If it is a duplicate, it is discarded. If a packet with a sequence number lower than the highest one seen so far ever arrives, it is rejected as being obsolete since the router has more recent data.

This algorithm has a few problems, but they are manageable. First, if the sequence numbers wrap around, confusion will reign. The solution here is to use a 32-bit sequence number. With one link state packet per second, it would take 137 years to wrap around, so this possibility can be ignored.

Second, if a router ever crashes, it will lose track of its sequence number. If it starts again at 0, the next packet will be rejected as a duplicate.

Third, if a sequence number is ever corrupted and 65,540 is received instead of 4 (a 1-bit error), packets 5 through 65,540 will be rejected as obsolete, since the current sequence number is thought to be 65,540.

The solution to all these problems is to include the age of each packet after the sequence number and decrement it once per second. When the age hits zero, the information from that router is discarded. Normally, a new packet comes in, say, every 10 sec, so router information only times out when a router is down (or six consecutive packets have been lost, an unlikely event). The Age field is also decremented by each router during the initial flooding process, to make sure no packet can get lost and live for an indefinite period of time (a packet whose age is zero is discarded).

Some refinements to this algorithm make it more robust. When a link state packet comes in to a router for flooding, it is not queued for transmission immediately. Instead it is first put in a holding area to wait a short while. If another link state packet from the same source comes in before the first packet is transmitted, their sequence numbers are compared. If they are equal, the duplicate is discarded. If they are different, the older one is thrown out. To guard against errors on the router-router lines, all link state packets are acknowledged. When a line goes idle, the holding area is scanned in round-robin order to select a packet or acknowledgement to send.

The data structure used by router B for the subnet shown in Fig. (a) is depicted in Fig. . Each row here corresponds to a recently-arrived, but as yet not fully-processed, link state packet. The table records where the packet originated, its sequence number and age, and the data. In addition, there are send and acknowledgement flags for each of B's three lines (to A, C, and F, respectively). The send flags mean that the packet must be sent on the indicated line. The acknowledgement flags mean that it must be acknowledged there.

			Send flags			ACK flags			
Source	Seq.	Age	A	C	F	A	C	F	Data
A	21	60	0	1	1	1	0	0	
F	21	60	1	1	0	0	0	1	
E	21	59	0	1	0	1	0	1	
C	20	60	1	0	1	0	1	0	
D	21	59	1	0	0	0	1	1	

In figure the link state packet from A arrives directly, so it must be sent to C and F and acknowledged to A, as indicated by the flag bits. Similarly, the packet from F has to be forwarded to A and C and acknowledged to F.

However, the situation with the third packet, from E, is different. It arrived twice, once via EAB and once via EFB. Consequently, it has to be sent only to C but acknowledged to both A and F, as indicated by the bits.

If a duplicate arrives while the original is still in the buffer, bits have to be changed. For example, if a copy of C's state arrives from F before the fourth entry in the table has been forwarded, the six bits will be changed to 100011 to indicate that the packet must be acknowledged to F but not sent there.

Computing the New Routes

Once a router has accumulated a full set of link state packets, it can construct the entire subnet graph because every link is represented. Every link is, in fact, represented twice, once for each direction. The two values can be averaged or used separately.

Now Dijkstra's algorithm can be run locally to construct the shortest path to all possible destinations. The results of this algorithm can be installed in the routing tables, and normal operation resumed.

For a subnet with n routers, each of which has k neighbors, the memory required to store the input data is proportional to kn. For large subnets, this can be a problem. Also, the computation time can be an issue. Nevertheless, in many practical situations, link state routing works well.

However, problems with the hardware or software can wreak havoc with this algorithm (also with other ones). For example, if a router claims to have a line it does not have or forgets a line it does have, the subnet graph will be incorrect. If a router fails to forward packets or corrupts them while forwarding them, trouble will arise.

Finally, if it runs out of memory or does the routing calculation wrong, bad things will happen. As the subnet grows into the range of tens or hundreds of thousands of nodes, the probability of some router failing occasionally becomes nonnegligible. The trick is to try to arrange to limit the damage when the inevitable happens. Perlman (1988) discusses these problems and their solutions in detail.

Link state routing is widely used in actual networks, so a few words about some example protocols using it are in order. The OSPF protocol, which is widely used in the Internet, uses a link state algorithm.

Another link state protocol is IS-IS (Intermediate System-Intermediate System), which was designed for DECnet and later adopted by ISO for use with its connectionless network layer protocol, CLNP. Since then it has been modified to handle other protocols as well, most notably, IP. IS-IS is used in some Internet backbones (including the old NSFNET backbone) and in some digital cellular systems such as CDPD. Novell NetWare uses a minor variant of IS-IS (NLSP) for routing IPX packets.

Basically IS-IS distributes a picture of the router topology, from which the shortest paths are computed. Each router announces, in its link state information, which network layer addresses it can reach directly. These addresses can be IP, IPX, AppleTalk, or any other addresses. IS-IS can even support multiple network layer protocols at the same time.

Many of the innovations designed for IS-IS were adopted by OSPF (OSPF was designed several years after IS-IS). These include a self-stabilizing method of flooding link state updates, the concept of a designated router on a LAN, and the method of computing and supporting path splitting and multiple metrics. As a consequence, there is very little difference between IS-IS and OSPF. The most important difference is that IS-IS is encoded in such a way that it is easy and natural to simultaneously carry information about multiple network layer protocols, a feature OSPF does not have. This advantage is especially valuable in large multiprotocol environments.

Group Management

Internet Group management protocol (IGMP), a multicasting protocol in the internet protocols family, is used by IP hosts to report their host group memberships to any immediately neighboring multicast routers. IGMP messages are encapsulated in IP datagrams, with an IP protocol number of 2. IGMP has versions IGMP v1, v2 and v3

IGMPv1: Hosts can join multicast groups. There were no leave messages. Routers were using a time-out based mechanism to discover the groups that are of no interest to the members.

IGMPv2: Leave messages were added to the protocol. Allow group membership termination to be quickly reported to the routing protocol, which is important for high-bandwidth multicast groups and/or subnets with highly volatile group membership.

IGMPv3: Major revision of the protocol. It allows hosts to specify the list of hosts from which they want to receive traffic from. Traffic from other hosts is blocked inside the network. It also allows hosts to block inside the network packets that come from sources that sent unwanted traffic.

The variant protocols of IGMP are:

DVMRP: Distance Vector Multicast Routing Protocol.

IGAP: IGMP for user Authentication Protocol.

RGMP: Router-port Group Management Protocol.

Protocol Structure -IGMP (Internet Group Management Protocol)

There are basically 5 types of messages in the IGMP that must be implemented in IGMP for the IGMP v3 functional properly and be compatible with previous versions:

0x11: membership query

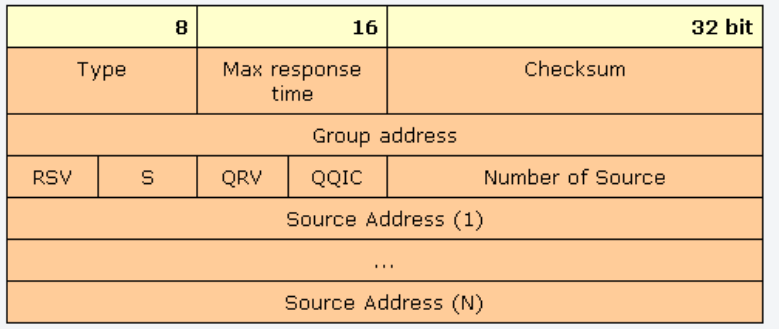
0x22: version 3 membership report

0x12: version 1 membership report

0x16: version 2 membership report

0x17 version 2 leave group

As an example, the message format for 0x11 (membership query) is displayed as follows



Type -- The message type: 0x11 (Membership query).

Max Response Time -- Used only in Membership query messages. Specifies the maximum time allowed before sending a responding report in units of 1/10 second. In all other messages, it is set to 0 by the sender and ignored by the receiver.

Checksum -- The checksum for messages errors

Group Address -- The Group address is set to 0 when sending a general query. It is set to the group address being queried, when sending a group specific query or group-and-source-specific query. In a membership report of a leave group message, it holds the IP multicast group address of the group being reported or left.

Group Record -- A block of fields containing information about the sender's membership in a single multicast group, on the interface from which the report is sent.

IGMP Messages

There are three message types used in IGMP. The IGMP 'type' field is set to the following values for each message type [in hex] :

MEMBERSHIP QUERY [0x11]

Membership Query messages are used by multicast enabled routers running IGMP to discover which hosts on attached networks are members of which multicast groups. Membership Query messages are sent to the 'all-systems' multicast group address of 224.0.0.1.

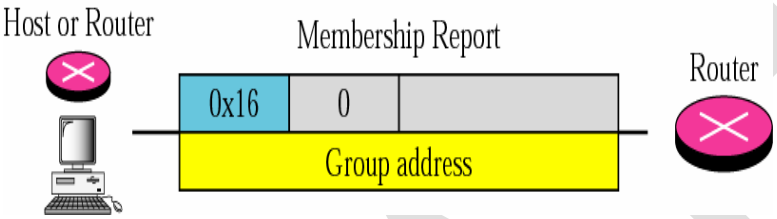
There are two types of Membership Queries:

General Query - used to learn which groups have members on an attached network.

Group-Specific Query - used to learn if a specific group has any members on an attached network.

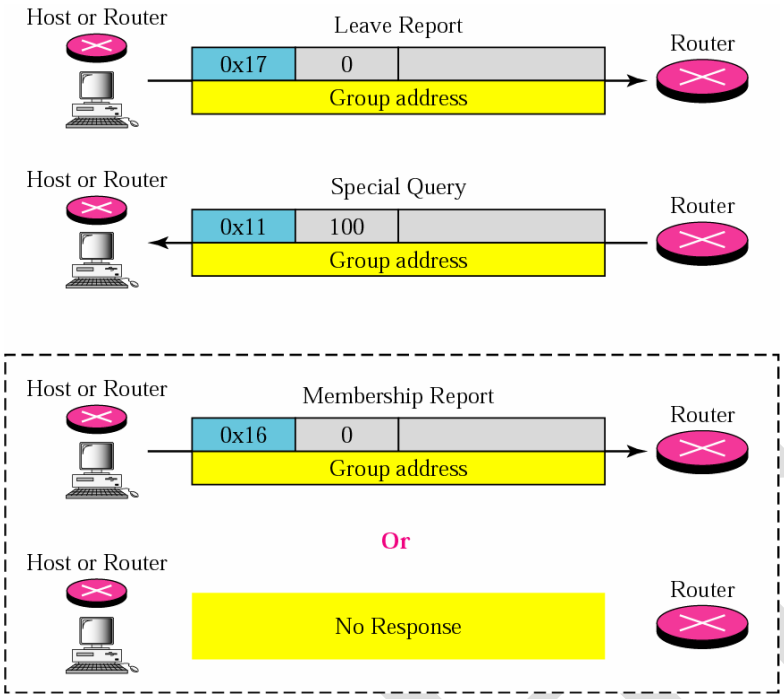
MEMBERSHIP REPORT (v1/v2) [0x12 / 0x16]

A membership report message is sent by a host whenever it joins a multicast group, and when responding to Membership Queries sent by an IGMP router that is functioning as a Querier.



LEAVE GROUP [0x17]

This message is sent when a host leaves a multicast group. This message is sent to the 'all-routers' multicast address of 224.0.0.2. The router then sends out a group-specific membership query to the network to verify if the last member of a group has left.



3. Process to process Communication

Traditionally, the TCP/IP protocol suite has specified two protocols for the transport layer: UDP and TCP. We studied UDP in Chapter11; we will study TCP in this chapter. A new transport-layer protocol called SCTP is now in use by some implementations and will be discussed in Chapter 13.

Figure 12.1 shows the relationship of TCP to the other protocols in the TCP/IP protocol suite. TCP lies between the application layer and the network layer and serves as the intermediary between the application programs and the network operations.

TCP, like UDP, is a process-to-process (program-to-program) protocol. TCP, therefore, like UDP, uses port numbers. Unlike UDP, TCP is a connection-oriented protocol; it creates a virtual connection between two TCPs to send data. In addition, TCP uses flow- and error-control mechanism at the transport level.

In brief, TCP is called a connection-oriented reliable transport protocol. It adds connection-oriented and reliability features to the service of IP.

TCP SERVICES

Before discussing TCP in detail, let us explain the services offered by TCP to the processes at the application layer.

Process-to-Process Communication

Like UDP, TCP provides process-to-process communication using port numbers Table lists some well-known port numbers used by TCP.

Table 12.1 Well-known ports used by TCP

Port	*Protocol*	*Description*
7	Echo	Echoes a received datagram back to the sender
9	Discard	Discards any datagram that is received
11	Users	Active users
13	Daytime	Returns the date and the time
17	Quote	Returns a quote of the day
19	Chargen	Returns a string of characters
20	FTP, Data	File Transfer Protocol (data connection)
21	FTP, Control	File Transfer Protocol (control connection)
23	TELNET	Terminal Network
25	SMTP	Simple Mail Transfer Protocol
53	DNS	Domain Name Server
67	BOOTP	Bootstrap Protocol
79	Finger	Finger
80	HTTP	Hypertext Transfer Protocol
111	RPC	Remote Procedure Call

Example 1

Each line in this file gives the name of the server and the well-known port number. We can use the grep utility to extract the line corresponding to the desired application. The following shows the ports for FTP.

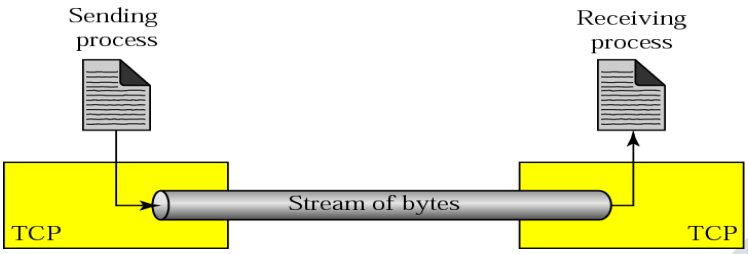
Stream Delivery Service

TCP, unlike UDP, is a stream-oriented protocol. IN UDP, a process (an application program) sends message, with predefined boundaries, to UDP for delivery. UDP adds its own header to each of these message and delivers it to IP for transmission. Each message from the process is called a user datagram, and becomes, eventually, one IP datagram. Neither IP or UDP recognizes any relationship between the datagrams.

TCP, on the other hand, 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 process seem to be connected by an imaginary “tube” that carries their data across the Internet. This

imaginary environment is depicted in Figure 12.2. The sending process produces (writes to) the stream of bytes and the receiving process consumes (reads from) them.

Figure 12.2 Stream delivery

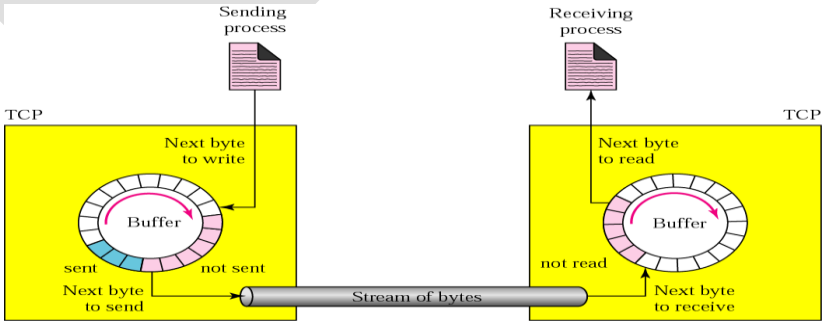


Sending and Receiving Buffers

Because the sending and the receiving process may not write or read at the same speed, TCP needs buffers for storage. There are two buffers, the sending buffer and the receiving buffer, one for each direction. (We will see later that these buffers are also necessary for flow- and error-control mechanism used by TCP). One way to implement a buffer is to use a circular array of 1-byte locations as shown in Figure 12.3. For simplicity, we have shown two buffers of 20 bytes each; normally the buffer are hundreds or thousands of bytes, depending on the implementation. We also show the buffers as the same size, which is not always the case.

The figure shows the movement of the data in one direction. At the sending site, the buffer has three types of chambers. The white section contains empty chambers that can be filled by the sending process (producer). The gray area holds bytes that have been sent but not yet acknowledged. TCP keeps these bytes in the buffer until it receives an acknowledgement. The colored area contains bytes to be sent by the sending TCP. However, as we will see later in this chapter, TCP may be able to send only part of this colored section.

Figure 12.3 Sending and Receiving buffers



The operation of the buffer at the receiver site is simpler. The circular buffer is divided into two areas (shown as white and colored). The white contain empty chambers to be filled by the receiving process. When a byte is read by the receiving process, the chambers is recycled and added to the pool of empty chambers.

Segments

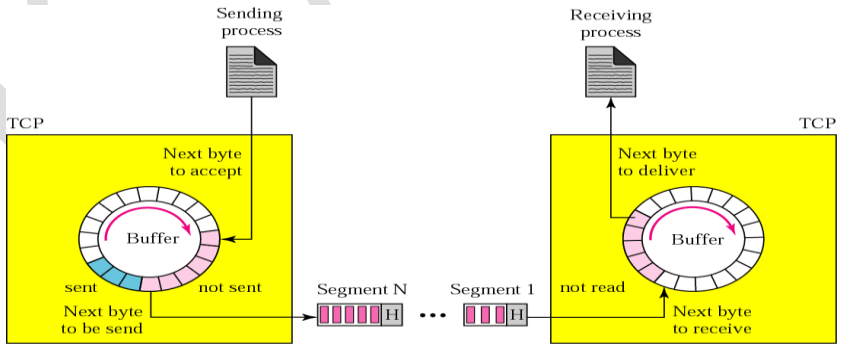
Although buffering handles the disparity between the speed of the producing and consuming process, we need one more step before we can send data. The IP layer, as a service provider for TCP, needs to send data in packets, not as a stream of bytes. At the transport layer, TCP groups a number of bytes together into a packet called segment. TCP adds a header to each segment (for control purpose) and delivers the segment to the IP transmission. The segments are encapsulated in an IP datagram and transmitted. This entire operation is transparent to the receiving process. Later we will see that segments may be received out of order, lost, or corrupted and resent. All of these are handled by TCP with the receiving process unaware of any activities. Figure 12.4 shows how segments are created from the bytes in the buffers.

Note that the segments are not necessary the same size. In the figure, for simplicity, we show one segment carrying 3 bytes and the other carrying 5 bytes. In reality segments carry hundreds if not thousands of bytes.

Full-Duplex Communication

TCP offers **full-duplex service,** where date can flow both directions at the same time. Each TCP then has a sending and the receiving buffer and segments move in both directions.

Figure 12.4 TCP Segments



Connection-Oriented Service

TCP, unlike UDP, is a connection-oriented protocol. When a process at site A wants to send and receive data from another process at the site B, the following occurs.

1. The two TCPs establish a connection between them.

2.Data are exchanged in both directions.

3.The connection is terminated

Note that this is a virtual connection, not a physical connection. The TCP segment is encapsulated in an IP datagram and can be sent out of order, or lost, or corrupted, and then resent. Each may use a different path to reach the destination. There is no physical connection. TCP creates a stream-oriented environment in which it accepts the responsibility of delivering the bytes in order to the other site. The situation is similar to creating a bridge that spans multiple islands ans passing all of the bytes from one island to another in one single connection. We will discuss this feature later in the chapter.

Reliable Service.

TCP is a reliable transport protocol. It uses an acknowledgement mechanism to check the safe and sound arrival of data. We will discuss this feature further in the section on error control.

TCP FEATURES

To provide the services mentioned in the previous section, TCP has several features that are briefly summarized in this section and discussed later in detail.

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 the byte number and not the segment number.

Byte Number

TCP numbers all data bytes that are transmitted in a connection. Numbering is independent direction. When TCP receives bytes of data from a process it stores them in the sending buffer and numbers them. The numbering does not necessarily start from 0. Instead, TCP generates a random number between 0 and 2^32-1 for the number of the first byte. For example, if the random number happens to be 1,057 and the total data to be sent is 6,000 bytes, the bytes are numbered from 1,057 to 7,056. We will see that byte numbering is used for flow and error control.

Sequence Number

After the bytes have been numbered, TCP assigns a sequence number to each segment that is being sent. The sequence number for each segment is the number of the first byte carried in that segment.

Example 2

Suppose a TCP connection is transferring a file of 5000 bytes. The first byte is numbered 10001. What are the sequence numbers for each segment if data is sent in five segments, each carrying 1000 bytes?

Solution

The following shows the sequence number for each segment:

Segment 1→Sequence number:10,001(range:10,001 to 11,000)

Segment 2→Sequence number:11,001(range:11,001 to 12,000)

Segment 3→Sequence number:12,001(range:12,001 to 13,000)

Segment 4→Sequence number:13,001(range:13,001 to 14,000)

Segment 5→Sequence number:13,001(range:14,001 to 15,000)

When a segment carries a combination of data and control information (piggybacking), it uses a sequence number. If a segment does not carry user data, it does not logically define a sequence number. The field is there, but the value is not valid. However, some segments, when carrying only control information need a sequence number to allow an acknowledgement from the receiver. These segments are used for connection establishment, termination, or abortion. Each of these segments consume one sequence number as though it carries one byte, but there is no actual data. If the random generated sequence number is x, the first data byte number is x+1. The byte x is considered as phony byte that is used for a control segment to open a connection, as we will see shortly.

Acknowledgement Number

As we discussed previously, communication in TCP is full duplex; when a connection is established, both parties can send and receive data at the same time. Each party numbers the bytes, usually with a different starting byte number. The sequence number in each direction shows the number of the first byte carried by the segment. Each party also uses an acknowledgement number to confirm the bytes it has received.

However, the acknowledgement number defines the number of the next byte that the party expects to receive. In addition, the acknowledgement number is cumulative here means that if a party uses 5,643 as an acknowledgement number, it has received all bytes from the beginning upto 5,642. Notr that this does not mean that the party has received 5,642 bytes because the first byte number does not have to start from 0.

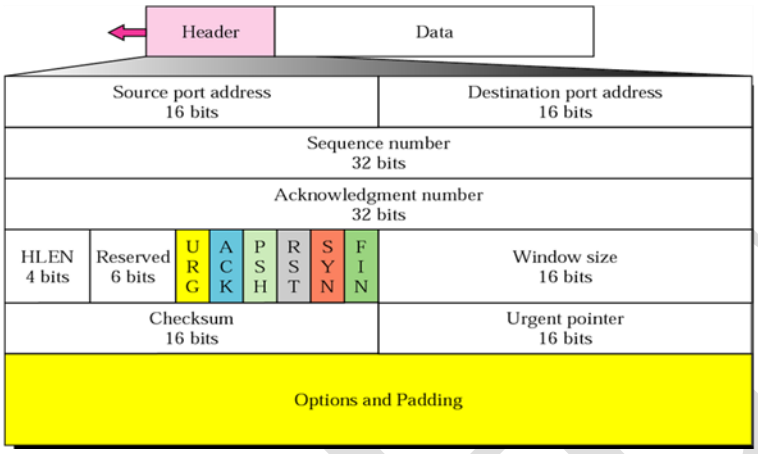
Flow Control

TCP, unlike UDP, provides flow control. The receiver of the data controls how much data are to be sent by the sender. This is done to prevent the receiver from being overwhelmed with data. The numbering system allows TCP to use a byte-oriented flow control.

SEGMENT

Before discussing TCP in more detail, let us discuss the TCP packets themselves. A packet in TCP is Calls a **segment.**

Figure 12.5 TCP segment format

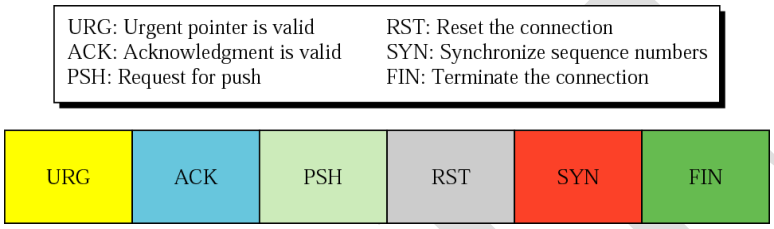


The segment consists of a 20- to 60-byte header, follwed by data from the application program. The header is 20 bytes if there are no options and up to 60 bytes if it contains options. We will discuss some of the header fields in this section. The meaning and purpose of these will become clearer as we proceed through the chapter.

- **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. This serves the same purpose as the source port address in the UDP header discussed in Chapter 11.
- **Destination 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. This serves the same purpose as the destination port address in the UDP header discussed in Chapter 11.
- **Sequence number.** This is a 32-bit field defines the number assigned to the first byte of data contained in this segment. As we said before, TCP is a stream transport protocol. To ensure connectivity, each byte to be transmitted is numbered. The sequence number tells the destination which byte in this sequence compromises the first byte in the segment. During connection establishment (see Section 12.12) each party uses a random number generator to create an **initial sequence number** (ISN), which is usually different in each direction.
- **Acknowledgement 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 the byte number x from the other party, it defines x+1 as the acknowledgement number. Acknowledgement 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 can be between 5 (5*4 = 20) and 15 (15 *4=60).
- **Reserved.** This is a 6-bit field reserved for future use.
- **Control**. This field defines 6 different control bits or flags as shown in Figure 12.6.One or more of these bits can be set at a time.

Figure 12.6 Control field.



These bits enable the flow control, connection establishment and termination, connection abortion, and the mode of data transfer in TCP. A brief description of each bit is shown in table 12.2. We will discuss them further when we study the detailed operation of TCP later in the chapter.

Table 12.2 Description of flags in the control field.

Flag	*Description*
URG	The value of the urgent pointer field is valid
ACK	The value of the acknowledgment field is valid
PSH	Push the data
RST	The connection must be reset
SYN	Synchronize sequence numbers during connection
FIN	Terminate the connection

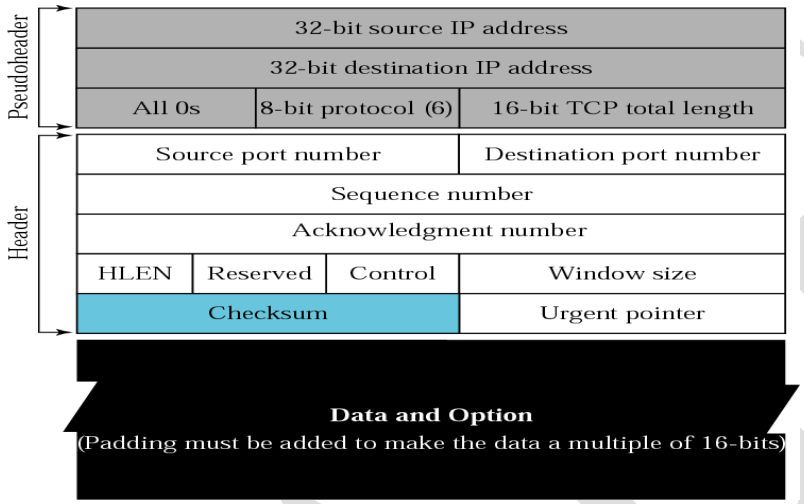
- **Window size.** This field defines the size of the window, in bytes, that the other party must maintain. Note that the length of this field is 16 bits, which means that the maximum size of window is 65,535 bytes. This value is normally referred to as the receiving window (rwnd) and is determined by the receiver. The sender must obey the dictation of the receiver in this case.
- **Checksum.** This 16-bit field contain the checksum. The calculation of the checksum for TCP follows the same procedure as the one described for UDP in Chapter 11. However, the inclusion of the checksum in the UDP datagram is optional, whereas the inclusion of the checksum for TCP is mandatory. The same pseudoheader, serving the same purpose, is added to the segment. For the TCP pseudoheader, the value for the protocol field is six. See Figure 12.7.

Urgent pointer.

This 16-bit field, which is valid only if the urgent flag is set, is used when the segment contains urgent data. It defines the number that must be added the sequence number to obtain the number of the 1 ast urgent byte in the data section of the segment. This will be discussed later in this chapter.

- **Options.** There can be up to 40 bytes of optional information in the TCP header. We will discuss the different options currently used in the TCP header later in the chapter.

Figure Pseudoheader added to the TCP datagram



Encapsulation

A TCP segment is encapsulated in an IP datagram, which in turn is encapsulated in a frame at the data-link layer as shown in Figure 12.8

Figure Encapsulation and decapsulation

TCP is a connection oriented. A connection-oriented transport protocol establishes a virtual path between the source and destination. All of the segments belonging to a message are then sent over this virtual path. Using a single virtual pathway for the entire message facilities the acknowledgement process as well as retransmission of damaged or lost frames. You may wonder how TCP, which uses the services of IP, a connectionless protocol, can be connection-oriented. The point is that a TCP uses the services of IP to deliver individual segments to the receiver, but it controls the connection itself. If a segment is lost or corrupted, it is retransmitted. Unlike TCP, IP is unaware of this retransmission. If a segment out of order, TCP holds it

until the missing segments arrive; IP is unaware of this reordering. 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 is transferred.

The "three-way handshake" happens thus. The originator (you, hopefully) sends an initial packet called a "SYN" to establish communication and "synchronize" sequence numbers in counting bytes of data which will be exchanged. The destination then sends a "SYN/ACK" which again "synchronizes" his byte count with the originator and acknowledges the initial packet. The originator then returns an "ACK" which acknowledges the packet the destination just sent him. The connection is now "OPEN" and ongoing communication between the originator and the destination are permitted until one of them issues a "FIN" packet, or a "RST" packet, or the connection times out. All the protocols of the Internet which need "connections" are built on the TCP protocol. The "three way handshake" establishes the communication. Much like you picking up your phone, getting a dial tone, dialing the number, hearing ringing, and then the other party saying "hello" or "mushi mushi."

UDP is the other major underlying communication protocol of the Internet (besides TCP) - but it does not use a handshake to establish a "connection." It is much like a letter dropped in the mail. There are no guarantees, and the post office makes a "best effort" to deliver the letter. But there is no final check to guarantee that the letter made it to its final destination. Of course there is a variant of the US mail which does this - registered mail.

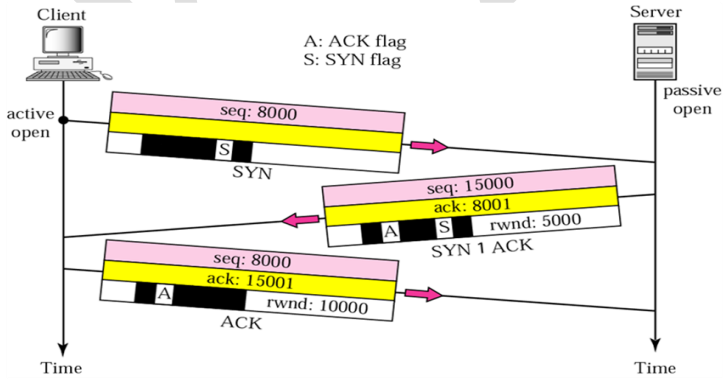
Operating behind a NAT/router has special challenges for UDP packets. A UDP IP packet just suddenly shows up at the router's doorstep, much like a letter, and it must try to decide whether to

let it pass or not. Outgoing UDP packets are permitted pretty much without question, but incoming UDP packets (unsolicited UDP) are typically not permitted unless they correspond to previous UDP packets outgoing. The IP address is used to see if it is one that was addressed in a previous outgoing packet. There is then a timer placed on this; if a UDP packet is not seen on this connection for a certain time, then the router reverts to denying the incoming UDP.

Now all the more familiar services of the Internet, such as web browsing (HTTP), email (POP3 and SMTP typically), FTP, Telnet, etc. etc. are built on top of the TCP and UDP protocols. And of course TCP and UDP are built on top of the fundamental IP protocol. IP protocols contain the IP addresses, TCP and UDP protocols contain the service addresses, to summarize it simply.

IP packets are the fundamental "currency of the Internet. If you were to look under the covers of the Internet, you would see "IP packets" in the most general sense. At different places and different levels, you may see "ethernet frames" or "ATM cells" or "Frame Relay packets" - but these are just ways to carry the fundamental currency which is IP packets. The thing that makes the IP packet the fundamental currency of the Internet is that the IP packet contains two "Internet addresses," or "IP addresses."

Figure 12.9 *Connection establishment using three-way handshaking*



TCP/IP Protocol Suite 22

Connection Establishment Functions

The connection establishment process actually accomplishes several things as it creates a connection suitable for data exchange:

- **Contact and Communication:** The client and server make contact with each other and establish communication by sending each other messages. The server usually doesn't even know what client it will be talking to before this point, so it discovers this during connection establishment.
- **Sequence Number Synchronization:** Each device lets the other know what initial sequence number it wants to use for its first transmission.
- **Parameter Exchange:** Certain parameters that control the operation of the TCP connection are exchanged by the two devices.
- TCP Connection Establishment Sequence Number Synchronization and Parameter Exchange (Page 1 of 3)
- The TCP three-way handshake describes the mechanism of message exchange that allows a pair of TCP devices to move from a closed state to a ready-to-use, established connection. Connection establishment is about more than just passing messages between devices to establish communication.
- The TCP layers on the devices must also exchange information about the sequence numbers each device wants to use for its first data transmission, as well as parameters that will control how the connection operates.
- The former of these two data exchange functions is usually called sequence number synchronization, and is such an important part of connection establishment that the messages that each device sends to start the connection are called SYN (synchronization) messages.
- The Problem With Starting Every Connection Using the Same Sequence Number
- In the example I gave in the topic describing the sliding windows system, I assumed for "simplicity" (ha ha, was that simple?) that each device would start a connection by giving the first byte of data sent sequence number 1. A valid question is, why wouldn't we always just start off each TCP connection by sending the first byte of data with a sequence number of 1? The sequence numbers are arbitrary, after all, and this is the simplest method.
- In an ideal world, this would probably work, but we don't live in an ideal world. J The problem with starting off each connection with a sequence number of 1 is that it introduces the possibility of segments from different connections getting mixed up. Suppose we established a TCP connection and sent a segment containing bytes 1 through 30. However, there was a problem with the internetwork that caused this segment to be delayed, and eventually, the TCP connection itself to be terminated. We then started up a new connection and again used a starting sequence number of 1.

As soon as this new connection was started, however, the old segment with bytes labeled 1 to 30 showed up. The other device would erroneously think those bytes were part of the new connection.

- This is but one of several similar problems that can occur. To avoid them, each TCP device, at the time a connection is initiated, chooses a 32-bit initial sequence number (ISN) for the connection. Each device has its own ISN, and they will normally not be the same.
- Selecting the Initial Sequence Number
- Traditionally, each device chose the ISN by making use of a timed counter, like a clock of sorts, that was incremented every 4 microseconds. This counter was initialized when TCP started up and then its value increased by 1 every 4 microseconds until it reached the largest 32-bit value possible (4,294,967,295) at which point it "wrapped around" to 0 and resumed incrementing. Any time a new connection is set up, the ISN was taken from the current value of this timer. Since it takes over 4 hours to count from 0 to 4,294,967,295 at 4 microseconds per increment, this virtually assured that each connection will not conflict with any previous ones.
- One issue with this method is that it makes ISNs predictable. A malicious person could write code to analyze ISNs and then predict the ISN of a subsequent TCP connection based on the ISNs used in earlier ones. This represents a security risk, which has been exploited in the past (such as in the case of the famous Mitnick attack). To defeat this, implementations now use a random number in their ISN selection process.

TCP Connection Management and Problem Handling, the Connection Reset Function, and TCP "Keepalives"

Once both of the devices in a TCP connection have completed connection setup and have entered the ESTABLISHED state, the TCP software is in its normal operating mode. Bytes of data will be packaged into segments for transmission using the mechanisms described in the section on message formatting and data transfer. The sliding windows scheme will be used to control segment size and to provide flow control, congestion handling and retransmissions as needed.

Once in this mode, both devices can remain there indefinitely. Some TCP connections can be very long-lived indeed—in fact, some users maintain certain connections like Telnet sessions for hours or even days at a time. There are two circumstances that can cause a connection to move out of the ESTABLISHED state:

Connection Termination: Either of the devices decides to terminate the connection. Connection Disruption: A problem of some sort occurs that causes the connection to be interrupted.

The TCP Reset Function

To allow TCP to live up to its job of being a reliable and robust protocol, it includes intelligence that allows it to detect and respond to various problems that can occur during an established connection. One of the most common is the half-open connection. This situation occurs when due to some sort of problem, one device closes or aborts the connection without the other one knowing about it. This means one device is in the ESTABLISHED state while the other may be in the CLOSED state (no connection) or some other transient state. This could happen if, for example, one device had a software crash and was restarted in the middle of a connection, or if some sort of glitch caused the states of the two devices to become unsynchronized.

To handle half-open connections and other problem situations, TCP includes a special reset function. A reset is a TCP segment that is sent with the RST flag set to one in its header. Generally speaking, a reset is generated whenever something happens that is "unexpected" by the TCP software. Some of the most common specific cases in which a reset is generated include:

Receipt of any TCP segment from any device with which the device receiving the segment does not currently have a connection (other than a SYN requesting a new connection.)

Receipt of a message with an invalid or incorrect Sequence Number or Acknowledgment Number field, indicating the message may belong to a prior connection or is spurious in some other way.

Receipt of a SYN message on a port where there is no process listening for connections.

Requirements and Issues In Connection Termination

Just as TCP follows an ordered sequence of operations to establish a connection, it includes a specific procedure for terminating a connection. As with connection establishment, each of the devices moves from one state to the next to terminate the connection. This process is more complicated than one might imagine it needs to be. In fact, an examination of the TCP finite state machine shows that there are more distinct states involved in shutting down a connection than in setting one up.

The reason that connection termination is complex is that during normal operation, both of the devices are sending and receiving data simultaneously. Usually, connection termination begins with the process on just one device indicating to TCP that it wants to close the connection. The matching process on the other device may not be aware that its peer wants to end the connection at all. Several steps are required to ensure that the connection is shut down gracefully by both devices, and that no data is lost in the process.

Ultimately, shut down of a TCP connection requires that the application processes on both ends of the connection recognize that "the end is nigh" for the connection and stop sending data. For this reason, connection termination is implemented so that each device terminates its end of the connection separately.

The act of closing the connection by one device means that device will no longer send data, but can continue to receive it until the other device has decided to stop sending. This allows all data that is pending to be sent by both sides of the communication to be flushed before the connection is ended

.5. User Datagram Protocol

User Datagram Protocol or UDP is part of the Internet Protocol suite, using which, programs running on different computers on a network can send short messages known as Datagrams to one another. UDP can be used in networks where TCP is traditionally implemented, but unlike TCP, it does not guarantee reliability or the correct sequencing of data. Datagrams may go missing without notice, or arrive in a different order from the one in which they were sent. The protocol was formulated by David P. Reed in 1980 and officially defined in RFC 768.

UDP makes use of a simple communication model without implicit transmission checks for guaranteeing reliability, sequencing, or datagram integrity. Though these factors might seem to suggest that UDP is not a useful protocol, it still finds wide usage in particular areas where speed, more than reliability, is of utmost importance. UDP considers that error checks and corrections

should be carried out in the communicating application, and not at the network layer. However, if error checks and corrections are needed at the network layer, the application can make use of Transmission Control Protocol (TCP) or Stream Control Transmission Protocol (SCTP), which are specifically formulated for this reason. Since UDP does not have the overhead of checking whether the data has reached the destination every time it is sent, it makes the protocol that much faster and more efficient. UDP is often used for time-sensitive applications where missing data is preferred to late-arriving data.

UDP is a stateless protocol which is useful for servers engaged in answering short queries from a large number of clients. While TCP is mainly used for communication between a server and a single client, UDP is used for packet broadcast or multicasting whereby the data is sent to all the clients in the network. Frequent network applications that use UDP include: Trivial File Transfer Protocol (TFTP) , Voice over IP (VoIP), IPTV, Domain Name System (DNS), etc.

Since UDP lacks any kind of mechanism to control or avoid network congestion, other forms of network-based control mechanisms need to be implemented to ensure smooth flow of traffic in a UDP network. One of the solutions being designed to tackle this problem is DCCP or Datagram Congestion Control Protocol which is aimed at monitoring and controlling traffic congestion in a UDP network.

A typical IP network consists of five layers:

The Physical Layer consisting of the actual channel for data flow like coaxial, twisted pair or fiber optic cables

The Data Link Layer implementing Wi-Fi, ISDN, GPRS etc

The Network / Internet Layer

Transport Layer implementing TCP, UDP etc

Application Layer running DNS, FTP, HTTP, POP3, SMTP, Telnet etc

As shown above, UDP belongs to the fourth layer. Although the entire amount of UDP traffic in a network is a small fraction of the whole, a number of key application in the fifth layer like DNS and SNMP or simple network management protocol use UDP.

UDP Packet

The UDP header comprises of only four fields. The use of two of those is optional (light red background in diagram).

Bits	0 - 15	16 - 31
0	Source Port	Destination Port
32	Length	Checksum
64	Data	

Source Port

Source port recognizes the sending port and should be understood to be the port to respond to if required. If not used, then its value should be zero.

DestinationPort

Destination port recognizes the destination port and is mandatory.

Length

A 16-bit length field indicates the length in bytes of the complete datagram: header and data.

Checksum

The 16-bit checksum field is implemented for error-checking of the header and data. The algorithm for computing the checksum is different for transmission over IPv4 and IPv6.

6. **OSPF Protocol**

Open Shortest Path First (OSPF) is a robust link-state interior gateway protocol (IGP). People use OSPF when they discover that Routing Information Protocol (RIP) just isn't going to work for their larger network, or when they need very fast convergence.

OSPF is the most widely used IGP. When we discuss IGPs, we're talking about one routing domain, or Autonomous System (AS). Imagine a medium-sized company with multiple buildings and departments, all connected and sharing two redundant Internet links. All of the buildings on-site are part of the same AS. With OSPF, however, we also have the concept of an Area, which allows further segmentation, perhaps by department in each building.

To understand the design needs for areas in OSPF, let's start with discussing how OSPF works. There is some terminology you may not have encountered before, including the following:

- Router ID: In OSPF this is a unique 32-bit number assigned to each router. This is chosen as the highest IP address on a router, and can be set large by configuring an address on a loopback interface of the chosen router.
- Neighbor Routers: two routers with a common link that can talk to each other.
- Adjacency: a two-way relationship between two neighbor routers. Neighbors don't always form adjacencies.
- LSA: Link State Advertisements are flooded; they describe routes within a given link.
- Hello Protocol: This is how routers on a network determine their neighbors and form LSAs.
- Area: a hierarchy. A set of routers that exchange LSAs, with others in the same area. Areas limit LSAs and encourage aggregate routes.

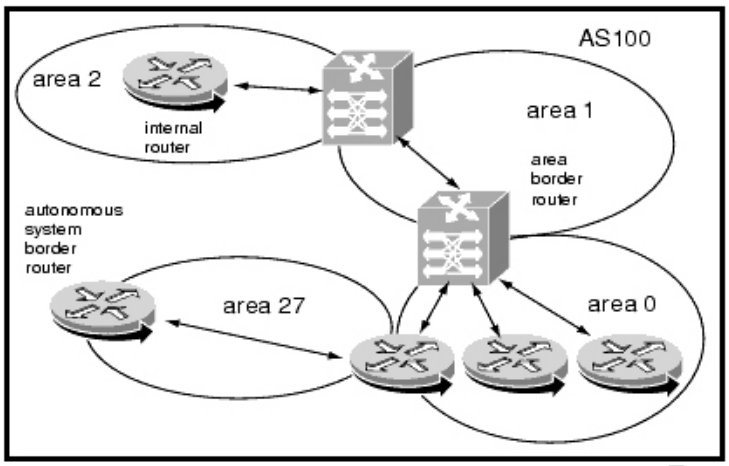
- OSPF is a link-state routing protocol, as we've said. Think of this as a distributed map of the network. To get this information distributed, OSPF does three things.

OSPF is a link-state routing protocol, as we've said. Think of this as a distributed map of the network. To get this information distributed, OSPF does three things.

First, when a router running OSPF comes up it will send hello packets to discover its neighbors and elect a designated router. The hello packet includes link-state information, as well as a list of neighbors. Providing information about your neighbor to that neighbor serves as an ACK, and proves that communication is bi-directional. OSPF is smart about the layer 2 topology: if you're on a point-to-point link, it knows that this is enough, and the link is considered "up." If you're on a broadcast link, the router must wait for an election before deciding if the link is operational.

The election ballot can be stuffed, with a Priority ID, so that you can ensure that your beefiest router is the Designated Router (DR). Otherwise, the largest IP address wins. The key idea with a DR and backup DR (BDR) is that they are the ones to generate LSAs, and they must do database exchanges with other routers in the subnet. So, non-designated routers form adjacencies with the DR. The whole DR/BDR design is used to keep the protocol scalable. The only way to ensure that all routers have the same information is to make them synchronize their databases. If you have 21 routers, and want to bring another one up, then you'd have to form 21 new adjacencies. If you centralize the database, with a backup (just in case), then adding more becomes an easy to manage linear problem.

The database exchange is part of bringing up adjacencies after the hello packets are exchanged, and it's very important. If the databases are out of sync, we could risk routing loops, blackholes and other perils. The third part of bringing up an adjacency is Reliable Flooding, or LSA exchange. The LSA area zero is special, and if you have multiple areas, they must all touch area zero. This is also called the Backbone Area. There are different types of areas in OSPF, and it can get really crazy when you throw in Virtual Links to allow two areas to speak without hitting area zero.



There also are different types of routers in OSPF:

- ABR: An Area Border Router is a router that is in area zero, and one or more other areas.
- DR, BDR: A Designated Router, as we said, is the router that keeps the database for the subnet. It sends and receives updates (via multicast) from the other routers in the same network.
- ASBR: The Autonomous System Boundary Router is very special, but confusing. The ASBR connects one or more AS, and exchanges routes between them. The ASBR's purpose is to redistribute routes from another AS into its own AS.

7. Poins to Remember

- Unicast is the process of forwarding unicasted traffic from a source to a destination on network.
- Modern computer networks generally use dynamic routing algorithms rather than the static ones
- The distance vector routing algorithm is sometimes called by other names, most commonly the distributed Bellman-Ford routing algorithm and the Ford-Fulkerson algorithm
- Internet Group management protocol (IGMP), a multicasting protocol in the internet protocols family
- TCP provides process-to-process communication using port numbers
- UDP, provides flow control
- TCP transmits data in full-duplex mode
- User Datagram Protocol or UDP is part of the Internet Protocol suite, using which, programs running on different computers on a network can send short messages known as Datagrams to one another.
- Open Shortest Path First (OSPF) is a robust link-state interior gateway protocol (IGP).

Possible Questions

Part B

(Each Question carries 6 Marks)

1. Write about the operation of distance vector routing.
2. Write about BGP sessions and packet types.
3. Give a brief description about RIP message format.
4. Discuss about multi cast routing and its applications
5. Discuss about IGMP Operations.
6. Explain Flat & Hierarchical Name Space
7. Discuss about links in OSPF.
8. Explain with an example process to process communication.
9. Explain path vector routing with an example.
10. Discuss in detail about Interior and Exterior Gateway Protocols

Part C

(Each Question carries 10 Marks)

1. Discuss about building of routing table and formation of shortest path tree in link state routing.

Karpagam Academy of Higher Education
Coimbatore - 21

DEPARTMENT OF COMPUTER APPLICATIONS
CLASS : II MCA

TCP/IP (18CAP405N)

Unit III

S.NO	Question	Choice1	Choice2	Choice3	Choice4	Choice 5	Choice 6
1	A _____ server gets its information from a primary server	root server	DNS server	BOOTP server	secondary server		
2	_____ server creates maintains and updates information about its zone	secondary	primary	root	BOOTP		
3	The domain name space is divided into _____ sections.	2	4	3	6		
4	The DNS client alled a _____maps a name to a address or an address to a name	resolver	register	server	client		
5	IN _____ resolution the client sends its request to a server	Iterative	recursive	non recursive	non iterative		
6	In _____resolution the client may send its request to multiple servers	iterative	recursive	non recursive	non iterative		
7	Two types of DNS messages are _____	questions and	Question and	queries and	resources and		
8	_____ is a method where by an answer to query is stored in memory	queuing	stacking	querying	caching		
9	DNS uses an __pointer for duplicated domain name information in its message	offset	inset	static	dynamic		
10	A_____ maps each address to a unique name	address space	domain space	offset pointer	name space		
11	_____ is a sequence of characters without structures	hierarchal name space	flat name space	address name	domain name		
12	In _____ name space the names are defined in a inverted tree structure	hierarchical	flat	address	domain		

13	_____ is a sting with the maximum of 63 characters	label	domain name	FQDN	domain name		
14	The tree can have only _____levels	126	127	128	138		
15	_____ is a sequence of labels separated by dots	domain name	FQDN	address space	PQDN		
16	_____ is a domain name that contains the full name of a host	PQDN	FQDN	QQDN	AQDN		
17	A _____ is sub tree of the domain name space	zone	PQDN	domain	FQDN		
18	____ is used when the name to be resolved belongs to the same site as client	PQDN	FQDN	address space	name space		
19	What a server is responsible for or has authority over is called a _____	segment	zone	domain	packed		
20	A _____is a server whose zone consists of the whole tree	primary server	secondary server	com server	root server		
21	The _____ define registered hosts according to their generic behavior	country domain	generic domain	inverse domain	net domain		
22	The _____section uses two character country abbreviations	country domain	generic domain	country domain	net domain		
23	_____ domain is used to map an address to a name	inverse	generic	country	net		
24	The new domains are added to the DNS by a _____	resolver	registrar	server	client		
25	_____record is used by the client to get information from a server	question	answer	query	authoritative		
26	Mapping a name to an address or an address to a name is called ____resolution	domain address	client address	name address	server address		
27	___ allows organizations to use the global internet for private and public communication.	VPN	DNS	FTP	SNMP		
28	A common technique to encrypt and authenticate in VPN is _____ .	internet security	IP security	network security	web security		

29	____ involves the encapsulation of an encrypted IP datagram in a second outer datagram.	Ipsec	VPN	Tunneling	Trimming		
30	Mobile IP is an enhanced version of _____	TCP	IP	ARP	FTP		
31	The protocol that binds IP address and physical address is _____.	RARP	ATMWAN	ATMRP	ATMARP		
32	_____ is a cell- switched network that can be a highway for an IP datagram.	ATM	SNMP	DNS	DHCP		
33	_____ router receives an IP datagram.	exit-point	middle-point	entering-point	ending-point		
34	_____ connection is established between two end points by the network provider.	PVC	TPA	SHA	VPN		
35	In a _____ connection each time a router wants to make a connection	PVC	SVC	SHA	VPN		
36	An ATM network can be divided into _____ subnetworks.	physical	dynamic	static	logical		
37	In mobile IP the host has its orginal address called _____ address.	physical	logical	home	care-of		
38	In mobile IP the host has its temporary address is called _____ address.	home	care of	physical	logical		
39	In mobile IP the care of address is associated with the ____ network.	home	LAN	foreign	WAN		
40	_____ is usually a router attached to the home network of the mobile host.	home agent	foreign agent	mobile agent	public agent		
41	_____ is usually a router attached to the foreign network of the mobile host.	home agent	foreign agent	mobile agent	public agent		
42	When the mobile host act as a foreign agent called a _____ care of addresss.	located	allocated	co-located	migrated		
43	The first phase in mobile communication is _____ discovery.	agent	user	server	client		
44	Mobile IP uses the router solicitation packet of _____.	IMCP	MICP	MCIP	ICMP		

45	_____ messages are encapsulated in a UDP user datagram.	resolution	registration	extension	identification		
46	_____ network private internet and access to the global internet.	hybrid	VPN	private	public		
47	The actual mail transfer is done through _____.	SDA	TTA	MTA	MTP		
48	The protocol that defines the MTA client and server in the internet is called __	SNMP	MTP	FTP	SMTP		
49	SMTP defines _____ commands.	13	14	24	15		
50	_____ is a TCP/IP client- server application for copying files	FTP	SMTP	SNMP	ICMP		
51	FTP uses _____ well- known TCP ports.	3	2	4	5		
52	In FTP port _____ is used for the control connection.	20	21	22	23		
53	In FTP port _____ is used for data connection.	20	21	22	23		
54	In FTP the client sents port number to the server using _____ command.	PASV	PORT	LIST	OPEN		
55	In FTP the control connection is made between _____ processes.	control	data transfer	data connectio	data store		
56	In FTP the data connection is made between _____ processes.	control	data transfer	data connectio	data store		
57	_____ is a general purpose client-server program.	NSFNET	CSNET	TELNET	ARPANET		
58	When a user logs into a local time sharing system it is called _____.	remote login	local login	security login	global login		
59	When a user wants to access an application program he performs _____.	remote login	local login	security login	global login		
60	___ driver pretends the characters are coming from a terminal.	non terminal	pseudo termainal	remote terminal	local termaina		

Answer
secondary server
primary
3
resolver
recursive
iterative
queries and response
caching
offset
name space
flat name space
domain

label
128
domain name
FQDN
PQDN
FQDN
zone
root server
generic domain
country domain
inverse
registrar
question
name address
VPN
IP security

tunneling
IP
ATMARP
ATM
entering-point
PVC
SVC
logical
home
care of
foreign
home agent
foreign agent
co-located
agent
ICMP

registration
hybrid
MTA
SMTP
14
FTP
2
21
20
PORT
control
data transfer
CSNET
local login
remote login
pseudo termainal

UNIT-IV

SYLLABUS

UDP Operation – TCP Services - Flow Control – Multicast Routing – Multicast Routing Protocols. BOOTP - DHCP – Address Discovery and Binding. DNS – Name Space – DNS in Internet – Resolution – Resource Records

UDP

Well-known Ports for UDP

Table 11.1 shows some well-known port numbers used by UDP. Some port numbers can be used by both UDP and TCP.

Table 11.1 Well-known ports used with UDP

Port	*Protocol*	*Description*
7	Echo	Echoes a received datagram back to the sender
9	Discard	Discards any datagram that is received
11	Users	Active users
13	Daytime	Returns the date and the time
17	Quote	Returns a quote of the day
19	Chargen	Returns a string of characters
53	Nameserver	Domain Name Service
67	Bootps	Server port to download bootstrap information
68	Bootpc	Client port to download bootstrap information
69	TFTP	Trivial File Transfer Protocol
111	RPC	Remote Procedure Call
123	NTP	Network Time Protocol
161	SNMP	Simple Network Management Protocol
162	SNMP	Simple Network Management Protocol (trap)

EXAMPLE 1

In UNIX, the well-known ports are stored in a file called /etc/services. Each line in this file gives the name of the server and the well-known port number. We can use the grep utility to extract the line corresponding to the desired application. The following shows the port for TFTP. Note TFTP can use port 69 on either UDP or TCP.

```
$ grep tftp /etc/services
tftp        69/tcp
tftp        69/udp
```

See Next Slide

EXAMPLE 1 (CONTINUED)

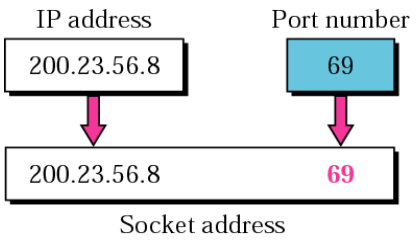
SNMP uses two port numbers (161 and 162), each for a different purpose, as we will see in Chapter 21.

```
$ grep snmp /etc/services
snmp        161/tcp     #Simple Net Mgmt Proto
snmp        161/udp     #Simple Net Mgmt Proto
snmptrap    162/udp     #Traps for SNMP
```

Socket Addresses

As we have seen, UDP needs two identifiers, the IP address and the port number, at each end to make a connection. The combination of an IP address and a port number is called a **socket address**. The client socket address defines the client process uniquely just as the server socket address defines the server process uniquely(see figure 11.6).

Figure 11.6 *Socket address*

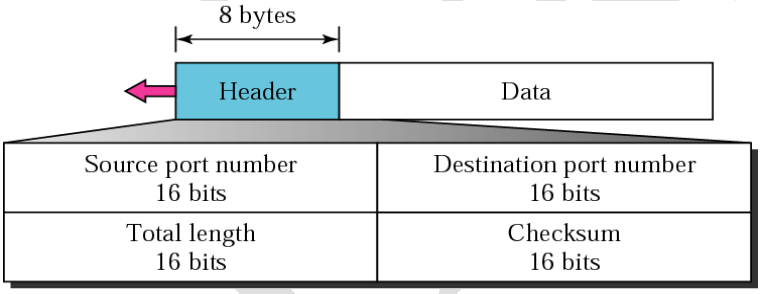


To use the services of UDP, we need a pair of socket addresses: the client socket address and the server socket address. These four pieces of information are part of the IP header and the UDP header. The ip header contains the IP addresses; the UDP header contains the port numbers.

USER DATA GRAM

UDP packets, called **user datagrams**, have a fixed-size header of 8 bytes. Figure 11.7 shows the format of a user datagram.

Figure 11.7 *User datagram format*



- **Source port number.** This is the port number used by the process running on the source host. It is 16 bits long, which means that the port number can range from 0 t0 65,535. If the source host is the client (a client sending a request), the port number, in most cases, is an ephemeral port number requested by the process and chosen by the UDP software running on the source host. If the source host is the server (a server sending a response), the port number, in most cases, is a well-known port number.
- **Destination port number.** This is the port number used by the process running on the destination host. It is also 16 bits long. If the destination host is the server (a client sending a request), the port number, in most cases,is well-known port number. If the destination host is

the client (a server sending a response), the port number, in most cases, is an ephemeral port number. In this case the server copies the ephemeral port number it has received in the request packet.

The fields are as follows:

- **Source port number.** This is the port number used by the process running on the source host. It is 16 bits long, which means that the port number can range from 0 t0 65,535. If the source host is the client (a client sending a request), the port number, in most cases, is an ephemeral port number requested by the process and chosen by the UDP software running on the source host. If the source host is the server (a server sending a response), the port number, in most cases, is a well-known port number.
- **Destination port number.** This is the port number used by the process running on the destination host. It is also 16 bits long. If the destination host is the server (a client sending a request), the port number, in most cases,is well-known port number. If the destination host is the client (a server sending a response), the port number, in most cases, is an ephemeral port number. In this case the server copies the ephemeral port number it has received in the request packet.
- **Length.** This is a 16-bit field that 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 much less because an UDP user datagram is stored in an IP datagram with the total length of 65535 bytes.
- The length field in a UDP user datagram is actually not necessary. A user datagram is encapsulated in an IP datagram. There is a field in the IP datagram that defines the total length. There is another field in the IP datagram that defines the length of the header. So if we subtract the value of the second field from the first,we can deduce the length of the UDP datagram that is encapsulated in an IP datagram.
- However, the designers of the UDP protocol felt that it was more efficient for the destination UDP to calculate the length of the data from the information provided in the UDP user datagram rather than asking the IP software to supply this information. We should remember that when the IP software delivers the UDP user datagram to the UDP layer, it has already dropped the IP header.

- **Checksum.** This field is used to detect errors over the entire user datagram (header plus data). The checksum is discussed in the next section.

- We have already talked about the concept of the checksum and the way it is calculated in chapter 8. we have also shown how to calculate the checksum for the IP and ICMP packet, we now show how this is done for UDP.

- UDP checksum calculation is different from the one for IP and ICMP. Here the checksum includes three sections:a pseudoheader,the UDP header, and the data coming from the application layer.

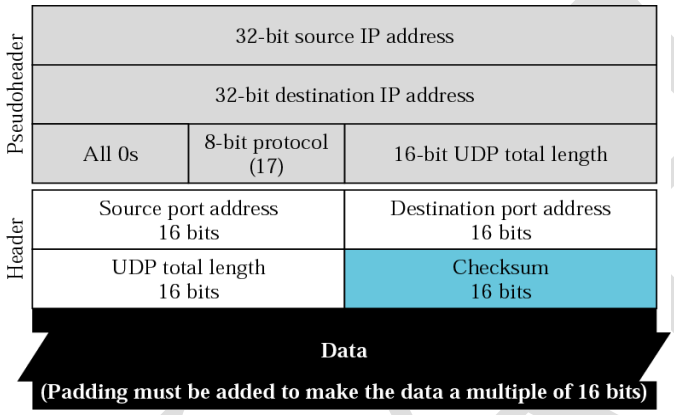
- The pseudoheader is the part of the header of the IP packet in which the user datagram is to be encapsulated with some fields filled with 0s(see Figure 11.8).

- UDP checksum calculation is different from the one for IP and ICMP. Here the checksum 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(see Figure 11.8).

- UDP checksum calculation is different from the one for IP and ICMP. Here the checksum includes three sections:a pseudoheader,the UDP header, and the data coming from the application layer.

Figure 11.8 *Pseudoheader for checksum calculation*



If the checksum does not include the pseudoheader, a user datagram may arrive safe and sound. However, if the IP address is corrupted, if may be delivered to the wrong host.

The protocol field is added to ensure that the packet belongs to UDP, and not to TCP. We will see later that if a process can use either UDP or TCP, the designation port number can be the same. The value of the protocol field for UDP is 17. if this value is changed during transmission, the checksum calculation at the receiver will detect it and UDP drops the packet. It is not delivered to the wrong protocol.

Note the similarities between the pseudoheader fields and the last 12 bytes of the IP header.

Checksum calculation

1. **Checksum calculation at sender**
2. **Checksum calculation at Receiver**

Checksum calculation at sender

The sender follow this eight steps to calculate the checksum:

1. Add the pseudoheader to the UDP user datagram.

2. Fill the checksums fields with zeros.
3. Divide the total bits into 16-bit(2-byte) words.
4. If the total number of bytes even, add 1 byte of padding(all 0s). The padding is only for the purpose of calculating the checksum and will be discarded afterwards.
5. Add all 16-bit sections using one's complement arithmetic.
6. Complement the result(change all 0s to 1s and all 1s to 0s), which is 16-bit number, and insert in the checksum field.
7. Drop the pseudoheader and any added padding.
8. Deliver the UDP user datagram to the IP software for encapsulation.

Note that the order of the rows in pseudoheader doesnot make any difference in checksum calculation. Also, adding 0s doesnot change the result. For this reason, the software that calculates the checksum can easily add the whole IP header (20 bytes) to the UDP datagram, set the first bytes to zero, set the TTL fields to zero, replace the IP checksum to UDP length, and calculate the checksum. The result would be the same.

Checksum Calculation at Receiver

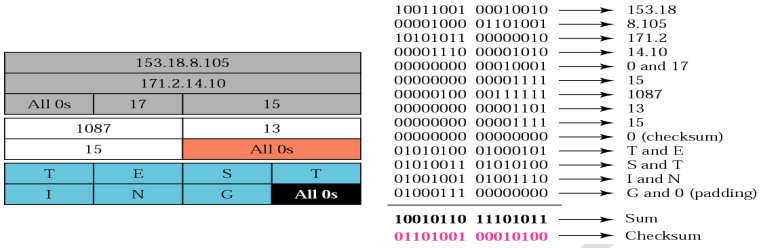
The receiver follows these six steps to calculate the checksum:

1. Add the pseudoheader to the UDP user datagram.
2. Add padding if needed.
3. Divide the total bits into 16-bit sections.
4. Add all 16-bit sections using one's complement arithmetic.
5. Complement the result.
6. If result is all 0s, drop the pseudoheader and any added padding and accept the user datagram. If the result is anything else, discard the user datagram.

An example

Figure 11.9 shows the checksum calculation for a very small user datagram with only 7 bytes of data. Because the number of bytes of data is odd, padding is added for checksum calculation. The pseudoheader as well as the padding will be dropped when the user datagram is delivered to IP.

Figure 11.9 *Checksum calculation of a simple UDP user datagram*



Optional use of the checksum

The calculation of the checksum and its inclusion in a user datagram is optional. If the checksum is not calculated, the field is filled with 0s. one might ask, when the UDP software on the destination computer receives a user datagram with a checksum value of zero, how can it determine if the checksum was not used or if it was used and the result happened to be all 0s? The answer is very simple. If the source does calculate the checksum and the result happens to be all 0s, it must be complemented. So what is sent is all 1s. Note that a calculated checksum can never be all 0s because this implies that the sum is all 1s which is impossible in two's complement arithmetic .

UDP Operation

UDP uses concept common to the transport layer. These concepts will be discussed here briefly, and then expanded in the next chapter on the TCP protocol.

Connectionless Services

As mentioned previously, UDP provides a **connectionless service.** This means that each user datagram sent by UDP is an independent datagram. There is no relationship between the different user datagram's even if they are coming from the same source process and going to the same destination program. The user datagram's are not numbered. Also, there is no connection establishment and no connection termination as is the case for TCP. This means that each user datagram can travel on a different path.

One of the ramifications of being connectionless is that the process that uses UDP cannot send a stream of data to UDP and expect UDP to chop them into different related user data grams. Instead

each request must be small enough to fit into one user datagram. Only those processes sending short messages should use UDP.

Flow and Error Control

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

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

The lack of **flow control and error control** means that the process using UDP should provide for these mechanisms.

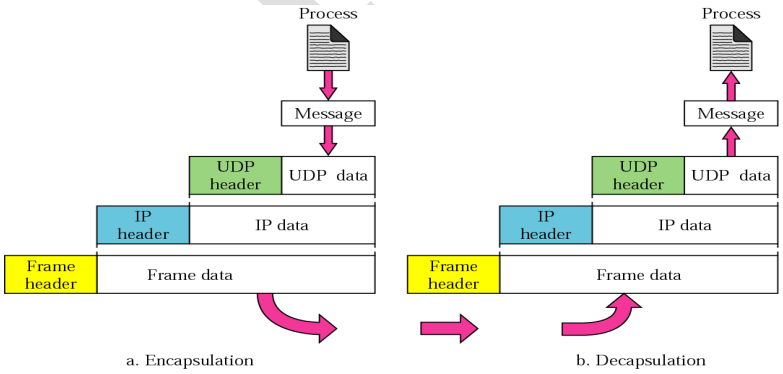
Encapsulation and Decapsulation

To send a message from one process to another, the UDP protocol encapsulates and decapsulates messages (see figure 11.10).

Encapsulation

When a process has a message to send through UDP, it passes the message to UDP along with a pair of socket addresses and the length of data. UDP receives the data and adds the UDP header.

Figure 11.10 *Encapsulation and decapsulation*



UDP then passes the user datagram to IP with the socket addresses. IP adds its own header, using the value 17 in the protocol field,indicating that the data has come from the UDP protocol. The IP datagram is then passed to the data link layer. The data link layer receives the IP datagram, adds its own header(and possibly a trailer), and passes it to the physical layer. The physical layer encodes the bits into electrical or optical signals and sends it to the remote machine.

Decapsulation :

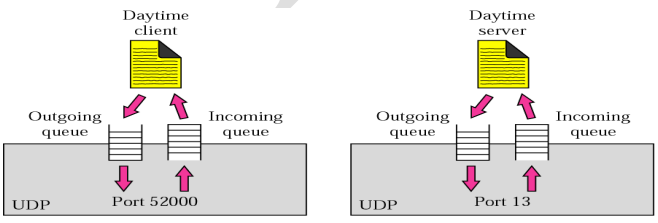
When the message arrives at the destination host, the physical layer decodes the signals into bits and passes it to the data link layer. The data link layer uses the header (and the trailer) to check the data. If there is no error, the header and trailer are dropped and the datagram is passed to **IP**. The IP software does its own checking. If there is no error, the header is dropped and the user datagram is passed to UDP with the sender and receiver IP addresses. UDP uses the checksum to check the entire user datagram. If there is no error, the header is dropped and the application data along with sender socket address is passed to the process. The sender socket address is passed to the process in case it needs to respond to the message received.

Queuing

We have talked about ports without discussing the actual implementation of them. In UDP queues are associated with ports(see figure 11.11).

At the client site, when a process starts, it requests aport number from the operating systems. Some implementations create both an incoming and an outgoing queue associated with each process. Other implementations create only an incoming queue associated with each process.

Figure 11.11 ***Queues in UDP***



Note that even if a process wants to communicate with multiple processes, it obtains only one port number and eventually one outgoing and one incoming queue. The queues opened by the client are, in most cases, identified by ephemeral port numbers. The queues function as long as the process is running. When the process terminates, the queues are destroyed.

The client process can send messages to the outgoing queue by using the source port number specified in the request. UDP removes the message one by one, and, after adding the UDP header, delivers them to IP. An outgoing queue can overflow. If this happens, the operating systems can ask the client process to wait before sending any more messages.

When a message arrives for a client, UDP checks to see if an incoming queue has been created for the port number specified in the destination port number field of the user datagram . If there is such a queue, UDP sends the received user datagram to the end of the queue.

If there is no such queue, UDP discards the user datagram and asks the ICMP protocol to send a port unreachable message to be sent to the server. All of the incoming messages for one particular client program, whether coming from the same or different server, are sent to the same queue. An incoming queue can overflow. If this happens, UDP drops the user datagram and asks for a port unreachable message to be sent to the server.

At the server site, the mechanisms of creating queues is different. In its simplest form, a server asks for incoming and outgoing queues using its well-known port when it starts running. The queues remain open as long as the server is running.

When a message arrives for a server, UDP checks to see if an incoming queue has been created for the port number specified in destination port number field of the user datagram. If there is such a queue, UDP sends the received user datagram to the end of the queue. If there is no such a queue, UDP discards the user datagram and asks the ICMP protocol to send a port unreachable message to the client. All of the incoming messages for one particular server, whether coming from the same or a different client, are sent to the same queue. An incoming queue can overflow. If this happens, UDP drops the user datagram and asks for a port unreachable message to be sent to the client.

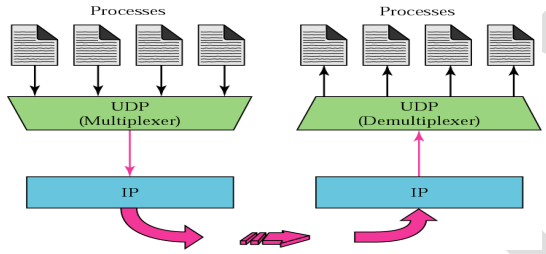
When a server wants to respond to a client , it sends messages to the outgoing queue using the source port number specified in the request. UDP removes the message one by one, and , after

adding the UDP header, delivers them to IP. An outgoing queue can overflow. If this happens, the operating system asks the server to wait before sending any more messages.

Multiplexing and Demultiplexing

In a host running a TCP/IP protocol suite,there is only one UDP but possibly several processes that may want use the services of UDP. To handle this situation, UDP multiplexes and demultiplexes(see figure 11.12).

Figure 11.12 *Multiplexing and demultiplexing*



Multiplexing

At the sender site, there may be several processes that need to send user datagram's.

However, there is only one UDP. This is a many-to-one relationship and requires multiplexing. UDP accepts messages from different processes, differentiated by their assigned port numbers. After adding the header, UDP passes the user datagram to the IP.

Demultiplexing

At the receiver site, there is only one UDP. However, we may have many processes that can receive user datagram's. This is a one- to-many relationship and requires demultiplexing. UDP receives user datagrams from IP. After error checking and dropping of the header, UDP delivers each message to the appropriate process based on the port numbers.

USES OF UDP

The following lists some of the uses of UDP protocol:

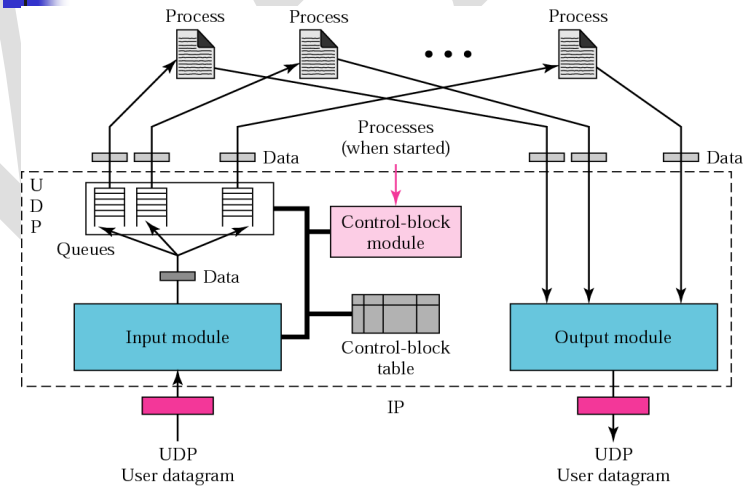
- UDP is suitable for a process that requires simple request-response communication with little concern for flow and error control. It is not usually used for a process such as FTP that needs to send bulk data

- UDP is suitable for a process with internal flow and error-control mechanisms. For example, the Trivial File Transfer Protocol(TFTP) (see chapter 19) process includes flow and error control. It can easily use UDP.
- UDP is suitable transport protocol for multicasting. Multicasting capability is embedded in the UDP software but not in the TCP software.
- UDP is used for management process such as SNMP
- UDP is used for some route updating protocols such as Routing Information Protocol (RIP) (see chapter 14).

UDP PACKAGE

- To show how UDP handles the sending and receiving of UDP packets, we present a simple version of the UDP package.
- We can say that the UDP package involves five components: a control-block table, input queues, a control-block module, an input module, and an output module. Figure 11.13 shows these five components and their interactions.

Figure 11.13 *UDP design*



Control-Block Table

In our package, UDP has a control-block table to keep track of the open ports. Each entry in this table has a minimum of four fields:the state, which can be FREE or IN-use, the process ID, the port number, and the corresponding queue number.

Input Queues

Our UDP packages uses a set of input queues, one for each process. In this design, we do not use output queues.

Control-Block Module

The control-block module is responsible for the management of the control-block table. When a process starts, it asks for a port number from the operating system.

The operating system assigns well-known port numbers to servers and ephemeral port numbers to clients. The process passes the process ID and the port number to the control-block module to create an entry in the table for the process. The module does not create the queues, the field for queue number has a value of zero. Note that we have not included a strategy to deal with a table that is full.

Control-Block Module

Receive: a process ID and a port number

1. search the control block table for a free entry.

    1. If (not found)

        1.Delete an entry using a predefined strategy.

    2. Create a new entry with the state IN-USE.

    3. Enter the process ID and the port number.

        2. Return.

Input Module

The input module receives a user datagram from the IP. It searches the control-block table to find an entry having the same port number as this user datagram. If the entry is found, the module uses the information in the entry to enqueue the data. If the entry is not found, it generates an ICMP message.

Input Module

Receive: a user datagram from IP

1. Look for the corresponding entry in the control-block table.
   1. If (found)

      1.Check the queue field to see if a queue is allocated.

         1.IF (no)

Allocate a queue.

         2. Enqueue the data in the corresponding queue.

   2. If (not found)

      1.Ask the ICMP module to send an "unreachable port" message.

      2. Discard the user datagram.

2. Return.

Output Module

The output module is responsible for creating and sending user datagrams.

Output Module

Receive: data and information from a process

1. Create a UDP user datagram.

2.Send a user datagram.

3.Return.

Examples

In this section we show some examples of how our package responds to input and output. The control-block table at the start of our examples is shown in Table 11.2

Table 11.2 *The control-block table at the beginning of examples*

State	Process ID	Port Number	Queue Number
IN-USE	2,345	52,010	34
IN-USE	3,422	52,011	
FREE			
IN-USE	4,652	52,012	38
FREE			

EXAMPLE 2

The first activity is the arrival of a user datagram with destination port number 52,012. The input module searches for this port number and finds it. Queue number 38 has been assigned to this port, which means that the port has been previously used. The input module sends the data to queue 38. The control-block table does not change.

Table 11.3 *Control-block table after Example 3*

State	Process ID	Port Number	Queue Number
IN-USE	2,345	52,010	34
IN-USE	3,422	52,011	
IN-USE	**4,978**	**52,014**	
IN-USE	4,652	52,012	38
FREE			

BOOTP Protocol

BOOTP is not a dynamic configuration protocol. When a client requests its IP addresses, the BOOTP server consults a table that matches the physical addresses of the client with its IP addresses. This implies that the binding between the physical addresses and the IP address of the client already exists. The binding is predetermined. However, what if a host moves from one physical network to another? What if a host wants a temporary IP address? BOOTP cannot handle these situations because the binding between the physical and IP addresses is static and fixed in a table until changed by the administrator. BOOTP is a static configuration protocol.

The **Dynamic Host Configuration Protocol (DHCP)** has been devised to provide static and dynamic address allocation that can be manual or automatic.

1.1 Static Address Allocation

In this capacity DHCP acts like BOOTP. It is backward compatible with BOOTP, which means a host running the BOOTP client can request a static address from a DHCP server. A DHCP server has a database that statically binds physical addresses to IP addresses.

Dynamic Address Allocation

DHCP has a second database with pool of available IP addresses. This second database makes DHCP dynamic. When a DHCP client requests a temporary IP address, the DHCP server goes to the pool of available IP addresses and assigns an IP address for a negotiable period of time.

When a DHCP client sends a request to a DHCP server, the server first checks its static database. If an entry with the requested physical address exists in the static database, the permanent IP address of the client is returned. On the other hand, if the entry does not exist in the static database, the server selects an IP address from the available pool, assigns the address to the client, and adds the entry to the dynamic database. The dynamic aspect of DHCP is needed when a host moves from network to network or is connected and disconnected from a network (like a subscriber to a service provider). DHCP provides temporary IP address for a limited period of time.

The addresses assigned from the pool are temporary addresses. The DHCP server issues a lease for a specific period of time. When the lease expires, the client must either stop using the IP address or renew the lease. The server has the choice to agree or disagree with the renewal. If the server disagrees, the client stops using the address.

Manual and Automatic Configuration

One major problem with BOOTP protocol is that the table mapping the IP addresses to physical adresses needs to be manually configured. This means that every time there is a change in a hysical or IP address, the administrator needs to manually enter the changes. DHCP, on the other hand, allows both manual and automatic configurations. Static addresses are created manually; dynamic addresses are created automatically.

1. 2 Packet Format

To make DHCP backward compatible with BOOTP, the designers of DHCP have decided to use almost the same packet format. They have only added a 1-bit flag to the packet. However, to allow different interactions with the server, extra options have been added to the option field. Figure 16.6 shows the format of a DHCP message. The new fields are as follows:

- **Flag.** A 1-bit flag has been added to the packet (the first bit of the unused field) to let the client specify a forced broadcast reply (instead of unicast) from the server. If the reply were to be unicast to the client, the destination IP address of the IP packet is the address assigned to the client. Since the client does not know

its IP address, it may discard the packet. However, if the IP datagram is broadcast, every host will receive and process the broadcast message.

- **Options.** Several options have been added to the list of options. One option, with value 53 for the tag subfield (see figure 16.5), is used to define the type of interaction between the client and server.(see table 16.2). Other options define parameters such as lease time and so on. The options field in DHCP can be up to 312 bytes.

Figure 16.6 *DHCP packet*

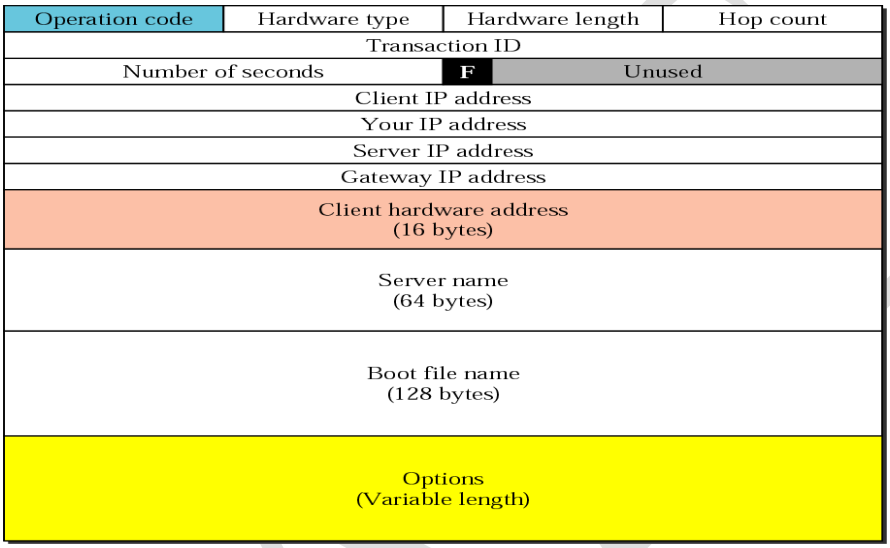


Table 16.2 Options for DHCP

Value	*Value*
1 DHCPDISCOVER	5 DHCPACK
2 DHCPOFFER	6 DHCPNACK
3 DHCPREQUEST	7 DHCPRELEASE
4 DHCPDECLINE	

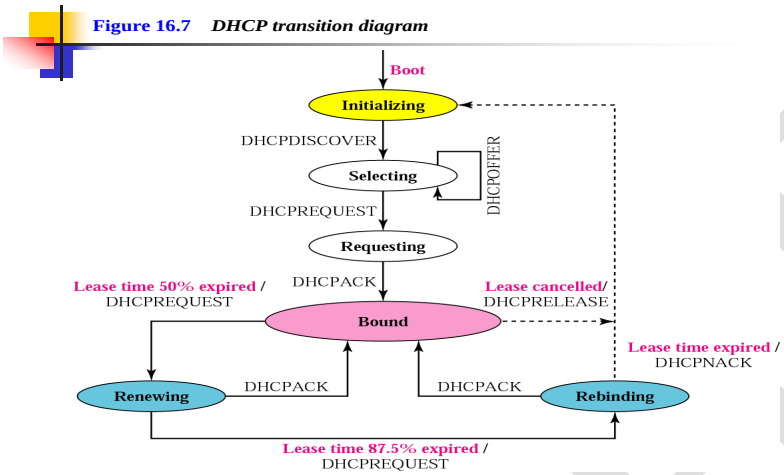
Transition states

The DHCP client transitions from one state to another

depending on the messages it receives or sends. See figure 16.7.

Initializing state

When the DHCP client first starts, it is in the initializing state. The client broadcasts a DHCPDISCOVER message (a request message with the DHCPDISCOVER option) using port 67.



Selecting state

After sending the DHCPDISCOVER message, the client goes to the selecting state. Those servers that can provide this type of service respond with a DHCPOFFER message. In these messages, the servers offer an IP address. The can also offer the lease duration. The default is 1 h. The server that sends a DHCPOFFER locks the offered IP address so that it is not available to any other clients. The client chooses one of the offers

and sends a DHCPREQUEST message to be selected server. It then goes to the requesting state. However, if the client receives no DHCPOFFER message, it tries four more times, each with a span of 2s. if there is no reply to any of these DHCPDISCOVER, the client sleeps for 5 minutes before trying again.

Requesting state

The client remains in the requesting state until it receives a DHCPPACKmessage from the server which creates the binding between the client physical address and its IP address . After the receipt of the DHCPACK, the client goes to the bound state

The client remains in the renewing state until one of two events happens. It can receive a DHCPACK, which renews the least agreement. In this case, the client resets and goes back to the bound state. Or, if a DHCPACK is not received, and 87.5% of the lease time expires; the client goes to the rebinding state.

Rebinding state

The client remains in the rebinding state until one of three events happens. If the client receives a CPNACK or the lease expires, it goes back to the initializing state and tries to get another IP address. If the client receives a DHCPACK it goes to the bound state and resets the timer.

Exchanging Messages

Figure 16.8 *Exchanging messages*

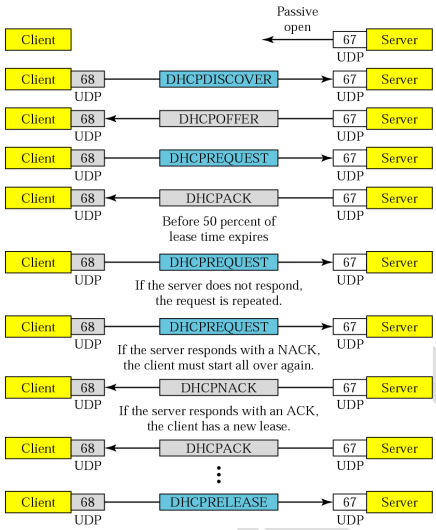
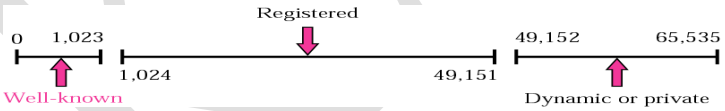


Figure 11.5 *ICANN ranges*



- **Well-known ports**. The ports ranging from 0 to 1,023 are assigned and controlled by ICANN. These are the well-known ports.
- **Registered ports.** The ports ranging from 1,024 to 1,023 are not assigned or controlled by ICANN. They can only be registered with ICANN to prevent duplication.
- **Dynamic ports**. The ports ranging from 49,152 to 65,535 are neither controlled nor registered. They can be used as temporary or private port numbers. The original recommendation was that the ephemeral port numbers for clients to be chosen from this range. However, most systems do not follow this recommendation.

DOMAIN NAME SYSTEM (DNS)

To identify an entity, TCP\IP protocols use the IP address, which uniquely identifies the connection of a host to the Internet. However, people prefer to use names instead of numeric addresses. Therefore, we need a system that can map a name to an address or an address to a name.

When the Internet was small, mapping was done using a *host file.* The host file had only two columns: name and address. Every host could store the host file on its disk and update it periodically from a master host file. When a program or a user wanted to map a name to an address, the host consulted the host file and found the mapping.

Today, however, it is impossible to have one single host file to relate every address with a name and vice versa. The host file would be too large to store in every host. In addition, it would be impossible to update all the host files every time there is a change.

One solution would be to store the entire host file in a single computer and allow access to this centralized information to every computer that needs mapping, But we know that this would create a huge amount of traffic on the Internet.

Another solution , the one used today, is to divide this huge amount of information into smaller parts and store each part on a different computer. In this method, the host that needs mapping can contact the closest computer holding the needed information. This method is used by the Domain Name System (DNS)

NAME SPACE

To be unambiguous, the names assigned to machines must be carefully selected from a name space with complete control over the binding between the names and IP addresses. In other words, the names must be unique because he addresses are unique. A name space that maps each addresses to a unique name can be organized in two ways: flat or hierarchical.

Flat Name Space

In a flat name space, a name is assigned to an address. A name in this space is a sequence of characters without structure. The names may or may not have a common section: if they do, it has no meaning. The main disadvantages of a flat name space is that it cannot be used in a large system such as the Internet because it must be centrally controlled to avoid ambiguity and duplication.

Hierarchical Name Space

In a hierarchical name space, each name is made of several parts. The first part can define the nature of the organization, the second part can define the name of an organization, the third part can define departments in the organization, and do on. In this case, the authority to assign and control the name spaces can be decentralized. A central authority can assign the part of the name that defines the

nature of the organization and the name of the organization. The responsibility of the rest of the name can be given to the organization itself. The organization can add suffixes (or prefixes) to the name to define its host or recourses. The management of the organization need not worry that the prefix chosen for a host is taken by another organization because, even if part of an address is the same, the whole address is different. For example, assume two colleges and a company call one of their computers challenger. The first college is given a name by the central authority such as fhda.edu, the second college is given the name smart.com. When each of these organizations adds the name challenger to the name they have already been given, the end result is three distinguishable names: challenger.fhda.edu, challenger.berkely.edu, and challenger.smart.com. The names are unique without the need for assignment by a central authority. The central authority controls only part of the name, not the whole.

DOMAIN NAME SPACE

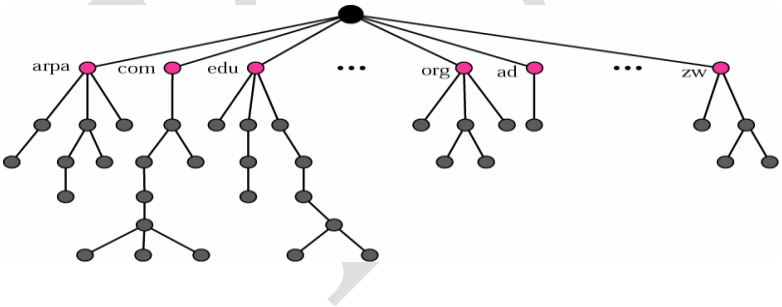
To have a hierarchical name space, a domain name space was designed. In this design the names are defined in an inverted-tree structure with the root at the top. The tree can have only 128 levels: level 0 (root) to level 127 (see Figure).

Label

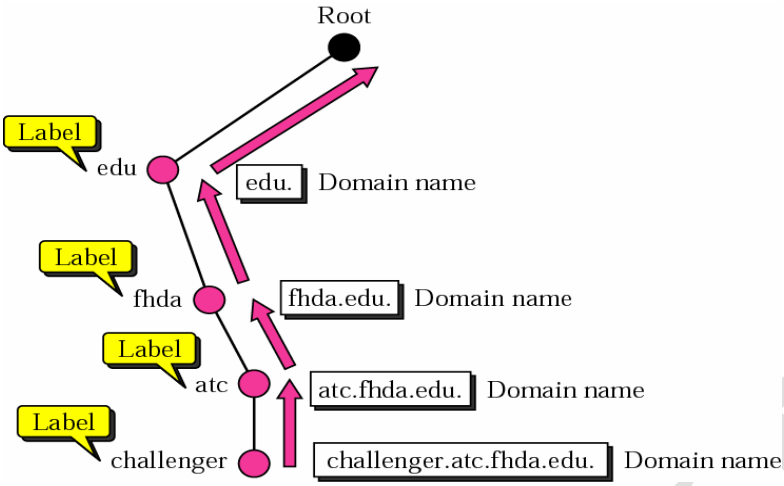
Each node in the tree has a label, which is a string with a maximum of 63 characters. The root label is a null string (empty string). DNS requires that children of a node (nodes that branch from the same node) have different labels, which guarantees the uniqueness of the domain names.

Domain Name

Each node in the tree has a domain name. A full domain name is a sequence of labels separated by dots (.). The domain names are always read from the node up to the root.



The last label is the label of the root (null). This means that a full domain name always ends in a null label, which means the last character is a dot because the null string is nothing.



Fully Qualified Domain Name (FQDN)

If a label is terminated by a null string, it is called a fully qualified domain name (FQDN). An FQDN is a domain name that contains the full name of a host. It contains all labels, from the most specific to the most general, that uniquely define the name of the host. For example, the domain name challenger.atc.fhda.edu. is the FQDN of a computer named challenger installed at the Advanced Technology Center (ATC) at De Anza College. A DNS server can only match an FQDN to an address. Note that the name must end with a null label, but because null means nothing, the label ends with a dot (.)

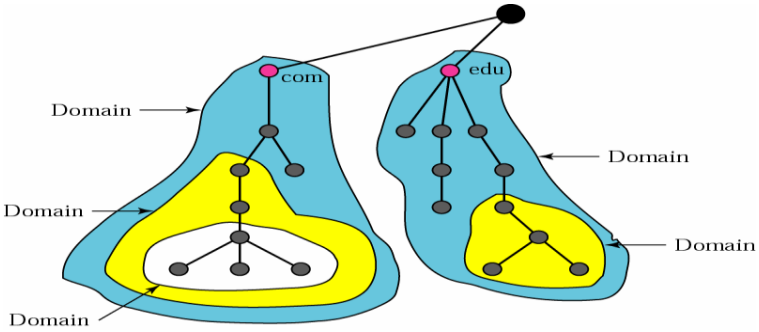
Partially Qualified Domain Name (PQDN)

If a label is not terminated by a null string, it is called a partially qualified domain name (PQDN). A PQDN starts from a node, but it does not reach the root. It is used when the name to be resolved belongs to the same site as the client. Here the resolver can supply the missing part, called the suffix, to create an FQDN. For example, if a user at the fhda.edu. site wants to get the IP address of the challenger computer,he or she can define the partial name

FQDN	PQDN
challenger.atc.fhda.edu. cs.hmme.com. www.funny.int.	challenger.atc.fhda.edu cs.hmme www

Domain

A domain is a subtree of the domain name space. The name of the domain is the domain name of the node at the top of the subtree. The above figure shows some domains. Note that a domain may itself be divided into domains(or subdomains as they are sometimes called).

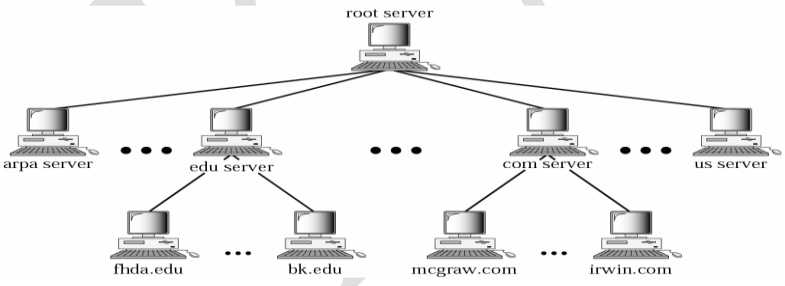


DISTRIBUTION OF NAME SPACE

The information contained in the domain space must be stored. However, it is very inefficient and also not reliable to have just one computer store such huge amount of information. It is inefficient because responding to request from all over the world places a heavy load on the system. It is not reliable because any failure makes the data inaccessible.

Hierarchy of Name Servers

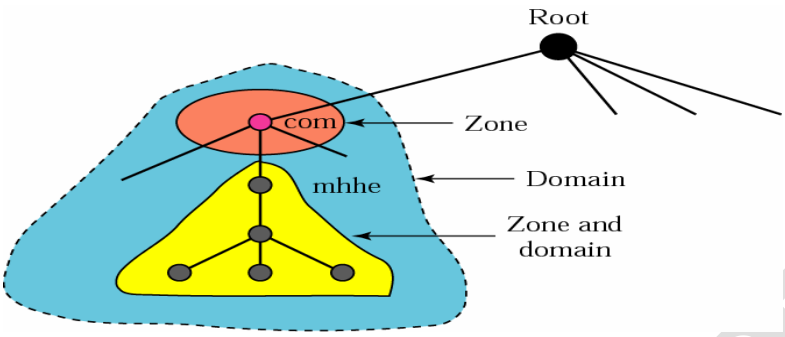
The solution to these problems is to distribute the information among many computers called DNS servers. One way to do this is to divide the whole space into many domains based on the first level. In other words, we let the root stand alone and create as many domains (subtrees) as there are first-level nodes. Because a domain created this way could be very large, DNS allows domains to be divided further into smaller domains (subdomains). Each server can be responsible (authoritative) for either a large or small domain. In other words, we have a hierarchy of servers in the same way that we have a hierarchy of names (see below figure).



Zone

Since the complete domain name hierarchy cannot be stored on a single server, it is divided among many servers. What a server is responsible for or has authority over is called a zone. We can define a zone as a contiguous part of the entire tree. If a server accepts responsibility for a domain and does not divide the domain into smaller domains, the “domain” and the “zone” refer to the same thing. The

server makes a database called a zone file and keeps all the information for every node under that domain. However, if a server divides its domain into subdomains and delegates part of its



Authority to others servers, “domain” and “zone” refer to different things. The information about the nodes in the subdomains is stored in the servers at the lower levels, with the original server keeping some sort of reference to these lower-level servers. Of course the original server does not free itself from responsibility totally: It still has a zone, but the detailed information is kept by the lower-level servers

A server can also divide part of its domain and delegate responsibility but still keep part of the domain for itself. In this case, its zone is made of detailed information for the part of the domain that is not delegated and references to those parts that are delegated.

Root Server

A root server is a server whose zone consists of the whole tree. A root server usually does not store any information about domains but delegates its authority to other servers, keeping references to those servers. There are several root servers, each covering the whole domain name space. The servers are distributed all around the world.

Primary and Secondary Servers

DNS defines two types of servers: primary and secondary. A primary server is a server that stores a file about the zone for which it is an authority. It is an authority. It is responsible for creating, maintaining, and updating the zone file. It stores the zone file on a local disk.

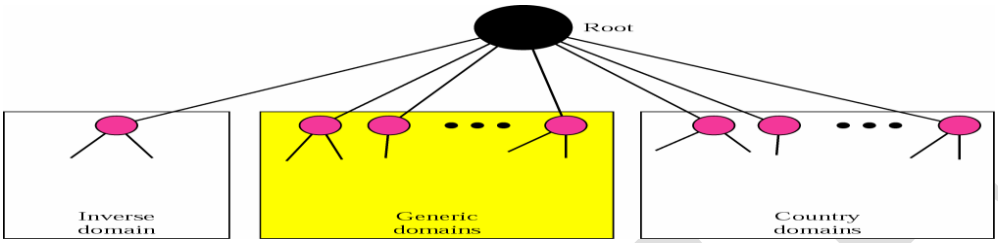
A secondary server is a server that transfers the complete information about a zone from another server (primary or secondary) and stores the file on its local disk. The secondary server neither creates nor updates the zone files. If updating is required, it must be done by the primary server, which sends the updated version to the secondary.

The primary and secondary servers are both authoritative for the zones they serve. The idea is not to put the secondary server at a lower level of authority but to create redundancy for the data so that if one server fails, the other can continue serving clients. Note also that a server can be a primary

server for a specific zone and a secondary server for another zone. Therefore, when we refer to a server as a primary or secondary server, we should be careful to which zone we refer.

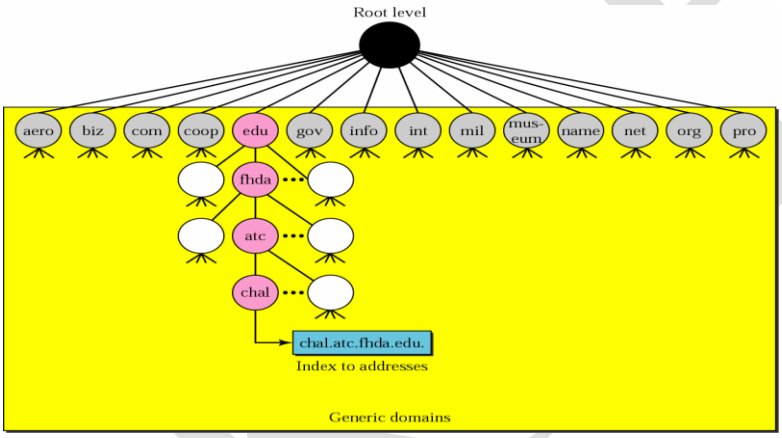
DNS in the internet

DNS is a protocol that can be used in different platforms. In the internet, the domain name space(tree) is divided into three different sections: generic domains, country domains, and the inverse domain.



Generic domains

The generic domains define registered hosts according to their generic behavior. Each node in the tree defines a domain, which is an index to the domain name space database.



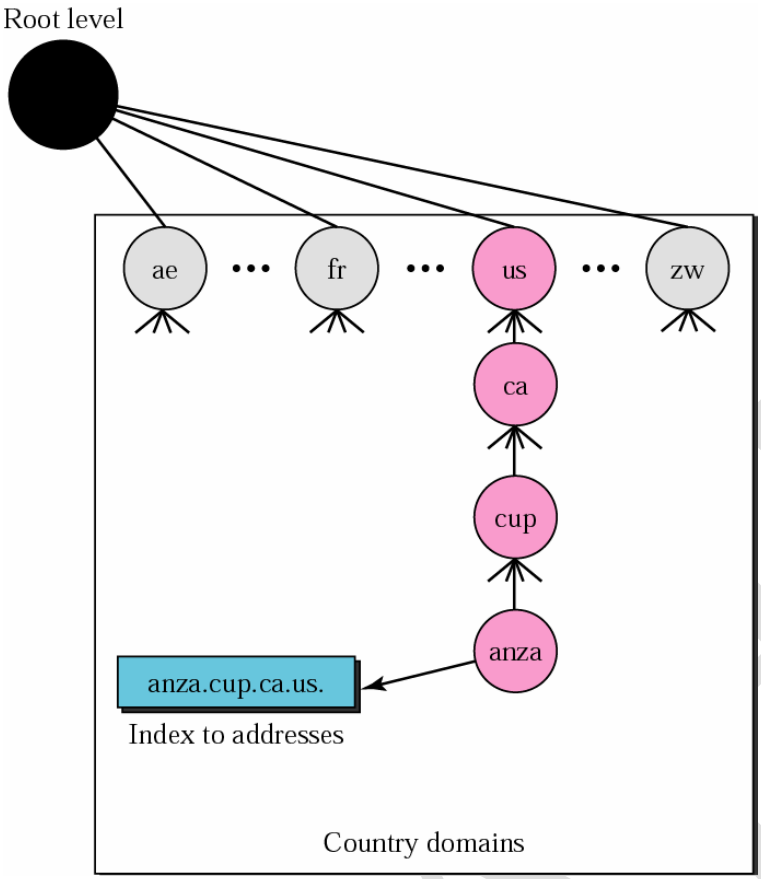
Looking at the tree, we see that the first level in the generic domains section allows 14 possible labels.These labels describe the organization types as listed in table.

Label	*Description*
aero	Airlines and aerospace companies
biz	Businesses or firms (similar to "com")
com	Commercial organizations
coop	Cooperative business organizations
edu	Educational institutions
gov	Government institutions
info	Information service providers

Country Domains

The country domains section uses two-character country abbreviations (e.g., us for United states). Second-labels can be organizational, or they can be more specific, national designations. The United States, for example, uses state abbreviations as a subdivision of us (e.g.,ca.us.).

The below figure shoes the country domains section. The address anza.cup.ca.us can be translated to De Anza College in Cupertino in California in the United States.



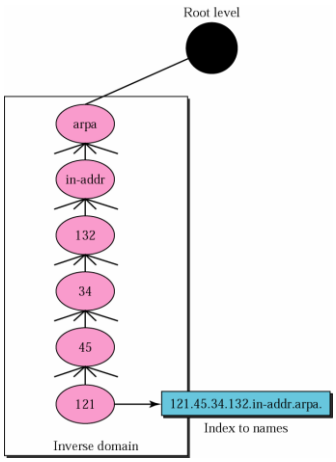
Inverse domain

The inverse domain is used to map as address to a name. This may happen, for example, when a server has received a request from a client to do a task. Although the server has a file that contains a list of authorized clients, only the IP address of the client (extracted from the received IPpacket) is listed. The server asks its resolver to send a query to the DNS server to map an address to a name to determine if the client is on the authorized list.

This type of query is called an inverse or pointer(PTR) query.To handle a pointer query, the inverse domain is added to the domain name space with the first-level node called arpa (for historical reason). The second level is also one single node named in-addr(for inverse address). The rest of the domain defines IP addresses.

The servers that handle the inverse domain are also hierarchical. This means the netid part of the address should be at a higher level than the subnetid part, and the subnetid part higher than the hostid part . In this way, a server serving the whole site is at a higher level than the servers serving each subnet. This configuration makes the domain look inverted when compared to a generic or country domain. To follow the convention of reading the domain labels from the bottom to the top, an IP

address such as 132.34.45.121(a classs B address with netid132.34) is read as 121.45.34.132.in-addr.arpa.



Registrar

How are the new domains added to DNS? This is done through a registrar, a commercial entity accredited by ICANN. A registrar first verifies that the requested domain name is unique and then enters into the DNS database. A fee is charged.

Resolution

Mapping a name to an address or an address to a name is called name-address resolution.

Resolver

DNS is designed as a client-server application. A host that needs to map an address to a name or a name to an address calls a DNS client called a resolver. The resolver accesses the closest DNS server with a mapping request. If the server has the information, it satisfies the resolver; otherwise, it either refers the resolver to other servers or asks other servers to provide the information.

After the resolver receives the mapping, it interprets the response to see if it is a real resolution or an error, and finally delivers the result to the process that requested it.

Mapping Names to Addresses

Most of the time, the resolver gives a domain name to the server and asks for the corresponding address. In this case, the server checks the generic domains or the country domains to find the mapping.

If the domain name is from the generic domains section, the resolver receives a domain name such as “chal.atc.fhda.edu.”. The query is sent by the resolver to the local DNS server for resolution. If the local server cannot resolve the query, it either refers the resolver to other servers or asks other servers directly.

If the domain name is from the country domains section, the resolver receives a domain name such as "ch.fhda.cu.ca.us.". The procedure is the same.

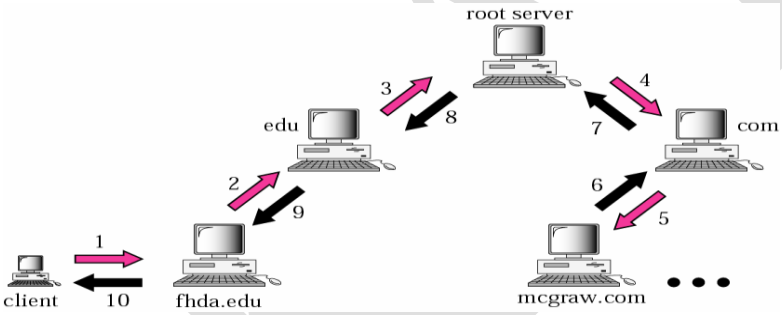
Mapping Addresses to Names

A client can send an IP address to a server to be mapped to a domain name. As mentioned before, this is called a PTR query. To answer queries of this kind, DNS uses the inverse domain. However, in the request, the IP address is reversed and two labels, in-addr and arpa, are appended to create a domain acceptable by the inverse domain section.

For example, if the resolver receives the IP address 132.34.45.121, the resolver first inverts the address and then adds the two labels before sending. The domain name sent is"121.45.34.132.in-addr.arpa.",which is received by the local DNS and resolved.

Recursive Resolution

The client(resolver) can ask for a recursive answer from a name server. This means that the resolver expects the server to supply the final answer. If the server is the authority for the domain name, it checks its database and responds. If the server is not the authority, it sends the request to another server (the parent usually) and waits for the response. If the parent is the authority, it responds; otherwise, it sends the query to yet another server. When the query is finally resolved, the response travels back until it finally reaches the requesting client.



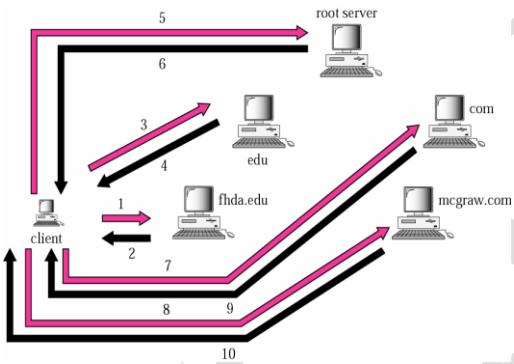
Iterative Resolution

If the client does not ask for a recursive answer, the mapping can be done iteratively. If the server is an authority for the name, it sends the answer. If it is not, it returns (to the client) the IP addresses of the server that it thinks can resolve the query. The client is responsible for repeating the query to this second server. If the newly addressed server can resolve the problem, it answers the query with the IP address; otherwise it the IP address of a new server to the client. Now the client must repeat the query to the third server. This process is called iterative because the client repeats the same query to multiple servers.

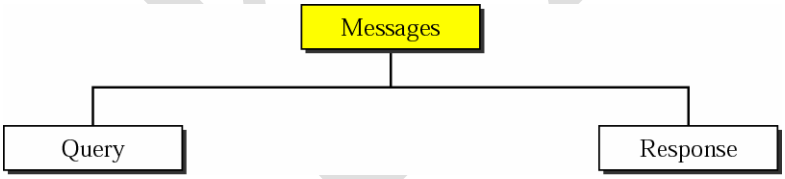
Caching

Each time a server receives a query for a name that is not in its domain, it needs to search its database for a server IP address. Reduction of this search time would increase efficiency. DNS handles this with a mechanism called caching. When a server asks for a mapping from another server and receives the response, it stores this information in its cache memory before sending it to the client. If the same or another client asks for the same mapping, it can check its cache memory and resolve the problem. However, to inform the client that the response is coming from the cache memory and not from an authoritative source, the server marks the response as unauthoritative.

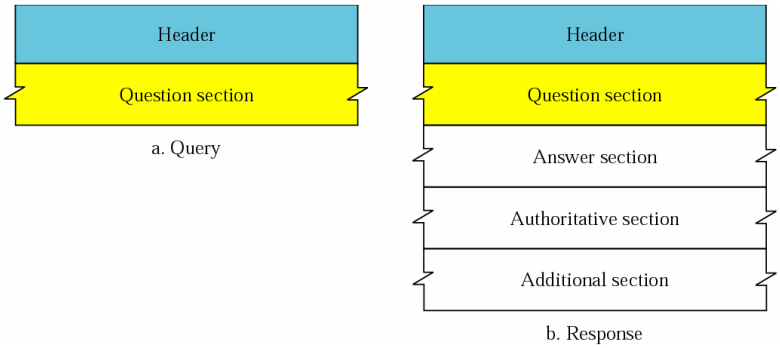
Caching speeds up resolution, but it can also be problematic. If a server caches a mapping for a long time, it may send an outdated mapping to the client. To counter this two techniques are used. Fist, the authoritative server always adds information to the mapping called time-to-live(TTL). It defines the time in seconds that the receiving server can cache the information. After that time, the mapping is invalid and any query must be sent again to the authoritative server. Second, DNS requires that each server keep a TTL counter for each mapping it caches. The cache memory must be searched periodically and those mappings with an expired TTL must be purged.



DNS has two types of messages: query and response (see below figure). Both types have the same format



The query message consists of a header and question records; the response message consists of a header, question records, answer records, authoritative records, and additional records (see below figure).



Header

Both query and response messages have the same header format with some fields set to zero for the query messages. The header is 12 bytes and its format is shown in below figure.

Identification	Flags
Number of question records	Number of answer records (All 0s in query message)
Number of authoritative records (All 0s in query message)	Number of additional records (All 0s in query message)

The header fields are as follows:

Identification. This is a 16-bit field used by the client to match the response with the query. The client uses a different identification number each time it sends a query.

The server duplicates this number in the corresponding

response.

Flags. This is a 16-bit field consisting of the subfields shown in the below figure.

A brief description of each flag subfield follows.

- QR (query/response). This is a 1-bit subfield that defines the type of message. If it is 0, the message is a query. If it is 1, the message is a response.
- OpCode. This is a 4-bit subfield that defines the type of query or response (0 if standard, 1 if inverse, and 2 if a server status request).
- AA (authoritative answer). This is a 1-bit subfield. When it is set (value of 1) it means that the name server is an authoritative server. It is used only in a response message.
- TC (truncated). This is a 1-bit subfield. When it is set (value of 1), it means that the response was more than 512 bytes and truncated to 512. it is used when DNS uses the services of UDP.
- RD (recursion desired). This is a 1-bit subfield. When it is set (value of 1) it means the client desires a recursive answer. It is set in the query message and repeated in the response message.

- RA (recursion available).This is a 1-bit subfield. When it is set in the response, it means that a recursive response is available. It is set only in the response message.
- Reserved. This is a 3-bit subfield set to 000.
- rCode. This is a 4-bit field that shows the status of the error in the response. Of course, only an authoritative server can make a judgment. Table below shows the possible values for this field.

Value	*Meaning*
0	No error
1	Format error
2	Problem at name server
3	Domain reference problem
4	Query type not supported
5	Administratively prohibited
6–15	Reserved

- queries in the section of the message.
- **Number of answer records.** This is 16-bit field containing the number of answer records in the answer section of the message. Its value is zero in the query message.
- **Number of authoritative records.** This is a 16-bit field containing the number of authoritative records in the authoritative section of a response message. Its value is zero in the query message.
- Number of additional records. This is a 16-bit field containing the number of additional records in the additional section of a response message. Its value is zero in the query message.

Question Section

This is a section consisting of one or more question records. It is present on both query and response message.

Answer Section

This is a section consisting of one or more resource records. It is a present only on response messages. This section includes the answer from the server to the client (resolver). We will discuss resource records in a following section.

Authoritative Section

This is a section consisting of one or more resource records. It is present only on response messages. This section gives information (domain name) about one or more authoritative servers for the query.

Additional Information Section

This is a section consisting of one or more resource records. It is present only on response messages. This section provides additional information that may help the resolver. For example, a server may give the domain name of an

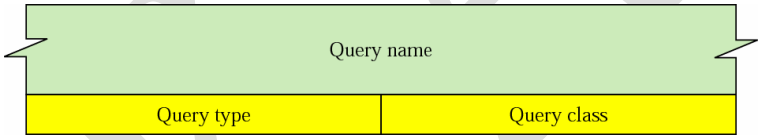
Authoritative server to the resolver in the authoritative section, and include the IP address of the same authoritative server in the additional information section.

TYPES OF RECORDS

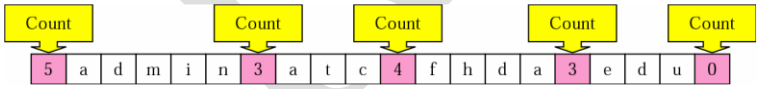
As we saw in the previous section, two types of records are used in DNS. The question records are used in the question section of the query and response messages. The resource records are used in the answer, authoritative and additional information sections of the response message.

Question Record

A question record is used by the client to get information from a server. This contains the domain name. The below figure shows a format of a question record. The list below describes question record fields.



Query name. This is a variable-length field containing a domain name (see below figure).

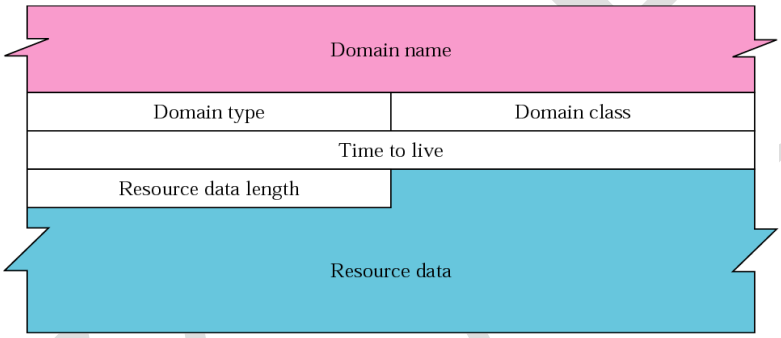


Query type. This is a 16-bit field defining the type of query. Table below shows some of the types commonly used. The last two can only be used in a query.

Type	*Mnemonic*	*Description*
1	A	Address. A 32-bit IPv4 address. It is used to convert a domain name to an IPv4 address.
2	NS	Name server. It identifies the authoritative servers for a zone.
5	CNAME	Canonical name. It defines an alias for the official name of a host.
6	SOA	Start of authority. It marks the beginning of a zone. It is usually the first record in a zone file.
11	WKS	Well-known services. It defines the network services that a host provides.
12	PTR	Pointer. It is used to convert an IP address to a domain name.
13	HINFO	Host information. It gives the description of the hardware and the operating system used by a host.
15	MX	Mail exchange. It redirects mail to a mail server.
28	AAAA	Address. An IPv6 address (see Chapter 27).
252	AXFR	A request for the transfer of the entire zone.
255	ANY	A request for all records.

Resource Record

Each domain name (each node on the tree) is associated with a record called the resource record. The server database consists of resource records. Resource records are also what is returned by the server to the client. The below figure shows the format of a resource record.



- **Domain name.** This is a variable-length field containing the domain name. It is a duplicate of the domain name in the question record. Since DNS requires the use of compression everywhere a name is repeated, this field is a pointer offset to the corresponding domain name field in the question record.
- **Domain type.** This field is the same as the query type field in the question record except the last two types are not allowed.
- **Domain class.** This field is the same as the query class field in the question record.
- **Time to live.** This is a 32-bit field that defines the number of seconds the answer is valid. The receiver can cache the answer for this period of time. A zero value means that the resource record is used only in a single transaction and is not cached.
- **Resource data length.** This is a 16-bit field defining the length of the resource data.
- **Resource data.** This is a variable-length field containing the answer to the query (in the answer section) or the domain name of the authoritative server (in the authoritative section) or additional information (in the additional information section). The format and contents of this field depend on the value of the type field. It can be one of the following:

a. **A number.** This is written in octets. For example, an IPv4 address is a 4-octet integer and an IPv6 address is a 16-octet integer.

A domain name. Domain names are expressed as a sequence of labels. Each label is preceded by a 1-byte length field that defines the number of characters in the label. Since every domain name ends with the null label, the last byte of every domain name is the length field with the value 0. To distinguish between a length field and an offset pointer (as we will discuss later), the two high-order bits of a length field always zero (00). This will not create a Problem because the length of a label cannot be more than 63, which is a maximum of 6 bits (111111).

- **An offset pointer.** Domain names can be replaced with an offset pointer. An offset pointer is a 2-byte fields with each of the 2 high-order bits set to 1 (11).
- **A character string.** A character string is represented by a 1-byte length field followed by the number of characters defined in the length field. The length field is not restricted like the domain name length field. The character string can be as long as 255 characters (including the length field).

Points to Remember

- **BOOTP** is not a dynamic configuration protocol.
- The **Dynamic Host Configuration Protocol (DHCP)** has been devised to provide static and dynamic address allocation that can be manual or automatic.
- DHCP has a second database with pool of available IP addresses
- One major problem with BOOTP protocol is that the table mapping the IP addresses to physical adresses needs to bually configured
- The combination of an IP address and a port number is called a **socket address**.
- UDP packets, called **user datagrams**, have a fixed-size header of 8 bytes.
- UDP provides a **connectionless service**s means that each user datagram sent by UDP is an independent datagram
- UDP is a very simple, unreliable transport protocol. There is no flow control, and hence no window mechanism
- To send a message from one process to another, the UDP protocol encapsulates and decapsulates messages
- If a label is terminated by a null string, it is called a fully qualified domain name (FQDN)
- If a label is not terminated by a null string, it is called a partially qualified domain name (PQDN)
- A domain is a subtree of the domain name space.

Possible Questions

Part B
(Each Question carries 6 Marks)

1. Discuss the operation of BOOTP.
2. Explain domain name space with an example.
3. Write a note on command processing in FTP.
4. Discuss about routing and carrying datagram in cells.
5. Give a detail description about DHCP
6. Explain BOOTP operation and Packet Format
7. Discuss about mapping names to address and address to names
8. Explain Static and dynamic address allocation in DHCP
9. Discuss the types of records used in DNS.
10. Explain distribution of name space

Part C:

(Each Question carries 10 Marks)

1. Analysis the different sections of DNS in internet

Karpagam Academy of Higher Education
Coimbatore - 21

DEPARTMENT OF COMPUTER APPLICATIONS
CLASS : II MCA

TCP/IP (18CAP405N)

Unit IV

S.NO	Question	Choice1	Choice2	Choice3	Choice4	Choice 5	Choice 6	Answer
1	Control characters can be used to handle _____ server.	local	global	remote	common			remote
2	In the _____ mode the client sends one character at a time to the server.	character	line	point	number			character
3	In the _____ mode the client sends one line at a time to the server.	character	line	point	number			line
4	A ___is a group of connected, communicating devices such asComputers and Printers.	Internet	Bridge	Network	Switch			Bridge
5	_____ Software provide communication between hosts	NCP	IMP	ACM	MIP			NCP
6	In 1972, _____ Project was Started.	Internet	Inter network	Internetting	Subneting			Internetting
7	Other than ARPANET, the other two Networks are _____ and _____	Packet ratio & Packet satellite	Packet radio & Packet Mobile	Packet mobile & Packet satellite	Packet Mobile & Packet ratio			Packet ratio & Packet satellite
8	_____ is a less expensive Network	CSNET	MILNET	ANSNET	NSFNET			CSNET
9	In NSFNET data is transferred at _____ rate.	1.439 Mbps	1.544Mbps	1.644 Mbps	1.437Mbps			1.544Mbps
10	Repeaters and Hub Operate in _____ Layer of Internet Model	First	Second	Third	First & Second.			First
11	Repeater receives _____ and regenerates _____ pattern.	Data, bytes	Signal, Bit	Decimal, Signal	Octal,Bit			Signal, Bit
12	Repeater connects _____ of a LAN.	Part	Mode	Mode2	Segment			Segment

13	Router connects _____ Lan to create _____	Segment, Network	Sector, Internet	Independent, Internetwork	Internet,Internetwork			Independent, Internetwork
14	Router changes physical address into _____	Datagram	Packet	Signal	bytes			Packet
15	_____notation uses Dot for separating bytes	Decimal	Binary	Hexa	Octal			Decimal
16	Migration is very fast in _____ addressing	Classful addressing	Classless addressing	LocalHost addressing	IP addressing			Classful addressing
17	If all the bits are one then it is a _____ address	Class A	Class B	Class C	Class E			Class E
18	A Block of address _____ is same.	First	Second	Third	Fourth.			First
19	The First address in a block is _____	Net-id	Host – id	Subnet	Network address			Network address
20	NIC stands for _____	Network Information Card	Network Information Center	Network Interface Card	Network Interface Center			Network Interface Card
21	_____ associates a logical address with physical address.	Static mapping	Dynamic mapping	Temporary mapping	Logical mapping			Static mapping
22	_____ addresses in the TCP/IP protocol suite are called IP address.	physical	static	dynamic	logical			logical
23	_____ defines the length of physical address in bytes.	protocol length	hardware length	software length	address length			hardware length
24	_____ define the length of logical address in bytes	protocol length	hardware length	software length	address length			protocol length
25	In cache table, _____ state means that the entry is complete.	Free	pending	resolved	complete			resolved
26	_____ module is responsible for maintain the cache table.	Input module	cache control module	output module	queue			cache control module
27	TOS bits means	Type Of Service	Type Of Security	Type Of System	Types of Select			Type Of Service

28	_____ option used for padding at the end of the option field	end of option	checksum	operation option	no operation option			end of option
29	SCMP messages are divided into _____ broad categories	2	4	1	3			2
30	Destination unreachable message is _____ type of message.	Query	error reporting	query reporting	delay reporting			error reporting
31	address mask requesting or reply is _____ type of message.	error reporting	query	delay reporting	query reporting			query
32	In destination _____represent the host is unreachable.	code1	code 2	code 3	code 4			code1
33	BOOTP stands for	Bootstrap protocol	Bootstrap project	Booting protocol	Booting project			Bootstrap protocol
34	_____ command that can create series of echo request and echo reply.	ping	pong	reply	echo			ping
35	_____ program in units can be used to trace the route of a packet	Tracer	Trace Route	Trace up	Trace Down			Trace Route
36	_____ programs in windows can be used to trace the route of a packet	Tracer	Trace Route	Trace up	Trace Down			Tracer
37	_____ msg in ICMP was designed to add a kind of flow control to the IP.	Source-Quench	Time-exceed	parameter problem	Re-direction			Source-Quench
38	In destination unreachable error reporting msg _____represent a protocol is unreachable.	code1	code 2	code 3	code 4			code 2
39	In service type TOS bit 0001 represent _____	Normal	Minimize Cost	Maximum reliability	Minimize delay			Minimize Cost
40	_____ is not a multicasting routing protocol.	ICMP	IGMP	TCP	TCP/IP			IGMP

UNIT-V

SYLLABUS

Remote Login - FTP – SMTP – SNMP. IP over ATM Wan – Cells – Routing the Cells. Mobile IP : Addressing – Agents – Agent discovery – Registration – Data Transfer – VPN

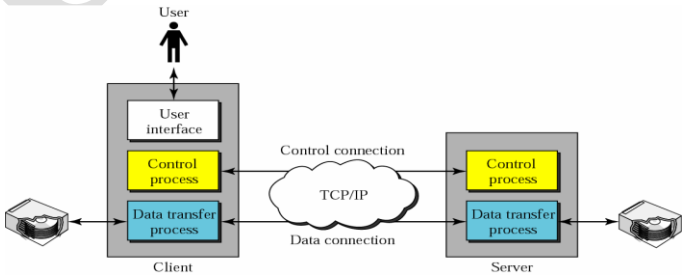
1 FILE TRANSFER PROTOCOL (FTP)

It is the standard mechanism provided by TCP\IP for copying a file from one host to another. Although transferring files from one system to another seems simple and straightforward, some problems must be dealt with first. For example, two systems may use different file name conventions. Two systems may have different ways to represent text and data. Two systems may have different directory structures. All of these problems have been solved by FTP in a very simple and elegant approach. FTP differs from other client-server applications in that it establishes two connections between the hosts. One connection is used for data transfer, the other for control information (commands and responses). Separation of commands and data transfer makes FTP more efficient. The control connection uses very simple rules of communication.

We need to transfer only a line of command or a line of response at a time. The data connection, on the other hand, needs more complex rules due to the variety of data types transferred.

FTP uses two well-known TCP Ports: Port 21 is used for the control connection, and port 20 is used for the data connection.

Figure 19.1 shows the basic model of FTP. The client has three components: user interface, client control process, and the client data transfer process. The server has two components: the server control process and the server data transfer process. The control connection is made between the control processes. The data connection is made between the data transfer processes.



The **control connection** remains connected during the entire interactive FTP session. The data connection is opened and then closed for each file transferred. It opens each time commands that involve transferring files are used, and it closes when the file is transferred. In other words, when a user starts an FTP session, the control connection opens. While the control connection is open, the data connection can be opened and closed multiple times if several files are transferred.

Connections

The two FTP connections control and data use different strategies and different port numbers.

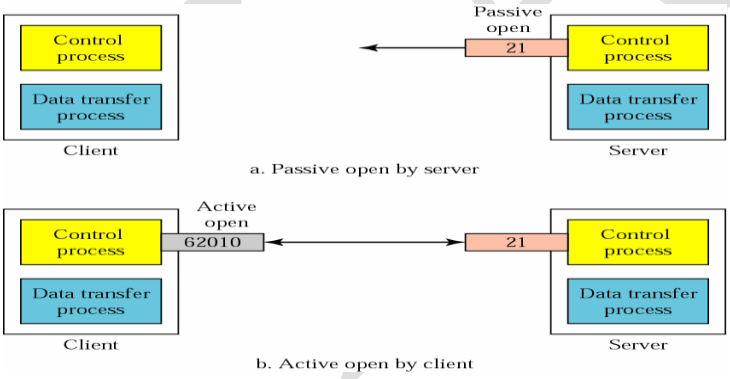
Control Connection

The control connection is created in the same way as other application programs described so far. There are two steps:

1. The server issues a passive open on the well-known port21 and waits for a client.
2. The client uses an ephemeral port and issues an active open.

The connection remains open during the entire process. The service type, used by the IP protocol, is *minimize delay* because this is an interactive connection between a user (human) and a server. The user types commands and expects to receive responses without significant delay. Figure 19.2 shows the initial connection between the server and the client.

Figure Opening the control connection



1.2 .Data Connection

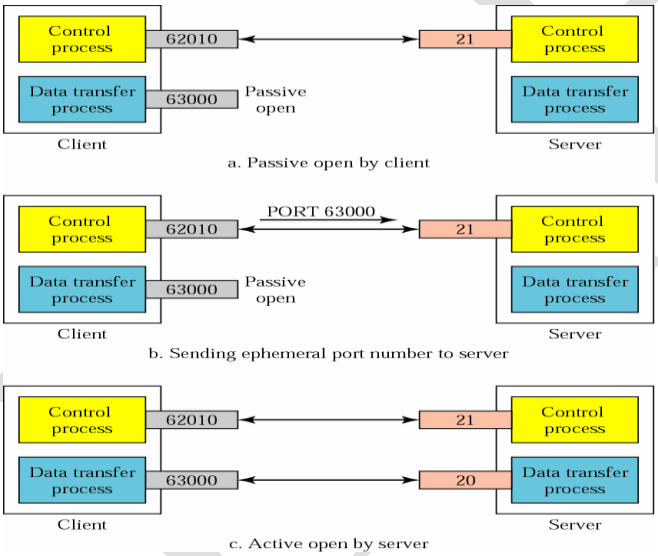
The **data connection** uses the well-known port 20 at the server site. However, the creation of a data connection is different from what we have seen so far. The following shows how FTP creates a data connection:

1. The client, not the server, issues a passive open using an ephemeral port. This must be done by the client because it is the client that issues the commands for transferring files.
2. The client sends this port number to the server using the PORT command (we will discuss this command shortly).
3. The server receives the port number and issues an active open using the well known port 20 and the received ephemeral port number.
4. **Communication**

5. The FTP client and server, which run on different computers, must communicate with each other. These two computers may use different file formats. FTP must make this heterogeneity compatible.

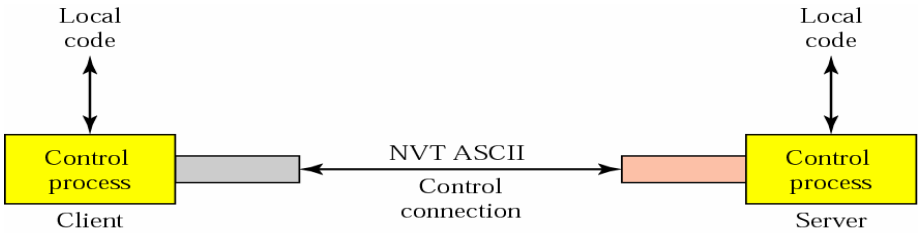
6. ` FTP has two different approaches, one for the control connection and one for the data communication. We will study each approach separately.

Figure Creating the data connection



Each line is terminated with a two-character (carriage return and line feed) end-of-line token.

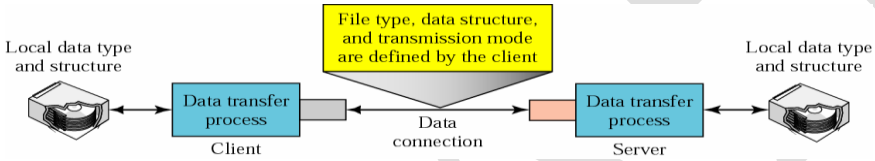
Figure Using the control connection



Communication over Data Connection

The purpose and implementation of the data connection are different from that of the control connection. We want to transfer files through the data connection. The client must define the type of file to be transferred, the structure of the data, and the transmission mode. Before sending the file through the data connection, we prepare for transmission through the control connection. The heterogeneity problem is resolved by defining three attributes of communication: file type, data structure, and transmission mode (see Figure 19.5).

Figure 19.5 Using the data connection



File Type FTP can transfer one of the following file types across the data connection:

- **ASCII file.** This is the default format for transferring text files. Each character is encoded using NVT ASCII. The sender transforms the file from its own representation into NVT ASCII characters and the receiver transforms the NVT ASDCII characters to its own representation.
- **EBCDIC file.** If one or both ends of the connection use EBCDIC encoding, the file can be transferred using EBCDIC encoding.
- **Image file.** This is the default format for transferring binary files. The file is sent as continuous streams of bits without any interpretation or encoding. This is mostly used to transfer binary files such as compiled programs.

If the file is encoded in ASCII or EBCDIC, another attribute must be added to define the printability of the file.

Nonprint. This is the default format for transferring a text file. The file contains no vertical specifications for printing. This means that the file cannot be printed without further processing because there are no characters to be interpreted for vertical movement of the print head. This format is used for files that will be stored and processed later**.**

- **File structure (default).** The file has no structure. It is a continuous stream of bytes.

- **Record structure.** The file is divided into records. This can be used only with text files.
- **Page structure.** The file is divided into pages, with each page having a page number and a page header. The pages can be stored and accessed randomly or sequentially.

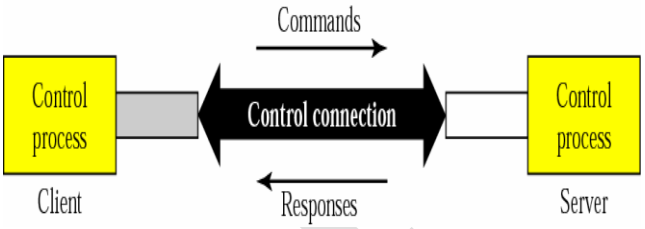
Transmission mode FTP can transfer a file across the data connection using one of the following three transmission modes:

- **Stream mode.** This is the default mode. Data are delivered from FTP to TCP as a continuous stream of bytes. TCP is responsible for chopping data into segments of appropriate size. If the data is simply a stream of bytes (file structure), no end-of-file is needed. End-of-file in this case is the closing of the data connection by the sender. If the data is divided into records (record structure), each record will have a 1-byte end-of-record (EOR) character and the end of the file will have a 1-byte end-of-file (EOF) character.
- **Block mode.** Data can be delivered from FTP to TCP in blocks. In this case, each block is preceded by a 3-byte header. the first byte called the *block descriptor;* the next two bytes define the size of the block in bytes.
- **Compressed mode.** If the file is big, the data can be compressed. The compression method normally used is run-length encoding. In this method, consecutive appearances of a data unit are replaced by one occurrence and the number of repetitions. In a text file, this is usually spaces (blanks). In a binary file, null characters are usually compressed.

Command processing

FTP uses the control connection to establish a communication between the client control process and the server control process. During this communication, the commands are sent from the client to the server and the responses are sent from the server to the client (see figure 19.6).

Figure 19.6 Command processing



Commands

Commands, which are sent from the FTP client control process, are in the form of ASCII uppercase, which may or may not be followed by an argument. We can roughly divide the commands into six groups: access commands, file management commands, data formatting commands, port defining commands, file transferring commands, and miscellaneous commands.

- **Access commands.** These commands let the user access the remote system.

Table lists common commands in this group

Command	Arguments(s)	Description
USER	User id	User information
PASS	User password	Password
ACCT	Account to be changed	Account information
REIN	Re install	Reinitialize
QUIT	Terminate	Log out of the system
ABOR	Cancel	Abort the previous command

- **File management commands.** These commands let the user access the file system on the remote computer. They allow the user to navigate through the directory structure, create new directories, delete files, and so on. Table 19.2 gives common commands in this group.

Table File management commands

Command	Argument(s)	Description
CWD	Directory name	Change to another directory
CDUP		Change to the parent directory
DELE	File name	Delete a file
LIST	Directory name	List subdirectories of files
NLIST	Directory name	List the names of subdirectories or files without other attributes
MKD	Directory name	Create a new directory
PWD		Display name of current directory
RMD	Directory name	Delete a directory
RNFR	File name (old file name)	Identify a file to be renamed

RNTO	File name (new file name)	Rename the file
SMNT	File system name	Mount a file system

- **Data formatting commands.** These commands let the user define the data structure, file type, and transmission mode. The defined format is then used by the file transfer commands. Table 19.3 shows common command in this group.

Table Data formatting commands

Command	Argument(s)	Description
Type	A(ASCII),E(EBCDIC),I(Image), N(Nonprint), or T (TELNET)	Define the file type and id necessary the print format
STRU	F(File),R(Record),or P(page)	Define the organization of the data
MODE	S(stream), B(Block), or C(Compressed)	Define the transmission mode

Port defining commands. These commands define the port number for the data connection on the client site. There are two methods to do this. In the first method, using the PORT command, the client can choose an ephemeral port number and send it to the server using passive open. The server uses that port number and creates an active open. In the second method, using PASV command, the client just asks the server to first choose a port number. The server does a passive open on that port and sends the port number in the Response (see response numbered 227 in Table 19.7). The client issues an active open using that port number. Table Port defining commands

Command	Argument(s)	Description
PORT	6-digit identifier	Client chooses a port
PASV		Server chooses a port

File transfer commands. These commands actually let the user transfer files. Table 19.5 lists common commands in this group.

Command	Argument(s)	Description
RETR	File name(s)	Retrieve files; file(s) are transferred from server to the client
STOR	File name(s)	Store files; file(s) are transferred from the client to the server
APPE	File name(s)	Similar to STOR except if the file exists, data must be appended to it

- **Miscellaneous commands.** These commands deliver information at the FTP user at the client site. Table 19.6 shows common commands in this group.

Table 19.6 Miscellaneous commands

Command	Argument(s)	Description
HELP		Ask information about he server
NOOP		Check if server is alive
SITE	Commands	Specify the site-specific commands
SYST		Ask about operating system used by the server

Responses

Every FTP command generates at least one response. A response has two parts: a three digit number followed by text. The numeric part defines the code; the text part defines needed parameters or extra explanations. We represent the three digits as xyz. The meaning of each digit is described below.

First digit The first digit defines the status of the command. One of five digits can be used in this position:

- **1yz (positive preliminary reply).** The action has started. The server will send another reply before accepting another command.
- **2yz (positive completions reply).** The action has been completed. The server will accept another command.
- **3yz (positive intermediate reply).** The command has been accepted, but further information is needed.

- **4yz (transient negative completion reply).** The action did not take place, but he error is temporary. The same command can be sent later.
- **5yz (permanent negative completion reply).** The command was not accepted and should not be retried again.

Second Digit

The second digit also defines the status of the command. One of six digits can be used in this position:

- **X0z (syntax).**
- **X1z (information).**
- **X2z (connections).**
- **X3z (authentication and accounting).**
- **X4z (unspecified)**
- **X5z (file system).**

Third digit The third digit provides additional information.

Table shows a brief list of possible responses (using all three digits).

Code	Description
Positive Preliminary Reply	
120	Service will be ready
125	Data connection open; data transfer will start shortly
150	File status is OK; data connection will be open shortly
Positive completion reply	
200	Command OK
211	System status or help reply

212	Directory status
213	File status
214	Help message
215	Naming the system type(operating system)
220	Service ready
221	Service closing
225	Data connection open
226	Closing data connection
227	Entering passive mode; server sends its IP address and port number
230	User login OK
250	Request file action OK
Positive intermediate reply	
331	User name OK; password is needed
332	Need account for logging
350	The file action is pending; more information needed
Transient negative completion reply	
425	Cannot open data connection
426	Connection closed; transfer aborted
450	File action not taken; file not available
451	Action aborted; local error

Table Responses (continued)

Code	Description

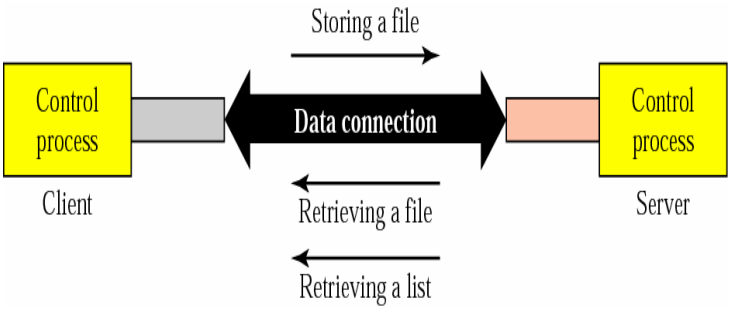
501	Syntax error in parameters or arguments
502	Command not implemented
503	Bad sequence of commands
504	Command parameter not implemented
530	User not logged in
532	Need account for storing file
550	Action is not done; file unavailable
552	Requested action aborted; exceed storage allocation
553	Requested action not taken; file name not allowed

File Transfer

File transfer occurs over the data connection under the control of the commands sent over the control connection. However, we should remember the file transfer in FTP means one of three things (see Figure 19.7).

- A file is to be copied from the server to the client. This is called retrieving a file. It is done under the supervision of the RETR command.
- A file is to be copied from the client to the server. This is called storing a file. It is done under the supervision of the STOR command.
- A list of directory or file names is to be sent from the server to the client. This is done under the supervision of the LIST command. Note that FTP treats a list of directory or file names as a file**.** It is sent over the data connection.

Figure File transfer



Anonymous FTP

To use FTP, a user needs an account (user name) and a password on the remote server. Some sites have a set of files available for public access. To access these files, a user does not need to have an account or password. Instead, the user can use anonymous as the user name and guest as the password.

User access to the system is very limited. Some sites allow anonymous users only a subset of commands. For example, most sites allow the user to copy some files, but do not allow navigation through the directories.

Flow and Error Control

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

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

The lack of **flow control and error control** means that the process using UDP should provide for these mechanisms.

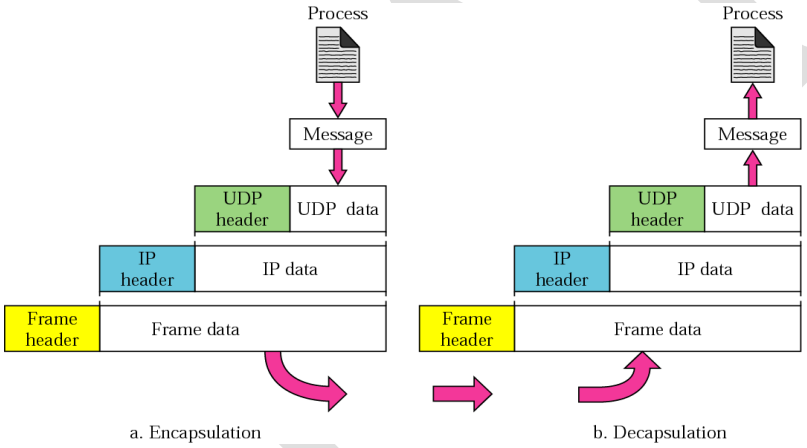
Encapsulation and Decapsulation

To send a message from one process to another, the UDP protocol encapsulates and decapsulates messages

Encapsulation

When a process has a message to send through UDP, it passes the message to UDP along with a pair of socket addresses and the length of data. UDP receives the data and adds the UDP header.

Figure 11.10 *Encapsulation and decapsulation*



2. Simple Mail Transfer Protocol

2.1. Introduction

2.1.1 Internet

Simple Mail Transfer Protocol is a protocol in the vast field of Internet. The Internet is a large network of interconnected computers all around the block. It is a reservoir of information sources and it provides service to everyone who has the access to it throughout the world. The Internet provides a spectrum of services such as electronic mailing, file transfer and file locators. It also act as a platform for many other activities such as publishing documents over the WWW in the form of web pages, conducting business over the net , forming discussion groups and bulletin board services, advertising of services and products. The other services that are provided by the Internet are access to databases. In addition, it allows the user to download enormous information and also provides resource haring among the computers connected on the network.

The hardware part of Internet is just the physical components like the computer either Client or Server, Modem and the connecting media. The hardware is just the body the life to it is given by the software, which is on the Internet. It provides access to the information n the Internet. There are many software products that are associated with Internet like MOSAIC, INTERNET EXPLORER, NETSCAPE NAVIGATOR etc. these are called web browsers. They are used to help the clients to download, and search the required information on the Internet.

World Wide Web

The Web is an architectural framework for accessing linked documents spread out over thousands of machines all over the Internet. The Web (also known as WWW) began in 1989 at CERN, the European center for nuclear research. CERN has large team of scientists from European countries carrying out research in particles physics. The Web grew out of the need to have these large teams of internationally dispersed researchers collaborate using a constantly changing collection of reports, blueprints, drawings, photos and other documents. Since the Web is basically a Client -Server system this Research Paper discusses both (i.e. .user side and server side [1]).

The Client Side

The Web consists of a vast, world wide collection of documents, called Web pages. Each page contains links (pointers) to other, related pages, anywhere in the world .Users can follow a link (e.g., by clicking on it), which then takes them to the page pointed to. This process can be repeated indefinitely possibly traversing hundreds of linked pages while doing so. Pages that point to other pages are said to use by hypertext.

Pages are viewed with a program called a browser, of which Mosaic and Netscape are two popular ones. The browser fetches the page requested, interprets the text and formatting commands that it contains and displays the page properly formatted on the screen. Strings of text that are links to other pages, called hyperlinks, are highlighted, either by underlining, displaying them in a special color or both. To follow the link, the user places the cursor on the highlighted are (using the mouse or the arrow keys) and select it (by clicking a mouse button or hitting ENTER).

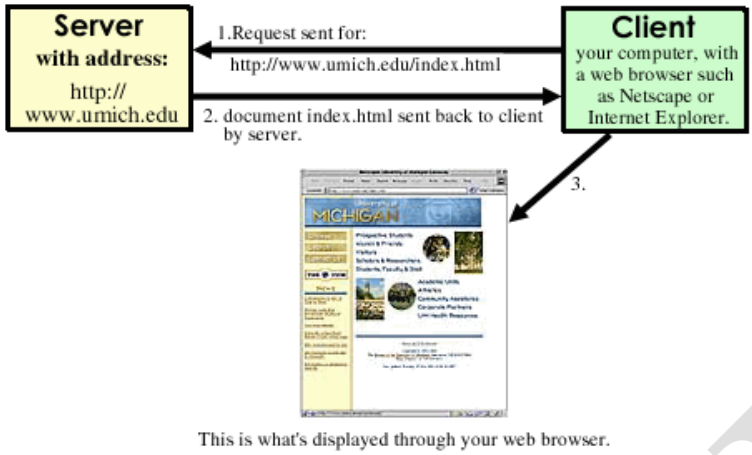
In addition to having ordinary text (not underlined) and hypertext (underlined), Web pages can also contain icons, line drawings, maps and photographs. Each of these can (optionally) be linked to another page. Clicking on one of these elements causes the browser to fetch the linked page and display it, the same as clicking on text .With images such as photos and maps , which page is fetched next may depend on what part of the image was clicked on.

Not all pages are viewable in the conventional way. For example, some pages consist of audio tracks, video clips or both. When hypertext pages are mixed with other media, the result is called hypermedia. To host a web browser, a machine must be directly on the internet or at least have a SLIP (Serial Line IP) or PPP (Point –to-Point Protocol) connection to a router or other machine that is directly on the Internet. This requirement exists because the way a browser fetches a page is to establish a TCP connection to the machine where the page is, and then send a message over the connection asking for the page. If it cannot establish a TCP connection to an arbitrary machine on the Internet, a browser will not work

The Server Side

Every website has a server process listening to TCP port 80 for incoming connections from clients (normally browsers). After a connection has been established, the client sends one request and the server sends one reply. Then the connection is released. The protocol that defines the legal requests and replies is called HTTP.

The user clicked on some pieces of text or perhaps on the icon those points to the URL (Uniform Resource Locator). URL has three parts: the name of the protocol, the name of machine where the page is loaded and the name of the file containing the page



This is what's displayed through your web browser.

The steps that occur between the user's click and the page being displayed are as follows.

1. The browser determines the URL.
2. The browser asks DNS for the IP address of the website.
3. DNS replies with IP address.
4. The browser makes a TCP connection to port 80 on IP address.
5. If then sends a GET command for getting information.
6. The server sends the requested file.
7. The TCP connection is released.
8. The browser displays all the text in the requested web site.
9. The browser fetches and displays all the images in the requested web site.

Many browsers display which step they are currently executing in a status line at the bottom of the screen. In this way, when the performance is poor, the user can see if it is due to DNS (Domain Name System) not responding, the server not responding, or simply network congestion during page transmission.

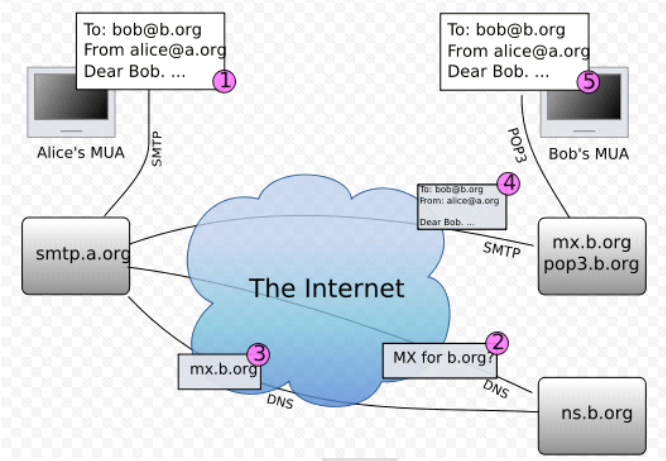
1.2 The E-Mail System

In present day life, communication plays a vital role. Some of the traditional communication facilities like postal service, fax, couriers have many drawbacks. These drawbacks are overcome by using the fast growing electronic communication called electronic mail or E-Mail. The term E-Mail simply refers to transfer of text messages from one computer to another over a type of network. E-mail can be used to send spread sheet files, word processing, documents, programs, images, voice messages and faxes. Almost anything can be stored electronically and can be sent.

1.2.1 The Working of E-Mail

To send a message from local computer to another user through E-Mail system, the client software creates the message in a temporary storage space. If the user tells the system to send the message, the system transfers that to a message database on the mail server. This Message is stored in a database in encrypted format. Thus casual browsing does not reveal its content. The mail system increments a counter in a recipient's mailbox to indicate the user of the new mail's presence.

The next time that the user checks his mail, he would see that he has one new message. If he issues the command to read that message, the mail system decrypts the message and sends it to his PC for display. If he chooses to save the message, it remains in the message database in its encrypted form. On the other hand, if he decides to delete it, the mail system remove it from the message database and decrements the counter in his mailbox. If he decides to save a cop to his local hard disk, it saves it in decrypted form.



Addressing and Standards

Addressing an e-mail message means putting information in the header that will enable the sending and receiving computers to deliver the message correctly. There are lots of different e-mail systems, each with its own addressing skills. The addressing skills depend on network protocol. [3] E -

mail would look pretty straightforward. There are different e-mail systems, each with its own addressing schemes.

The Internet Standard

Most of the WAN e-mail sent within North America is arranged in the form at known as Internet format. Because the Internet is the largest and most well known network, which uses this format, hence it's often called the internet format, but the actual name of the standard is RFC – 822. Examples of e-mail addresses that use this format are

Mrobbins@bga.com

Etittel@zilker.net

The naming system on the internet is called the Domain Name System abbreviated as DNS. Thus, addresses in the Internet format are also called DNS addresses. DNS addressing will be the only one standard on which the world will settle, largely because there is already a considerable volume of software that uses it; and also it has been deployed in so many networks throughout the words.

Store and Forward

There are two types of messaging systems. They are

1. Real –time systems

2. Store –and-forward systems.

In the real-time systems one can send messages only to users who are currently logged in to the system. In Store-and-Forward systems messages are held up in a central repository until users retrieve them. This system also queues up mail coming from different networks and sends groups of messages in batches, instead of sending each message as soon as it arrives. The delays introduced by intermediate storage are usually invisible to the users.

A big advantage of store-and-forward systems is that they are less vulnerable to network problems than real time systems. If a link foes down in a real-time-system, no message can get through until the link recovers. If a link goes down in a store-and-forward system, the system queue incoming messages and delivers after the link are back up.

Mailbox

A mailbox is a place where the e-mail system stores messages for that user. It contains messages received and read messages, which are not yet deleted. The user needs to clean out his mailbox from time to time by deleting unwanted messages because storing in his mailbox would clog the mailbox.

1.2.2 E-Mail Architecture

E-mail systems consist of two subsystems: the user agents, allows people to read and send e-mail, and the message transfer agents, moves the messages from the source to the destination. The user agents are local programs that provide a command-based, menu based or graphical method for interacting with the email system. The message transfer agents are typically system daemons that run in the background and move email through the system.

E – Mail Using Browser

Most users log on to a web page using corresponding web servers for sending mails. As the client requests for that page the web server responds using HTTP protocol. For sending mails, the browser used by the client gives the date to the web server, which in turn gives to the middle ware. The middle ware converts the data into a mail format which mail server can understand. To make the mail server understand the data, it should be converted into commands [3].

SMTP commands are used for sending mails. So the data from the web server are converted into various SMTP commands by the middle ware (Java Server Pages [JSP], Active Server Pages [ASP]) used. These commands are given to the mail server and the responses are evaluated. This mail server is called as Sender SMTP server. This server transfers the mails to the corresponding destination servers (Fig.1.2).

E-Mail Using Mail Client

Mail client software is designed in such a way that it abides the protocols. As mail client follows the protocols, for sending mails the SMTP commands are directly send to the source mail server. This in turn passes the mail to the destination mail server. (Fig.1.3)[3]

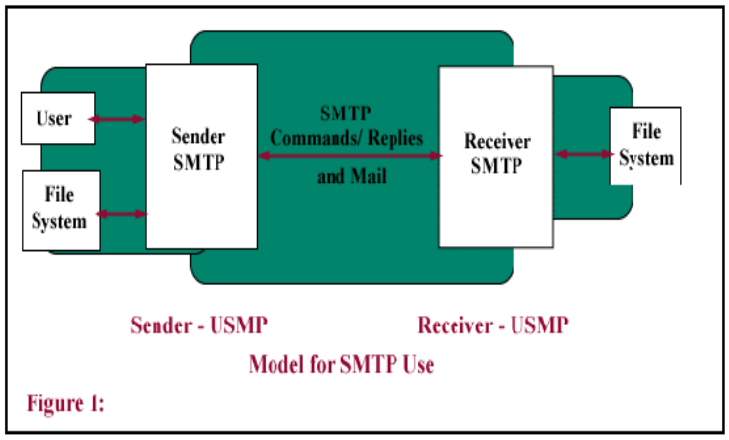


Fig.1.3 E-mail Using Mail Client

E – Mail Sending Process

The SMTP provides mechanisms for the transmission of mail directly from the sending user's host to the receiving user's host when the two hosts are connected via one or more SMTP servers. These transactions triggered by various SMTP commands.

The conversation between client and mail server are given below:

Sender: MAILFROM:Smith@Alpha.ARPA

Receiver: 250 OK

Sender: RCPT TO:Jones@Beta.ARPA

Receiver: 260 OK

Sender: DATA

Receiver: 354 Start mail input; end with<CRLF>.<CRLF>

Sender: “God is great” [Message]

Receiver: 250 OK

In this conversation there is no authentication happening. The sender can enter anything at the HELO prompt during identification. The same applies to the MAIL FROM prompt

Services of E-Mail

E-Mail system supports basic functions.

1 Composition refers to the process of creating messages and answers.
2 Transfer refers to moving messages from the originator to the recipient.

3 Reporting has to do with telling the originator what happened to the message.

4 Displaying incoming messages is needed so people can read their email.

5 Disposition is the final step and concerns what the recipient does with the message after receiving it. [1]

1.3 Mail Server and Mail Clients

1.3.1 Mail Server Details

Mail server is a software program that distributes files or information in response to requests sent via e-mail. The mail server sends and receives email messages are directed by the client. For those technical types, the Post Office Protocol version 3(POP3) servers receive mail and send them to the client. The Simple Mail Transfer Protocol (SMTP) server sends off e-mails to other computers. The mail server handles mailboxes for any number of users (or purposes).

These mailboxes can be accessed by the POP3 protocol , which is currently the most popular method for mail access ,using this mail's can be left on the server or downloaded to the user's own machine.

A server is used to store mail in a store-and-forward architecture. Workstations send mail to a mail server for forwarding, and retrieve any mail that might have been received from their electronic postbox. [24] Powerful administration facilities allow the automatic expiry of mail that has been left in the mailbox for too long, it also reduces the resource Demand without having to persuade the users to do these themselves.

Courier Mail Server

The courier mail transfer agent (MTA) is an integrated Mail/Group Ware server based on open commodity protocols, such as IM AP (Internet Mail Accessing protocol), POP3, SSL (Secure Socket Layer), LDAP (Lightweight Directory Access Protocol) and HTTP. Courier provides IMAP, POP3, Web Mail, and mailing lists services within a single, consistent, framework. Courier's source code should compile on POSIX- based operating systems based on Linux, and BSD – Derived Kernels. Courier should also compile on Solaris and AIX, with some help from Sun or IBM's freeware add-on tools for their respective operating systems. [17]. Courier implements SMTP extensions for mailing list management and spam filtering , Courier can function as an intermediate mail relay , relaying mail between an internal LAN and the Internet , or perform final delivery to mailboxes. Courier's configuration is set by plain text files and Perl scripts.

Courier can also provide mail services for operating system accounts, virtual mail accounts, managed by an LDAP, My SQL, or Postures-based authentication database. Certain portions of Courier - the mail filtering engine, the Web mail server and the IMAP server are also available in separate, smaller, packages that can be used with other mail servers.

Commands in SMTP protocol

Command Semantics

The SMTP commands define the mail transfer or the mail system function requested by the user. SMTP commands are character strings terminated by <CRLF>. The command codes themselves are

alphabetic characters terminated by <SP> if parameters follow and <CRLG> otherwise. The syntax of mailboxes must conform to receiver site conventions. The SMTP commands are discussed below

A mail transaction involves several data objects, which are communicated as arguments to different commands. The reverse – path is the argument of the MAIL command, the forward –path is the argument of the RCPT command, and the mail data is the argument of the DATA command. Those arguments or data objects must be transmitted and held pending till the confirmation is communicated by the end of email, which finalizes the transaction. The model for this is that distinct buffers are provided to hold the types of data objects, that is, there is a reverse – path buffer, a forward-path buffer, and a mil data buffer. Specific commands cause information to be appended to a specific buffer, or cause one or more buffers to be cleared.

Hello (HELO)

This command is used to identify the sender – SMTP to the receiver –SMTP. The arguments field contains the host name of the sender-SMTP.

The receiver – SMTP identifies itself to the sender-SMTP through the greeting reply, and through then response to this command. This command and an OK reply to it confirm that both the sender – SMTP and the receiver-SMTP are in the initial state, that is, there is no transaction in progress and all state tables and buffers are cleared.

Mail (MAIL)

This command is used to initiate a mail transaction in which the mail data is delivered to one or more mailboxes. The argument field contains a reverse-path.

The reverse-path consists of an optional list of hosts and the sender mailbox. When the lists of hosts are present, it is a “reverse” source route and indicates that the mail was relayed through each host on the list (the first host in the last was the most recent relay). This list is used as a source route to return non-delivery notices to the sender. As each relay host adds itself to the beginning of the list , it must use its name as known in the IPCE to which it is relaying the mail rather than the IPCE from which the mail came.(if they are different). In some types or error reporting messages (for e.g.., undeliverable mail notification) i.e... The reverse –path may be null.

This command cleans the reverse – path buffer, the forward – path buffer, and the mail data buffer; and inserts the reverse – path information from this command into the reverse-path buffer.

This command is used to identify an individual recipient of the mail data; multiple recipients are specified by multiple use of this command. The forward – path consists of an optional list of hosts and a required destination mailbox. When the list of hosts is present, it is a source route and indicates that the mail must be relayed to the next host on the list. If the receiver – SMTP does not implement the relay function it may be send the user the same reply.

When the mail is relayed, the relay host must remove from the beginning forward – path and put itself at the beginning of the reverse- path. When the mail reaches its ultimate destination (the forward path contains only the destination mail box,) the receiver SMTP inserts it into the destination mail box in accordance with its host mail conventions.

For egg; mail received at the relay host A with arguments

FROM : <USERX@HOSTY.ARPA

TO :< @HOSTA.ARPA, @HOSTB.ARPA:USERC@HOSTD.ARPA>

Will be relayed on to host B with arguments

FROM :< @HOSTA.ARPA:USERX@HOSTY.ARPA>

TO: <@HOSTB.ARPA:USERC@HOSTD.ARPA>

Data (DATA)

The receiver treats the following the command as mail data from the sender. This command causes the mail data from this command to be appended to the mail data buffer

The mail data may contain any of the 128 ASCII character codes. The mail data is terminated by a line containing only a period. , that the character sequence “<CRLF>, <CRLF>. This is the end of the mail data indication.

The end of the mail data indication that the receiver must now process the stored mail transaction information. This processing consumes the information in the reverse-path buffer, the forward path buffer and the mail data buffer, and the completion of this command this buffers are

cleared. If the processing, is successful the receiver must send an OK reply. If the processing is fails completely receiver must send a failure reply.

When the receiver – SMTP accepts a message either for relaying or for final delivery it inserts at the beginning of the mail data a time stamp line. The time stamp line indicates the identity of the host machine that send the message, and identity that receive the message (and is inserting the time stamp), and the time and date the message was received. Relayed message will have multiple time stamp lines.

When the receiver-SMTP makes the 'final delivery' of a message it inserts at the beginning of the mail data a return path line. The return path line preserves the information in the <reverse-path> from the mail command. Here, final delivery means the message leaves the SMTP world. Normally, this would be meaning it has been delivered to the destination user, but in some cases it may be further processed and transmitted by another mail system. It is possible for the mailbox in the return path be different from the actual sender's mailbox, for example if error responses are to be delivered a special error handling mailbox rather than the message senders.

The preceding two paragraphs imply that the final mail data will begin a return path line, followed by one or more time stamp lines. Special mention is needed of the response and further action required when the processing following the end of the mail data indication is partially successful. This could arise if after accepting several recipients and the mail data , the receiver-SMTP finds that the mail data can be successfully delivered to some of the recipients , but it cannot be to others (for egg ; due to mailbox space allocation problems). In such a situation, the response to the DATA command must be an OK reply. But, the receiver –SMTP must compose and send an "undeliverable mail" notification message to the originator of the message. Either a single which lists all of the notifications which lists all of the recipients that failed to get the message, or separate notification messages are sent using the MAIL command.

Example of RETURN path and RECEIVED time stamps

Return path :<@GHI.ARPA,@DEF.ARPA,@ABC.ARPA:JOE@ABC.ARPA>

Received : from GHI.ARPA by JKL.ARPA ; 27 OCT 81 15:27:39 PST

Received : from DEF.ARPA BY GHI.ARPA ; 27 OCT 81 15:15:13 PST

Received :from ABC.ARPA by DEF.ARPA ; 27 OCT 81 15:01:59 PST

Date : 27 OCT 81 15:01:01 PST

From :JOE@ABC.ARPA

Subject : Improved mail system installed

To :SAM@JKL.ARPA

Send(SEND)

This command is used to initiate a mail transaction in which the mail data is delivered to one or more terminals. The argument field contains a reverse path. This command is successful if the message is delivered to a terminal. The reverse path consists of an optional list of hosts and the sender mailbox. When the list of hosts is present, it is a reverse source route and indicates that the mail was relayed through each host on the list (the first host in the list was the most recent relay).

This list is used as a source route to return non-delivery notices to the sender , as each relay host adds itself to the beginning of the list , it must use its name as known on the IPCE to which it is relaying the mail than the IPCE from which the mail came (if they are different)

This command clears the reverse path buffer, the forward path buffer, and the mail data buffer, and inserts the reverse path information from this command into the reverse path buffer.

Verify (VRFY)

This command asks the receiver to conform that the argument identifies a user. If it is a user name, the full name of the user (if known) and the fully specified mailbox are returned.

This command has no effect on any of the reverse path buffer, the forward path buffer, or the mail data buffer.

Quit (QUIT)

This command specified that the receiver must send an OK reply and then close the transmission channel. The receiver should not close the transmission channel until it receives and replies to a QUIT command. The sender should not close the transmission channel until it sends a QUIT command and receives the reply. If the connection is closed prematurely the receiver should acts as a RSET command had been received (canceling any pending transaction, but not undoing any previously completed transaction), the sender should act as if the command or transaction in progress had received a temporary error.

Command Syntax

The command consists of a command code followed by an argument field. Command codes are four alphabetic characters. Upper and lower case alphabetic characters are to be treated identically. Thus, any of the following may represent the mail command: MAIL mail Mall mail

This also applies to any symbols representing parameter values such as “TO” or “to” for the forward path. Command codes and the argument fields are separated by one or more spaces. However, with in the reverse path and forward path arguments case is important. In particular, in some hosts the user “smith” is different from the user “SMITH”.

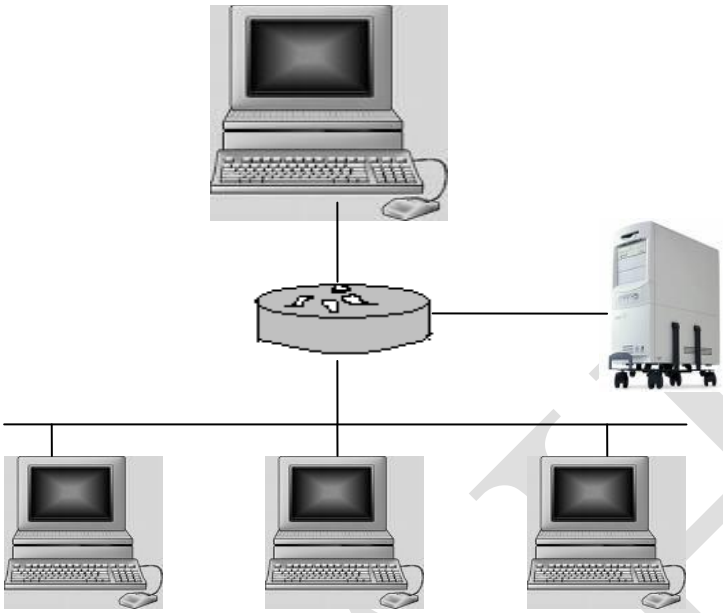
***3.* Simple Network Management Protocol**

Background

The Simple Network Management protocol (SNMP) is an application layer protocol that facilitates the exchange of the management information between network devices. It is part of the Transmission Control Protocol / Internet Protocol (TCP/IP) protocol suite. SNMP enables network administrators to manage network performance, find and solve network problems, and plan for network growth.

Two versions of SNMP exist: SNMP version 1 (SNMPv1) and SNMP version 2 (SNMPv2). Both versions have a number of features in common. but SNMPv2 offers enhancements , such as additional protocol operations. Standardization of yet another version of SNMP - SNMP version 3 (SNMPv3) – is

pending. This chapter provides descriptions of the SNMPv1 and SNMPv2 protocol operations. Figure 56-1 illustrates a basic network managed by SNMP.

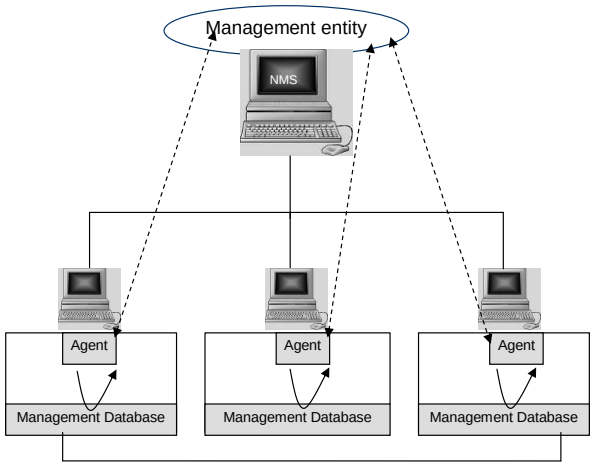


SNMP Basic Components

An SNMP - managed network consists of three key components: managed devices, agents, and network – management systems (NMSs).

A managed device is a network node that contains an SNMP agent and that resides on a managed network. Managed devices collect and store management information and make this information available to NMSs using SNMP. Managed devices, sometimes called network elements, can be routers and access servers, switches and bridges, hubs, computer hosts, or printers.

An agent is a network management software module that resides in a managed device. An agent ghas local knowledge of management information and translates that information into a form compatible with SNMP. An NMS executes applications that monitor and control managed devices. NMSs provide the bulk of the processing and memory resources required for network management. One or more NMSs must exist on any managed network.



SNMP Commands

Managed devices are monitored and controlled using four basic SNMP commands: read, write, trap, and traversal operations.

The read command is used by an NMS to monitor managed devices. The NMS examines the different variables that are maintained by the managed devices. The write command is used by an NMS to control managed devices. The NMS changes the values of the variables stored within managed devices. The trap command is used by the managed devices to asynchronously report events to the NMS. When certain types of events occur, a managed device sends a trap to the NMS.

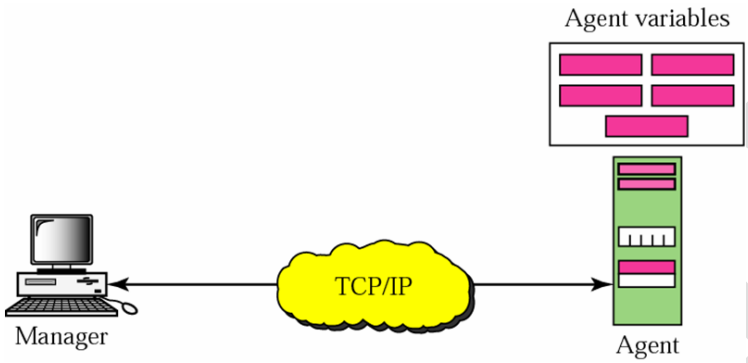
Traversal operations are used by the NMS to determine which variables a managed device supports and to sequentially gather information in variable tables, such as a routing table.

Network Management Architecture

Network management system contains two primary elements. A manager and agents. The manager is the console through which the network administrator performs network management functions. Agents are the entities that interface to the actual device being managed. Bridges, hubs, routers or network servers are examples of managed devices that contain managed objects. These managed objects might be hardware, configuration parameters, performance statistics, and so on, that directly relate to the current operation of the device in question. These objects are arranged in what is

known as a virtual information database, called a management information base, also called MIB. SNMP allows managers and agents to communicate for the purpose of accessing these objects.

The model of network management architecture looks like this:



typical agent usually:

1 Implements full SNMP protocol.
2 Stores and retrieves management data as defined by the MIB.
3 Can asynchronously signal an event to the manager.
4 Can be a proxy for some non-SNMP manageable network node. Click here to see typical proxy architecture.

Atypical manager usually:

1 Implemented as a Network Management Station (the NMS)
2 Implements full SNMP protocol
3 Able to send Query
4 Get responses from agents
5 Set variables in agents
6 Acknowledge asynchronous events from agents

Some prominent vendors offer network management platforms which implement the role of the manager (listed in alphabetic order):

1 Dec PolyCenter Network Manager

2 Hewlett – Packard Open View
3 IBM AIX NetView/6000
4 SunConnect SunNet Manager

Management Information Base

Management Information Bases (MIBs) are a collection of definitions, which define the properties of the managed object within the device to be managed. Every managed device keeps a database of values for each of the definitions written in the MIB. It is not the actual database itself – it is the implementation dependent. Definition of the MIB conforms to the SMI given in RFC 1155. Latest Internet MIB is given in RFC 1213 sometimes called the MIB-II. Click here to see MIB architecture. You can think of a MIB as an information warehouse.

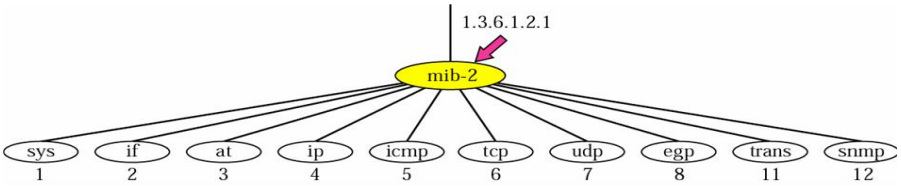
Criteria and Philosophy for standardized MIB

1 Objects have to be uniquely named
2 Objects have to be essential
3 Abstract structure of the MIB needed to be universal
4 For the standard MIB maintain only a small number of objects
5 Allow for private extensions
6 Object must be general and not too device dependent
7 Objects cannot be easily derivable from their objects
8 If agent is to be SNMP manageable then it is mandatory to implement the Internet MIB (currently given as MIB-II in RFC 1157)

Naming an object

1. Universal unambiguous identification of arbitrary objects
2. Can be achieved by using an hierarchical tree
3. Based on the Object Identification Scheme defined by OSI

The Registered Tree



Identifiers

1. Object name is given by its name in the tree.
2. All child nodes are given by the unique integer values within the new sub-tree.
3. Children can be parents of further child sub-tree (ie: they have subordinates) where the numbering scheme is recursively applied.
4. The Object Identifier (or name) of an object is the sequence of non-negative Integer values traversing the tree to the node required.
5. Allocation of an integer value for a node in the tree is an act of registration by whoever has delegated authority for that sub tree.
6. This process can go to an arbitrary depth.
7. If a node ha children then it is an aggregate node.
8. Children of the same parent cannot have the same integer value.

Object and Object Identifiers

1. Object is named or identified by the sequence of integers in traversing the tree to the object type required
2. This does not identify an instance of the object
3. The Object Identifier(OID) is shown in afew ways with a.b.c.d.e being the preferred
4. OIDs can name many types of objects:

The Internet Sub – tree

- Directory sub-tree if for future directory services
- Experimental sub-tree is for experimental MIB work – still
- Has to be registered with the authority (IESG)
- MIB sub-tree is the actual mandatory Internet MIB for all
- Agents to implement (currently MIB-II RFC 1156- this is the Only sub-tree for management)
- Enterprise sub-tree (of private) are MIBs of proprietary objects And are of course not mandatory (sub-tree registered with Internet assigned numbers authority) for example: CISCO
- Router OID: 1.3.6.1.4.1.9.1.1

- SNMP management nearly always Internet in MIB and specific enterprises MIBs.

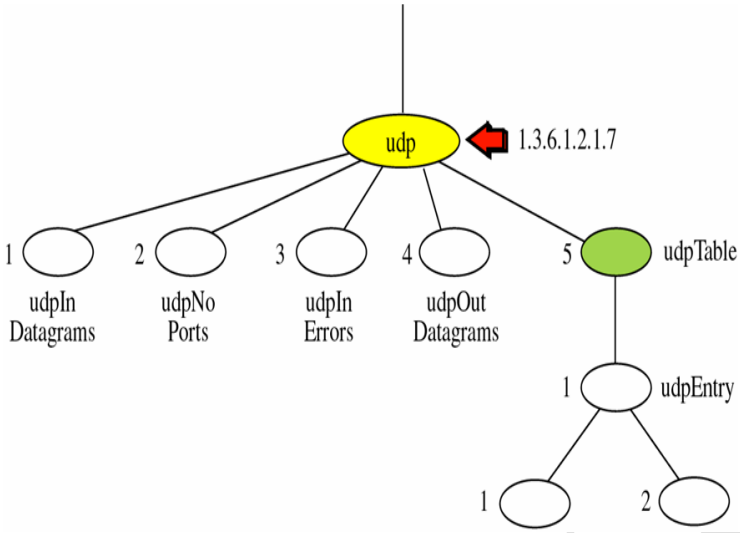
MIB-II Standard Internet MIB

1. Definition follows structure given in SMI
2. MIB-II (RFC 1213) is current standard definition of the virtual file store for SNMP manageable objects
3. Has 10 basic groups
    - System
    - Interfaces
    - At
    - Ip
    - Icmp
    - Tcp
    - Udp
    - Egp
    - Transmission
    - Snmp

If agent implements any group then is has to implement all of the managed objects within the group. An agent does not have to implement all groups. Note: MIB –I and MIB-II have some OID (position in the Internet sub-tree)

MIB-II

The MIB sub-tree



Note: there is an object cm OT (9) under the MIB but it has become almost superfluous and for all intense and purposes is not one of the SNMP manageable groups within MIB.

SNMP Protocol

SNMP is based on the managers/ agent model. SNMP is referred to as "simple" because the agent requires minimal software. Most of the processing power and the data storage reside on the management system, while a complementary subset of those functions resides in the managed system.

To achieve its goal of being simple, SNMP includes a limited set of management commands and responses. The management system issues Get, GetNext and Set messages to retrieve single or multiple object variables or to establish the value of a single variable. The managed agent sends a response message to complete the Get, GetNext or Set. The managed agents send an event notification, called a trap to the management system to identify the occurrence of conditions such as threshold that exceeds a predetermined value. In short there are only five primitive operations:

1 Get(retrieve operation)
2 Getnext(traversal operation)
3 Getresponse(indicative operation)
4 Set(alter operation)
5 Trap(asynchronous trap operation)

SNMP Message Construct

Each SNMP message has the format:

1. Version number
2. Community name – kind of a password
3. One or more SNMP PDUs – assuming trivial authentication

Each SNMP PDU except trap has the following format:

1. Request id – request sequence number
2. Error status – zero if no error otherwise one of a small set
3. Error index – if non zero indicates which of the OIDs in the PDU caused the error 2

4. List of OIDs and values - values are null for get and getnext

Trap PDUs have the following format:

1. Enterprise – identifies the type of object causing the trap
2. Agent address – IP address of agent which sent a the trap
3. Generic trap id – the common standard traps
4. Specific trap id – proprietary or enterprise trap
5. Time stamp – when trap occurred in time ticks
6. List of OIDs and values – OIDs that may be relevant to Send to the NMS

Outline of the SNMP protocol

1. Each SNMP managed object belongs to a community
2. NMS station may belong to multiple communities
3. A community is defined by a community name which is an OctetString with 0 to 255 octets in length.

Security levels with basic SNMP

Authentication

1. Trivial authentication based on plain text community name exchanged in SNMP message
2. Authentication is based on the assumption that the message is not tampered with or interrogated

Authorization

1. Once community name is validated then agent or manager checks to see if sending address is permitted or has the rights for the requested operation
2. "View" or "cut" of the objects together with permitted access rights is then derived for the pair(community name , sending address)

Summary

1 not very secure
2 SNMP version2 is addressing this
3 Extended security is possible with current protocol (eg: DES and MD5)
4 Does not reduce its power for monitoring

4. Mobile IP

4.1 Introduction to Mobile IP

Mobile IP is an open standard, defined by the Internet Engineering Task Force (IETF) RFC 2002, that allows users to keep the same IP address, stay connected, and maintain ongoing applications while roaming between IP networks. Mobile IP is scalable for the Internet because it is based on IP—any media that can support IP can support Mobile IP.

The number of wireless devices for voice or data is projected to surpass the number of fixed devices. Mobile data communication will likely emerge as the technology supporting most communication including voice and video. Mobile data communication will be pervasive in cellular systems such as 3G and in wireless LAN such as 802.11, and will extend into satellite communication.

Though mobility may be enabled by link-layer technologies, data crossing networks or different link layers is still a problem. The solution to this problem is a standards-based protocol, Mobile IP.

Mobile IP Overview

In IP networks, routing is based on stationary IP addresses, similar to how a postal letter is delivered to the fixed address on the envelope. A device on a network is reachable through normal IP routing by the IP address it is assigned on the network.

The problem occurs when a device roams away from its home network and is no longer reachable using normal IP routing. This results in the active sessions of the device being terminated. Mobile IP was created to enable users to keep the same IP address while traveling to a different network (which may even be on a different wireless operator), thus ensuring that a roaming individual could continue communication without sessions or connections being dropped.

Because the mobility functions of Mobile IP are performed at the network layer rather than the physical layer, the mobile device can span different types of wireless and wireline networks while maintaining connections and ongoing applications. Remote login, remote printing, and file transfers are some examples of applications where it is undesirable to interrupt

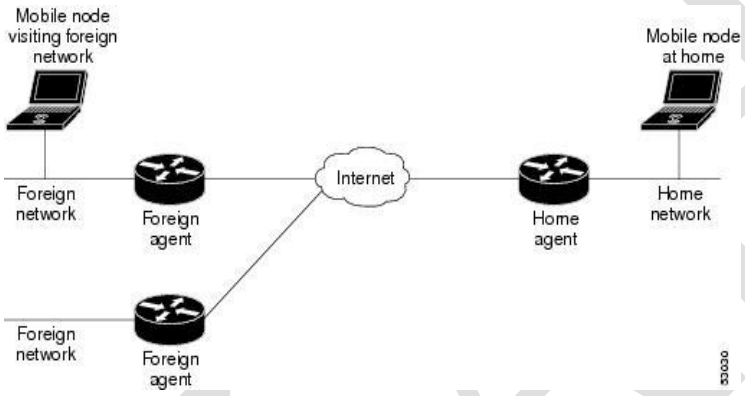
communications while an individual roams across network boundaries. Also, certain network services, such as software licenses and access privileges, are based on IP addresses. Changing these IP addresses could compromise the network services.

Components of a Mobile IP Network

Mobile IP has the following three components, as shown in Figure 1:

- Mobile Node
- Home Agent
- Foreign Agent

Figure 1 Mobile IP Components and Relationships



The Mobile Node is a device such as a cell phone, personal digital assistant, or laptop whose software enables network roaming capabilities.

The Home Agent is a router on the home network serving as the anchor point for communication with the Mobile Node; it tunnels packets from a device on the Internet, called a Correspondent Node, to the roaming Mobile Node. (A tunnel is established between the Home Agent and a reachable point for the Mobile Node in the foreign network.)

The Foreign Agent is a router that may function as the point of attachment for the Mobile Node when it roams to a foreign network, delivering packets from the Home Agent to the Mobile Node.

The care-of address is the termination point of the tunnel toward the Mobile Node when it is on a foreign network. The Home Agent maintains an association between the home IP address of the Mobile Node and its care-of address, which is the current location of the Mobile Node on the foreign or visited network

How Mobile IP Works

This section explains how Mobile IP works. The Mobile IP process has three main phases, which are discussed in the following sections.

Agent Discovery

A Mobile Node discovers its Foreign and Home Agents during agent discovery.

Registration
The Mobile Node registers its current location with the Foreign Agent and Home Agent during registration.

Tunneling
A reciprocal tunnel is set up by the Home Agent to the care-of address (current location of the Mobile Node on the foreign network) to route packets to the Mobile Node as it roams.

Agent Discovery

During the agent discovery phase, the Home Agent and Foreign Agent advertise their services on the network by using the ICMP Router Discovery Protocol (IRDP). The Mobile Node listens to these advertisements to determine if it is connected to its home network or foreign network.

The IRDP advertisements carry Mobile IP extensions that specify whether an agent is a Home Agent, Foreign Agent, or both; its care-of address; the types of services it will provide such as reverse tunneling and generic routing encapsulation (GRE); and the allowed registration lifetime or roaming period for visiting Mobile Nodes. Rather than waiting for agent advertisements, a Mobile Node can send out an agent solicitation. This solicitation forces any agents on the link to immediately send an agent advertisement.

If a Mobile Node determines that it is connected to a foreign network, it acquires a care-of address. Two types of care-of addresses exist:

- Care-of address acquired from a Foreign Agent
- Colocated care-of address

A Foreign Agent care-of address is an IP address of a Foreign Agent that has an interface on the foreign network being visited by a Mobile Node. A Mobile Node that acquires this type of care-of address can share the address with other Mobile Nodes. A colocated care-of address is an IP address temporarily assigned to the interface of the Mobile Node itself. A colocated care-of address represents the current position of the Mobile Node on the foreign network and can be used by only one Mobile Node at a time.

When the Mobile Node hears a Foreign Agent advertisement and detects that it has moved outside of its home network, it begins registration.

Registration

The Mobile Node is configured with the IP address and mobility security association (which includes the shared key) of its Home Agent. In addition, the Mobile Node is configured with either its home IP address, or another user identifier, such as a Network Access Identifier.

The Mobile Node uses this information along with the information that it learns from the Foreign Agent advertisements to form a Mobile IP registration request. It adds the registration request to its pending list and sends the registration request to its Home Agent either through the Foreign Agent or directly if it is using a colocated care-of address and is not required to register through the Foreign Agent. If the registration request is sent through the Foreign Agent, the Foreign Agent checks the validity of the registration request, which includes checking that the requested lifetime does not exceed its limitations, the requested tunnel encapsulation is available, and that reverse tunnel is supported. If the registration request is valid, the Foreign Agent adds the visiting Mobile Node to its pending list before relaying the request to the Home Agent. If the registration request is not valid, the Foreign Agent sends a registration reply with appropriate error code to the Mobile Node.

The Home Agent checks the validity of the registration request, which includes authentication of the Mobile Node. If the registration request is valid, the Home Agent creates a mobility binding (an association of the Mobile Node with its care-of address), a tunnel to the care-of address, and a routing entry for forwarding packets to the home address through the tunnel.

The Home Agent then sends a registration reply to the Mobile Node through the Foreign Agent (if the registration request was received via the Foreign Agent) or directly to the Mobile Node. If the registration request is not valid, the Home Agent rejects the request by sending a registration reply with an appropriate error code.

The Foreign Agent checks the validity of the registration reply, including ensuring that an associated registration request exists in its pending list. If the registration reply is valid, the Foreign Agent adds the Mobile Node to its visitor list, establishes a tunnel to the Home Agent, and creates a routing entry for forwarding packets to the home address. It then relays the registration reply to the Mobile Node.

Finally, the Mobile Node checks the validity of the registration reply, which includes ensuring an associated request is in its pending list as well as proper authentication of the Home Agent. If the registration reply is not valid, the Mobile Node discards the reply. If a valid registration reply specifies that the registration is accepted, the Mobile Node is confirmed that the mobility agents are aware of its roaming. In the colocated care-of address case, it adds a tunnel to the Home Agent. Subsequently, it sends all packets to the Foreign Agent.

The Mobile Node reregisters before its registration lifetime expires. The Home Agent and Foreign Agent update their mobility binding and visitor entry, respectively, during reregistration. In the case where the registration is denied, the Mobile Node makes the necessary adjustments and attempts to register again. For example, if the registration is denied because of time mismatch and the Home Agent sends back its time stamp for synchronization, the Mobile Node adjusts the time stamp in future registration requests.

Thus, a successful Mobile IP registration sets up the routing mechanism for transporting packets to and from the Mobile Node as it roams.

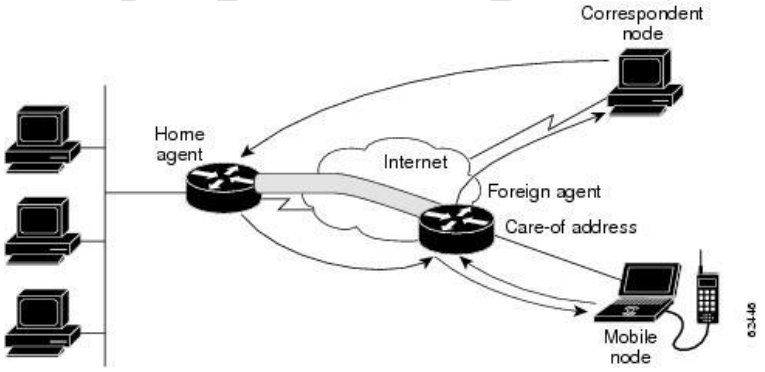
Tunneling

The Mobile Node sends packets using its home IP address, effectively maintaining the appearance that it is always on its home network. Even while the Mobile Node is roaming on foreign networks, its movements are transparent to correspondent nodes.

Data packets addressed to the Mobile Node are routed to its home network, where the Home Agent now intercepts and tunnels them to the care-of address toward the Mobile Node. Tunneling has two primary functions: encapsulation of the data packet to reach the tunnel endpoint, and decapsulation when the packet is delivered at that endpoint. The default tunnel mode is IP Encapsulation within IP Encapsulation. Optionally, GRE and minimal encapsulation within IP may be used.

Typically, the Mobile Node sends packets to the Foreign Agent, which routes them to their final destination, the Correspondent Node, as shown in Figure 2

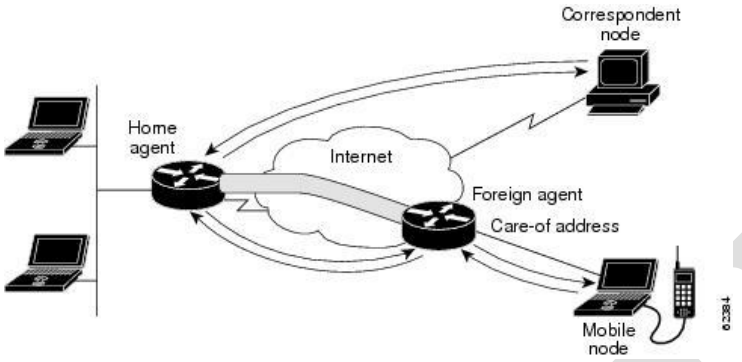
Figure 2 Packet Forwarding



However, this data path is topologically incorrect because it does not reflect the true IP network source for the data—rather, it reflects the home network of the Mobile Node. Because the packets show the home network as their source inside a foreign network, an access control list on routers in the network called ingress filtering drops the packets instead of forwarding

them. A feature called reverse tunneling solves this problem by having the Foreign Agent tunnel packets back to the Home Agent when it receives them from the Mobile Node.

Figure 3 Reverse Tunnel



Tunnel MTU discovery is a mechanism for a tunnel encapsulator such as the Home Agent to participate in path MTU discovery to avoid any packet fragmentation in the routing path between a Correspondent Node and Mobile Node. For packets destined to the Mobile Node, the Home Agent maintains the MTU of the tunnel to the care-of address and informs the Correspondent Node of the reduced packet size. This improves routing efficiency by avoiding fragmentation and reassembly at the tunnel endpoints to ensure that packets reach the Mobile Node.

Security

Mobile IP uses a strong authentication scheme for security purposes. All registration messages between a Mobile Node and Home Agent are required to contain the Mobile-Home Authentication Extension (MHAE).

The integrity of the registration messages is protected by a preshared 128-bit key between a Mobile Node and Home Agent. The keyed message digest algorithm 5 (MD5) in "prefix+suffix" mode is used to compute the authenticator value in the appended MHAE, which is mandatory. Mobile IP also supports the hash-based message authentication code (HMAC-MD5). The receiver compares the authenticator value it computes over the message with the value in the extension to verify the authenticity.

Optionally, the Mobile-Foreign Authentication Extension and Foreign-Home Authentication Extension are appended to protect message exchanges between a Mobile Node and Foreign Agent and between a Foreign Agent and Home Agent, respectively.

Replay protection uses the identification field in the registration messages as a timestamp and sequence number. The Home Agent returns its time stamp to synchronize the Mobile Node for registration.

Cisco IOS software allows the mobility keys to be stored on an authentication, authorization, and accounting (AAA) server that can be accessed using TACACS+ or RADIUS protocols. Mobile IP in Cisco IOS software also contains registration filters, enabling companies to restrict who is allowed to register.

Solution to Network Mobility

Network mobility is enabled by Mobile IP, which provides a scalable, transparent, and secure solution. It is scalable because only the participating components need to be Mobile IP aware—the Mobile Node and the endpoints of the tunnel. No other routers in the network or any hosts with which the Mobile Node is communicating need to be changed or even aware of the movement of the Mobile Node. It is transparent to any applications while providing mobility. Also, the network layer provides link-layer independence, interlink layer roaming, and link-layer transparency. Finally, it is secure because the set up of packet redirection is authenticated.

5. Points to Remember

- FTP uses two well-known TCP Ports: Port 21 is used for the control connection, and port 20 is used for the data connection.
- The two FTP connections control and data use different strategies and different port numbers.
- The **data connection** uses the well-known port 20 at the server site.
- The FTP client and server, which run on different computers, must communicate with each other
- FTP uses the control connection to establish a communication between the client control process and the server control process.
- UDP is a very simple, unreliable transport protocol
- There are many software products that are associated with Internet like MOSAIC, INTERNET EXPLORER, and NETSCAPE NAVIGATOR
- The Web is an architectural framework for accessing linked documents spread out over thousands of machines all over the Internet.
- There are two types of messaging systems. They are Real –time systems and Store –and-forward systems
- E-mail systems consist of two subsystems: the user agents, allows people to read and send e-mail, and the message transfer agents, moves the messages from the source to the destination
- The Simple Network Management protocol (SNMP) is an application layer protocol that facilitates the exchange of the management information between network devices.
- Each SNMP managed object belongs to a community
- NMS station may belong to multiple communities

Possible Questions

Part B
(Each Question carries 6 Marks)

1. Describe in detail about message transfer agent.
2. Explain the different phases of mobile host in detail.
3. Discuss in detail about file transfer protocol.
4. Discuss in detail about ATM ARP.
5. Analysis commands and responses in SMTP
6. Analysis the operation of SNMP.
7. Write notes on logical IP subnet and ATM ARP operation.
8. Explain the operation of ATM WAN
9. Explain techniques to guarantee privacy for an organization in VPN technology
10. Discuss on Security levels in basic SNMP

Part C
(Each Question carries 10 Marks)

1. Compare the strategies used in an organization to achieve privacy and give a note on VPN technology.

Karpagam Academy of Higher Education
Coimbatore - 21

DEPARTMENT OF COMPUTER APPLICATIONS
CLASS : II MCA

TCP/IP (18CAP405N)

Unit V

S.NO	Question	Choice1	Choice2	Choice3	Choice4	Choice 5	Choice 6	**Answer**
1	IGMP stands for	Internet Group Management Protocol	Internet Group Maintenance Protocol	Information Group Management Protocol	Information Group Maintenance Protocol			Internet Group Management Protocol
2	_____ managers group membership	ICMP	IGMP	TCP	TCP/IP			IGMP
3	_____ is called a connectionless, unreliable transport protocol.	UDP	TCP	SCTP	FTP			UDP
4	UDP means _____	User Datagram Protocol	User Defined Protocol	User Derived Protocol	User Device Protocol			User Datagram Protocol
5	In Multicasting _____process multicast packet are encapsulated network.	Tunneling	Trimming	Transporting	Terminate			Tunneling
6	UDP provides _____ service.	connection-oriented	connectionless	managing	Transfer			connectionless
7	_____ protocol provide domain name services	Echo	daytime	name server	quote			name server
8	_____ protocol returns string of characters.	daytime	quote	chargen	RPC			daytime
9	RPC means	Remote Procedure Call	Remote Packet Call	Resource Procedure Call	Response Procedure Call			Remote Procedure Call

10	The connection establishment in ____is called three way handshaking.	UDP	FTP	TCP	TCP/IP			FTP
11	ISN means _____.	Initial Sequence Number	Initial Service Number	Initial Segment Number	Initial Segment Node.			Initial Sequence Number
12	_____ is a string of characters that hold some information	country domain	compression	cookie	Cookies			cookie
13	In TCP one end can store sending data while still receiving data is called.	full-close	half close	two-way handshaking	three way handshaking			half close
14	A _____ is a machine that goes through a limited no of rates.	Infinite state machine	finite state machine	unlimited statemachine	limited state machine			finite state machine
15	When client process has no more data to send issues an	active close	passive close	full close	half close			active close
16	In ____ protocol host uses a window for outbound communication.	UDP	FTP	Sliding window protocol	SMTP			Sliding window protocol
17	RTO means _____	Remote Time out	Retransmission Time in	Retransmission Time in	Remote timing			Retransmission Time in
18	One of the algorithms used in TCP congestion control is _____	Fast Start	Fast Stop	Slow Start	Slow stop			Slow Start
19	_____ defines the size of the buffer in the local TCP	Hardware Type	Protocol size	Software Size	Buffer Size			Buffer Size
20	_____ Protocol returns the Quote of the Day	Quote	Daytime	Users	Discard.			Quote
21	_____ address is used for Multicasting.	Class B	Class D	Class A	Class E			Class D
22	_____ address is used for future purpose	Class B	Class D	Class A	Class E			Class E

23	_____ refers to finding network address	Multihome	Mask	Routing	Host			Multihome
24	Vaiable length block is used in _____ address	Class address	classless addressing	Classfl address	LocalHostAddress			classless addressing
25	The two terms often used in classless addressing is _____ and _____	address, type	length, prefix	Prefix, prefix length	suffix, suffix length.			Prefix, prefix length
26	In fixed length subnetting, the number of subnets is power of _____	4	5	3	2			5
27	Occasionally used term in classless addressing are _____ and _____	address, type	length, prefix	Prefix, prefix length	suffix, suffix length.			suffix, suffix length.
28	FTP uses the service of _____	IP	TCP	SMTP	IGMP			TCP
29	In FTP Port 21 is used for _____	Control connection	Data connection	Transformation connection	Transfer connection			Control connection
30	FTP uses_____ character set	NVTA ASCII	VTM ASCII	Binary	Unary			VTM ASCII
31	_____ mode data is delivered from FTP to TCP as continuous stream of bytes.	Block mode	Compress mode	Stream mode	Composite mode			Compress mode
32	_____ Command terminates the message in SNMP	END	QUIT	LOOP	EXIT			QUIT
33	_____ is a permanent negative completion reply	4YZ	5YZ	3YZ	YZ			3YZ
34	ATM has _____ formats for header	2	1	4	3			1
35	The switches inside he ATM Network route he cells on the _____	VPI	VCI	ARP	RARP			VPI

36	OPER is a _____bit field	8	45	12	34			45
37	SPA stands for _____	Sender Protocol Address	Seder Protocol Access	Sender Private Application	Sender Public Address			Sender Protocol Address
38	ATM accepts _____ bytes and transfers it into _____ bytes.	40 to 50	30 to 50	48 t65	48 to 53.			48 t65
39	First Phase in the Mobile Communication is _____	Agent advertisement	Agent finding	Agent discovery	Agent Filling			Agent advertisement
40	_____ Network is designed to be used only inside an organization	Protected	Private	Sensitive	Non Sensitive			Sensitive
41	_____ Networks have its private network with global internet access.	Globalization	Interknitting	Hybrid	Inter			Interknitting
42	_____ is a simple and efficient adaptation layer.	AAL6	AAL6	AAL3	AAL4			AAL6
43	_____ Command in SMTP is used to ask Receipt to send information about the command as the argument.	TURN	HELP	SEEK	QUIT.			TURN
44	The first and second stage of Mail delivery use _____	FTP	SMTP	SNMP	DNS			FTP
45	POP stands for _____	POP office Protocol	Presentation of POP	POP Office Presentation	Presentation POP office			POP office Protocol
46	Account information is accessed by using _____ command in FTP	ABORT	ACOV	ACCT	ACOC			ACOV
47	the Compression method normally used in _____ encoding.	File Length	Byte Length	Run Length	Bits Length			Run Length
48	In Class A, First byte refers to _____	Netid	hosted	Mask	Subnet id			Netid

49	The Third level of hierarchy in the subnet is _____	Site id	Host id	Mask	Subnet id			Subnet id
50	_____ joins several Classes	Subnet	Supernet	Milnet	Arpranet			Subnet
51	The class of 208.34.55.12 is _____	Class A	Class B	Class C	Class D			Class C
52	The Net id of 114.34.2.8 IP Address is _____	114	114.34	114.34.2	14.43.28			114
53	The Value of Default Mask is _____	255.0.0.0	255.255.0.0	255.255.255.0	255.255.255.255			255.255.0.0
54	The Hardware deice that is used transfer data from one network to another is _____	Router	Bridge	Gateway	Repeater			Gateway
55	If the IP Address is 20.1.1.02 then the class of the address is _____	Class A	Class B	Class C	Class D			Class A
56	NTP refers to _____	Network Transfer Protocol	Network Traffic Protocol	Network Time Protocol	Network Transmission Protocol			Network Time Protocol
57	PV6 refers to _____	IP Version Number	ARP Version Number	RARP Version Number	FTP Version Number			IP Version Number
58	Two types of Subnetting are _____ and _____	Static, Variable	Dynamic, Static	Fixed, Static	Fixed,Dynamic			Static, Variable
59	If the IP address is 123.12.3.2, then Host id is _____	3.2	12.3.2	12.3	13.3.2			3.2
60	The Network Class of 187.12.12.6 is _____	Class A	Class B	Class C	Class D			Class B
61	All ARP request packets are transmitted with the _____	Ethernet broadcast address	Ethernet Unicast address	Ethernet Multicast address	Ethernet Address			Ethernet broadcast address

62	ARP reply packets is directed to the _____	Host	Router	Bridge	Hub			Host
63	The size of an ARP request or reply packet is _____	24 Bytes	12 Bytes	6 Bytes	28 Bytes.			28 Bytes.
64	IP belongs to the _____ in the OSI model.	Network Layer	Session Layer	Transport Layer	Presentation Layer			Network Layer
65	ICMP Refers to _____	Internet Control Management Protocol	Internet Control Message Protocol	Internet Control Middleware Protocol	Internet Control Machine Protocol			Internet Control Management Protocol
66	Private networks can provide ___for organizations	Efficiency	Privacy	Public	Protected			Privacy
67	Both private and hybrid networks have a major drawback: _____	Lack of privacy	Cost	Lack of security	Time consuming			Cost
68	VPN is a network that is _____ but _____.	private; public	private; virtual	Public ; virtual	private;public			private; virtual
69	VPN is physically _____ but virtually _____.	public; private	private; virtual	Public ; virtual	private;public			public; private
70	An _____ is a private access that uses the TCP/IP protocol suite	Internet	Extranet	Intranet	Ethernet			Intranet
71	A _____ network is totally isolated from the global Internet	Private	Hybrid	Virtual	Public			Private
72	A VPN can use _____ to guarantee privacy	IP Sec	Tunneling	IP	Tunnelling			Tunneling
73	On a network that uses NAT, the _____ has a translation table.	Bridge	Router	Server	Switch			Router
74	On a network that uses NAT, _____ initiates the communication	An internal host	An External host	Router	Hub			An internal host

75	On a network that uses NAT, the router can use _____ global address	1	2	3	4			1
76	An IPv6 address is _____ bits long.	32	64	128	82			128
77	IPv6 allows _____ security provisions than IPv4.	More	Less	Same Level	Medium			More
78	An IPv6 address consists of _____ bytes	4	8	16	32			4
79	An IPv6 address can have up to _____ colons	8	7	4	9			8
80	_____ Address defines a group of computers.	Unicast	Multicast	Broadcast	Supercast			Multicast