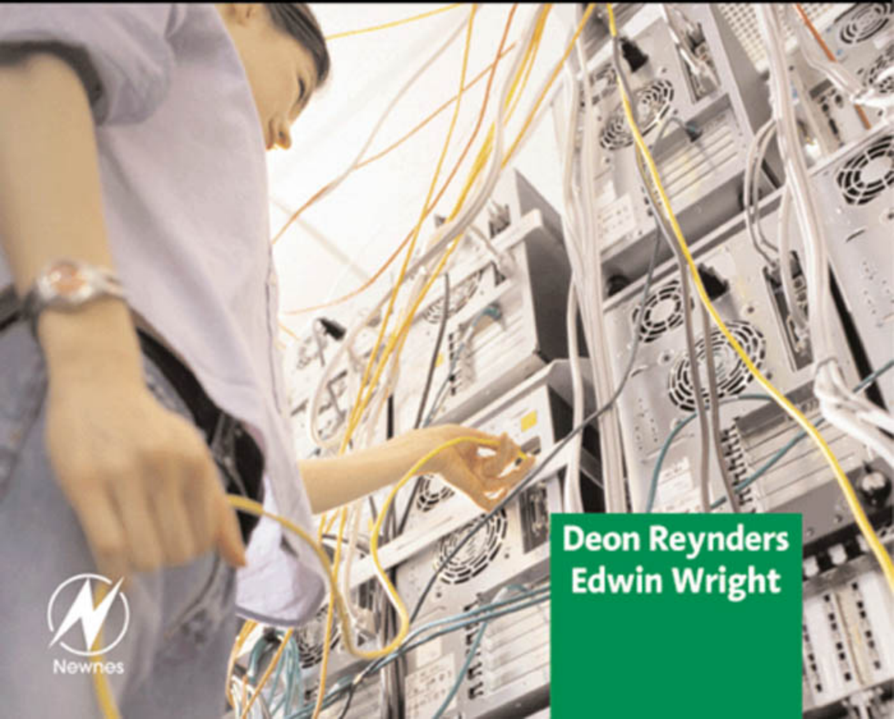
ELSEVIER

Practical TCP/IP and Ethernet Networking



Deon Reynders
Edwin Wright

Newnes

Practical TCP/IP and Ethernet Networking

Titles in the series

Practical Cleanrooms: Technologies and Facilities (David Conway)

Practical Data Acquisition for Instrumentation and Control Systems (John Park, Steve Mackay)

Practical Data Communications for Instrumentation and Control (Steve Mackay, Edwin Wright, John Park)

Practical Digital Signal Processing for Engineers and Technicians (Edmund Lai)

Practical Electrical Network Automation and Communication Systems (Cobus Strauss)

Practical Embedded Controllers (John Park)

Practical Fiber Optics (David Bailey, Edwin Wright)

Practical Industrial Data Networks: Design, Installation and Troubleshooting (Steve Mackay, Edwin Wright, John Park, Deon Reynders)

Practical Industrial Safety, Risk Assessment and Shutdown Systems for Instrumentation and Control (Dave Macdonald)

Practical Modern SCADA Protocols: DNP3, 60870.5 and Related Systems (Gordon Clarke, Deon Reynders)

Practical Radio Engineering and Telemetry for Industry (David Bailey)

Practical SCADA for Industry (David Bailey, Edwin Wright)

Practical TCP/IP and Ethernet Networking (Deon Reynders, Edwin Wright)

Practical Variable Speed Drives and Power Electronics (Malcolm Barnes)

Practical TCP/IP and Ethernet Networking

Deon Reynders Pr Eng, BSc BEng, BSc Eng (Elec)(Hons), MBA

Edwin Wright MIPENZ, BSc(Hons), BSc(Elec Eng), IDC Technologies, Perth, Australia
.

ELSEVIER

AMSTERDAM • BOSTON • HEIDELBERG • LONDON • NEW YORK • OXFORD
PARIS • SAN DIEGO • SAN FRANCISCO • SINGAPORE • SYDNEY • TOKYO

Newnes is an imprint of Elsevier

Newnes

Newnes
An imprint of Elsevier
Linacre House, Jordan Hill, Oxford OX2 8DP
200 Wheeler Road, Burlington, MA 01803

First published 2003

British Library Cataloguing in Publication Data
A catalogue record for this book is available from the British Library

ISBN 07506 58061

Typeset and Edited by Vivek Mehra, Mumbai, India
(vivekmehra@tatanova.com)

Printed and bound in Great Britain

Contents

Preface

One of the great protocols that has been inherited from the Internet is TCP/IP and this is being used as the open standard today for all network and communications systems. The reasons for this popularity are not hard to find. TCP/IP and Ethernet are truly open standards available to competing manufacturers and providing the user with a common standard for a variety of products from different vendors. In addition, the cost of TCP/IP and Ethernet is low. Initially TCP/IP was used extensively in military applications and the purely commercial world such as banking, finance, and general business. But of great interest has been the strong movement to universal usage by the hitherto disinterested industrial and manufacturing spheres of activity who have traditionally used their own proprietary protocols and standards. These proprietary standards have been almost entirely replaced by the usage of the TCP/IP suite of protocols.

This is a hands-on book that has been structured to cover the main areas of TCP/IP and Ethernet in detail, while going through the practical implementation of TCP/IP in computer and industrial applications. Troubleshooting and maintenance of TCP/IP networks and communications systems in an office and industrial environment will also be covered.

After reading this book we would hope you would be able to:

- Understand the fundamentals of the TCP/IP suite of protocols
- Gain a practical understanding of the application of TCP/IP
- Learn how to construct a robust local area network (LAN)
- Learn the basic skills in troubleshooting TCP/IP and LANs
- Apply the TCP/IP suite of protocols to both an office and industrial environment

Typical people who will find this book useful include:

- Network technicians
- Data communications managers
- Communication specialists
- IT support managers and personnel
- Network planners
- Programmers
- Design engineers
- Electrical engineers
- Instrumentation and control engineers
- System integrators
- System analysts
- Designers
- IT and MIS managers
- Network support staff
- Systems engineers

You should have a modicum of computer knowledge and know how to use the Microsoft Windows operating system in order to derive maximum benefit from this book.

The structure of the book is as follows.

Chapter 1: Overview. This chapter gives a brief overview of what is covered in the book with an outline of the essentials of communications systems.

Chapter 2: Networking fundamentals. An overview of network communication, types of networks, the OSI model, network topologies and media access methods.

Chapter 3: Ethernet networks. A description of the operation and performance of Ethernet networks commencing with the basic principles.

Chapter 4: Fast and gigabit Ethernet Systems. A minimum speed of 100 Mbps is becoming *de rigeur* on most Ethernet networks and this chapter examines the design and installation issues for fast Ethernet and gigabit Ethernet systems, which go well beyond the traditional 10 Mbps speed of operation.

Chapter 5: Introduction to TCP/IP. A brief review of the origins of TCP/IP to lay the foundation for the following chapters.

Chapter 6: Internet layer protocols. Perhaps the workhorse of the TCP/IP suite of protocols this chapter fleshes out the Internet protocol (both Ipv4 and Ipv6) and also examines the operation of ARP, RARP and ICMP.

Chapter 7: Host-to-host (transport) layer protocols. The TCP (transmission control protocol) and UDP (user datagram protocol) are both covered in this chapter.

Chapter 8: Application layer protocols. A thorough coverage of the most important application layer protocols such as FTP (file transfer protocol), TFTP (trivial file transfer protocol), TELNET, Rlogin, network file system, domain name system, WINS, simple network management protocol (SNMP), SMTP, POP, BOOTP and DHCP.

Chapter 9: TCP/IP utilities. A coverage focussing on the practical application of the main utilities such as Ping, ARP, NETSTAT NBTSTAT, IPCONFIG, WINIPCFG, tracert, ROUTE and the hosts file.

Chapter 10: LAN system components. A discussion on the key components in connecting networks together such as repeaters, bridges, switches and routers.

Chapter 11: The Internet. A brief discussion on the origins of the Internet and the various associated standards organizations.

Chapter 12: Internet access. The typical methods of connecting to the Internet are outlined here with a discussion on connecting a single host to the Internet, connecting multiple remote hosts to a corporate LAN, and in connecting multiple hosts to the Internet.

Chapter 13: The Internet for communications. Speed/bandwidth issues, the different options for E-mail, voice over IP and voice mail will be described in this chapter.

Chapter 14: Security considerations. The security problem and methods of controlling access to a network will be examined in this chapter. This is a growing area of importance due to the proliferation attacks on computer networks by external parties.

Chapter 15: Process automation. The legacy architectures and the factory of the future will be examined here together with an outline of the key elements of the modern Ethernet and TCP/IP architecture.

Chapter 16: Installing and troubleshooting Ethernet systems. The functions of the various types of network driver software together with the parameters to set the network card to match up to the software for correct operation will be described here.

Chapter 17: Troubleshooting TCP/IP. Maintenance of a TCP/IP network together with three typical methods requiring troubleshooting and the use of the utilities such as NETSTAT, Ping, tracert, and ripquery.

Chapter 18: Satellites and TCP/IP. An overview of satellites and TCP/IP with satellites.

1

Introduction to communications

Objectives

When you have completed study of this chapter you should be able to:

- Understand the main elements of the data communication process
- Understand the difference between analog and digital transmission
- Explain how data transfer is affected by attenuation, bandwidth and noise in the channel
- Know the importance of synchronization of digital data systems
- Describe the basic synchronization concepts used with asynchronous and synchronous systems
- Explain the following types of encoding:
  - Manchester
  - RZ
  - NRZ
  - MLT-3
  - 4B/5B
  - Describe the basic error detection principles.

1.1 Data communications

Communications systems exist to transfer information from one location to another. The components of the information or message are usually known as data (derived from the Latin word for items of information). All data are made up of unique code symbols or other entities on which the sender and receiver of the messages have agreed. For example binary data is represented by two states '0' and '1'. These are referred to as Binary digiTS or 'bits'. These bits are represented inside our computers by the level of the electrical signals within storage elements; a high level could represent a ‘1’, and a low-level represent a ‘0’. Alternatively, the data may be represented by the presence or absence of light in an optical fiber cable.

1.2 Transmitters, receivers and communication channels

A communications process requires the following components:

- A source of the information
- A transmitter to convert the information into data signals compatible with the communications channel
- A communications channel
- A receiver to convert the data signals back into a form the destination can understand
- The destination of the information

This process is shown in Figure 1.1.

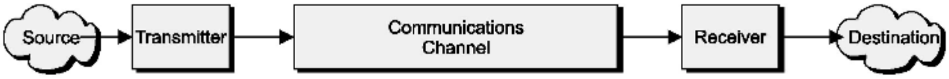


Figure 1.1
Communications process

The transmitter encodes the information into a suitable form to be transmitted over the communications channel. The communications channel moves this signal as electromagnetic energy from the source to one or more destination receivers. The channel may convert this energy from one form to another, such as electrical to optical signals, whilst maintaining the integrity of the information so the recipient can understand the message sent by the transmitter.

For the communications to be successful the source and destination must use a mutually agreed method of conveying the data.

The main factors to be considered are:

- The form of signaling and the magnitude(s) of the signals to be used
- The type of communications link (twisted pair, coaxial, optic fiber, radio etc)
- The arrangement of signals to form character codes from which the message can be constructed
- The methods of controlling the flow of data
- The procedures for detecting and correcting errors in the transmission

The form of the physical connections is defined by interface standards, some agreed coding is applied to the message and the rules controlling the data flow and detection and correction of errors are known as the **protocol**.

1.2.1 Interface standards

An interface standard defines the electrical and mechanical aspects of the interface to allow the communications equipment from different manufacturers to operate together.

A typical example is the **EIA/TIA-232-E** interface standard. This specifies the following three components:

- **Electrical signal characteristics** – defining the allowable voltage levels, grounding characteristics etc
- **Mechanical characteristics** – defining the connector arrangements and pin assignments
- **Functional description of the interchange circuits** – defining the function of the various data, timing and control signals used at the interface

It should be emphasized that the interface standard only defines the electrical and mechanical aspects of the interface between devices and does not cover how data is transferred between them.

1.2.2 Coding

A wide variety of codes have been used for communications purposes. Early telegraph communications used Morse code with human operators as transmitter and receiver. The Baudot code introduced a constant 5-bit code length for use with mechanical telegraph transmitters and receivers. The commonly used codes for data communications today are the **Extended Binary Coded Decimal Interchange Code** (EBCIDIC) and the **American Standards Committee for Information Interchange** (ASCII).

1.2.3 Protocols

A protocol is essential for defining the common message format and procedures for transferring data between all devices on the network. It includes the following important features:

- **Initialization:** Initializes the protocol parameters and commences the data transmission
- **Framing and synchronization:** Defines the start and end of the frame and how the receiver can synchronize to the data stream
- **Flow control:** Ensures that the receiver is able to advise the transmitter to regulate the data flow and ensure no data is lost.
- **Line control:** Used with half-duplex links to reverse the roles of transmitter and receiver and begin transmission in the other direction.
- **Error control:** Provides techniques to check the accuracy of the received data to identify transmission errors. These include Block Redundancy checks and cyclic redundancy checks
- **Time out control:** Procedures for transmitters to retry or abort transmission when acknowledgments are not received within agreed time limits

1.2.4 Some commonly used communications protocols

- Xmodem or Kermit for asynchronous file transmission
- Binary synchronous protocol (BSC), synchronous data link control (SDLC) or high level data link control (HDLC) for synchronous transmissions
- Industrial protocols such as manufacturing automation protocol (MAP), technical office protocol (TOP), Modbus, Data Highway Plus, HART, Profibus, Foundation Fieldbus, etc

1.3 Types of communication channels

1.3.1 Analog communications channels

An analog communications channel conveys analog signals that are changing continuously in both frequency and amplitude. These signals are commonly used for audio and video communication as illustrated in Figure 1.2 and Figure 1.3.

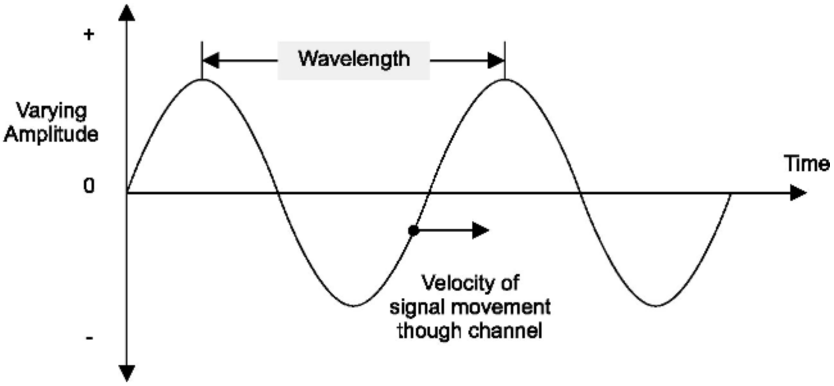


Figure 1.2
Analog signal

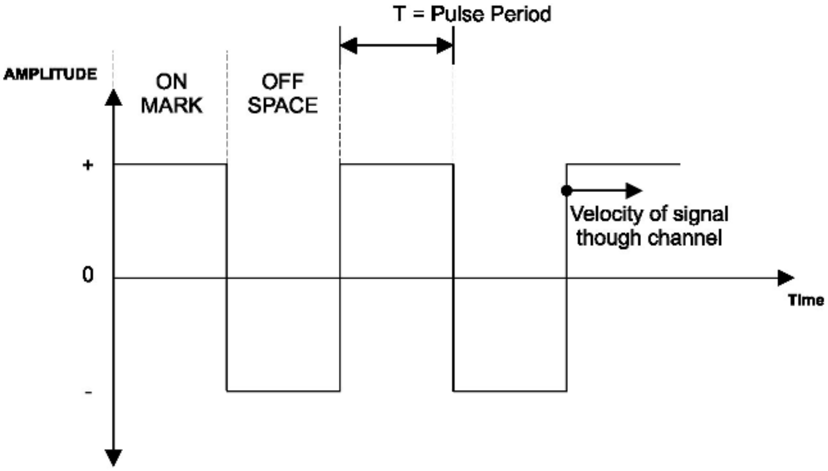


Figure 1.3
Digital signal

1.4 Communication channel properties

1.4.1 Signal attenuation

As the signal travels along a communications channel its amplitude decreases as the physical medium resists the flow of the electromagnetic energy. This effect is known as signal attenuation. With electrical signaling some materials such as copper are very efficient conductors of electrical energy. However, all conductors contain impurities that resist the movement of the electrons that constitute the electric current. The resistance of the conductors causes some of the electrical energy of the signal to be converted to heat energy as the signal progresses along the cable resulting in a continuous decrease in the electrical signal. The signal attenuation is measured in terms of signal loss per unit length of the cable, typically dB/km.

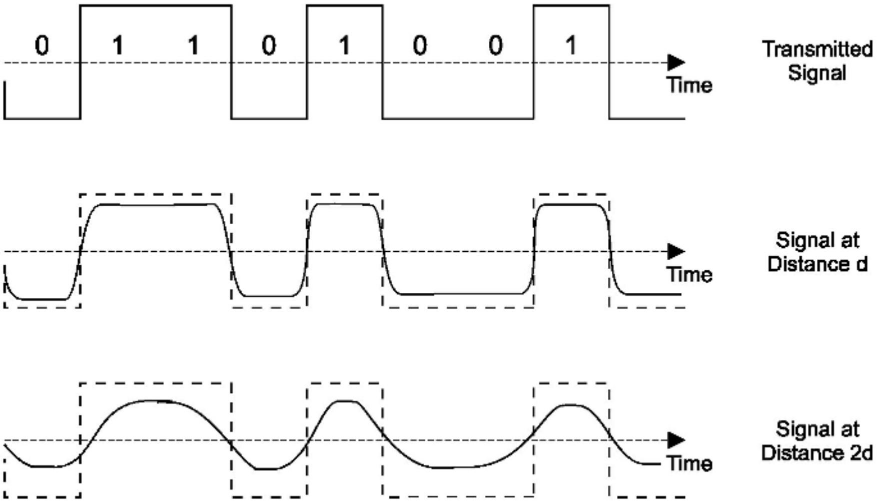


Figure 1.4
Signal attenuation

To allow for attenuation, a limit is set for the maximum length of the communications channel. This is to ensure that the attenuated signal arriving at the receiver is of sufficient amplitude to be reliably detected and correctly interpreted. If the channel is longer than this maximum length, amplifiers or repeaters must be used at intervals along the channel to restore the signal to acceptable levels.

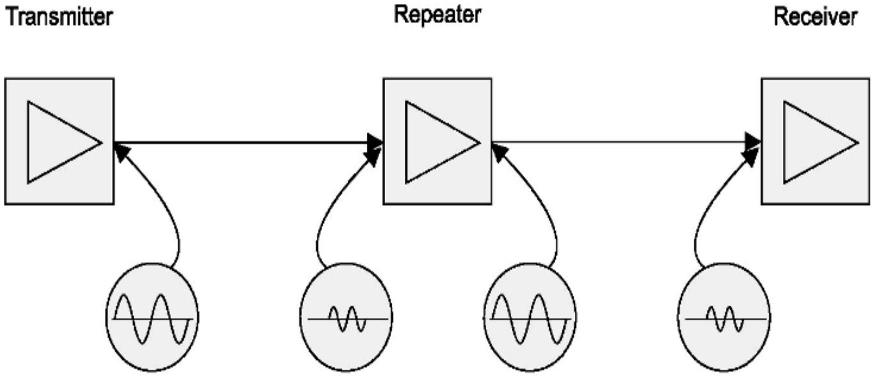


Figure 1.5
Signal repeaters

Signal attenuation increases as the frequency increases. This causes distortion to practical signals containing a range of frequencies. This is illustrated in Figure 1.4 where the rise-times of the attenuated signals progressively decrease as the signal travels through the channel, caused by the greater attenuation of the high frequency components. This problem can be overcome by the use of amplifiers that amplify the higher frequencies by greater amounts.

1.4.2 Channel bandwidth

The quantity of information a channel can convey over a given period is determined by its ability to handle the rate of change of the signal, that is its frequency. An analog signal varies between a minimum and maximum frequency and the difference between those frequencies is the bandwidth of that signal. The bandwidth of an analog channel is the difference between the highest and lowest frequencies that can be reliably received over the channel. These frequencies are often those at which the signal has fallen to half the power relative to the mid-band frequencies, referred to as 3 dB points. In which case the bandwidth is known as the 3 dB bandwidth.

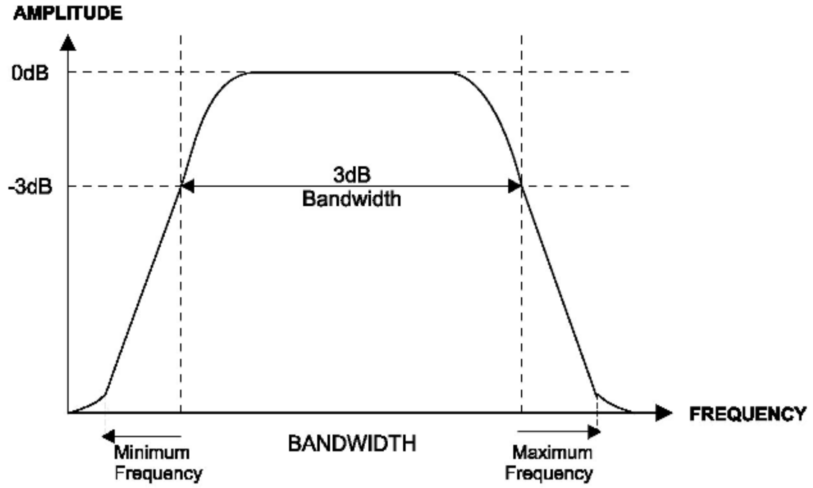


Figure 1.6
Channel bandwidth

Digital signals are made up of a large number of frequency components, but only those within the bandwidth of the channel will be able to be received. It follows that the larger the bandwidth of the channel, the higher the data transfer rate can be and more high frequency components of the digital signal can be transported, and so a more accurate reproduction of the transmitted signal can be received.

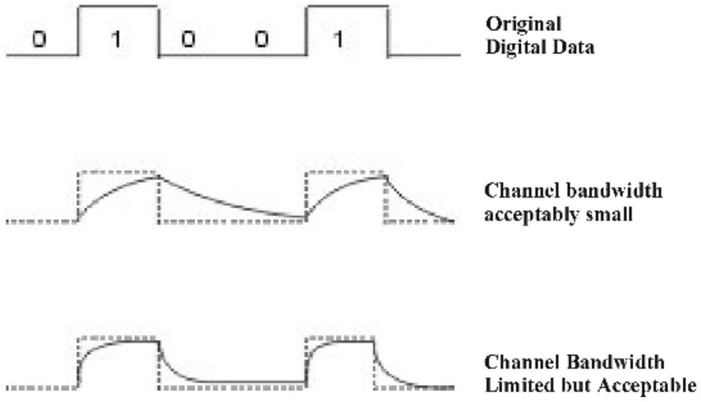


Figure 1.7
Effect of channel bandwidth on digital signal

The maximum data transfer rate (C) of the transmission channel can be determined from its bandwidth, by use of the following formula derived by Shannon.

$\boldsymbol{C = 2B\log_2 M}$ **bps**

Where

B = bandwidth in hertz and M levels are used for each signaling element.

In the special case where only two levels, 'ON' and 'OFF' are used (binary), $M = 2$ and $C = 2\ B$. As an example, the maximum data transfer rate for a PSTN channel of bandwidth 3200 hertz carrying a binary signal would be 2 × 3200 = 6400 bps. The achievable data transfer rate is reduced to ½ of 6400 because of the Nyquist rate. It is further reduced in practical situations because of the presence of noise on the channel to approximately 2400 bps unless some modulation system is used.

1.4.3 Noise

As the signals pass through a communications channel the atomic particles and molecules in the transmission medium vibrate and emit random electromagnetic signals as noise. The strength of the transmitted signal is normally large relative to the noise signal. However, as the signal travels through the channel and is attenuated, its level can approach that of the noise. When the wanted signal is not significantly higher than the background noise, the receiver cannot separate the data from the noise and communication errors occur.

An important parameter of the channel is the ratio of the power of the received signal (S) to the power of the noise signal (N). The ratio S/N is called the signal to noise ratio, which is normally expressed in decibels, abbreviated to dB.

$\boldsymbol{S/N = 10 \log_{10} (S/N)}$ **dB**

A high signal to noise ratio means that the wanted signal power is high compared to the noise level, resulting in good quality signal reception. The theoretical maximum data transfer rate for a practical channel can be calculated using the Shannon-Hartley law, which states:

$\boldsymbol{C = B \log_2 (1+S/N)}$ **bps**

Where

C = data rate in bps

B = bandwidth of the channel in hertz

S = signal power in watts and N is the noise power in watts

It can be seen from this formula that increasing the bandwidth or increasing the signal to noise ratio will allow increases to the data rate, and that a relatively small increase in bandwidth is equivalent to a much greater increase in signal to noise ratio.

Digital transmission channels make use of higher bandwidths and digital repeaters or regenerators to regenerate the signals at regular intervals and maintain acceptable signal to noise ratios. The degraded signals received at the regenerator are detected, then re-timed and retransmitted as nearly perfect replicas of the original digital signals, as shown in Figure 1.8. Provided the signal to noise ratios are maintained in each link, there is no accumulated noise on the signal, even when transmitted thousands of kilometers.

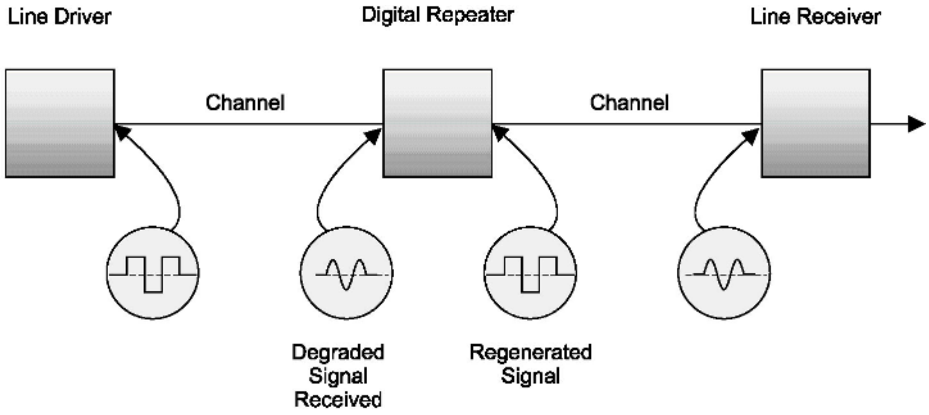


Figure 1.8
Digital link

1.5 Data transmission modes

1.5.1 Direction of signal flow

Simplex

A simplex channel is unidirectional and allows data to flow in one direction only, as shown in Figure 1.9. Public radio broadcasting is an example of a simplex transmission. The radio station transmits the broadcast program, but does not receive any signals back from your radio receiver.

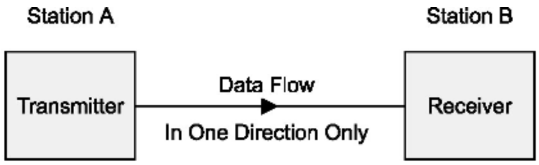


Figure 1.9
Simplex transmission

This has limited use for data transfer purposes, as we invariably require the flow of data in both directions to control the transfer process, acknowledge data etc.

Half-duplex

Half-duplex transmission allows us to provide simplex communication in both directions over a single channel, as shown in Figure 1.10. Here the transmitter at station 'A' sends data to a receiver at station 'B'. A line turnaround procedure takes place whenever transmission is required in the opposite direction. The station 'B' transmitter is then enabled and communicates with the receiver at station 'A'. The delay in the line turnaround procedures reduces the available data throughput of the communications channel.

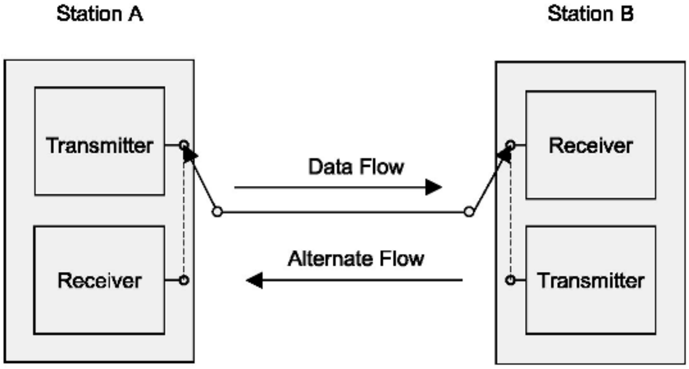


Figure 1.10
Half-duplex transmission

Full-duplex

A full-duplex channel gives simultaneous communications in both directions, as shown in Figure 1.11.

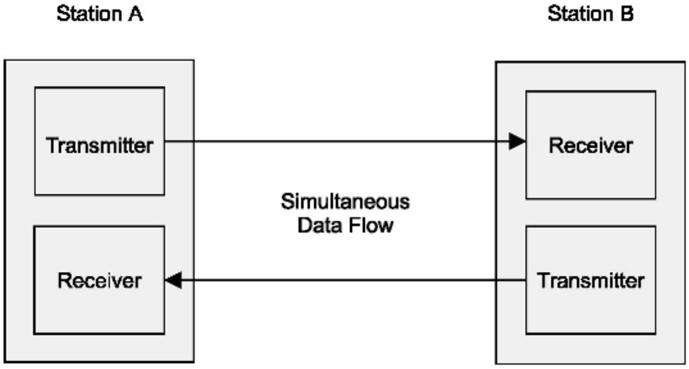


Figure 1.11
Full-duplex transmission

1.5.2 Synchronization of digital data signals

Data communications depends on the timing of the signal generation and reception being kept correct throughout the message transmission. The receiver needs to look at the incoming data at the correct instants before determining whether a ‘1’ or ‘0’ was transmitted. The process of selecting and maintaining these sampling times is called synchronization.

In order to synchronize their transmissions, the transmitting and receiving devices need to agree on the length of the code elements to be used, known as the bit time. The receiver needs to extract the transmitted clock signal encoded into the received data stream. By synchronizing the bit time of the receiver’s clock with that encoded by the sender, the receiver is able to determine the right times to detect the data transitions in the message and correctly receive the message. The devices at both ends of a digital channel can synchronize themselves using either asynchronous or synchronous transmission as outlined below.

1.5.3 Asynchronous transmission

Here the transmitter and receiver operate independently, and exchange a synchronizing pattern at the start of each message code element (frame). There is no fixed relationship between one message frame and the next, such as a computer keyboard input with potentially long random pauses between keystrokes.

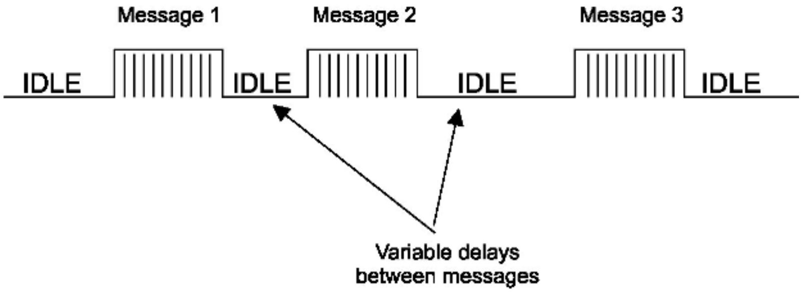


Figure 1.12
Asynchronous data transmission

At the receiver the channel is sampled at a high rate, typically in excess of 16 times the bit rate of the data channel, to accurately determine the centers of the synchronizing pattern (start bit) and its duration (bit time).

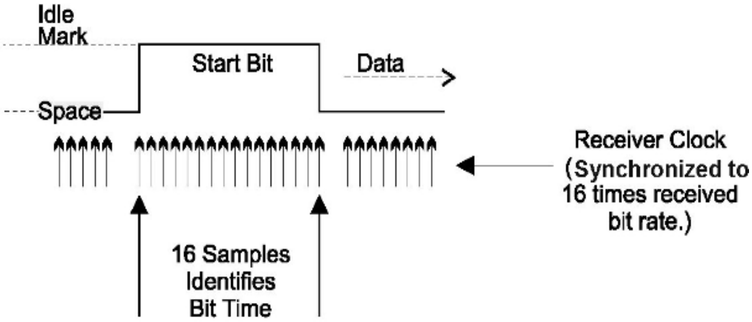
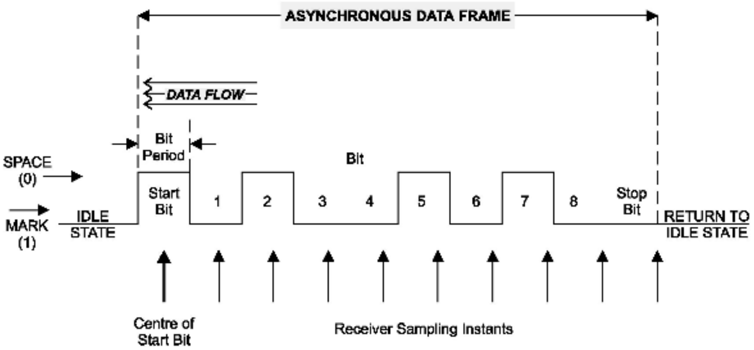

Idle
Mark
Start Bit
Data
Space
Receiver Clock (Synchronized to 16 times received bit rate.)
16 Samples Identifies Bit Time

Figure 1.13
Clock estimation at receiver

The data bits are then determined by the receiver sampling the channel at intervals corresponding to the centers of each transmitted bit. These are estimated by delaying multiples of the bit time from the centers of the start bit. For an eight-bit serial transmission, this sampling is repeated for each of the eight data bits then a final sample is made during the ninth time interval. This sample is to identify the stop bit and confirm that the synchronization has been maintained to the end of the message frame. Figure 1.14 illustrates the asynchronous data reception process.



Data Bit	1 2 3 4 5 6 7 8
Received Data	1 0 1 1 0 1 0 1

Figure 1.14
Asynchronous data reception

1.5.4 Synchronous transmission

The receiver here is initially synchronized to the transmitter then maintains this synchronization throughout the continuous transmission. This is achieved by special data coding schemes, such as Manchester encoding, which ensure that the transmitted clock is continuously encoded into the transmitted data stream. This enables the synchronization to be maintained at any receiver right to the last bit of the message, which could be as large as 4500 bytes (36 000 bits). This allows larger frames of data to be efficiently transferred at higher data rates. The synchronous system packs many characters together and sends them as a continuous stream, called a block. For each transmission block there is a preamble, containing the start delimiter for initial synchronization purposes and information about the block, and a post-amble, to give error checking, etc. An example of a synchronous transmission block is shown in Figure 1.15. Understandably all high-speed data transfer systems utilize synchronous transmission systems to achieve fast, accurate transfers of large blocks of data.

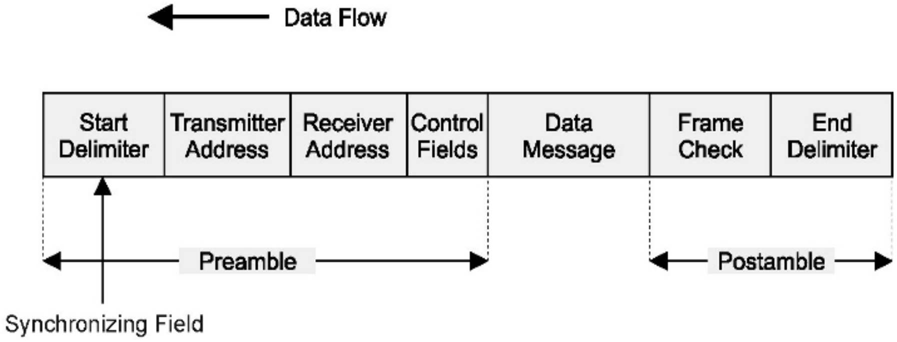


Figure 1.15
Synchronous transmission block

1.6 Encoding methods

1.6.1 Manchester

Manchester is a bi-phase signal-encoding scheme used in Ethernet LANs. The direction of the transition in mid-interval (negative to positive or positive to negative) indicates the value (1 or 0, respectively) and provides the clocking.

The Manchester codes have the advantage that they are self-clocking. Even a sequence of one thousand '0s' will have a transition in every bit; hence the receiver will not lose synchronization. The price paid for this is a bandwidth requirement double that which is required by the RZ-type methods.

The Manchester scheme follows these rules:

- +V and –V voltage levels are used
- There is a transition from one to the other voltage level halfway through each bit interval
- There may or may not be a transition at that start of each bit interval, depending on whether the bit value is a 0 or 1
- For a 1 bit, the transition is always from a –V to +V; for a 0 bit, the transition is always from a +V to a –V

In Manchester encoding, the beginning of a bit interval is used merely to set the stage. The activity in the middle of each bit interval determines the bit value: upward transition for a 1 bit, downward for a 0 bit.

1.6.2 Differential Manchester

Differential Manchester is a bi-phase signal-encoding scheme used in token ring LANs. The presence or absence of a transition at the beginning of a bit interval indicates the value; the transition in mid-interval just provides the clocking.

For electrical signals, bit values will generally be represented by one of three possible voltage levels: positive (+V), zero (0 V), or negative (–V). Any two of these levels are needed – for example, + V and –V.

There is a transition in the middle of each bit interval. This makes the encoding method self-clocking, and helps avoid signal distortion due to DC signal components.

For one of the possible bit values but not the other, there will be a transition at the start of any given bit interval. For example, in a particular implementation, there may be a signal transition for a 1 bit.

In differential Manchester encoding, the presence or absence of a transition at the beginning of the bit interval determines the bit value. In effect, 1 bit produces vertical signal patterns; 0 bits produce horizontal patterns. The transition in the middle of the interval is just for timing.

1.6.3 RZ (return to zero)

The RZ-type codes consume only half the bandwidth taken up by the Manchester codes. However, they are not self-clocking since a sequence of a thousand '0s' will result in no movement on the transmission medium at all.

RZ is a bipolar signal-encoding scheme that uses transition coding to return the signal to a zero voltage during part of each bit interval. It is self-clocking.

In the differential version, the defining voltage (the voltage associated with the first half of the bit interval) changes for each 1 bit, and remains unchanged for each 0 bit.

In the non-differential version, the defining voltage changes only when the bit value changes, so that the same defining voltages are always associated with 0 and 1. For example, +5 volts may define a 1, and –5 volts may define a 0.

1.6.4 NRZ (non-return to zero)

NRZ is a bipolar encoding scheme. In the non-differential version it associates, for example, +5 V with 1 and –5 V with 0.

In the differential version, it changes voltages between bit intervals for 1 values but not for 0 values. This means that the encoding changes during a transmission. For example, 0 may be a positive voltage during one part, and a negative voltage during another part, depending on the last occurrence of a 1. The presence or absence of a transition indicates a bit value, not the voltage level.

1.6.5 MLT-3

MLT-3 is a three-level encoding scheme that can also scramble data. This scheme is one proposed for use in FDDI networks. The MLT-3 signal-encoding scheme uses three voltage levels (including a zero level) and changes levels only when a 1 occurs.

It follows these rules:

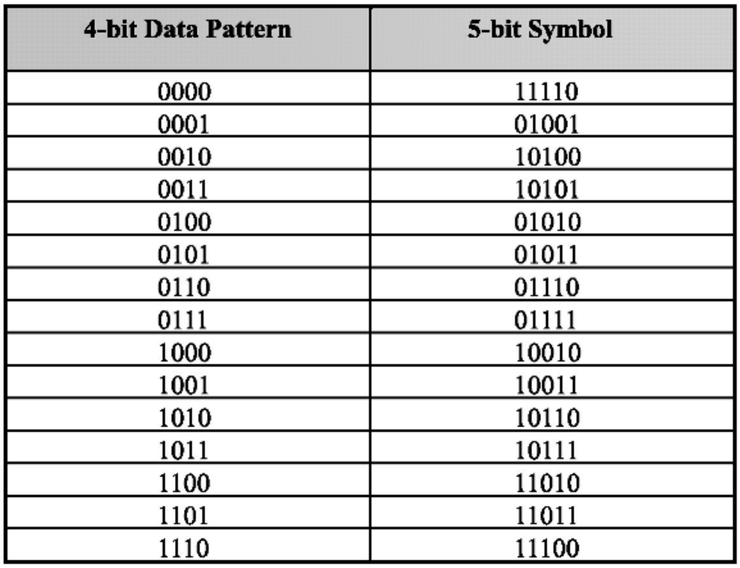
- +V, 0 V, and –V voltage levels are used
- The voltage remains the same during an entire bit interval; that is, there are no transitions in the middle of a bit interval
- The voltage level changes in succession; from +V to 0 V to –V to 0 V to +V, and so on
- The voltage level changes only for a 1 bit

MLT-3 is not self-clocking, so that a synchronization sequence is needed to make sure the sender and receiver are using the same timing.

1.6.6 4B/5B

The Manchester codes, as used for 10 Mbps Ethernet, are self-clocking but consume unnecessary bandwidth. For this reason, it is not possible to use it for 100 Mbps Ethernet over CAT5 cable. A solution to the problem is to revert back to one of the more bandwidth efficient methods such as NRZ or RZ. The problem with these, however, is that they are not self-clocking and hence the receiver loses synchronization if several zeros are transmitted sequentially. This problem, in turn, is overcome by using the 4B/5B technique.

The 4B/5B technique codes each group of four bits into a five-bit code. For example, the binary pattern 0110 is coded into the five-bit pattern 01110. This code table has been designed in such a way that no combination of data can ever be encoded with more than 3 zeros on a row. This allows the carriage of 100 Mbps data by transmitting at 125 MHz, as opposed to the 200 Mbps required by Manchester encoding.



4-bit Data Pattern	5-bit Symbol
0000	11110
0001	01001
0010	10100
0011	10101
0100	01010
0101	01011
0110	01110
0111	01111
1000	10010
1001	10011
1010	10110
1011	10111
1100	11010
1101	11011
1110	11100

Table 1.1
4B/5B data coding

1.7 Error detection

All practical data communications channels are subject to noise, particularly where equipment is situated in industrial environments with high electrical noise, such as electromagnetic radiation from adjacent equipment or electromagnetic induction from adjacent cables. As a consequence the received data may contain errors. To ensure reliable data communication we need to check the accuracy of each message.

Asynchronous systems often use a single bit checksum, the parity bit, for each message, calculated from the seven or eight data bits in the message. Longer messages require more complex checksum calculations to be effective. For example the **longitudinal redundancy check** (LRC) calculates an additional byte covering the content of the message (up to 15 bytes) while an arithmetic checksum (calculates two additional bytes) can be used for messages up to 50 bytes in length. Most high-speed local area networks uses a 32-bit **cyclic redundancy check** (CRC).

1.7.1 Cyclic redundancy check (CRC)

The cyclic redundancy check (CRC) enables detection of errors with very high accuracy in messages of any length. So, for example, we can detect the presence of a single bit in error in a synchronous data frame containing 36 000 bits. The CRC works by treating all the bits of the message block as one binary number that is then divided by a known polynomial. For a 32-bit CRC this is a specific 32-bit generator, specially chosen to detect very high percentages of errors, including all error sequences of less than 32 bits. The remainder found after this division process is the CRC. Calculation of the CRC is carried out by the hardware in the transmission interface of LAN adapter cards.

2

Networking fundamentals

Objectives

When you have completed study of this chapter you should be able to:

- Explain the difference between circuit switching and packet switching
- Explain the difference between connectionless and connection oriented communication
- Explain the difference between a datagram service and a virtual circuit
- List the differences between local area networks, metropolitan area networks, wide area networks and virtual private networks
- Describe the concept of layered communications model
- Describe the functions of each layer in the OSI reference model
- Indicate the structure and relevance of the IEEE 802 (ISO 8802) series of Standards and Working Groups
- Identify hub, ring and bus topologies – from a physical as well as from a logical point of view
- Describe the basic mechanisms involved in contention, token passing and polling media access control methods

2.1 Overview

Linking computers and other devices together to share information is nothing new. The technology for **local area networks** (LANs) was developed in the 1970s by minicomputer manufacturers to link widely separated user terminals to computers. This allowed the sharing of expensive peripheral equipment as well as data that may have previously existed in only one physical location.

A LAN is a communications path between one or more computers, file-servers, terminals, workstations and various other intelligent peripheral equipment, which are generally referred to as devices or hosts. A LAN allows access to devices to be shared by several users, with full connectivity between all stations on the network. It is usually owned and administered by a private owner and is located within a localized group of buildings.

The connection of a device into a LAN is made through a node. A node is any point where a device is connected and each node is allocated a unique address number. Every message sent on the LAN must be prefixed with the unique address of the destination. All devices connected to nodes also watch for any messages sent to their own addresses on the network. LANs operate at relatively high speeds (Mbps range and upwards) with a shared transmission medium over a fairly small geographical (i.e. local) area.

In a LAN, the software controlling the transfer of messages among the devices on the network must deal with the problems of sharing the common resources of the network without conflict or corruption of data. Since many users can access the network at the same time, some rules must be established on which devices can access the network, when and under what conditions. These rules are covered under the general subject of media access control.

When a node has access to the channel to transmit data, it sends the data within a packet (or frame), which includes, in its header, the addresses of both the source and the destination. This allows each node to either receive or ignore data on the network.

2.2 Network communication

There are two basic types of communications processes for transferring data across networks, viz. circuit switching and packet switching. These are illustrated in Figure 2.1

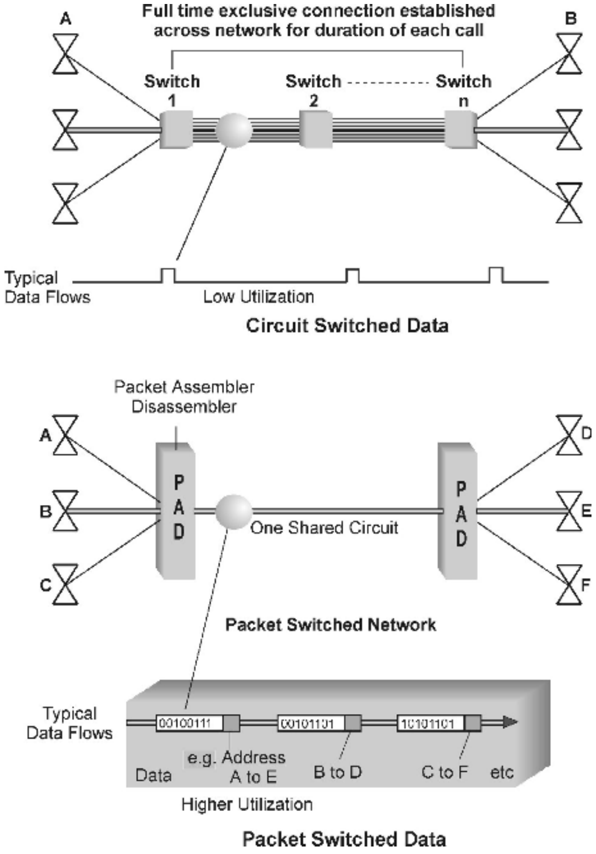


Figure 2.1
Circuit switched and packet switched data

2.2.1 Circuit switching

In a circuit switched process a continuous connection is made across the network between the two different points. This is a temporary connection, which remains in place as long as both parties wish to communicate, that is until the connection is terminated. All the network resources are available for the exclusive use of these two parties whether they are sending data or not. When the connection is terminated the network resources are released for other users. A telephone call is an example of a circuit switched connection.

The advantage of circuit switching is that the users have an exclusive channel available for the transfer of their data at any time while the connection is made. The obvious disadvantage is the cost of maintaining the connection when there is little or no data being transferred. Such connections can be very inefficient for the bursts of data that are typical of many computer applications.

2.2.2 Packet switching

Packet switching systems improve the efficiency of the transfer of bursts of data, by sharing the one communications channel with other similar users. This is analogous to the efficiencies of the mail system as discussed in the following paragraph.

When you send a letter by mail you post the stamped, addressed envelope containing the letter in your local mailbox. At regular intervals the mail company collects all the letters from your locality and takes them to a central sorting facility where the letters are sorted in accordance with the addresses of their destinations. All the letters for each destination are sent off in common mailbags to those locations, and are subsequently delivered in accordance with their addresses. Here we have economies of scale where many letters are carried at one time and are delivered by the one visit to your street/locality. Efficiency is more important than speed, and some delay is normal – within acceptable limits.

Packet switched messages are broken into a series of packets of certain maximum size, each containing the destination and source addresses and a packet sequence number. The packets are sent over a common communications channel, possibly interleaved with those of other users. All the receivers on the channel check the destination addresses of all packets and accept only those carrying their address. Messages sent in multiple packets are reassembled in the correct order by the destination node.

All packets do not necessarily follow the same path. As they travel through the network they may get separated and handled independently from each other, but eventually arrive at their correct destination. For this reason, packets often arrive at the destination node out of their transmitted sequence. Some packets may even be held up temporarily (stored) at a node, due to unavailable lines or technical problems that might arise on the network. When the time is right, the node then allows the packet to pass or be ‘forwarded’.

2.2.3 Datagrams and virtual circuits

Packet switched services generally support two types of service viz. datagrams and virtual circuits.

In a self contained local area network all packets will eventually reach their destination. However, if the packet is to be switched ACROSS networks i.e. on an internetwork – such as a wide area network – then a routing decision must be made.

There are two approaches that can be taken. The first is referred to as a **DATAGRAM** service. The approach is to allow each packet to be independently routed. The destination address incorporated in the data header will allow the routing to be performed. There is no guarantee when any packet will arrive at its destination, and they may well be out of sequence. The principle is similar to the mail service. You may send four postcards from your holiday in the South of France, but there is no guarantee that they will arrive in the same order that you posted them. If the recipient does not have a telephone, there is no easy method of determining that they have, in fact, been delivered. Such a service is called an UNRELIABLE service. This word is not used in its everyday context, but rather refers to the fact that there is no mechanism for informing the sender that the packet had not been delivered. The service is also called connectionless since a connection is not made for each packet.

The second approach is to setup a connection between transmitter and receiver, and to send all packets of data along this connection or **VIRTUAL CIRCUIT**. Whilst this might seem to be in conflict with the earlier statements on circuit switching, it should be quite clear that this does NOT imply a permanent circuit being dedicated to the one packet stream of data. Rather, the circuit shares its capacity with other traffic. The important point to note is that the route for the data packets to follow is taken up-front when all the routing decisions are taken. The data packets just follow that pre-established route. This service is known as RELIABLE and is also referred to as a connection oriented service or COS.

2.3 Types of networks

2.3.1 Local area networks (LANs)

LANs are characterized by high-speed transmission over a restricted geographical area. Thick Ethernet (10Base5), for example, operates at 10 Mb/s over a maximum distance of 500 m before the signals need to be boosted. This is illustrated in Figure 2.2.

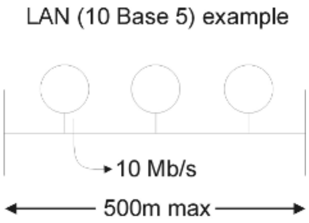


Figure 2.2
Example of LAN

2.3.2 Wide area networks (WANs)

While LANs operate where distances are relatively small, wide area networks (WANs) are used to link LANs that are separated by large distances that range from a few tens of meters to thousands of kilometers. WANs normally use the public telecommunication system to provide cost-effective connection between LANs. Since these links are supplied by independent telecommunications utilities, they are commonly referred to (and illustrated as) a ‘communications cloud’. Special equipment called gateways have been developed for this type of activity, which store the message at LAN speed and transmit it across the ‘communications cloud’ at a lower speed. When the entire message has been received at the remote LAN, the message is reinserted at LAN speed. A typical speed at which a WAN interconnects is 9600 bps to 45 Mbps. This is shown in Figure 2.3.

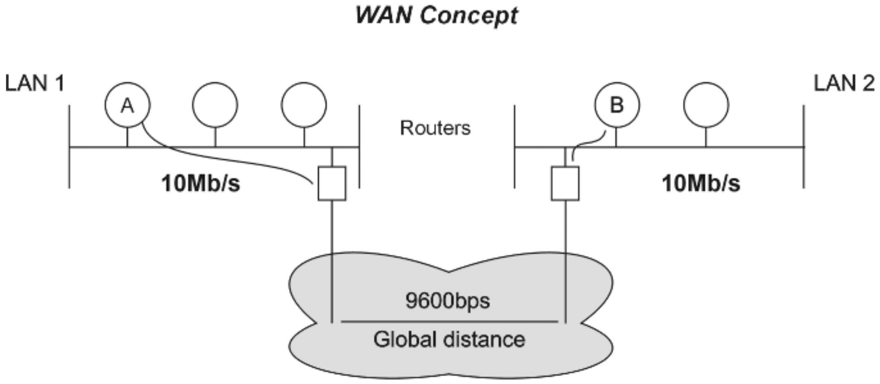


Figure 2.3
WAN concept

If reliability is needed for a time critical application, WANs can be considered quite unreliable, as delay in the information transmission is varied and wide. For this reason, WANs can only be used if the necessary error detection/ correction software is in place, and if propagation delays can be tolerated within certain limits.

2.3.3 Metropolitan area networks (MANs)

An intermediate type of a network – MANs – operate at speeds ranging from 56 kbps to 100Mbps – typically a higher speed than WANs but slower than LANs. MANs use fiber optic technology to communicate over distances of up to several hundred kilometers. They are normally used by telecommunication service providers within cities.

2.3.4 Coupling ratio

The coupling ratio provides an academic yardstick for comparing the performance of these different kinds of networks. It is useful to give us an insight as to the way that each network needs to operate.

Coupling ratio $\alpha = \tau / T$

Where

τ	Propagation delay for packet
T	Average packet transmission time
α =<<1	indicates a LAN
α = 1	indicates a MAN
α =>>1	indicates a WAN

This is illustrated in the following examples and Figure 2.4.

200 m LAN: With a propagation delay of about 1 mS, a 1000 byte packet takes about 0.8 ms to transmit at 10 Mbps.

Therefore α is about 1 mS/0.8 ms or 1/800 which is very much less than 1. This means that for a LAN the packet quickly reaches the destination and the transmission of the packet then takes say hundreds of times longer to complete.

200 km MAN: With a propagation delay of about 1 mS, a 4000 byte packet takes about 0.4 ms to transmit at 100 Mbps.

Therefore α is about 1 mS/0.4 ms or 2.4 which is about the order of 1. This means that for a MAN the packet reaches the destination then may only take about the same time again to complete the transmission.

100 000 km WAN: Propagation delay about 0.5–2 seconds, a packet of 128 bytes takes about 10 ms to transmit at 1 Mbps.

Therefore α is about 1 S/10 ms or 100. This means that for a WAN the packet reaches the destination after a delay of 100 times the packet length.

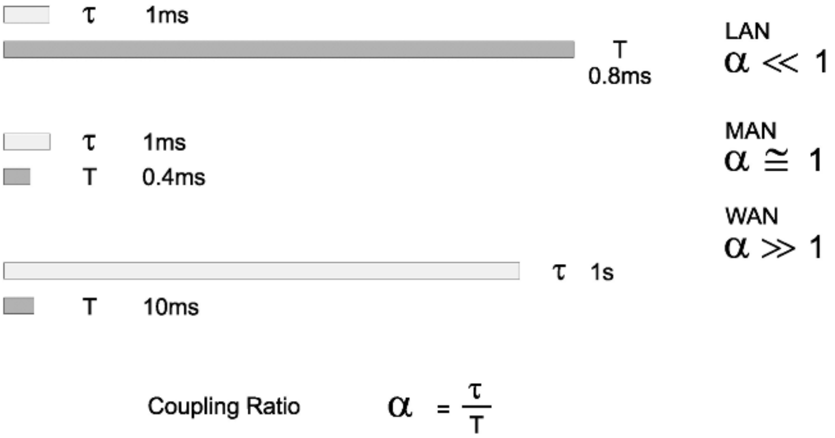


Figure 2.4
Coupling ratios

2.3.5 Virtual private networks (VPNs)

A cheaper alternative to a WAN, which uses dedicated packet switched links (such as X.25) to interconnect two or more LANs, is the virtual private network, which interconnects several LANs by utilizing the existing Internet infrastructure.

A potential problem is the fact that the traffic between the networks shares all the other Internet traffic and hence all communications between the LANs are visible to the outside world. This problem is solved by utilizing encryption techniques to make all communications between the LANs transparent (i.e. illegible) to other Internet users.

2.4 The open systems interconnection model

A communication framework that has had a tremendous impact on the design of LANs is the **open systems interconnection** (or OSI) model. The objective of this model is to provide a framework for the coordination of standards development and allows both existing and evolving standards activities to be set within that common framework.

2.4.1 Open and closed systems

The wiring together of two or more devices with digital communication is the first step towards establishing a network. In addition to the hardware requirements, which have been discussed above, the software problems of communication must also be overcome. Where all the devices on a network are from the same manufacturer, the hardware and software problems are usually easily overcome because the system is usually designed within the same guidelines and specifications.

When devices from several manufacturers are used on the same application, the problems seem to multiply. Networks that are specific to one manufacturer and which work with specific hardware connections and protocols are called closed systems. Usually, these systems were developed at a time before standardization or when it was considered unlikely that equipment from other manufacturers would be included in the network.

In contrast, open systems are those, which conform to specifications and guidelines, which are ‘open’ to all. This allows equipment from any manufacturer, who claims to comply with that standard, to be used interchangeably on the standard network. The benefits of open systems include wider availability of equipment, lower prices and easier integration with other components.

2.4.2 The open systems interconnection reference model (OSI model)

Faced with the proliferation of closed network systems, in 1978 the International Standards Organization (ISO) defined a ‘Reference Model for Communication between Open Systems’, which has become known as the open systems interconnection (OSI) model, or simply as the ISO/OSI model (ISO 7498). OSI is essentially a data communications management structure, which breaks data communications down into a manageable hierarchy of seven layers. Each layer has a defined purpose and interfaces with the layers above it and below it. By laying down standards for each layer, some flexibility is allowed so that the system designers can develop protocols for each layer independent of each other. By conforming to the OSI standards, a system is able to communicate with any other compliant system, anywhere in the world.

It should be realized at the outset that the OSI reference model is not a protocol or set of rules for how a protocol should be written but rather an overall framework in which to define protocols. The OSI model framework specifically and clearly defines the functions or services that have to be provided at each of the seven layers (or levels).

Since there must be at least two sites to communicate, each layer also appears to converse with its peer layer at the other end of the communication channel in a virtual (‘logical’) communication. These concepts of isolation of the process of each layer, together with standardized interfaces and peer-to-peer virtual communication, are fundamental to the concepts developed in a layered model such as the OSI model. The OSI layering concept is shown in Figure 2.5.

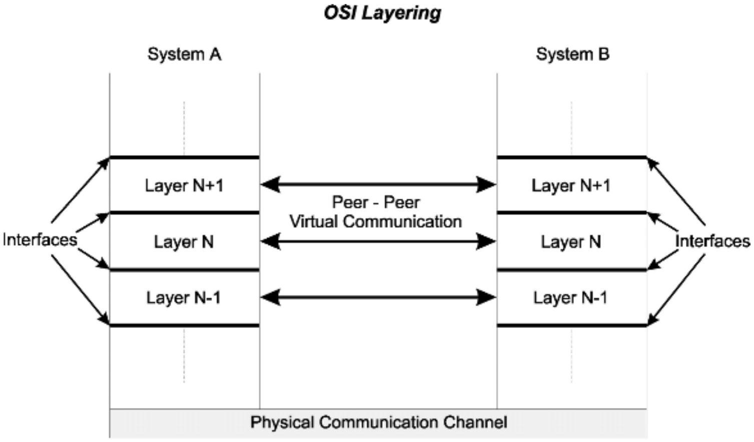


Figure 2. 5
OSI layering concept

The actual functions within each layer are provided by entities which are abstract devices, such as programs, functions, or protocols, that implement the services for a particular layer on a single machine. A layer may have more than one entity – for example a protocol entity and a management entity. Entities in adjacent layers interact through the common upper and lower boundaries by passing physical information through service access points (SAPs). A SAP could be compared to a predefined 'postbox' where one layer would collect data from the previous layer. The relationship between layers, entities, functions and SAPs is shown in Figure 2.6.

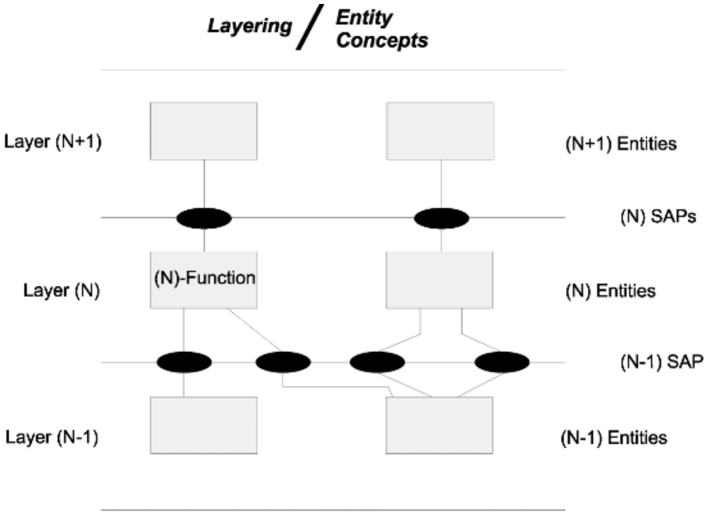


Figure 2.6
Relationship between layers, entities, functions and SAPs

In the OSI model, the entity in the next higher layer is referred to as the N+1 entity and the entity in the next lower layer as N–1. The services available to the higher layers are the result of the services provided by all the lower layers.

The functions and capabilities expected at each layer are specified in the model. However, the model does not prescribe how this functionality should be implemented. The focus in the model is on the 'interconnection' and on the information that can be passed over this connection. The OSI model does not concern itself with the internal operations of the systems involved.

When the OSI model was being developed, a number of principles were used to determine exactly how many layers this communication model should encompass. These principles are:

- A layer should be created where a different level of abstraction is required
- Each layer should perform a well-defined function
- The function of each layer should be chosen with thought given to defining internationally standardized protocols
- The layer boundaries should be chosen to minimize the information flow across the boundaries
- The number of layers should be large enough that distinct functions need not be thrown together in the same layer out of necessity and small enough that the architecture does not become unwieldy

The use of these principles led to seven layers being defined, each of which has been given a name in accordance with its process purpose. The diagram below shows the seven layers of the OSI model.

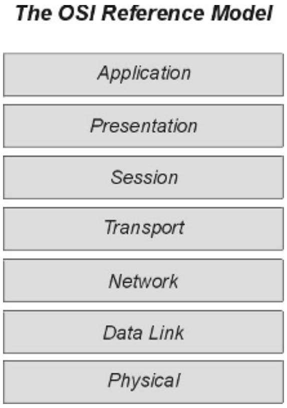


Figure 2.7
The OSI reference model

The service provided by any layer is expressed in the form of a service primitive with the data to be transferred as a parameter. (A service primitive is a fundamental service request made between protocols. For example, layer W may sit on top of layer X. If W wishes to invoke a service from X, it may issue a service primitive in the form of X.Connect.request to X. An example of a service primitive is shown in Figure 2.8. Service primitives are normally used to transfer data between processes within a node.

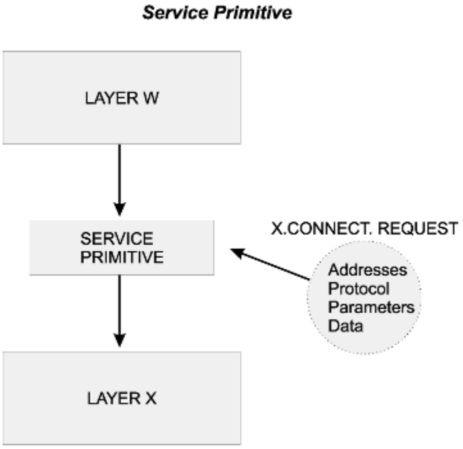


Figure 2.8
Service primitive

Typically, each layer in the transmitting site, with the exception of the lowest, adds header information, or protocol control information (PCI), to the data before passing it through the interface between adjacent layers. This interface defines which primitive operations and services the lower layer offers to the upper one. The headers are used to establish the peer-to-peer sessions across the sites and some layer implementations use the headers to invoke functions and services at the N+1 or N–1 adjacent layers.

At the transmitter, the user invokes the system by passing data, primitive names and control information physically to the highest layer of the protocol stack. The system then passes the data physically down through the seven layers, adding headers (and possibly trailers), and invoking functions in accordance with the rules of the protocol. At each level, this combined data and header ‘packet’ is termed a protocol data unit or PDU. At the receiving site, the opposite occurs with the headers being stripped from the data as it is passed up through the layers. These header and control messages invoke services and a peer-to-peer logical interaction of entities across the sites. Generally, layers in the same site communicate with parameters passed through primitives, and peer layers across sites communicate with the use of the protocol control information, or header.

At this stage, it should be quite clear that there is NO connection or direct communication between the peer layers of the network. Rather, all physical communication is across the physical layer, or the lowest layer of the stack. Communication is down through the protocol stack on the transmitting stack and up through the stack on the receiving stack. Figure 2.9 shows the full architecture of the OSI model, whilst Figure 2.10 shows the effects of the addition of PCI to the respective PDUs at each layer. As will be realized, the net effect of this extra information is to reduce the overall bandwidth of the communications channel, since some of the available bandwidth is used to pass control information.

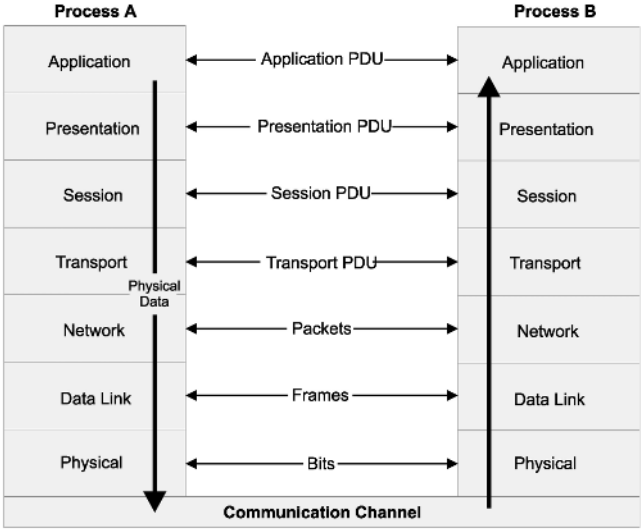


Figure 2.9
Full architecture of OSI model

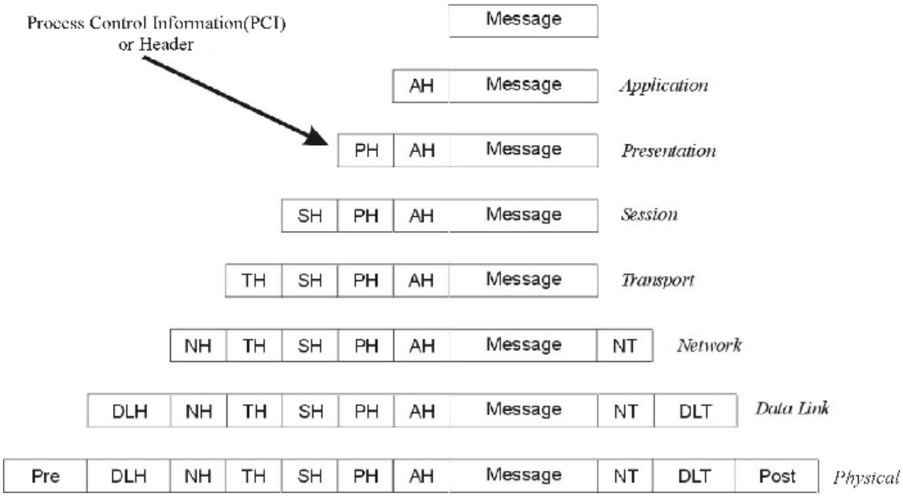

Process Control Information(PCI) or Header
Message
AH Message Application
PH AH Message Presentation
SH PH AH Message Session
TH SH PH AH Message Transport
NH TH SH PH AH Message NT Network
DLH NH TH SH PH AH Message NT DLT Data Link
Pre DLH NH TH SH PH AH Message NT DLT Post Physical

Figure 2.10
OSI message passing

2.4.3 OSI layer services

Briefly, the services provided at each layer of the stack are:

- **Application**
  Provision of network services TO the user's application programs
  Note: the user's actual application programs do NOT reside here
- **Presentation**
  Maps the data representations into an external data format that will enable correct interpretation of the information on receipt. The mapping can also possibly include encryption and/or compression of data
- **Session**
  Control of the communications between the users. This includes the grouping together of messages and the coordination of data transfer between grouped layers. It also affects checkpoints for (transparent) recovery of aborted sessions
- **Transport**
  The management of the communications between the two end systems
- **Network**
  Responsible for the control of the communications network. Functions include routing of data, network addressing, fragmentation of large packets, congestion and flow control.
- **Data link**
  Responsible for sending a frame of data from one system to another. Attempts to ensure that errors in the received bit stream are not passed up into the rest of the protocol stack. Error correction and detection techniques are used here
- **Physical**
  Defines the electrical and mechanical connections at the physical level, or the communication channel itself. Functional responsibilities include modulation, multiplexing and signal generation.

A more specific discussion of each layer is now presented.

2.4.4 Application layer

The application layer is the topmost layer in the OSI reference model. This layer is responsible for giving applications access to the network. Examples of application-layer tasks include file transfer, electronic mail (e-mail) services, and network management. Application-layer services are much more varied than the services in lower layers, because the entire range of application and task possibilities is available here. The specific details depend on the framework or model being used. For example, there are several network management applications. Each of these provides services and functions specified in a different framework for network management. Programs can get access to the application-layer services through application service elements (ASEs). There are a variety of such application service elements; each designed for a class of tasks. To accomplish its tasks, the application layer passes program requests and data to the presentation layer, which is responsible for encoding the application layer's data in the appropriate form.

2.4.5 Presentation layer

The presentation layer is responsible for presenting information in a manner suitable for the applications or users dealing with the information. Functions such as data conversion from EBCDIC to ASCII (or vice versa), use of special graphics or character sets, data compression or expansion, and data encryption or decryption are carried out at this layer. The presentation layer provides services for the application layer above it, and uses the session layer below it. In practice, the presentation layer rarely appears in pure form, and it is the least well defined of the OSI layers. Application- or session-layer programs will often encompass some or all of the presentation layer functions.

2.4.6 Session layer

The session layer is responsible for synchronizing and sequencing the dialog and packets in a network connection. This layer is also responsible for making sure that the connection is maintained until the transmission is complete, and ensuring that appropriate security measures are taken during a 'session' (that is, a connection). The session layer is used by the presentation layer above it, and uses the transport layer below it.

2.4.7 Transport layer

In the OSI reference model, the transport layer is responsible for providing data transfer at an agreed-upon level of quality, such as at specified transmission speeds and error rates. To ensure delivery, outgoing packets are assigned numbers in sequence. The numbers are included in the packets that are transmitted by lower layers. The transport layer at the receiving end checks the packet numbers to make sure all have been delivered and to put the packet contents into the proper sequence for the recipient. The transport layer provides services for the session layer above it, and uses the network layer below it to find a route between source and destination. The transport layer is crucial in many ways, because it sits between the upper layers (which are strongly application-dependent) and the lower ones (which are network-based).

The layers below the transport layer are collectively known as the subnet layers. Depending on how well (or not) they perform their function, the transport layer has to interfere less (or more) in order to maintain a reliable connection.

Subnet service classes

Three types of subnet service are distinguished in the OSI model:

- **Type A:** Very reliable, connection-oriented service
- **Type B:** Unreliable, connection-oriented service
- **Type C:** Unreliable, possibly connectionless service

Transport layer protocols

To provide the capabilities required for whichever service type applies, several classes of transport layer protocols have been defined in the OSI model:

- **TP0 (transfer protocol class 0)**
  It is the simplest protocol. It assumes type A service; that is, a subnet that does most of the work for the transport layer. Because the subnet is reliable, TP0 requires neither error detection or error correction. Because the connection is connection-oriented, packets do not need to be numbered before transmission

- **TP1 (transfer protocol class 1)**
  It assumes a type B subnet; that is, one that may be unreliable. To deal with this, TP1 provides its own error detection, along with facilities for getting the sender to retransmit any erroneous packets
- **TP2 (transfer protocol class 2)**
  It also assumes a type A subnet. However, TP2 can multiplex transmissions, so that multiple transport connections can be sustained over the single network connection
- **TP3 (transfer protocol class 3)**
  It also assumes a type B subnet. TP3 can also multiplex transmissions, so that this protocol has the capabilities of TP1 and TP2
- **TP4 (transfer protocol class 4)**
  It is the most powerful protocol, in that it makes minimal assumptions about the capabilities or reliability of the subnet. TP4 is the only one of the OSI transport-layer protocols that supports connectionless service

2.4.8 Network layer

The network layer is the third lowest layer, or the uppermost subnet layer. It is responsible for the following tasks:

- Determining addresses or translating from hardware to network addresses. These addresses may be on a local network or they may refer to networks located elsewhere on an internetwork. One of the functions of the network layer is, in fact, to provide capabilities needed to communicate on an internetwork
- Finding a route between a source and a destination node or between two intermediate devices
- Establishing and maintaining a logical connection between these two nodes, to establish either a connectionless or a connection-oriented communication. The data is processed and transmitted using the data link layer below the network layer. Responsibility for guaranteeing proper delivery of the packets lies with the transport layer, which uses network layer services
- Fragmentation of large packets of data into frames which are small enough to be transmitted by the underlying data link layer (fragmentation). The corresponding network layer at the receiving node undertakes reassembly of the packet

2.4.9 Data link layer

The data link layer is responsible for creating, transmitting, and receiving data packets. It provides services for the various protocols at the network layer, and uses the physical layer to transmit or receive material. The data link layer creates packets appropriate for the network architecture being used. Requests and data from the network layer are part of the data in these packets (or frames, as they are often called at this layer). These packets are passed down to the physical layer and from there, the data is transmitted to the physical layer on the destination machine. Network architectures (such as Ethernet, ARCnet, Token Ring, and FDDI) encompass the data link and physical layers, which is why these architectures support services at the data link level. These architectures also represent the most common protocols used at the data link level.

The IEEE (802.x) networking working groups have refined the data link layer into two sub layers:

- Logical-link control (LLC) sub layer at the top
- Media-access control (MAC) sub layer at the bottom

The LLC sub layer must provide an interface for the network layer protocols, and control the logical communication with its peer at the receiving side. The MAC sub layer must provide access to a particular physical encoding and transport scheme.

2.4.10 Physical layer

The physical layer is the lowest layer in the OSI reference model. This layer gets data packets from the data link layer above it, and converts the contents of these packets into a series of electrical signals that represent 0 and 1 values in a digital transmission. These signals are sent across a transmission medium to the physical layer at the receiving end. At the destination, the physical layer converts the electrical signals into a series of bit values. These values are grouped into packets and passed up to the data link layer.

Transmission properties defined

The mechanical and electrical properties of the transmission medium are defined at this level. These include the following:

- The type of cable and connectors used. Cable may be coaxial, twisted-pair, or fiber optic. The types of connectors depend on the type of cable
- The pin assignments for the cable and connectors. Pin assignments depend on the type of cable and also on the network architecture being used
- Format for the electrical signals. The encoding scheme used to signal 0 and 1 values in a digital transmission or particular values in an analog transmission depend on the network architecture being used. Most networks use digital signaling, and most use some form of Manchester encoding for the signal

2.5 Interoperability and internetworking

Interoperability is the ability for users of a network to transfer information between different communications systems; irrespective of the way those systems are supported. One definition of interoperability is:

‘*The capability of using similar devices from different manufacturers as effective replacements for each other without losing functionality or sacrificing the degree of integration with the host system. In other words, it is the capability of software and hardware systems on different devices to communicate together. This results in the user being able to choose the right devices for an application independent of the supplier, control system and the protocol.*’

It describes how networks can communicate with each other, as well as how they can share data.

Internetworking is a term that is used to describe the interconnection of differing networks so that they retain their own status as a network. What is important in these concepts is that internetworking devices be made available so that the exclusivity of each of the linked networks is retained, but that the ability to share information, and physical resources if necessary, becomes both seamless and transparent to the end user.

The problems that can be observed through the inability to consider these important concepts can be seen in a typical plant wide situation. For example, consider a manufacturing industry that wishes to connect a series of networks from the plant equipment through to the corporate management level. Equipment will have been purchased from a variety of vendors, most of who will not have previously considered the ability to interact with other vendors, let alone other levels of information system equipment. The difficulties have led to the introduction of a number of standardization schemes, which to a greater or lesser degree comply with the OSI reference model. In the United States, both Boeing Aircraft Company and General Motors – two large manufacturing organizations – have developed schemes to allow interoperability between equipment differing manufacturers. These standards are known as the Technical Office Protocol (TOP) and the Manufacturing Automation Protocol (MAP), and are designed as a subset of the OSI model. At the field sensor level, a standard that is being used is the international Fieldbus standard. These attempts at interoperability are shown in diagrammatic form below. The MAP/TOP approaches were never successful; but their design and implementation have been built into many of the protocol standards used today.

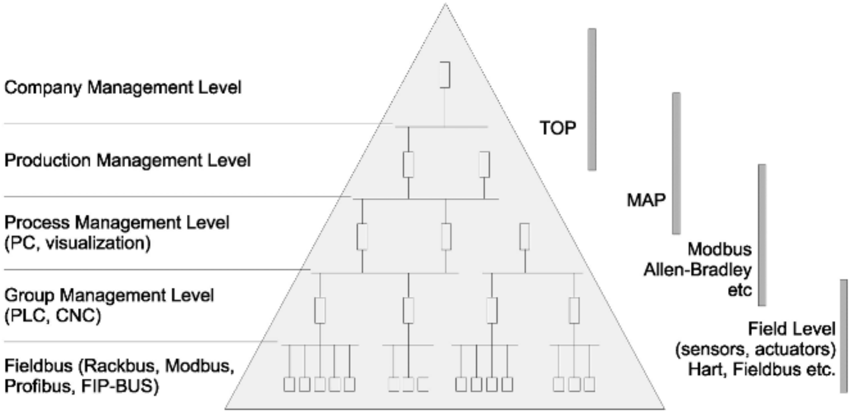


Figure 2.11

It should be noted that at the plant level, the requirement for all seven layers of the OSI model is not appropriate if real time communications are to take place. Hence a simplified OSI model is often preferred for industrial applications where time critical communications is more important than full communications functionality provided by the full seven layer model. Such a protocol stack is acceptable since there will be no internetworking at this level. Two well-known stacks are the Mini-MAP and the Fieldbus standard, which is shown in Figure 2.12.

Generally most industrial protocols are written around three layers:

- The physical layer
- The data link layer
- The application layer

When the reduced OSI model is implemented the following limitations exist:

- The maximum size of the application messages is limited by the maximum size allowed on the channel (as there is no network layer to fragment large packets)
- No routing of messages is possible between different networks (as there is no network layer)
- Only half-duplex communications is possible (as there is no session layer)

- Message formats must be the same for all nodes (as there is no presentation layer)

MiniMAP and the Fieldbus protocol standards use the reduced OSI model with only three layers. Similarly other industrial protocols such as the Allen Bradley Data Highway Plus protocol, Modbus Plus and the HART smart instrumentation protocols have all standardized on the three layers only.

One of the challenges with the use of the OSI model is the concept of interoperability and the need for definition of another layer above the application layer, called the 'user' layer.

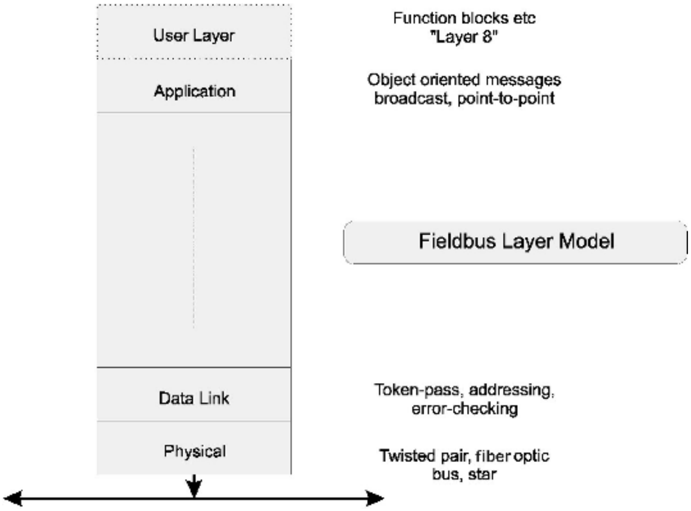


Figure 2.12
'Collapsed' OSI stack

However, it is the so-called user layer that actually specifies the type of data in information and how it is to be used. Specification of the user layer is essential to ensure complete performance of a fieldbus system.

From the point of view of internetworking, TCP/IP operates as a set of programs that interacts at the transport and network layer levels without needing to know the details of the technologies used in the underlying layers. As a consequence this has developed as a *de facto* industrial internetworking standard. Many manufacturers of proprietary equipment are using TCP/IP to facilitate internetworking.

2.6 Protocols and protocol standards

A protocol has already been defined as the rules for exchanging data in a manner that is understandable to both the transmitter and the receiver. There must be a formal and agreed set of rules if the communication is to be successful. The rules generally relate to such responsibilities as error detection and correction methods, flow control methods, and voltage and current standards. However, there are other properties such as the size of the data packet that are important in the protocols that are used in LANs.

Another important responsibility is the method of routing the packet, once it has been assembled. In a self contained local area network i.e. intranet work, this is not a problem, since all packets will eventually reach their destination by virtue of design. However, if the packet is to be switched across networks i.e. on an internetwork – such as a wide area network – then a routing decision must be made. In this regard we have already examined the use of a datagram service *vis à vis* a virtual circuit.

There are two other classes of service provision that you might encounter. These are the acknowledged connectionless service ALS and the unconfirmed connection oriented service UOS, sometimes called send-and-pray. The ALS service is used for real-time communications. It is similar to the datagram or connectionless service, except it provides the transmitter with an acknowledgment that the data has been delivered. The UOS service is a connection oriented service that insists a link be established before data packets are transmitted. However, subsequent delivery of the packets is not acknowledged.

In summary, there are many different types of protocol, but they can be classified in terms of their functional emphasis. One scheme of classification is:

- **Master/slave vs peer-to-peer**
  A master slave relationship requires that one of the communicators act as a master controller. Peer-to-peer protocols allow all communications to take place as and when required
- **Connection oriented**
  Connectionless; acknowledged connectionless; unconfirmed connection oriented. These are described above
- **Asynchronous vs synchronous**
  Synchronous protocols send data at the clock rate of the network. Asynchronous protocols send data one byte at a time, with a varying delay between each byte
- **Layered vs monolithic**
  The OSI model illustrates a layered approach to protocols. The monolithic approach uses a single layer to provide all functionality
- **Heavy vs light**
  A heavy protocol has a wide range of functions built in, and consequently incurs a high processing delay overhead. A light protocol incurs low processing delay but only provides minimal functionality

2.7 IEEE/ISO standards

The Institute of Electrical and Electronic Engineers in the United States has been given the task of developing standards for local area networking under the auspices of the IEEE 802 committees. Once a draft standard has been agreed and completed, it is passed to the International Standards Organization ISO for ratification. The corresponding ISO standard, which is generally internationally accepted, is given the same committee number as the IEEE committee, with the addition of an extra ‘8’ in front of the number i.e. the IEEE 802 committees are equivalent to the ISO 8802 committees.

These IEEE committees, consisting of various technical, study and working groups, provide recommendations for various features within the networking field. Each committee is given a specific area of interest, and a separate subnumber to distinguish it. The main committees and the standards that they are working on are described below.

IEEE 802.1 High level interface

The HILI sub committee is concerned with issues such as high level interfaces, internetworking and addressing.

There are a series of sub committees, such as:

• 802.1B	LAN management
• 802.1D	Local bridging

- 802.1E System load protocol
- 802.1F Guidelines for layer management standards
- 802.1G Remote MAC bridges
- 802.1I MAC bridges (FDDI supplement)

IEEE 802.2 Logical link control

This is the interface between the network layer and the specific network environments at the physical layer. The IEEE has divided the data link layer in the OSI model into two sub layers – the media access MAC sub layer, and the logical link layer LLC. The logical link control protocol is common for all IEEE 802 standard network types. This provides a common interface to the network layer of the protocol stack. The protocol used at this sub layer is based on IBM's SDLC protocol, and can be used in three modes, or types.

These are:

- Type 1: Unacknowledged connectionless link service
- Type 2: Connection oriented link service
- Type 3: Acknowledged connectionless link service, used in real time applications such as manufacturing control

IEEE 802.3 CSMA/CD

The carrier sense, multiple accesses with collision detection type LAN is commonly – but strictly speaking incorrectly – known as an Ethernet LAN. Ethernet refers to the original DEC/INTEL/XEROX product known as Version II (or Bluebook) Ethernet.

Subsequent to ratification this system has been known as IEEE 802.3. IEEE 802.3 is virtually identical, but not absolutely identical to Bluebook Ethernet, in that they differ in two bytes within the frame. The following chapter will deal with this anomaly.

Subsequently, two additional specifications have been approved viz. IEEE 802.3u (100 Mbps or 'fast' Ethernet) and IEEE 8023z (1000 Mbps or 'gigabit' Ethernet).

IEEE 802.4 Token bus

The other major access method for a shared medium is the use of a token. This is a type of data frame that a station must possess before it can transmit messages. The stations are connected to a passive bus, although the token logically passes around in a cyclic manner.

This standard is the ratification of the Token Bus LAN developed by General Motors for its manufacturing automation protocol (MAP). The media used is usually broadband coax, and speeds vary from 1 Mbps to 10 Mbps.

IEEE802.5 Token ring

As in 802.4, data transmission can only occur when a station holds a token. The logical structure of the network wiring is in the form of a ring, and each message must cycle through each station connected to the ring.

This standard is the ratified version of the IBM token ring LAN. However, where IBM token ring supports speed of 4 and 16 Mbps, IEEE 802.5 supports 1 and 4 Mbps. The physical media for the token ring can be unshielded twisted pair, coaxial cable or optical fiber.

The original specification called for a single ring, which creates a problem if the ring gets broken. A subsequent enhancement of the specification, called IEEE 802.5u, introduces the concept of a dual redundant ring, which enables the system to continue operating in case of a cable break.

Work is currently underway on a 100 Mbps token ring specification.

IEEE 802.6 Metropolitan area networks

This committee is responsible for defining the standards for MANs. It has recommended that a system known as distributed queue data bus (DQDB) be utilized as a MAN standard.

The DQDB network is sponsored by Telecom Australia and defines the protocol for integrated voice and data on the same medium, within an area up to 15 km in diameter.

IEEE 802.7 Broadband LANs technical advisory group (TAG)

The 802.7 Committee provides technical advice on broadband technique.

IEEE 802.8 Fiber optic LANs TAG

The fiber optic equivalent of the 802.7 broadband TAG. The committee is attempting to standardize physical compatibility with FDDI and synchronous optical networks (SONET). It is also investigating single mode fiber and multimode fiber architectures.

IEEE 802.9 Integrated voice and data LANs

This committee has recently released a specification for isochronous Ethernet as IEEE 802.9a. It provides a 6.144 Mbps voice service (96 channels at 64 kbps) multiplexed with 10 Mbps data on a single cable. It is designed for multimedia applications.

IEEE 802.10 Secure LANs

Current proposals include two methods to address the lack of security in the original specifications. These are:

- A secure data exchange sub layer SDE sitting between the LLC and the MAC sub layer. There will be different SDEs for different systems i.e. military and medical
- A secure interoperable LAN system architecture SILS. This will define system standards for secure LAN communications

IEEE 802.11 Wireless LANs

The IEEE802.11 Wireless LAN standard uses the 2.4 GHz band and allows operation to 1 or 2 Mbps. The 802.11b standard also uses the 2.4 GHz band, but allows operation at 11 Mbps. The latest IEEE 802.11a specification use the 5.7 GHz band instead and allows operation at 54 Mbps.

IEEE 802.12 Fast LANs

This specification covers the system known as 100VG AnyLAN. Developed by Hewlett-Packard, this system operates on voice grade (CAT3) cable – hence the VG in the name. The AnyLAN indicates that the system can interface with both IEEE 802.3 and IEEE 802.5 networks (by means of a special speed adaptation bridge).

2.8 Network topologies

2.8.1 Broadcast and point-to-point topologies

The way the nodes are connected to form a network is known as its topology. There are many topologies available but they form two basic types, broadcast and point-to-point.

Broadcast topologies are those where the message ripples out from the transmitter to reach all nodes. There is no active regeneration of the signal by the nodes and so signal propagation is independent of the operation of the network electronics. This then limits the size of such networks.

Figure 2.13 shows an example of a broadcast topology.

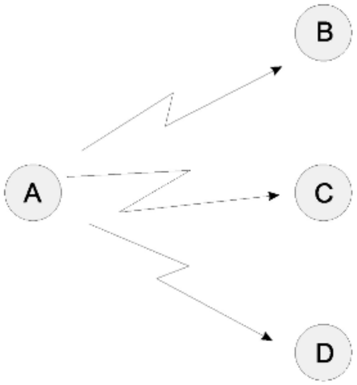


Figure 2.13
Broadcast topology

In a point-to-point communications network, however, each node is communicating directly with only one node. That node may actively regenerate the signal and pass it on to its nearest neighbor. Such networks have the capability of being made much larger. Figure 2.14 shows some examples of point-to-point topologies.

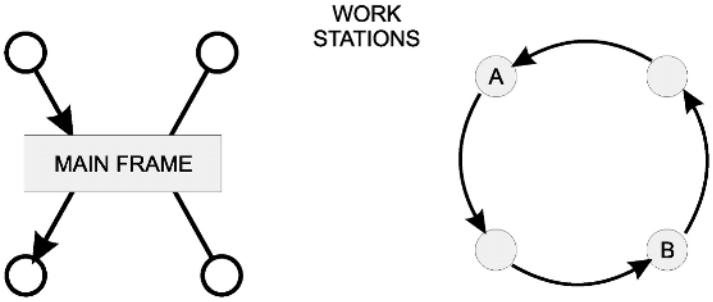


Figure 2.14
Point-to-point topologies

2.8.2 Logical and physical topologies

A logical topology defines how the elements in the network communicate with each other, and how information is transmitted through a network. The different types of media-access methods, determine how a node gets to transmit information along the network. In a bus topology, information is broadcast, and every node gets the same information within the amount of time it actually takes a signal to cover the entire length of cable. This time interval limits the maximum speed and size for the network. In a ring topology, each node hears from exactly one node and talks to exactly one other node. Information is passed sequentially, in an order determined by a predefined process. A polling or token mechanism is used to determine who has transmission rights, and a node can transmit only when it has this right.

A physical topology defines the wiring layout for a network. This specifies how the elements in the network are connected to each other electrically. This arrangement will determine what happens if a node on the network fails. Physical topologies fall into three main categories... bus, star, and ring topology. Combinations of these can be used to form hybrid topologies to overcome weaknesses or restrictions in one or other of these three component topologies.

2.9 Bus topology

A bus refers to both a physical and a logical topology. As a physical topology, a bus describes a network in which each node is connected to a common single communication channel or ‘bus’. This bus is sometimes called a backbone, as it provides the spine for the network. Every node can hear each message packet as it goes past.

Logically, a passive bus is distinguished by the fact that packets are broadcast and every node gets the message at the same time. Transmitted packets travel in both directions along the bus, and need not go through the individual nodes, as in a point-to-point system. Rather, each node checks the destination address that is included in the message packet to determine whether that packet is intended for the specific node. When the signal reaches the end of the bus, an electrical terminator absorbs the packet energy to keep it from reflecting back again along the bus cable, possibly interfering with other messages already on the bus. Each end of a bus cable must be terminated, so that signals are removed from the bus when they reach the end.

In a bus topology, nodes should be far enough apart so that they do not interfere with each other. However, if the backbone bus cable is too long, it may be necessary to boost the signal strength using some form of amplification, or repeater. The maximum length of the bus is limited by the size of the time interval that constitutes ‘simultaneous’ packet reception. Figure 2.15 illustrates the bus topology.

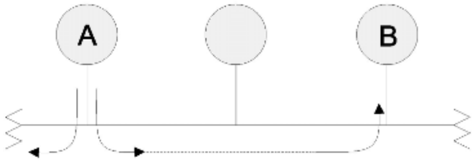


Figure 2.15
Bus topology

Bus topology advantages

Bus topologies offer the following advantages:

- A bus uses relatively little cable compared to other topologies, and arguably has the simplest wiring arrangement
- Since nodes are connected by high impedance tappings across a backbone cable, it’s easy to add or remove nodes from a bus. This makes it easy to extend a bus topology
- Architectures based on this topology are simple and flexible
- The broadcasting of messages is advantageous for one-to-many data transmissions

Bus topology disadvantages

These include the following:

- There can be a security problem, since every node may see every message, even those that are not destined for it
- Diagnosis/troubleshooting (fault-isolation) can be difficult, since the fault can be anywhere along the bus

- There is no automatic acknowledgment of messages, since messages get absorbed at the end of the bus and do not return to the sender
- The bus cable can be a bottleneck when network traffic gets heavy. This is because nodes can spend much of their time trying to access the network

2.10 Star topology

A star topology is a physical topology in which multiple nodes are connected to a central component, generally known as a hub. The hub of a star usually is just a wiring center; that is, a common termination point for the nodes, with a single connection continuing from the hub. In some cases, the hub may actually be a file server (a central computer that contains a centralized file and control system), with all its nodes attached directly to the server. As a wiring center, a hub may, in turn, be connected to the file server or to another hub.

All signals, instructions, and data going to and from each node must pass through the hub to which the node is connected. The telephone system is doubtless the best known example of a star topology, with lines to individual customers coming from a central telephone exchange location. There are not many LAN implementations that use a logical star topology. The low impedance ARCnet networks are probably the best examples. However, you will see that the physical layout of many other LANs look like a star topology even though they are considered to be something else. An example of a star topology is shown in Figure 2.16.

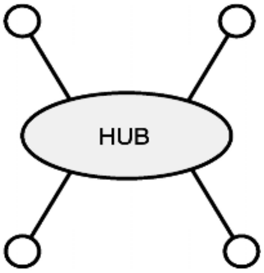


Figure 2.16
Star topology

Star topology advantages

- Troubleshooting and fault isolation is easy
- It is easy to add or remove nodes, and to modify the cable layout
- Failure of a single node does not isolate any other node
- The inclusion of a central hub allows easier monitoring of traffic for management purposes

Star topology disadvantages

- If the hub fails, the entire network fails. Sometimes a backup central machine is included, to make it possible to deal with such a failure
- A star topology requires a lot of cable

2.11 Ring topology

A ring topology is both a logical and a physical topology. As a logical topology, a ring is distinguished by the fact that message packets are transmitted sequentially from node to node, in a predefined order, and as such it is an example of a point-to-point system. Nodes are arranged in a closed loop, so that the initiating node is the last one to receive a packet. As a physical topology, a ring describes a network in which each node is connected to exactly two other nodes.

Information traverses a one-way path, so that a node receives packets from exactly one node and transmits them to exactly one other node. A message packet travels around the ring until it returns to the node that originally sent it. In a ring topology, each node can act as a repeater, boosting the signal before sending it on. Each node checks whether the message packet's destination node matches its address. When the packet reaches its destination, the destination node accepts the message, then sends it back to the sender, to acknowledge receipt.

As you will see later in this chapter, since ring topologies use token passing to control access to the network, the token is returned to sender with the acknowledgment. The sender then releases the token to the next node on the network. If this node has nothing to say, the node passes the token on to the next node, and so on. When the token reaches a node with a packet to send, that node sends its packet. Physical ring networks are rare, because this topology has considerable disadvantages compared to a more practical star-wired ring hybrid, which is described later.

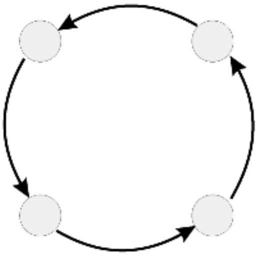


Figure 2.17
Ring topology

Ring topology advantages

- A physical ring topology has minimal cable requirements
- No wiring center or closet is needed
- The message can be automatically acknowledged
- Each node can regenerate the signal

Ring topology disadvantages

- If any node goes down, the entire ring goes down
- Diagnosis/troubleshooting (fault isolation) is difficult because communication is only one-way
- Adding or removing nodes disrupts the network
- There will be a limit on the distance between nodes

As well as these three main topologies, some of the more important variations will now be considered. Once again, you should be clear that these are just variations, and should not be considered as topologies in their own right.

2.12 Other types of topology

2.12.1 Star-wired ring topology

A star-wired ring topology, also known as a hub topology, is a hybrid physical topology that combines features of the star and ring topologies. Individual nodes are connected to a central hub, as in a star network. Within the hub, however, the connections are arranged into an internal ring. Thus, the hub constitutes the ring, which must remain intact for the network to function. The hubs, known as multistation access units (MAUs) in IBM token ring network terminology, may be connected to other hubs. In this arrangement, each internal ring is opened and connected to the attached hubs, to create a larger, multi-hub ring.

The advantage of using star wiring instead of simple ring wiring is that it is easy to disconnect a faulty node from the internal ring. The IBM data connector is specially designed to close a circuit if an attached node is disconnected physically or electrically. By closing the circuit, the ring remains intact, but with one less node. The IBM token ring networks are the best-known example of a star-wired ring topology. In token ring networks, a secondary ring path can be established and used if part of the primary path goes down. The star-wired ring is illustrated in Figure 2.18.

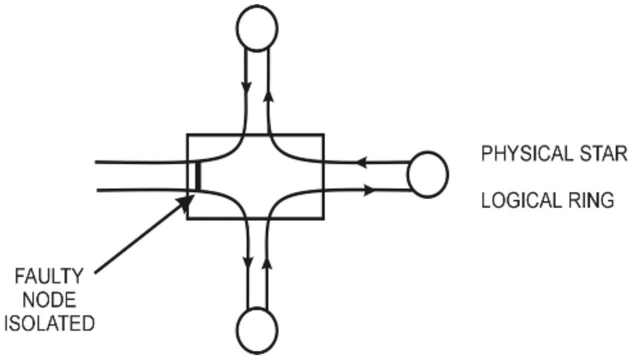


Figure 2.18
Star-wired ring

The advantages of a star-wired ring topology include:

- Troubleshooting, or fault isolation, is relatively easy
- The modular design makes it easy to expand the network, and makes layouts extremely flexible
- Individual hubs can be connected to form larger rings
- Wiring to the hub is flexible

The disadvantages of a star-wired ring topology include:

- Configuration and cabling may be complicated because of the extreme flexibility of the arrangement.

2.12.2 Distributed star topology

A distributed star topology is a physical topology that consists of two or more hubs, each of which is the center of a star arrangement. A good example of such a topology is an

ARCnet network with at least one active hub and one or more active or passive hubs. The 100VG ANYLAN utilizes a similar topology.

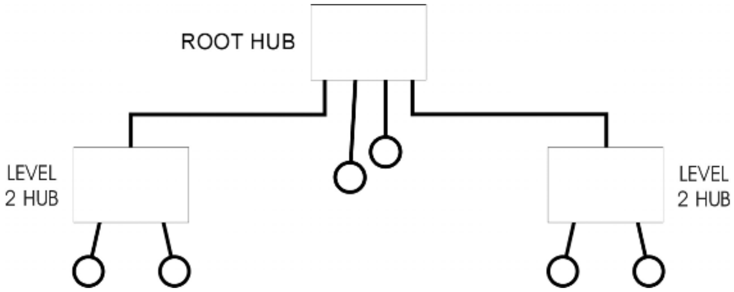


Figure 2.19
Distributed star topology

2.12.3 Mesh topology

A mesh topology is a physical topology in which there are at least two paths to and from every node. This type of topology is advantageous in hostile environments in which connections are easily broken. If a connection is broken, at least one substitute path is always available. A more restrictive definition requires each node to be connected directly to every other node. Because of the severe connection requirements, such restrictive mesh topologies are feasible only for small networks.

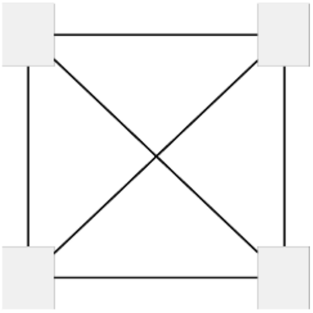


Figure 2.20
Mesh network

2.12.4 Tree topology

A tree topology, also known as a distributed bus or a branching tree topology, is a hybrid physical topology that combines features of star and bus topologies. Several buses may be daisy-chained together, and there may be branching at the connections (which will be hubs). The starting end of the tree is known as the root or head end. This type of topology is used in delivering cable television services.

The advantages of a tree topology are:

- The network is easy to extend by just adding another branch, and that fault isolation is relatively easy

The disadvantages include:

- If the root goes down, the entire network goes down
- If any hub goes down, all branches off that hub go down
- Access becomes a problem if the entire network becomes too big

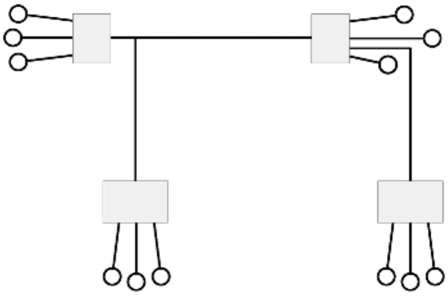


Figure 2.21
Tree topology

2.13 Media access methods

A common and important method of differentiating between different LAN types is to consider their media access methods. Since there must be some method of determining which node can send a message, this is a critical area that determines the efficiency of the LAN. There are a number of methods, which can be considered, of which the two most common in current LANs are the **contention method** and the **token passing** method. You will become familiar with these as part of your study of LANs, although some of the other methods will also be briefly discussed.

2.13.1 Contention systems

The basis for a first-come-first-served media accesses method. This operates in a similar manner to polite human communication. We listen before we speak, deferring to anyone who already is speaking. If two of us start to speak at the same time, we recognize that fact and both stop, before starting our messages again a little later. In a contention-based access method, the first node to seek access when the network is idle will be able to transmit. Contention is at the heart of the **carrier sense multiple access/collision detection** (CSMA/CD) access method used in the IEEE 802.3 and Ethernet V2 networks.

The carrier sense component involves a node wishing to transmit a message listening to the transmission media to ensure there is no 'carrier' present. In fact, the signaling method used on Ethernet type systems that make use of this method do not use a carrier in its true sense, and the name relates back to the original Aloha project in Hawaii that used radio links for transmission. The length of the channel and the finite propagation delay means that there is still a distinct probability that more than one transmitter will attempt to transmit at the same time, as they both will have heard 'no carrier'. The collision detection logic ensures that more than one message on the channel simultaneously will be detected and transmission, from both ends, eventually stopped. The system is a probabilistic system, since access to the channel cannot be ascertained in advance.

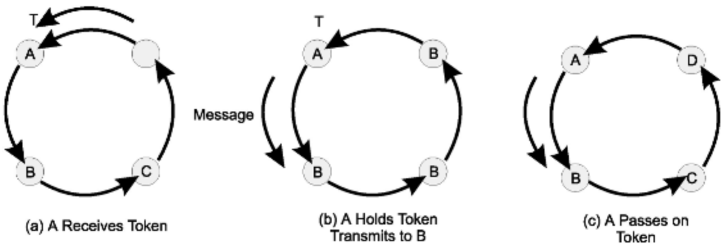
2.13.2 Token passing

Token passing is a deterministic media access method in which a token is passed from node to node, according to a predefined sequence. A token is a special packet, or frame, consisting of a signal sequence that cannot be mistaken for a message. At any given time, the token can be available or in use. When an available token reaches a node, that node can access the network for a maximum predetermined time, before passing the token on.

This deterministic access method guarantees that every node will get access to the network within a given length of time, usually in the order of a few milliseconds. This is in contrast to a probabilistic access method (such as CSMA/CD), in which nodes check for network activity when they want to access the network, and the first node to claim the idle network gets access to it. Because each node gets its turn within a fixed period, deterministic access methods are more efficient on networks that have heavy traffic. With such networks, nodes using probabilistic access methods spend much of their time competing to gain access and relatively little time actually transmitting data over the network. Network architectures that support the token passing access method include token bus, ARCnet, FDDI, and token ring.

To transmit, the node first marks the token as ‘in use’, and then transmits a data packet, with the token attached. In a ring topology network, the packet is passed from node to node, until the packet reaches its destination. The recipient acknowledges the packet by sending the message back to the sender, who then sends the token on to the next node in the network.

In a bus topology network, the next recipient of a token is not necessarily the node that is nearest to the current token passing node. Rather, the next node is determined by some predefined rule. The actual message is broadcast on to the bus for all nodes to ‘hear’. For example, in an ARCnet or token bus network, the token is passed from a node to the node with the next lower network address. Networks that use token passing generally have some provision for setting the priority with which a node gets the token. Higher-level protocols can specify that a message is important and should receive higher priority.



(a) A Receives Token

(b) A Holds Token Transmits to B

(c) A Passes on Token

Figure 2.22
Token passing

A token ring network requires an active monitor (AM) and one or more standby monitors (SMs). The AM keeps track of the token to make sure it has not been corrupted, lost, or sent to a node that has been disconnected from the network. If any of these things happens, the AM generates a new token, and the network is back in business. The SM makes sure the AM is doing its job and does not break down and get disconnected from the network. If the AM is lost, one of the SMs becomes the new AM, and the network is again in business. These monitoring capabilities make for complex circuitry on network interface cards that use this media access method.

2.13.3 Polling

Polling refers to a process of checking elements, such as computers or queues, in some defined order, to see whether the polled element needs attention (wants to transmit, contains jobs, and so on). In roll call polling, the polling sequence is based on a list of elements available to the controller. In contrast, in hub polling, each element simply polls the next element in the sequence.

In LANs, polling provides a deterministic media access method in which the server polls each node in succession to determine whether that node wants to access the network. In some systems, the polling is done by means of software messages being passed to and fro, which could slow down the process. In order to overcome this problem, systems such as 100VG Any LAN employ a hardware-polling message, which uses voltage levels to determine whether a node wants to be serviced.

3

Ethernet networks

Objectives

When you have completed study of this chapter you should be able to:

- Describe the major hardware components of an IEEE 802.3 CSMA/CD network
- Explain the method of connection of 10Base5, 10Base2 and 10BaseT networks
- Explain the operation of the CSMA/CD protocol
- List the fields in the Ethernet data frames
- Describe the causes of Ethernet collisions and how to reduce them
- Demonstrate how to apply the Ethernet design rules

3.1 IEEE 802.3 CSMA/CD ('Ethernet')

The Ethernet network concept was developed by Xerox Corporation at its Palo Alto Research Center (PARC) in the mid-seventies. It was based on the work done by researchers at the University of Hawaii where there were campus sites on the various islands. Its ALOHA network was set up using radio broadcasts to connect the various sites. This was colloquially known as their 'Ethernet' since it used the 'ether' as the transmission medium and created a network 'net' between the sites.

The philosophy was quite straightforward. Any station that wanted to broadcast to another station would do so immediately. The receiving stations then had a responsibility to acknowledge the message; thus advising the original transmitting station of a successful reception of the original message. This primitive system did not rely on any detection of collisions (two radio stations transmitting at the same time) but rather waited for an acknowledgment back within a predefined time.

The initial system installed by Xerox was so successful that they soon applied the system to their other sites typically connecting office equipment to shared resources such as printers and large computers acting as repositories of large databases, for example.

In 1980, the Ethernet Consortium consisting of Xerox, Digital Equipment Corporation and Intel (sometimes called the DIX consortium) issued a joint specification based on the Ethernet concepts and known as the Ethernet Blue Book 1 specification. This was later superseded by the Ethernet Blue Book 2 specification, which was offered to the IEEE as a standard. In 1983, the IEEE issued the 802-3 standard for Carrier Sense; Multiple Access; Collision Detect LANs based on the Ethernet standard, which gave this networking standard even more credibility.

As a result of this, there are three standards in existence. The first – often-termed Ethernet Version 1 – can be disregarded as very little equipment based on this standard is still in use. Ethernet Version 2 or 'Blue Book Ethernet' is, however, still in use and there is a potential for incompatibility with the IEEE 802.3 standard. Whilst these differences are minor, they are nonetheless significant. Despite the generic term 'Ethernet' being applied to all CSMA/CD networks, it should be reserved for the original DIX standard. This book will continue with popular use and refer to all the LANs of this type as Ethernet, unless it is important to distinguish between them.

The original Ethernet specification was developed around CSMA/CD. Later versions (from 100 Mbps upwards) also support full-duplex, although they support CSMA/CD for the sake of backward compatibility. Industrial versions of Ethernet typically operate at 100 Mbps and above in full-duplex mode, and support the IEEE 802.1p/Q modified frame structure. This allows highly deterministic operation.

3.2 Physical layer

802.3 standard defines a range of cable types that can be used for a network based on this standard. They include coaxial cable, twisted pair cable and fiber optic cable. In addition, there are different signaling standards and transmission speeds that can be utilized. These include both base band and broadband signaling, and speeds of 1 Mbps and 10 Mbps. The standard is continuing to evolve, and this manual will look at 100 Mbps CSMA/CD systems in the next chapter.

The IEEE 802.3 standard documents (ISO 8802.3) support various cable media and transmission rates up to 10 Mb/s as follows:

- **10Base2**

  Thin wire coaxial cable (0.25 inch diameter), 10 Mbps, single cable bus
- **10Base5**

  Thick wire coaxial cable (0.5 inch diameter), 10 Mbps, single cable bus
- **10BaseT**

  Unscreened twisted pair cable (0.4 to 0.6 mm conductor diameter), 10 Mbps, twin cable bus
- **10BaseF**

  Optical fiber cables, 10 Mbps, twin fiber bus
- **1Base5**

  Unscreened twisted pair cables, 1 Mbps, twin cable bus
- **10Broad36**

  Cable television (CATV) type cable, 10 Mbps, broadband

3.2.1 10Base5 systems

This is a coaxial cable system and uses the original cable for Ethernet systems – generically called ‘Thicknet’. It is a coaxial cable, of 50-ohm characteristic impedance, and yellow or orange in color. The naming convention for 10Base5: means 10 Mbps; base band signaling on a cable that will support 500-meter segment lengths. It is difficult to work with, and so cannot normally be taken to the node directly. Instead, it is laid in a cabling tray etc and the transceiver electronics (medium attachment unit, MAU) is installed directly on the cable. From there an intermediate cable, known as an attachment unit interface (AUI) cable is used to connect to the NIC. This cable can be a maximum of 50 meters long, compensating for the lack of flexibility of placement of the segment cable. The AUI cable consists of 5 individually shielded pairs – two each (control and data) for both transmit and receive; plus one for power.

Cutting the cable and inserting an N-connector and a coaxial Tee or more commonly by using a ‘bee sting’ or ‘vampire’ tap can make the MAU connection to the cable. This is a mechanical connection that clamps directly over the cable. Electrical connection is made via a probe that connects to the center conductor and sharp teeth, which physically puncture the cable sheath to connect to the braid. These hardware components are shown in Figure 3.1.

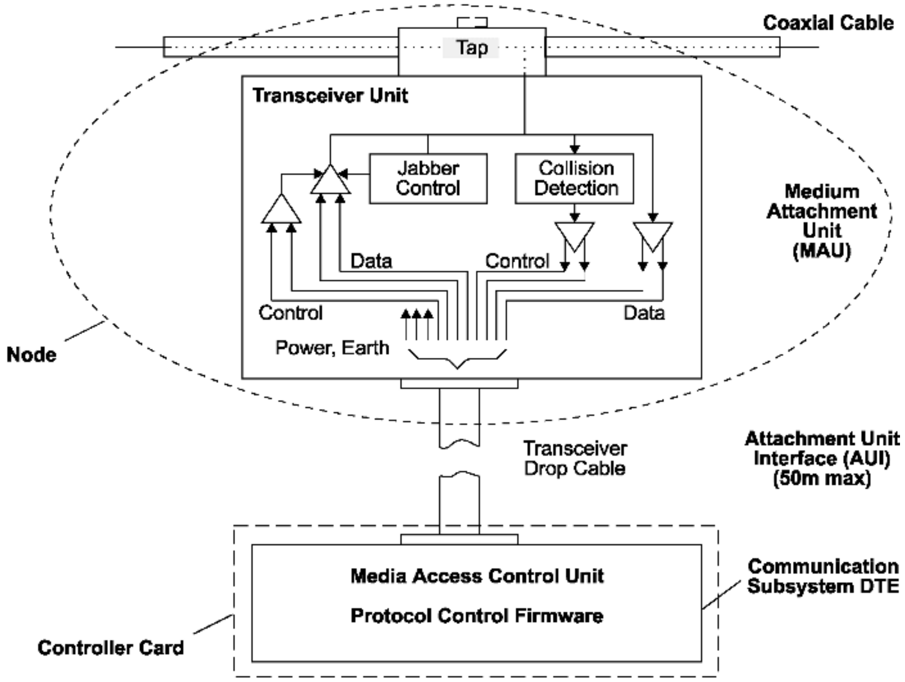

Coaxial Cable
Tap
Transceiver Unit
Jabber Control
Collision Detection
Data
Control
Control
Data
Power, Earth
Medium Attachment Unit (MAU)
Node
Transceiver Drop Cable
Attachment Unit Interface (AUI) (50m max)
Media Access Control Unit
Protocol Control Firmware
Communication Subsystem DTE
Controller Card

Figure 3.1
10Base5 hardware components

The location of the connection is important to avoid multiple electrical reflections on the cable, and the Thicknet cable is marked every 2.5 meters with a black or brown ring to indicate where a tap should be placed. Fan out boxes can be used if there are a number of nodes for connection, allowing a single tap to feed each node as though it was individually connected. The connection at either end of the AUI cable is made through a 25-pin D-connector, with a slide latch, often called a DIX connector after the original consortium.

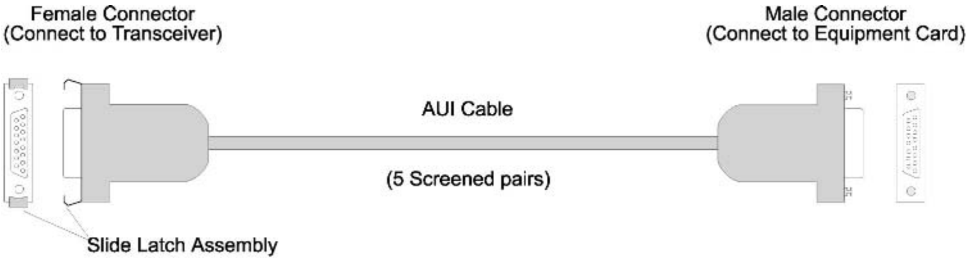


Figure 3.2
AUI cable connectors

There are certain requirements if this cable architecture is used in a network. These include:

- Segments must be less than 500 meters in length to avoid signal attenuation problems
- No more than 100 taps on each segment i.e. not every potential connection point can support a tap
- Taps must be placed at integer multiples of 2.5 meters
- The cable must be terminated with a 50-ohm terminator at each end
- One end of the cable shield must be earthed

The physical layout of a 10Base5 Ethernet segment is shown in Figure 3.3.

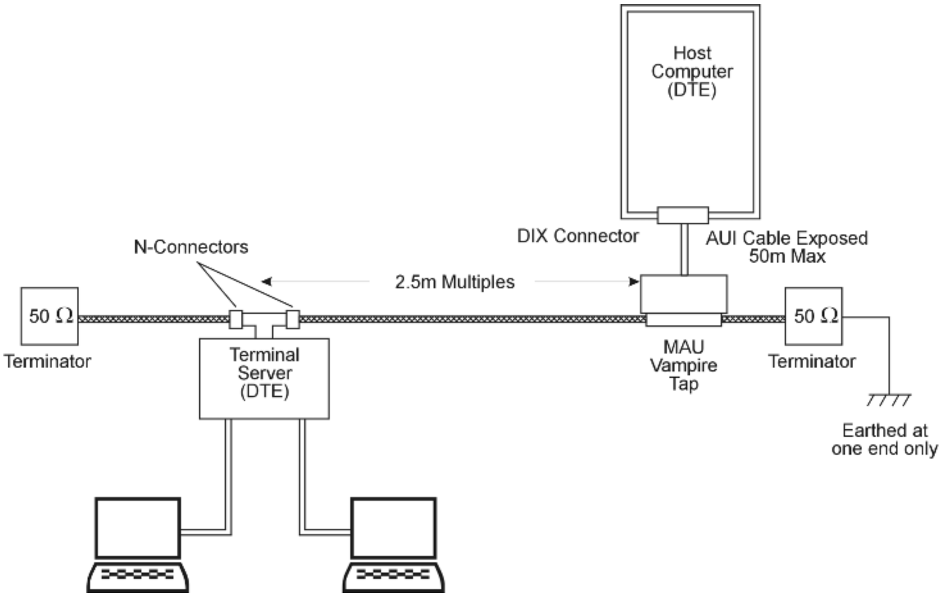


Figure 3.3
10Base5 Ethernet segment

The Thicknet cable was extensively used as a backbone cable until recently but 10BaseT and fiber is becoming more popular. Note that when a MAU (tap) and AUI cable is used, the on board transceiver on the NIC is not used. Rather, there is a transceiver in the MAU and this is fed with power from the NIC via the AUI cable. Since the transceiver is remote from the NIC, the node needs to be aware that the termination can detect collisions if they occur. This confirmation is performed by a signal quality error (SQE), or heartbeat, test function in the MAU. The SQE signal is sent from the MAU to the node on detecting a collision on the bus. However, on completion of every frame transmission by the MAU, the SQE signal is asserted to ensure that the circuitry remains active, and that collisions can be detected. You should be aware that not all components support SQE test and mixing those that do with those that don't could cause problems. Specifically, if a NIC was to receive a SQE signal after a frame had been sent, and it was not expecting it, the NIC could think it was seeing a collision. In turn, as you will see later in the manual, the NIC will then transmit a jam signal.

3.2.2 10Base2 systems

The other type of coaxial cable Ethernet networks is 10Base2 and often referred to as 'Thinnet' or sometimes 'thinwire Ethernet'. It uses type RG-58 A/U or C/U with a 50-ohm characteristic impedance and of 5 mm diameter. The cable is normally connected to the NICs in the nodes by means of a BNC T-piece connector, and represents a daisy chain approach to cabling.

Connectivity requirements include:

- It must be terminated at each end with a 50-ohm terminator
- The maximum length of a cable segment is 185 meters and NOT 200 meters
- No more than 30 transceivers can be connected to any one segment
- There must be a minimum spacing of 0.5 meters between nodes.
- It may not be used as a link segment between two 'Thicknet' segments
- The minimum bend radius is 5 cm

The physical layout of a 10Base2 Ethernet segment is shown in Figure 3.4.

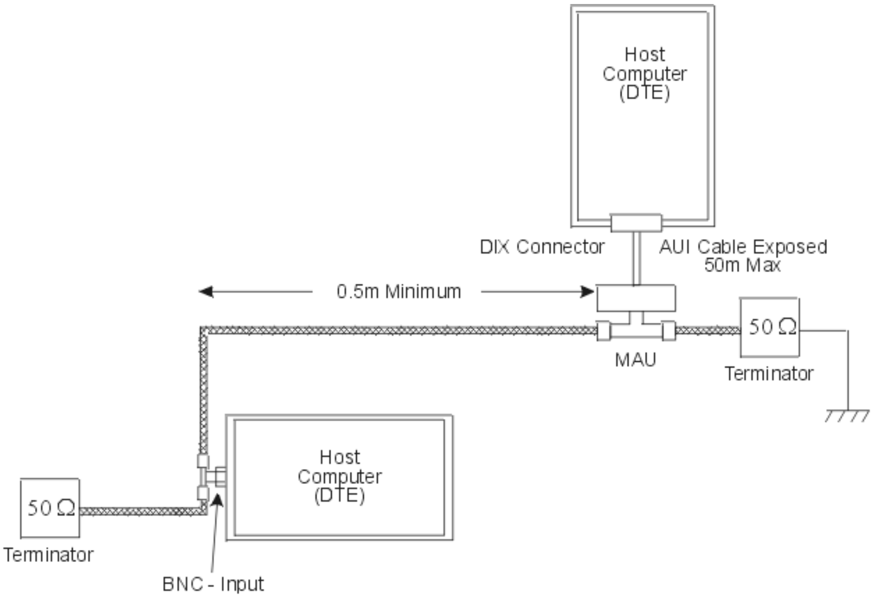


Figure 3.4
10Base2 Ethernet segment

The use of Thinnet cable was, and remains, very popular as a cheap and relatively easy way to set up a network. However, there are disadvantages with this approach. A cable fault can bring the whole system down very quickly. To avoid such a problem, the cable is often taken to wall connectors with a make–break connector incorporated. The connection to the node can then be made by 'fly leads' of the same cable type. It is important to take the length of these fly leads into consideration in any calculation on cable length! There is also provision for remote MAUs in this system, with AUI cables making the node connection, in a similar manner to the Thicknet connection.

3.2.3 10BaseT

The 10BaseT standard for Ethernet networks uses AWG24 unshielded twisted pair (UTP) cable for connection to the node. The physical topology of the standard is a star, with nodes connected to a wiring hub, or concentrator. Concentrators can then be connected to a backbone cable that may be coax or fiber optic. The node cable can be category 3 or category 4 cable, although you would be well advised to consider category 5 for all new installations. This will allow an upgrade path as higher speed networks become more common, and given the small proportion of cable cost to total cabling cost, will be a worthwhile investment. The node cable has a maximum length of 100 meters; consists of two pairs for receive and transmit and is connected via RJ45 plugs. The wiring hub can be considered as a local bus internally, and so the topology is still considered as a logical bus topology. Figure 3.5 shows schematically how the 10BaseT nodes are interconnected by the hub.

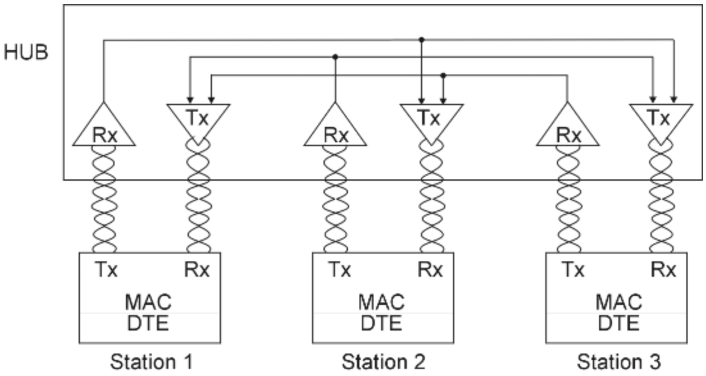


Figure 3.5
Schematic 10BaseT system

Collisions are detected by the NIC and so an input signal must be retransmitted by the hub on all output pairs. The electronics in the hub must ensure that the stronger retransmitted signal does not interfere with the weaker input signal. The effect is known as far end crosstalk (FEXT), and is handled by special adaptive crosstalk echo cancellation circuits.

The standard has become increasingly popular for new networks, although there are some disadvantages that should be recognized:

- The cable is not very resistant to electrostatic electrical noise, and may not be suitable for some industrial environments

Whilst the cable is inexpensive, there is the additional cost of the associated wiring hubs to be considered:

- The node cable is limited to 100 m

Advantages of the system include:

- Intelligent hubs are available that can determine which spurs from the hub receive information. This improves on the security of the network – a feature that has often been lacking in a broadcast, common media network such as Ethernet

- Flood wiring can be installed in a new building, providing many more wiring points than are initially needed, but giving great flexibility for future expansion. When this is done, patch panels – or punch down blocks – are often installed for even greater flexibility

3.2.4 10BaseF

This standard, like the 10BaseT standard, is based on a star topology using wiring hubs. The actual standard has been delayed by development work in other areas, and was ratified in September 1993. It consists of three architectures.

These are:

- **10BaseFL**

  The fiber link segment standard that is basically a 2 km upgrade to the existing fiber optic inter repeater link (FOIRL) standard. The original FOIRL as specified in the 802.3 standard was limited to a 1 km fiber link between two repeaters, with a maximum length of 2.5 km if there are 5 segments in the link. Note that this is a link between two repeaters in a network, and cannot have any nodes connected to it
- **10BaseFP**

  A star topology network based on the use of a passive fiber optic star coupler. Up to 33 ports are available per star, and each segment has a maximum length of 500 m. The passive hub is completely immune to external noise and is an excellent choice for noisy industrial environments
- **10BaseFB**

  A fiber backbone link segment in which data is transmitted synchronously. It is designed only for connecting repeaters, and for repeaters to use this standard, they must include a built in transceiver. This reduces the time taken to transfer a frame across the repeater hub. The maximum link length is 2 km, although up to 15 repeaters can be cascaded, giving great flexibility in network design

3.2.5 10Broad36

This architecture, whilst included in the 802.3 standard, is no longer installed as a new system. This is a broadband version of Ethernet, and uses a 75-ohm coaxial cable for transmission. Each transceiver transmits on one frequency and receives on a separate one. The Tx/Rx streams require a 14 MHz bandwidth and an additional 4 MHz is required for collision detection and reporting. The total bandwidth requirement is thus 36 MHz. The cable is limited to 1800 meters because each signal must traverse the cable twice, so the worst-case distance is 3600 m. It is this figure that gives the system its nomenclature.

3.2.6 1Base5

This architecture, whilst included in the 802.3 standard, is no longer installed as a new system. It is hub based and uses UTP as a transmission medium over a 500-meter maximum length. However, signaling is 1 Mbps, and this means special provision must be made if it is to be incorporated in a 10 Mbps network. It has been superseded by 10BaseT.

3.3 Signaling methods

Ethernet signals are encoded using the Manchester encoding scheme. This method allows a clock to be extracted at the receiver end and synchronize the transmission/reception process. The encoding is performed by an exclusive-or between a 20MHz clock signal and the data stream. In the resulting signal, a 0 is represented by a high to low change at the center of the bit cell, whilst a 1 is represented by a low to high change at the center of the bit cell. There may or may not be transitions at the beginning of a cell as well, but these are ignored at the receiver. The transitions in every cell allow the clock to be extracted, and synchronized with the transmitter.

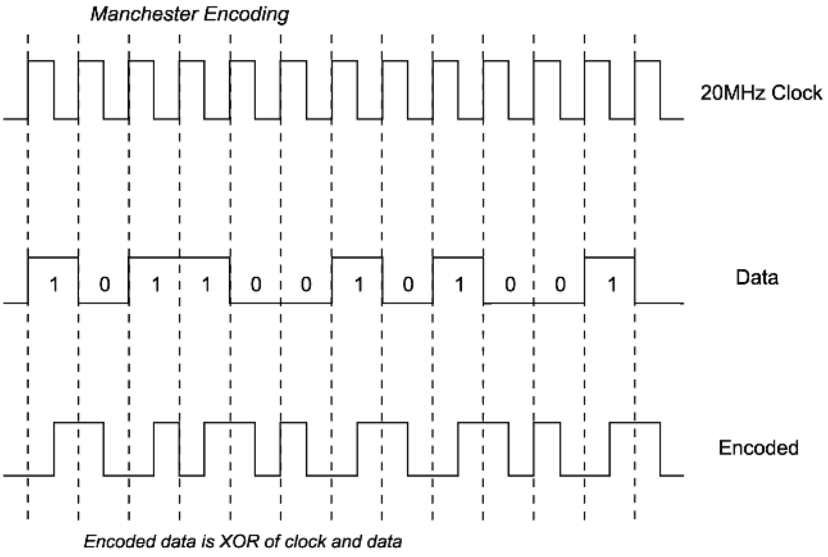


Figure 3.6
Manchester encoding

The voltage swings were from –0.225 to –1.825 volts in the original Ethernet specification. In the 802.3 standard, voltages on coax cables are specified to swing between 0 and –2.05 volts with a rise and fall time of 25 ns at 10 Mbps.

3.4 Medium access control

Essentially, the method used is one of contention. As was described in the first section on this architecture, each node has a connection via a transceiver to the common bus. As a transceiver, it can both transmit and receive at the same time. Each node can be in any one of three states at any time.

These states are:

- Idle, or listen
- Transmit
- Contention

In the idle state, the node merely listens to the bus, monitoring all traffic that passes. If a node then wishes to transmit information, it will defer whilst there is any activity on the bus, since this is the ‘carrier sense’ component of the architecture. At some stage, the bus will become silent, and the node, sensing this, will then commence its transmission. It is now in the transmit mode, and will both transmit and listen at the same time. This is because there is no guarantee that another node at some other point on the bus has not also started transmitting having recognized the absence of traffic.

After a short delay as the two signals propagate towards each other on the cable, there will be a collision of signals. Quite obviously, the two transmissions cannot coexist on the common bus, since there is no mechanism for the mixed analog signals to be ‘unscrambled’. The transceiver quickly detects this collision, since it is monitoring both its input and output and recognizes the difference. The node now goes into the third state of contention. The node will continue to transmit for a short time – the jam signal – to ensure the other transmitting node detects the contention, and then performs a back-off algorithm to determine when it should again attempt to transmit its waiting frames.

3.5 Frame transmission

When a frame is to be transmitted, the medium access control monitors the bus and defers to any passing traffic. After a period of 96 bit times, known as the interframe gap, to allow the passing frame to be received and processed by the destination node, the transmission process commences. Since there is a finite time for this transmission to propagate to the ends of the bus cable, and thus ensure that all nodes recognize that the medium is busy, the transceiver turns on a collision detect circuit whilst the transmission takes place. In fact, once a certain number of bits (576 bits in a 10 Mbps system) have been transmitted, provided that the network cable segment specifications have been complied with, the collision detection circuitry can be disabled. If a collision should take place after this, it will be the responsibility of higher protocols to request retransmission – a far slower process than the hardware collision detection process.

Here is a good reason to comply with cable segment specifications! This initial ‘danger’ period is known as the collision window, and is effectively twice the time interval for the first bit of a transmission to propagate to all parts of the network. The slot time for the network is then defined as the worst-case time delay that a node must wait before it can reliably know that a collision has occurred.

It is defined as:

Slot time = 2 * (transmission path delay) + safety margin

For a 10 Mbps system, the slot time is FIXED as 512 bits or 64 octets.

3.6 Frame reception

The transceiver of each node is constantly monitoring the bus for a transmission signal. As soon as one is recognized, the NIC activates a carrier sense signal to indicate that transmissions cannot be made. The first bits of the MAC frame are a preamble and consist of 56 bits of 1010 etc. On recognizing these, the receiver synchronizes its clock, and converts the Manchester encoded signal back into binary form. The eighth octet is a start of frame delimiter, and this is used to indicate to the receiver that it should strip off the first eight octets and commence determining whether this frame is for its node by reading the destination address. If the address is recognized, the data is loaded into a frame buffer within the NIC.

Further processing then takes place, including the calculation and comparison of the frame CRC, checking with the transmitted CRC. Checking that the frame contains an integral number of octets and is either too short or too long. Provided all is correct, the frame is passed to the LLC layer for further processing.

3.7 Collisions

You should recognize that collisions are a normal part of a CSMA/CD network. The monitoring and detection of collisions is the method by which a node ensures unique access to the shared medium. It is only a problem when there are excessive collisions. This reduces the available bandwidth of the cable and slows the system down while retransmission attempts occur. There are many reasons for excessive collisions and you will investigate these shortly.

The principle of collision cause and detection is shown in the following diagram.

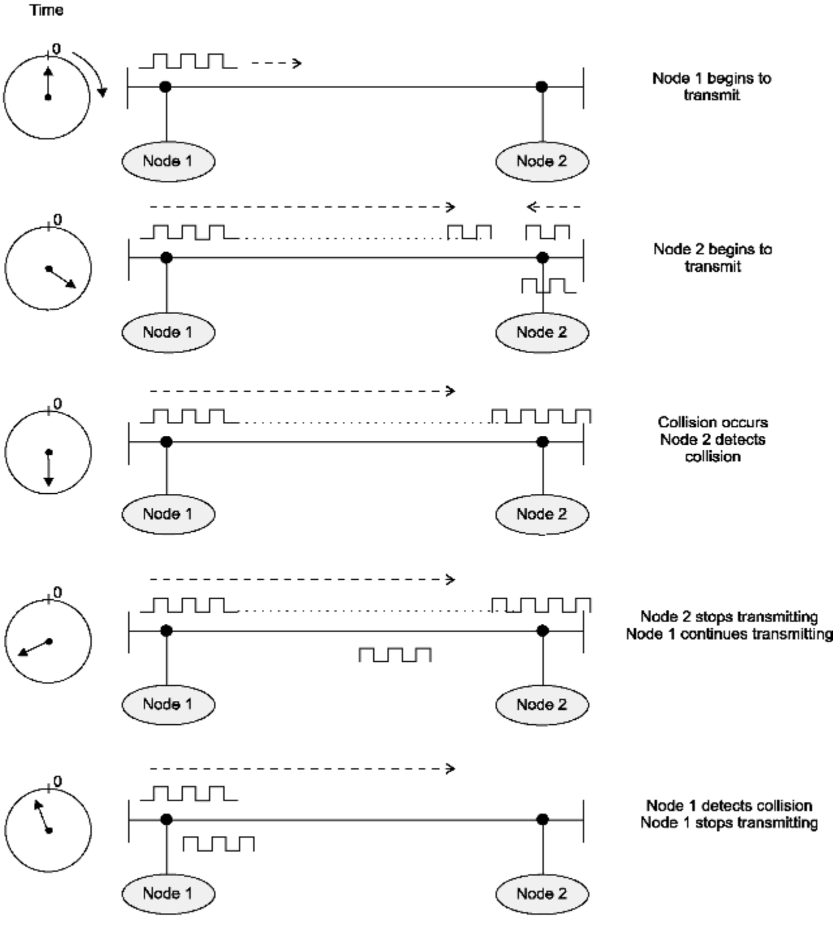

Time
0
Node 1 begins to transmit
Node 1
Node 2
Node 2 begins to transmit
Collision occurs Node 2 detects collision
Node 2 stops transmitting Node 1 continues transmitting
Node 1 detects collision Node 1 stops transmitting

Figure 3.7
CSMA/CD collisions

Assume that both node 1 and node 2 are in listen mode and node 1 has frames queued to transmit. All previous traffic on the medium has ceased i.e. there is no carrier, and the interframe gap from the last transmission has timed out. Node 1 now commences to transmit its preamble signal, which immediately commences to propagate both left and right on the cable. At the left end, the termination resistance absorbs the transmission, but the signal continues to propagate to the right. However, the MAC sub layer in node 2 has also been given a frame to transmit from the LLC sub layer, and since the node 'sees' a free cable, it too commences to transmit its preamble. Again, the signals propagate on to the cable, and some short time later they 'collide'. Almost immediately, node 2's transceiver recognizes that the signals on the cable are corrupted, and the logic incorporated on the NIC asserts a collision detect signal. This causes node 2 to send a jam signal of 32 bits of random data, and then stop transmitting. In fact, the standard allows any data to be sent as long as, by design, it is not the value of the CRC field of the frame. It appears that most nodes will send the next 32 bits of the data frame as a jam, since that is instantly available.

This jam signal continues to propagate along the cable, as a contention signal since it is 'mixed' with the signal still being transmitted from node 1. Eventually, node 1 recognizes the collision, and goes through the same jam process as node 2. You can see from this that the frame from node 1 must be at least twice the end to end propagation delay of the network, or else the collision detection will not work correctly. The jam signal from node 1 will continue to propagate across the network until absorbed at the far end terminator, meaning that the system vulnerable period is three times the end to end propagation delay.

After the jam sequence has been sent, the transmission is halted. The node then schedules a retransmission attempt after a random delay controlled by a process known as the truncated binary exponential back off algorithm. The length of the delay is chosen so that it is a compromise between reducing the probability of another collision and delaying the retransmission for an unacceptable length of time. The delay is always an integer multiple of the slot time. In the first attempt, the node will choose, at random, either one or zero slot times delay. If another collision occurs, the delay will be chosen at random from 0, 1, 2 or 3 slot times, thus reducing the probability that a further collision will occur. This process can continue for up to 10 attempts, with a doubling of the range of slot times available for the node to delay transmission at each attempt. After ten attempts, the node will attempt 6 more retries, but the slot times available for the delay period will remain as they were at the tenth attempt. After 16 attempts, it is likely that there is a problem on the network and the node will cease attempting to retransmit.

3.8 MAC frame format

The basic frame format for an 802.3 network is shown below. There are eight fields in each frame, and they will be described in detail.

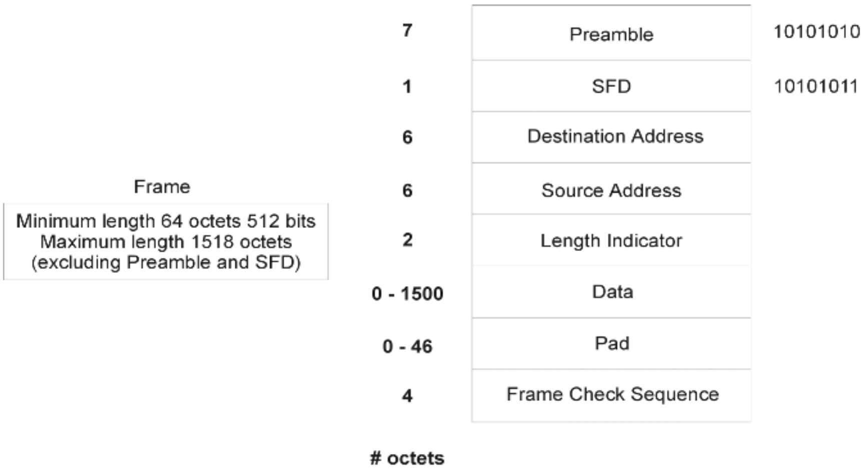


Figure 3.8
MAC frame format

Preamble

This field consists of 7 octets of the data pattern 10101010. The preamble is used by the receiver to synchronize its clock to the transmitter.

Start frame delimiter

This single octet field consists of the data 10101011. It enables the receiver to recognize the commencement of the address fields.

Source and destination address

These are the physical addresses of both the source and destination nodes. The fields can be 2 or 6 octets long, although the six-octet standard is the most common. The six-octet field is split into two three octet blocks. The first three octets describe the block number to which all NICs of this type belong. This number is the license number and all cards made by this company have the same number. The second block refers to the device identifier, and each card will have a unique address under the terms of the license to manufacture. This means there are 248 unique addresses for Ethernet cards.

There are three addressing modes that are available. These are:

- **Broadcast**

  Destination address is set to all 1s or FFFFFFFFFFFF
- **Multicast**

  First bit of the destination address is set to a 1. It provides group restricted communications
- **Individual, or point-to-point**

  First bit of the address set to 0, and the rest set according to the target destination node

Length

A two-octet field that contains the length of the data field. This is necessary since there is no end delimiter in the frame.

Information

The information that has been handed down from the LLC sub layer.

Pad

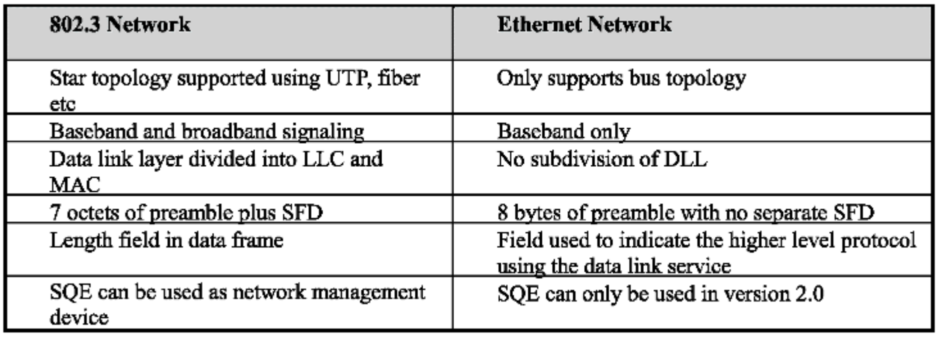
Since there is a minimum length of frame of 64 octets (512 bits or 576 bits if the preamble is included) that must be transmitted to ensure that the collision mechanism works, the pad field will pad out any frame that does not meet this minimum specification. This pad, if incorporated, is normally random data. The CRC is calculated over the data in the pad field. Once the CRC checks OK, the receiving node discards the pad data, which it recognizes by the value in the length field.

FCS

A 32-bit CRC value that is computed in hardware at the transmitter and appended to the frame. It is the same algorithm used in the 802.4 and 802.5 standard.

3.9 Difference between 802.3 and Ethernet

As has already been discussed, there is a difference between an 802.3 network and a Blue Book Ethernet network. These differences are primarily in the frame structure and are tabulated below.



802.3 Network	Ethernet Network
Star topology supported using UTP, fiber etc	Only supports bus topology
Baseband and broadband signaling	Baseband only
Data link layer divided into LLC and MAC	No subdivision of DLL
7 octets of preamble plus SFD	8 bytes of preamble with no separate SFD
Length field in data frame	Field used to indicate the higher level protocol using the data link service
SQE can be used as network management device	SQE can only be used in version 2.0

Table 3.1
Differences between IEEE 802.3 and Blue Book Ethernet (V2)

The significant difference in the frame is the length field in 802.3 is interpreted as the higher protocol field in Ethernet. Since an 802.3 frame cannot be longer than 1500 bytes, the values in the protocol type field of the Ethernet V2 frame commences at 1500. This allows protocol analyzers to recognize one type of frame as opposed to the other.

3.10 Reducing collisions

The main reasons for collision rates on an Ethernet network are:

- The number of packets per second
- The signal propagation delay between transmitting nodes
- The number of stations initiating packets
- The bandwidth utilization

A few suggestions on reducing collisions in an Ethernet network are:

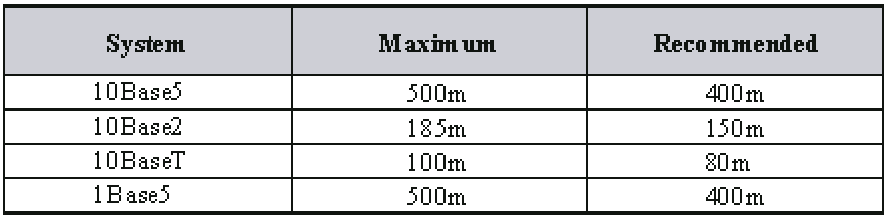
- Keep all cables as short as possible
- Keep all high activity sources and their destinations as close as possible. Possibly isolate these nodes from the main network backbone with bridges/routers to reduce backbone traffic
- Use buffered repeaters rather than bit repeaters
- Check for unnecessary broadcast packets that are aimed at non existent nodes
- Remember that the monitoring equipment to check out network traffic can contribute to the traffic (and the collision rate)

3.11 Ethernet design rules

The following design rules on length of cable segment, node placement and hardware usage should be strictly observed.

3.11.1 Length of the cable segment

It is important to maintain the overall Ethernet requirements as far as length of the cable is concerned. Each segment has a particular maximum length allowable. For example, 10Base2 allows 200 m maximum length. The recommended maximum length is 80% of this figure. Some manufacturers advise that you can disregard this limit with their equipment. This can be a risky strategy and should be carefully considered.



System	Maximum	Recommended
10Base5	500m	400m
10Base2	185m	150m
10BaseT	100m	80m
1Base5	500m	400m

Table 3.2
Length of the cable segment

Cable segments need not be made from a single homogenous length of cable, and may comprise multiple lengths joined by coaxial connectors (two male plugs and a connector barrel). Although Thicknet (10Base5) and Thinnet (10Base2) cables have the same nominal 50-ohm impedance they can only be mixed within the same 10Base2 cable segment to achieve greater segment length.

To achieve maximum performance on 10Base5 cable segments, it is preferable that the total segment be made from one length of cable or from sections off the same drum of cable. If multiple sections of cable from different manufacturers are used, then these should be standard lengths of 23.4 m, 70.2 m or 117 m (± 0.5 m), which are odd multiples of 23.4 m (half wavelength in the cable at 5 MHz). These lengths ensure that reflections from the cable-to-cable impedance discontinuities are unlikely to add in phase. Using these lengths exclusively a mix of cable sections should be able to be made up to the full 500 m segment length.

If the cable is from different manufacturers and you suspect potential mismatch problems, you should check that signal reflections due to impedance mismatches do not exceed 7% of the incident wave.

3.11.2 Maximum transceiver cable length

In 10Base5 systems the maximum length of the transceiver cables is 50 m but it should be noted that this only applies to specified IEEE 802.3 compliant cables. Other AUI cables using ribbon or office grade cables can only be used for short distances (less than 12.5 m) so check the manufacturer specifications for these!

3.11.3 Node placement rules

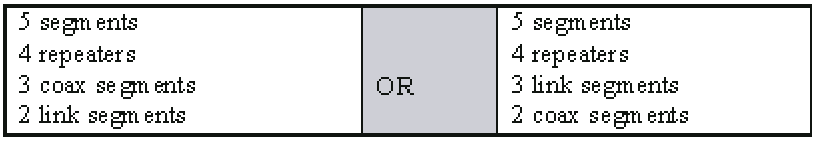
Connection of the transceiver media access units (MAU) to the cable causes signal reflections due to their bridging impedance. Placement of the MAUs must therefore be controlled to ensure that reflections from them do not significantly add in phase.

In 10Base5 systems the MAUs are spaced at 2.5 m multiples, coinciding with the cable markings.

In 10Base2 systems the minimum MAU spacing is 0.5 m.

3.11.4 Maximum transmission path

The maximum transmission path is made of five segments connected by four repeaters. The total number of segments can be made up of a maximum of three coax segments containing station nodes and two link segments, having no intermediate nodes. This is summarized as the 5-4-3-2 rule. These link segments are 10BaseFL fiber links as specified in IEEE 802.3.



5 segments 4 repeaters 3 coax segments 2 link segments	OR	5 segments 4 repeaters 3 link segments 2 coax segments

Table 3.3
5-4-3-2 rule

It is important to verify that the above transmission rules are met by all paths between any two nodes on the network.

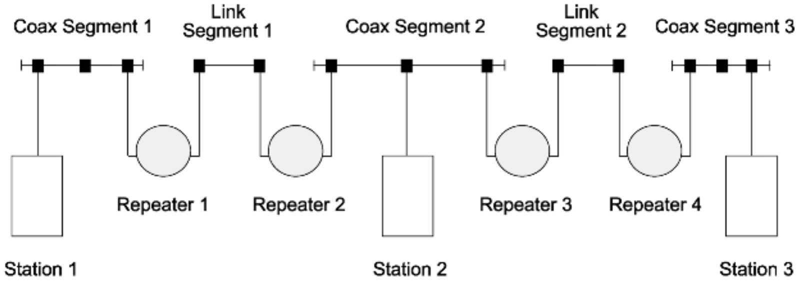


Figure 3.9
Maximum transmission path

Note that the maximum sized network of four repeaters supported by IEEE 802.3 can be susceptible to timing problems. The maximum configuration is limited by propagation delay.

Note that 10Base2 segments should not be used to link 10Base5 segments.

3.11.5 Maximum network size

10Base5 = 2800 m node to node
(5 × 500 m segments + 4 repeater cables + 2 AUI)
10Base2 = 925 m node to node (5 × 185 m segments)
10BaseT = 100 m node to hub

3.11.6 Repeater rules

Repeaters are connected to transceivers that count as one node on the segments.

Special transceivers are used to connect repeaters and these do not implement the signal quality error test (SQE).

Fiber optic repeaters are available giving up to 3000 m links at 10 Mbps. Check the vendor’s specifications for adherence with IEEE 802.3 repeater performance and compliance with the fiber optic inter repeater link (FOIRL) standard.

3.11.7 Cable system grounding

Grounding has safety and noise connotations. IEEE 802.3 states that the shield conductor of each coaxial cable shall make electrical contact with an effective earth reference at one point only.

The single point earth reference for an Ethernet system is usually located at one of the terminators. Most terminators for Ethernet have a screw terminal to which a ground lug can be attached using a braided cable preferably to ensure good earthing.

Ensure that all other splices taps or terminators are jacketed so that no contact can be made with any metal objects. Insulating boots or sleeves should be used on all in-line coaxial connectors to avoid unintended earth contacts.

4

Fast and gigabit Ethernet systems

Objectives

When you have completed study of this chapter you should be able to:

- List the basic methods used to achieve high transmission speeds on UTP cables
- Describe the operation of the 100Base-TX system
- List the different physical media options for 100Base-T systems
- Explain the basic differences between a Class I and Class II repeater
- Explain the packet bursting technique used by gigabit Ethernet
- List the different media options used by gigabit Ethernet

4.1 Achieving higher speed

Although Ethernet with over 200 million installed nodes world-wide is the most popular method of linking computers on a network, its 10 Mbps speed is too slow for very data intensive or real-time applications.

From a philosophical point of view there are several ways to increase speed on a network. The easiest, conceptually, is to increase the bandwidth and allow faster changes of the data signal. This requires a high bandwidth medium and generates a considerable amount of high frequency electrical noise on copper cables, which is difficult to suppress. The second approach is to move away from the serial transmission of data on one circuit to a parallel method of transmitting over multiple circuits at each instant. A third approach is to use data compression techniques to enable more than one bit to be transferred for each electrical transition. A fourth approach used with 1000 Mbps gigabit Ethernet is to operate circuits full-duplex, enabling simultaneous transmission in both directions.

All of the three approaches are used to achieve 100 Mbps fast Ethernet and 1000 Mbps gigabit Ethernet transmission on both fiber optic and copper cables using the current high-speed LAN technologies.

4.1.1 Cabling limitations

Typically most LAN systems use either coaxial cable, shielded (STP) or unshielded twisted pair (UTP) or fiber optic cables. The capacitance of the coaxial cable imposes a serious limit to the distance over which the higher frequencies that can be handled. Consequently 100 Mbps systems do not use coaxial cables.

The unshielded twisted pair is obviously popular because of ease of installation and low cost. This is the basis of the 10Base-T Ethernet standard. The category 3 cable enables us to achieve only 10 Mbps while category 5 cables can attain 100 Mbps data rates, whilst the four pairs in the standard cable enable several parallel data streams to be handled.

As we have seen fiber optic cables have enormous bandwidths and excellent noise immunity so are the obvious choice for high-speed LAN systems.

4.2 100Base-T (100Base-TX, -T4, -FX, -T2)

This is the preferred approach to 100 Mbps transmission, which uses the existing Ethernet MAC layer with various enhanced **physical media dependent** (PMD) layers to improve the speed. These are described in the IEEE 802.3u and 802.3y standards as follow.

IEEE 802.3u defines three different versions based on the physical media:

- 100Base-TX which uses two pairs of category 5 UTP or STP
- 100Base-T4 which uses four pairs of wires of category 3, 4 or 5 UTP
- 100Base-FX which uses multimode or single-mode fiber optic cable

IEEE 802.3y:

- 100Base-T2 which uses two pairs of wires of category 3, 4 or 5 UTP

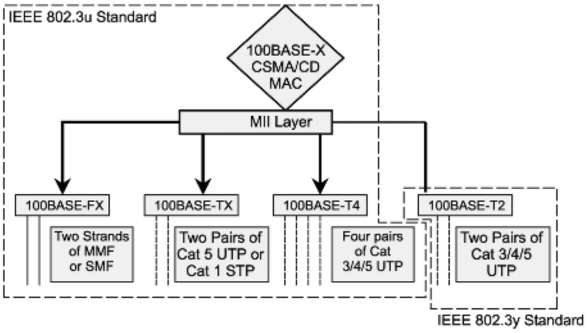

IEEE 802.3u Standard
100BASE-X CSMA/CD MAC
MII Layer
100BASE-FX
100BASE-TX
100BASE-T4
100BASE-T2
Two Strands of MMF or SMF
Two Pairs of Cat 5 UTP or Cat 1 STP
Four pairs of Cat 3/4/5 UTP
Two Pairs of Cat 3/4/5 UTP
IEEE 802.3y Standard

Figure 4.1
Summary of 100Base-T standards

This approach is possible because the original 802.3 specifications defined the MAC layer independently of the various physical PMD layers it supports. As you will recall, the MAC layer defines the format of the Ethernet frame and defines the operation of the CSMA/CD access mechanism. The time dependent parameters are defined in the 802.3 specifications in terms of bit-time intervals and so is speed independent. The 10 Mbps Ethernet interframe gap is actually defined as an absolute time interval of 9.60 microseconds, equivalent to 96 bit times; while the 100 Mbps system reduces this by ten times to 960 nanoseconds.

One of the limitations of the 100Base-T systems is the size of the collision domain, which is 250 m. This is the maximum sized network in which collisions can be detected; being one tenth of the size of the maximum 10 Mbps network. This limits the distance between our workstation and hub to 100 m, the same as for 10Base-T, but usually only one hub is allowed in a collision domain. This means that networks larger than 200 m must be logically connected together by store and forward type devices such as bridges, routers or switches. However, this is not a bad thing, since it segregates the traffic within each collision domain, reducing the number of collisions on the network. The use of bridges and routers for traffic segregation, in this manner, is often done on industrial CSMA/CD networks.

The dominant 100Base-T system is 100Base-TX, which accounts for about 95% of all fast Ethernet shipments. The 100Base-T4 systems were developed to use four pairs of category 3 cable; however few users had the spare pairs available and T4 systems are not capable of full-duplex operation, so this system has not been widely used. The 100Base-T2 system has not been marketed at this stage, however its underlying technology using digital signal processing (DSP) techniques is used for the 1000Base-T systems on two category 5 pairs. With category 3 cable diminishing in importance, it is not expected that the 100Base-T2 systems will become significant.

4.2.1 IEEE 802.3u 100Base-T standards arrangement

The IEEE 802.3u standard fits into the OSI model as shown in Figure 4.2. You will note that the unchanged IEEE 802.3 MAC layer sits beneath the LLC as the lower half of the data link layer of the OSI model.

Its physical layer is divided into the following two sub layers and their associated interfaces:

- PHY – physical medium independent layer
- MII – medium independent interface
- PMD – physical medium dependent layer
- MDI – medium dependent interface

A convergence sub layer is added for the 100Base-TX and -FX systems, which use the ANSI X3T9.5 PMD layer which was developed for the reliable transmission of 100 Mbps over the twisted pair version of FDDI. The FDDI PMD layer operates as a continuous full-duplex 125 Mbps transmission system, so a convergence layer is needed to translate this into the 100 Mbps half-duplex data bursts expected by the IEEE 802.3 MAC layer.

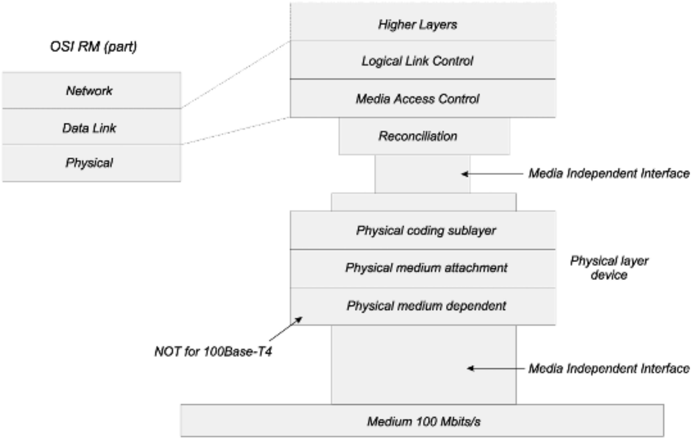


Figure 4.2
100Base-T standards architecture

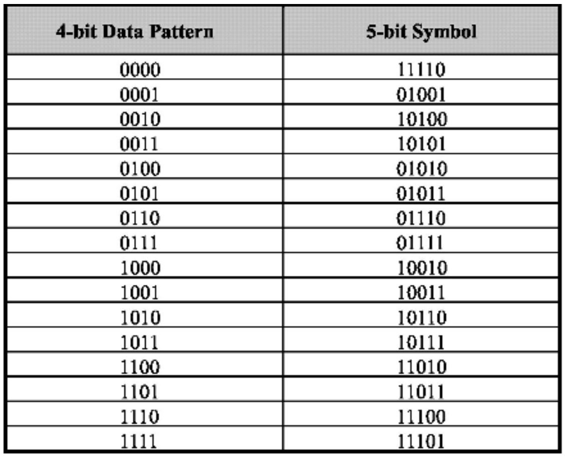
4.2.2 Physical medium independent (PHY) sub layer

The PHY layer specifies the 4B/5B coding of the data, data scrambling and the non return to zero – inverted (NRZI) data coding together with the clocking, data and clock extraction processes.

The 4B/5B technique selectively codes each group of four bits into a five-bit cell symbol. For example, the binary pattern 0110 is coded into the five-bit pattern 01110. In turn, this symbol is encoded using non return to zero – inverted (NRZ-I) where a ‘1’ is represented by a transition at the beginning of the cell, and a ‘0’ by no transition at the beginning. This allows the carriage of 100 Mbps data by transmitting at 125 MHz, and gives a consequent reduction in component cost of some 80%.

With a five-bit pattern, there are 32 possible combinations. Obviously, there are only 16 of these that need to be used for the four bits of data, and of these, each is chosen so that there are no more than three consecutive zeros in each symbol. This ensures there will be sufficient signal transitions to maintain clock synchronization. The remaining 16 symbols are used for control purposes.

This selective coding is shown in Table 4.1.



4-bit Data Pattern	5-bit Symbol
0000	11110
0001	01001
0010	10100
0011	10101
0100	01010
0101	01011
0110	01110
0111	01111
1000	10010
1001	10011
1010	10110
1011	10111
1100	11010
1101	11011
1110	11100
1111	11101

Table 4.1
4B/5B data coding

This coding scheme is not self clocking so each of the receivers maintains a separate data receive clock which is kept in synchronization with the transmitting node, by the clock transitions in the data stream. Hence the coding cannot allow more than three consecutive zeros in any symbol.

4.2.3 100Base-TX and -FX physical media dependent (PMD) sub layer

This uses the ANSI TP-X3T9.5 PMD layer and operates on two pairs of category 5 twisted pair. It uses stream cipher scrambling for data security and MLT-3 bit encoding. The multilevel threshold-3 (MLT-3) bit coding uses three voltage levels: +1 volts, 0 volts and –1 volts. The level remains the same for consecutive sequences of the same bit, i.e. continuous ‘1’ s. When a bit changes, the voltage level changes to the next state in the circular sequence 0 V, +1 V, 0 V, –1 V, 0 V etc. This results in a coded signal, which resembles a smooth sine wave of much lower frequency than the incoming bit stream. Hence for a 31.25 MHz baseband signal this allows for a 125 Mbps signaling bit stream providing a 100 Mbps throughput (4 B/5B encoder). The MAC outputs a NRZ code. This code is then passed to a scrambler, which ensures that there are no invalid groups in its NRZI output. The NRZI converted data is passed to the three level code block and the output is then sent to the transceiver. The code words are selectively chosen so the mean line signal line zero, in other words the line is DC balanced.

The three level code results in a lower frequency signal. Noise tolerance is not as high as 10Base-T because of the multilevel coding system; hence data grade (category 5) cable is required.

Two pair wire, RJ45 connectors and a hub are requirements for 100Base-TX. These factors and a maximum distance of 100 m between the nodes and hubs make for a very similar architecture to 10Base-T.

4.2.4 100Base-T4 physical media dependent (PMD) sub layer

The 100Base-T4 systems use four pairs of Category 3 UTP. It uses data encoded in an eight binary six ternary (8B/6T) coding scheme similar to the MLT-3 code. The data is encoded using three voltage levels per bit time of +V, 0 volts and –V, these are usually written as simply +, 0 and –. This coding scheme allows the eight bits of binary data to be coded into six ternary symbols, and reduces the required bandwidth to 25 MHz. The 256 code words are chosen so the line has a mean line signal of zero. This helps the receiver to discriminate the positive and negative signals relative to the average zero level. The coding utilizes only those code words, which have a combined weight of 0 or +1, as well as at least two signal transitions for maintaining clock synchronization. For example, the code word for the data byte 20H is –++–00, which has a combined weight of 0 while 40H is –00+0+, which has a combined weight of +1.

If a continuous string of code words of weight +1 is sent then the mean signal will move away from zero, known as DC wander. This causes the receiver to misinterpret the data since it is assuming the average voltage it is seeing, which is now tending to '+1', is its zero reference. To avoid this situation, a string of code words of weight +1 is always sent by inverting alternate code words before transmission. Consider a string of consecutive data bytes 40H, the codeword is –00+0+ which has weight +1. This is sent as the sequence –00+0+, +00–0–, –00+0+, +00–0– etc, which results in a mean signal level of zero. The receiver consequently reinverts every alternate codeword prior to decoding.

These signals are transmitted in half-duplex over three parallel pairs of category 3, 4 or 5 UTP cable, while a fourth pair is used for reception of collision detection signals. This is shown in Figure 4.3.

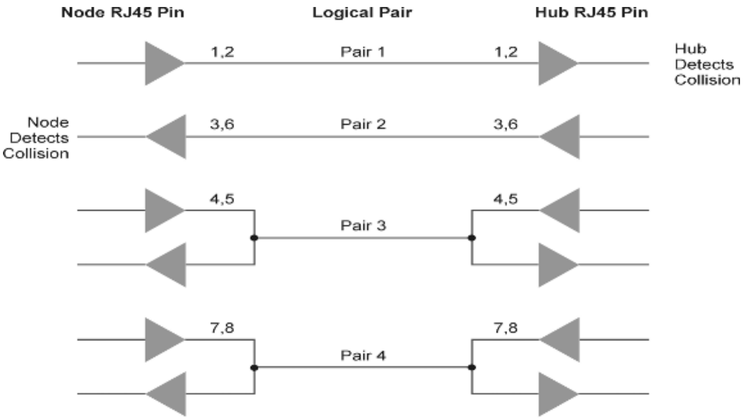


Figure 4.3
100Base-T4 wiring

100Base-TX and 100Base-T4 are designed to be interoperable at the transceivers using a media independent interface and compatible (class 1) repeaters at the hub. Maximum node to hub distances of 100 m and 250 m maximum network diameter are supported. Maximum hub-to-hub distance of 10 m.

4.2.5 100Base-T2

The 100Base-T2 system was published by the IEEE in 1996 as the IEEE 802.3y standard but has not been marketed at this stage. It was designed to address the shortcomings of 100Base-T4, making full-duplex 100 Mbps accessible to installations with only two category 3 cable pairs available. The standard was completed two years after 100Base-TX, at which stage the -TX had such market dominance that the -T2 products were not commercially produced. However it is mentioned here for reference and because its underlying technology using digital signal processing (DSP) techniques and five-level coding (PAM-5) is used for the 1000Base-T systems on two category 5 pairs. These are discussed in detail under 1000Base-T systems in Section 4.5.

The features of 100Base-T2 are:

- Uses two pairs of category 3, 4 or 5 UTP
- Uses both pairs for simultaneously transmitting and receiving – commonly known as dual-duplex transmission. This is achieved by using digital signal processing (DSP) techniques
- Uses a five-level coding scheme with five phase angles called pulse amplitude modulation (PAM 5) to transmit two bits per symbol

4.2.6 100Base-T hubs

The IEEE 802.3u specification defines two classes of 100Base-T hubs, which are normally called repeaters:

- Class I, or translational repeaters which can support both TX/FX and T4 systems
- Class II, or transparent repeaters which support only one signaling system

The class I repeaters have greater delays (0.7 μs maximum) in supporting both signaling standards and so only permit one hub in each collision domain. The class I repeater fully decodes each incoming TX or T4 packet into its digital form at the media independent interface (MII) and then sends the packet out as an analog signal from each of the other ports in the hub. Repeaters are available with all T4 ports, all TX ports or combinations of TX and T4 ports, called translational repeaters. Their layout is shown in Figure 4.4.

The class II repeaters operate like a 10Base-T repeater connecting the ports (all of the same type) at the analog level. These then have lower inter-repeater delays (0.46 μs maximum) and so two repeaters are permitted in the same collision domain, but only 5 m apart. Alternatively, in an all fiber network, the total length of all the fiber segments is 228 meters. This allows two 100 m segments to the nodes with 28 m between the repeaters or any other combination. Most fast Ethernet repeaters available today are class II.

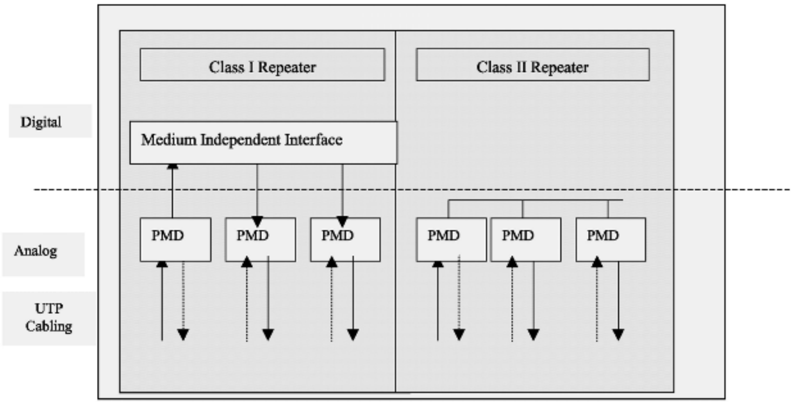


Figure 4.4
Class I and class II fast Ethernet repeaters

4.2.7 100Base-T adapters

Adapter cards are readily available as standard 100Mbps and as 10/100Mbps. These latter cards are interoperable at the hub on both speed systems.

4.3 Fast Ethernet design considerations

4.3.1 UTP cabling distances 100Base-TX/T4

Maximum distance between UTP hub and desktop NIC is 100 meters, made up as follows:

- 5 meters from hub to patch panel
- 90 meters horizontal cabling from patch panel to office punch-down block
- 5 meters from punch-down block to desktop NIC

4.3.2 Fiber optic cable distances 100Base-FX

The following maximum cable distances are in accordance with the 100Base-T bit budget (see Section 4.4.3).

Node to hub: maximum distance of multimode cable (62.5/125) is 160 meters (for connections using a single Class II repeater).

Node to switch: maximum multimode cable distance is 210 meters.

Switch-to-switch: maximum distance of multimode cable for a backbone connection between two 100Base-FX switch ports is 412 meters.

Switch-to-switch, full-duplex: maximum distance of multimode cable for a full-duplex connection between two 100Base-FX switch ports is 2000 meters.

Note The IEEE have not included the use of single mode fiber in the 802.3u standard. However numerous vendors have products available enabling switch-to-switch distances of up to ten to twenty kilometers using single mode fiber.

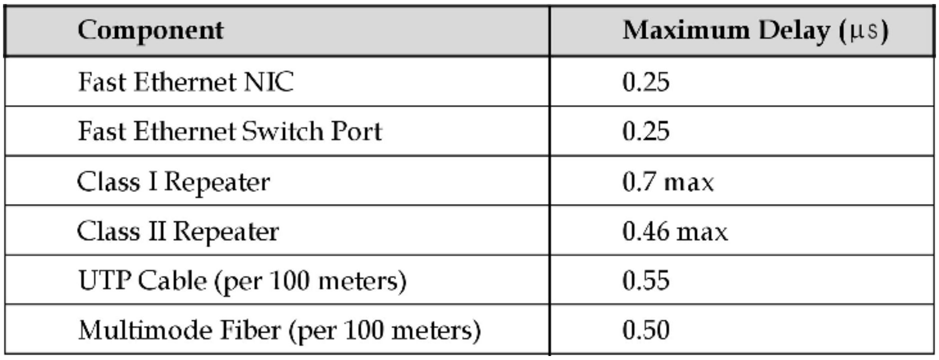
4.3.3 100Base-T repeater rules

The cable distance and the number of repeaters, which can be used in a 100Base-T collision domain, depends on the delay in the cable and the time delay in the repeaters and NIC delays. The maximum round-trip delay for 100Base-T systems is the time to transmit 64 bytes or 512 bits and equals 5.12 µs. A frame has to go from the transmitter to the most remote node then back to the transmitter for collision detection within this round trip time. Therefore the one-way time delay will be half this.

The maximum sized collision domain can then be determined by the following calculation:

Repeater delays + Cable delays + NIC delays + Safety factor (5 bits minimum)< 2.56 µs

The following Table 4.2 gives typical maximum one-way delays for various components. Repeater and NIC delays for specific components can be obtained from the manufacturer.



Component	Maximum Delay (µs)
Fast Ethernet NIC	0.25
Fast Ethernet Switch Port	0.25
Class I Repeater	0.7 max
Class II Repeater	0.46 max
UTP Cable (per 100 meters)	0.55
Multimode Fiber (per 100 meters)	0.50

Table 4.2
Maximum one-way fast Ethernet component delays

Notes

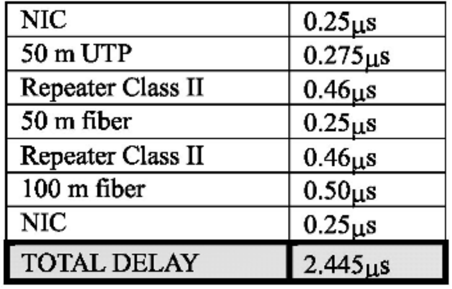
If the desired distance is too great it is possible to create a new collision domain by using a switch instead of a repeater.

Most 100Base-T repeaters are stackable, which means multiple units can be placed on top of one another and connected together by means of a fast backplane bus. Such connections do not count as a repeater hop and make the ensemble function as a single repeater.

4.3.4 Sample calculation

Can two fast Ethernet nodes be connected together using two class II repeaters connected by 50 m fiber? One node is connected to the first repeater with 50 m UTP while the other has a 100 m fiber connection.

Calculation: Using the time delays in Table 4.3



NIC	0.25µs
50 m UTP	0.275µs
Repeater Class II	0.46µs
50 m fiber	0.25µs
Repeater Class II	0.46µs
100 m fiber	0.50µs
NIC	0.25µs
TOTAL DELAY	2.445µs

Table 4.3
Time delays

The total one-way delay of 2.445 μs is within the required interval (2.56 μs) and allows at least 5 bits safety factor, so this connection is permissible.

4.4 Gigabit Ethernet 1000Base-T

4.4.1 Gigabit Ethernet summary

Gigabit Ethernet uses the same 802.3 frame format as 10 Mbps and 100 Mbps Ethernet systems. This operates at ten times the clock speed of Fast Ethernet at 1Gbps. By retaining the same frame format as the earlier versions of Ethernet, backward compatibility is assured with earlier versions, increasing its attractiveness by offering a high bandwidth connectivity system to the Ethernet family of devices.

Gigabit Ethernet is defined by the IEEE 802.3z standard. This defines the gigabit Ethernet media access control (MAC) layer functionality as well as three different physical layers: 1000Base-LX and 1000Base-SX using fiber and 1000Base-CX using copper. These physical layers were originally developed by IBM for the ANSI Fiber channel systems and used 8B/10B encoding to reduce the bandwidth required to send high-speed signals. The IEEE merged the fiber channel to the Ethernet MAC using a gigabit media independent interface (GMII), which defines an electrical interface, enabling existing fiber channel PHY chips to be used and enabling future physical layers to be easily added.

1000Base-T is being developed to provide service over four pairs of category 5 or better copper cable. As discussed earlier this uses the same technology as 100Base-T2. This development is defined by the IEEE 802.3ab standard.

These gigabit Ethernet versions are summarized in Figure 4.5.

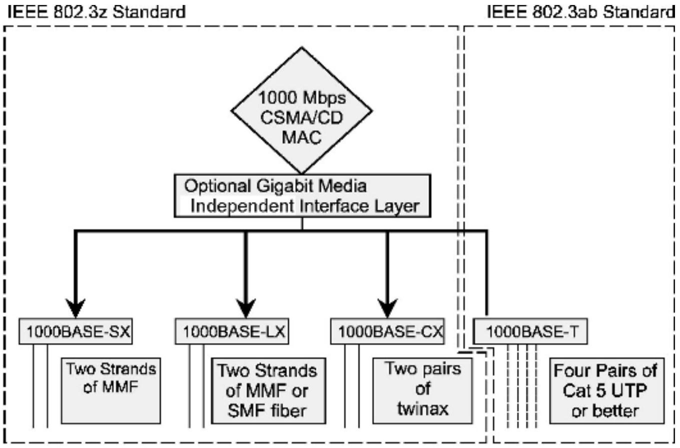


Figure 4.5
Gigabit Ethernet versions

4.4.2 Gigabit Ethernet MAC layer

Gigabit Ethernet retains the standard 802.3 frame format, however the CSMA/CD algorithm has had to undergo a small change to enable it to function effectively at 1 Gbps. The slot time of 64 bytes used with both 10 Mbps and 100 Mbps systems has been increased to 512 bytes. Without this increased slot time the network would have been impractically small at one tenth of the size of fast Ethernet – only 20 meters!

The slot time defines the time during which the transmitting node retains control of the medium, and in particular is responsible for collision detection. With gigabit Ethernet it was necessary to increase this time by a factor of eight to 4.096 μs to compensate for the tenfold speed increase. This then gives a collision domain of about 200 m.

If the transmitted frame is less than 512 bytes the transmitter continues transmitting to fill the 512-byte window. A carrier extension symbol is used to mark frames, which are shorter than 512 bytes, and to fill the remainder of the frame. This is shown in Figure 4.6.

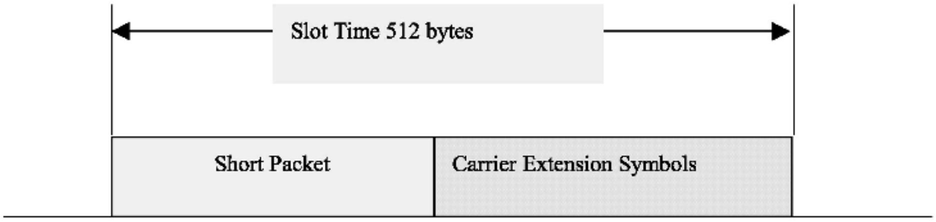


Figure 4.6
Carrier extension

While this is a simple technique to overcome the network size problem, it could cause problems with very low utilization if we send a lot of short frames, typical of some industrial control systems. For example, a 64-byte frame would have 448 carrier extension symbols attached and result in a utilization of less than 10%. This is unavoidable, but its effect can be minimized if we are sending a lot of small frames by a technique called packet bursting. Once the first frame in a burst has successfully passed through the 512-byte collision window, using carrier extension if necessary, transmission continues with additional frames being added to the burst until the burst limit of 1500 bytes is reached. This process averages the time wasted sending carrier extension symbols over a number of frames. The size of the burst varies depending on how many frames are being sent and their size. Frames are added to the burst in real-time with carrier extension symbols filling the inter packet gap. The total number of bytes sent in the burst is totaled after each frame and transmission continues until at least 1500 bytes have been transmitted. This is shown in Figure 4.7.

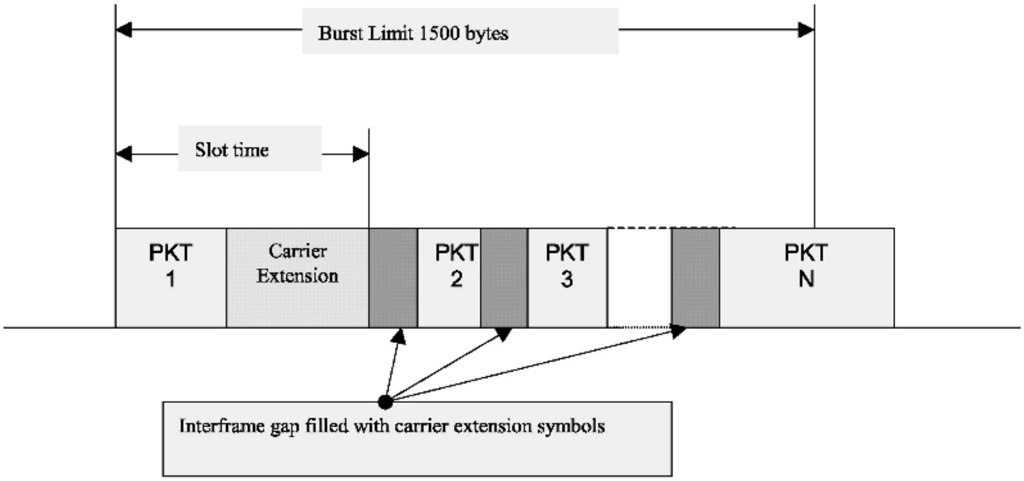


Figure 4.7
Packet bursting

4.4.3 Physical medium independent (PHY) sub layer

The 802.3z Gigabit Ethernet standard used the three PHY sub layers from the ANSI X3T11 fiber channel standard for the 1000Base-SX and 1000Base-LX versions using fiber optic cable and 1000Base-CX using shielded 150-ohm twinax copper cable.

The Fiber Channel PMD sub-layer ran at 1 Gbaud and specifies the 8B/10B coding of the data, data scrambling and the non return to zero – inverted (NRZI) data coding together with the clocking, data and clock extraction processes. This translated to a data rate of 800 Mbps. The IEEE then had to increase the speed of the fiber channel PHY layer to 1250 Mbaud to obtain the required throughput of 1 Gbps.

The 8B/10B technique selectively codes each group of eight bits into a ten-bit symbol. Each symbol is chosen so that there are at least two transitions from ‘1’ to ‘0’ in each symbol. This ensures there will be sufficient signal transitions to allow the decoding device to maintain clock synchronization from the incoming data stream. The coding scheme allows unique symbols to be defined for control purposes, such as denoting the start and end of packets and frames as well as instructions to devices.

The coding also balances the number of ‘1’s and ‘0’s in each symbol, called DC balancing. This is done so that the voltage swings in the data stream would always average to zero, and not develop any residual DC charge, which could result in any AC-coupled devices distorting the signal. This phenomenon is called ‘baseline wander’.

4.4.4 1000Base-SX for horizontal fiber

This gigabit Ethernet version was developed for the short backbone connections of the horizontal network wiring. The SX systems operate full-duplex with multimode fiber only, using the cheaper 850 nm wavelength laser diodes. The maximum distance supported varies between 200 and 550 meters depending on the bandwidth and attenuation of the fiber optic cable used. The standard 1000Base-SX NICs available today are full-duplex and incorporate SC fiber connectors.

4.4.5 1000Base-LX for vertical backbone cabling

This version was developed for use in the longer backbone connections of the vertical network wiring. The LX systems can use single mode or multimode fiber with the more expensive 1300 nm laser diodes. The maximum distances recommended by the IEEE for these systems operating in full-duplex is 5 km for single mode cable and 550 meters for multimode fiber cable. Many 1000Base-LX vendors guarantee their products over much greater distances, typically 10 km. Fiber extenders are available to give service over as much as 80 km.The standard 1000Base-LX NICs available today are full-duplex and incorporate SC fiber connectors.

4.4.6 1000Base-CX for copper cabling

This version of gigabit Ethernet was developed for the short interconnection of switches, hubs or routers within a wiring closet. It is designed for 150-ohm twinax cable similar to that used for IBM token ring systems. The IEEE specified two types of connectors: the high-speed serial data connector (HSSDC) known as the fiber channel style 2 connector and also the 9-pin D-subminiature connector from the IBM token ring systems. The maximum cable length is 25 meters for both full- and half-duplex systems.

These systems are not currently available in the marketplace for connecting different switches. The preferred connection arrangements are to connect chassis-based products via the common back plane and stackable hubs via a regular fiber port.

4.4.7 1000BaseT for category 5 UTP

This version of the gigabit Ethernet is developed under the IEEE 802.3ab standard for transmission over four pairs of category 5 or better cable. This is achieved by simultaneously sending and receiving over each of the four pairs. Compare this to the existing 100Base-TX system which has individual pairs for transmitting and receiving. This is shown in Figure 4.8.

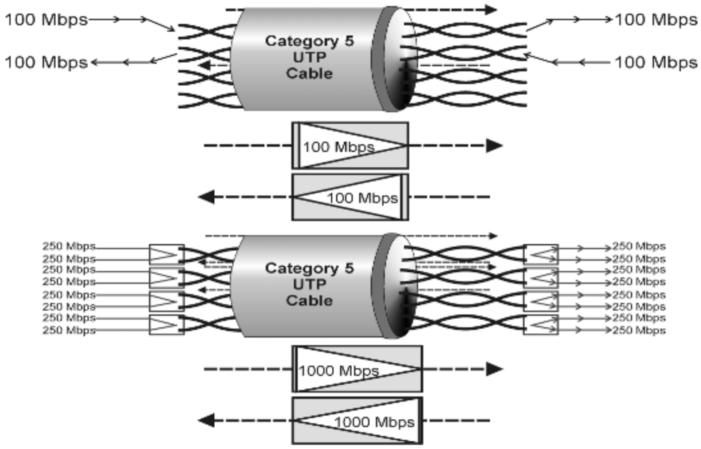


Figure 4.8
Comparison of 100Base-TX and 1000Base-T

This system uses the same data encoding scheme developed for 100Base-T2 which is PAM5. This utilizes five voltage levels so has less noise immunity, however the digital signal processors (DSP) associated with each pair overcomes any problems in this area. The system achieves its tenfold speed improvement over 100BaseT2 by transmitting on twice as many pairs (4) and operating at five times the clock frequency (125 MHz).

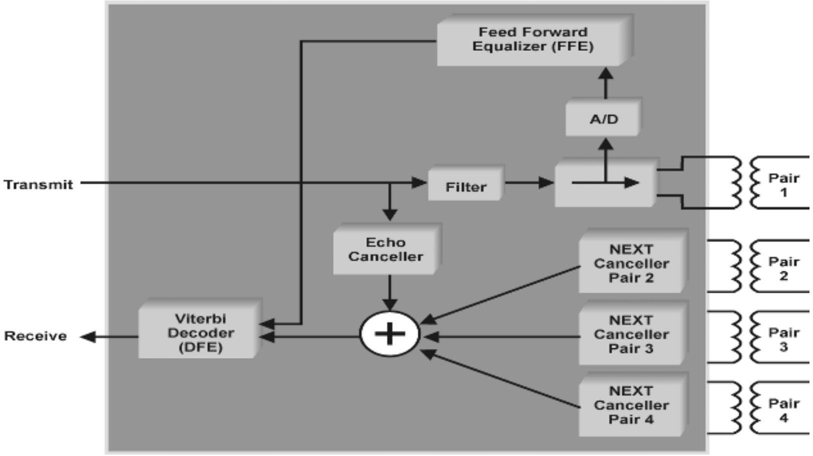


Figure 4.9
1000Base-T receiver uses DSP technology

4.4.8 Gigabit Ethernet full-duplex repeaters

Gigabit Ethernet nodes are connected to full-duplex repeaters also known as non-buffered switches or buffered distributors. As shown in Figure 4.14 these devices have a basic

MAC function in each port, which enables them to verify that a complete frame is received and compute its frame check sequence (FCS) to verify the frame validity. Then the frame is buffered in the internal memory of the port before being forwarded to the other ports of the repeater. It is therefore combining the functions of a repeater with some features of a switch.

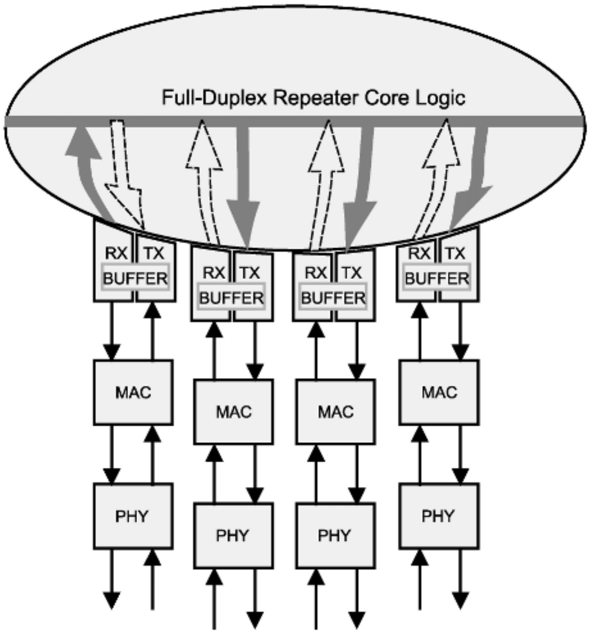


Figure 4.10
Gigabit Ethernet full-duplex repeaters

All ports on the repeater operate at the same speed of 1 Gbps, and operate in full-duplex so it can simultaneously send and receive from any port. The repeater uses 802.3x flow control to ensure the small internal buffers associated with each port do not overflow. When the buffers are filled to a critical level, the repeater tells the transmitting node to stop sending until the buffers have been sufficiently emptied. The repeater does not analyze the packet address fields to determine where to send the packet, like a switch does, but simply sends out all valid packets to all the other ports on the repeater.

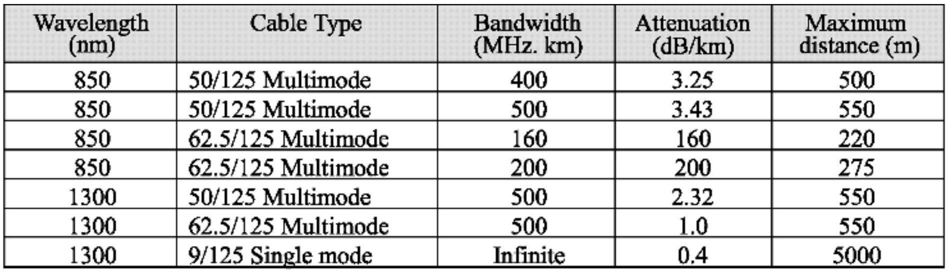
The IEEE does allow for half-duplex gigabit repeaters – however none exist at this time.

4.5 Gigabit Ethernet design considerations

4.5.1 Fiber optic cable distances

The maximum cable distances which can be used between the node and a full-duplex 1000Base-SX and -LX repeater depend mainly on the chosen wavelength, the type of cable, and its bandwidth. The maximum transmission distances on multimode cable are limited by the differential mode delay (DMD). The very narrow beam of laser light injected into the multimode fiber results in a relatively small number of rays going through the fiber core. These rays each have different propagation times because they are going through differing lengths of glass by zigzagging through the core to a greater or lesser extent. These pulses of light can cause jitter and interference at the receiver. This is overcome by using a conditioned launch of the laser into the multimode fiber. This spreads the laser light evenly over the core of the multimode fiber so the laser source looks more like a light emitting diode (LED) source. This spreads the light in a large number of rays across the fiber resulting in smoother spreading of the pulses, so less interference. This conditioned launch is done in the 1000Base-SX transceivers.

The following table gives the maximum distances for full-duplex 1000Base-X repeaters.



Wavelength (nm)	Cable Type	Bandwidth (MHz. km)	Attenuation (dB/km)	Maximum distance (m)
850	50/125 Multimode	400	3.25	500
850	50/125 Multimode	500	3.43	550
850	62.5/125 Multimode	160	160	220
850	62.5/125 Multimode	200	200	275
1300	50/125 Multimode	500	2.32	550
1300	62.5/125 Multimode	500	1.0	550
1300	9/125 Single mode	Infinite	0.4	5000

Table 4.4
Maximum fiber distances for 1000Base-X (full-duplex)

4.5.2 Gigabit repeater rules

The cable distance and the number of repeaters, which can be used in a half-duplex 1000Base-T collision domain, depend on the delay in the cable and the time delay in the repeaters and NIC delays. The maximum round-trip delay for 1000Base-T systems is the time to transmit 512 bytes or 4096 bits and equals 4.096 μs. A frame has to go from the transmitter to the most remote node then back to the transmitter for collision detection within this round trip time. Therefore the one-way time delay will be half this.

The maximum sized collision domain can then be determined by the following calculation:

Repeater delays + Cable delays + NIC delays + Safety factor (5 bits minimum) < 2.048 μs

The following Table 4.5 gives typical maximum one-way delays for various components. Repeater and NIC delays for your specific components can be obtained from the manufacturer.

System	Maximum collision diameter point-to-point Half-duplex	Maximum collision diameter One repeater segment
1000Base-CX	25 m	50m
1000Base-T	100 m	200m
1000Base-SX or LX	316 m	220 m

Table 4.5
Maximum one-way gigabit Ethernet component delays

These calculations give the maximum collision diameter for IEEE 802.3z half-duplex Gigabit Ethernet systems. The maximum gigabit Ethernet network diameters specified by the IEEE are shown in Table 4.6.

System	Maximum collision diameter point-to-point Half-duplex	Maximum collision diameter One repeater segment
1000Base-CX	25 m	50 m
1000Base-T	100 m	200 m
1000Base-SX or LX	316 m	220 m

Table 4.6
Maximum half-duplex gigabit Ethernet network diameters

Note half-duplex gigabit Ethernet repeaters are not available for sale. Use full duplex repeaters with the point-to-point cable distances between node and repeater or node and switch.

5

Introduction to TCP/IP

Objectives

When you have completed study of this chapter you should be able to:

- Describe the origins of TCP/IP
- Compare the OSI and DARPA (DOD) models
- Describe the overall structure of the TCP/IP suite of protocols

5.1 The origins of TCP/IP

In the early 1960s The US **Department Of Defense** (DOD) indicated the need for a wide-area communication system, covering the United States and allowing the interconnection of heterogeneous hardware and software systems.

In 1967 the Stanford Research Institute was contracted to develop the suite of protocols for this network, initially to be known as ARPANet. Other participants in the project included the University of Berkeley (California) and the private company BBN (Bolt, Barenek and Newman). Development work commenced in 1970 and by 1972 approximately 40 sites were connected via TCP/IP. In 1973 the first international connection was made and in 1974 TCP/IP was released to the public.

Initially the network was used to interconnect governments; military and educational sites together. Slowly, as time progressed, commercial companies were allowed access and by 1990 the backbone of the Internet, as it was now known, was being extended into one country after the other.

One of the major reasons why TCP/IP has become the *de facto* standard world-wide for industrial and telecommunications applications is the fact that the Internet was designed around it in the first place and that, without it, no Internet access is possible.

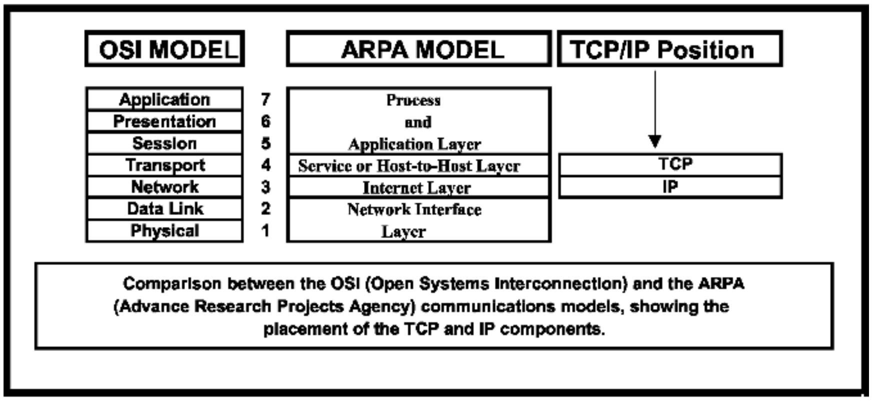
5.2 The ARPA model vs the OSI model

Whereas the OSI model was developed in Europe by the **International Standards Organization** (ISO), the ARPA model (also known as the DoD or Department of Defense model) was developed in the USA by the Advanced Projects Research Agency. Although they were developed by different bodies and at different points in time, both serve as models for a communications infrastructure and hence provide ‘abstractions’ of the same reality. The remarkable degree of similarity is therefore not surprising.

Whereas the OSI model has 7 layers, the ARPA model has 4 layers. The OSI layers map onto the ARPA model as follows:

- The OSI session, presentation and applications layers are contained in the ARPA process and application layer (nowadays referred to by the Internet community as the application level)
- The OSI transport layer maps onto the ARPA host-to-host layer (nowadays referred to by the Internet community as the host level)
- The OSI network layer maps onto the ARPA Internet layer (nowadays referred to by the Internet community as the network level)
- The OSI physical and data link layers map onto the ARPA network interface layer

The relationship between the two models is depicted in Figure 5.1.



Comparison between the OSI (Open Systems Interconnection) and the ARPA (Advance Research Projects Agency) communications models, showing the placement of the TCP and IP components.

Figure 5.1
OSI vs ARPA models

5.3 The TCP/IP protocol suite vs the ARPA model

TCP/IP, or rather – the TCP/IP protocol suite – is not limited to the TCP and IP protocols, but consist of a multitude of interrelated protocols that occupy the upper three layers of the ARPA model. TCP/IP does NOT include the bottom network access layer, but depends on it for access to the medium.

The network interface layer

The network interface layer is responsible for transporting data (frames) between hosts on the same physical network. It is implemented in the network interface card or NIC, using both hardware and ‘firmware’ (i.e. software resident in read only memory).

The NIC employs the appropriate medium access control methodology, such as CSMA/CA, CMSA/CD, token passing or polling, and is responsible for placing the data received from the upper layers within a frame before transmitting it. The frame format is dependent on the system being used, for example Ethernet or frame relay, and holds the hardware address of the source and destination hosts as well as a checksum for data integrity.

RFCs that apply to the network interface layer include:

- Asynchronous transfer mode (ATM), described in RFC 1438
- Switched multimegabit data service (SMDS), described in RFC 1209
- Ethernet, described in RFC 894,
- ARCNET, described in RFC 1201
- Serial line internet protocol (SLIP), described in RFC 1055
- Frame relay, described in RFC 1490
- Fiber distributed data interface (FDDI), described in RFC 1103

(Note: Any Internet-related specification is originally submitted as a request for comments or RFC. As time progresses an RFC may become a standard, or a recommended practice, and so on. Regardless of the status of an RFC, it can be obtained from various sources on the Internet such as http://www.rfc-editor.org.

The Internet layer

This layer is primarily responsible for the routing of packets from one host to another. The emphasis is on ‘packets’ as opposed to frames, since at this level the data has not yet been placed in a frame for transmission. Each packet contains the address information needed for its routing through the Internet work to the receiving host.

The dominant protocol at this level is the IP (as in TCP/IP), namely the Internet protocol.

There are, however, several other additional protocols required at this level. These protocols include:

- **Address resolution protocol** (ARP), RFC 826. This is a protocol used for the translation of an IP address to a hardware (MAC) address, such as required by Ethernet.
- **Reverse address resolution protocol** (RARP), RFC 903. This is the complement of ARP and translates a hardware address to an IP address.
- **Internet control message protocol** (ICMP), RFC 792. This is a protocol used for sending control or error messages between routers or hosts. One of the best-known applications here is the ping or echo request that is used to test a communications link.

The host-to-host layer

This layer is primarily responsible for data integrity between the sender host and receiver host regardless of the path or distance used to convey the message. Communications errors are detected and corrected at this level.

It has two protocols associated with it, these being:

- **User data protocol** (UDP). This is a connectionless (unreliable) protocol used for higher layer port addressing. It offers minimal protocol overhead and is described in RFC 768
- **Transmission control protocol** (TCP). This is a connection-oriented protocol that offers vastly improved protection and error control. This

protocol, the TCP component of TCP/IP, is the heart of the TCP/IP suite of applications. It provides a very reliable method of transferring data in byte (octet) format, between applications. This is described in RFC 793.

The process and application layer

This layer provides the user or application programs with interfaces to the TCP/IP stack. At this level there are many protocols used, some of the more common ones being:

- **File transfer protocol** (FTP), which as the name implies, is used for the transfer of files between two hosts using TCP. It is described in RFC 959
- **Trivial file transfer protocol** (TFTP), which is an economic version of FTP and uses UDP instead of TCP for, reduced overhead. It is described in RFC 783
- **Simple mail transfer protocol** (SMTP), which is an example of an application, which provides access to the TCP and IP for programs sending e-mail. It is described in RFC 821
- **TELNET** (telecommunications network), which is used to emulate terminals and for remote access to servers. It can, for example, emulate a VT100 terminal across a network

Other process/application layer protocols include POP3, RPC, RLOGIN, IMAP, Berbers, HTTP and NTP. Users can also develop their own application layer protocols by means of a developer’s kit such as Winsock.

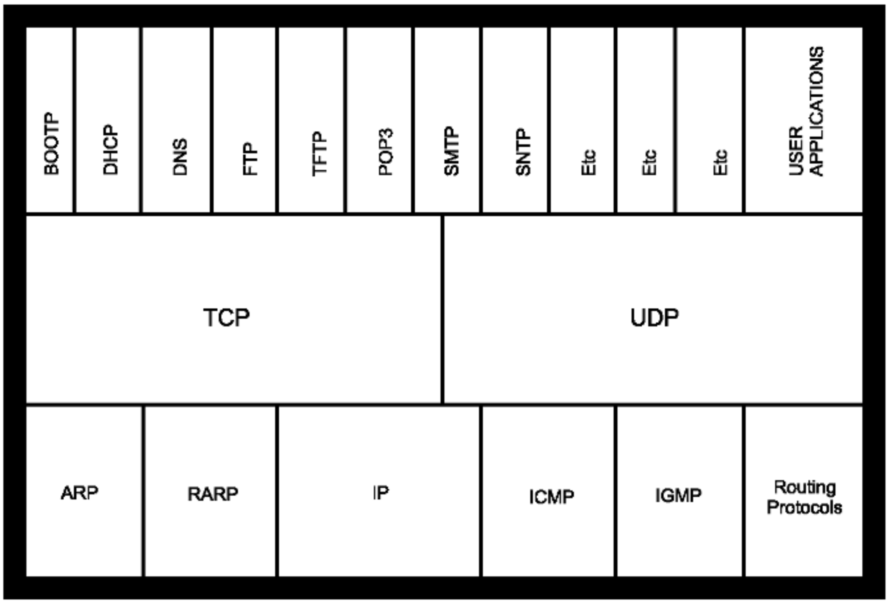


Figure 5.2
The TCP/IP protocol suite

6

Internet layer protocols

Objectives

When you have completed the study of this chapter, you should be able to:

- Explain the basic operation of all Internet layer protocols including IP, ARP, RARP, and ICMP
- Explain the purpose and application of the different fields in the IPv4 header
- Invoke the following protocols, capture their headers with a protocol analyzer, and compare the headers with those in your notes: IPv4, ARP and ICMP. You should be able to interpret the fundamental operations taking place and verify the different fields in each header
- Demonstrate the fragmentation capability of IPv4 using a protocol analyzer
- Explain the differences between class A, B and C addresses, and the relationship between class numbers, network ID and host ID
- Explain the concept of classless addressing and CIDR
- Explain the concept of subnet masks and prefixes
- Explain the concept of subnetting by means of an example
- Explain, in very basic terms, the concept of supernetting
- Set up hosts in terms of IP addresses, subnet masks and default gateways
- Understand the principles of routing, the difference between interior and exterior gateway protocols, name some examples of both and explain, in very basic terms, their principles of operation
- Explain the basic concepts of IPv6, the ‘new generation’ IP protocol

6.1 Overview

As pointed out in the previous chapter, the Internet layer is not populated by a single protocol, but rather by a collection of protocols.

They include:

- The Internet protocol (IP)
- The Internet control message protocol (ICMP),
- The address resolution protocol (ARP),
- The reverse address resolution protocol (RARP), and
- Routing protocols (such as RIP, OSPF, BGP-4, etc)

Two particular protocols that are difficult to 'map' on the DOD model are the **dynamic host configuration protocol** (DHCP) and the **boot protocol** (BootP).

DHCP was developed out of BootP and for that reason could be perceived as being resident at the same layer as BootP. BootP exhibits a dualistic behavior. On the one hand, it issues IP addresses and therefore seems to reside at the Internet Layer, as is the case with RARP. On the other hand, it allows a device to download the necessary boot file via TFTP and UDP, and in this way behaves like an application layer protocol. In the final analysis, the perceived location in the model framework is not that important, as long as the functionality is understood. In this manual both DHCP and BootP have been grouped under application layer protocols.

6.2 Internet protocol version 4 (IPv4)

The **Internet protocol** (IP) is at the core of the TCP/IP suite. It is primarily responsible for routing packets towards their destination, from router to router. This routing is performed on the basis of the IP addresses, embedded in the header attached to each packet forwarded by IP.

The most prevalent version of IP in use today is version 4 (IPv4), which uses a 32-bit address. However, IPv4 is at the end of its lifetime and is being superseded by version 6 (IPv6 or IPng), which uses a 128-bit address.

This chapter will focus primarily on version 4 as a vehicle of explaining the fundamental processes involved, but will also provide an introduction to version 6.

6.2.1 Source of IP addresses

The ultimate responsibility for the issuing of IP addresses is vested in the **Internet Assigned Numbers Authority** (IANA). This responsibility is, in turn, delegated to the three **Regional Internet Registries** (RIRs).

They are:

- **APNIC**

  Asia-Pacific Network Information Center (http://www.apnic.net)
- **ARIN**

  American Registry for Internet Numbers (http://www.arin.net)
- **RIPE NCC**

  Reseau IP Europeens (http://www.ripe.net)

The Regional Internet Registries allocate blocks of IP addresses to Internet service providers (ISPs) under their jurisdiction, for subsequent issuing to users or sub-ISPs.

The version of IP used this far, IPv4, is in the process of being superseded by IPv6. On July 14, 1999 IANA advised the Internet community that the RIRs have been authorized to commence world-wide deployment of IPv6 addresses.

The use of 'legitimate' IP addresses is a prerequisite for connecting to the Internet. For systems NOT connected to the Internet, any IP addressing scheme may be used. It is, however, recommended that so-called 'private' Internet addresses are used for this purpose, as outlined in this chapter.

6.2.2 The purpose of the IP address

The MAC or hardware address (also called the media address or Ethernet address) discussed earlier is unique for each node, and has been allocated to that particular node e.g. network interface card at the time of its manufacture. The equivalent for a human being would be its ID or Social Security number. As with a human ID number, the MAC address belongs to that node and follows it wherever it goes. This number works fine for identifying hosts on a LAN where all nodes can 'see' (or rather, 'hear') each other.

With human beings the problem arises when the intended recipient is living in another city, or worse, in another country. In this case the ID number is still relevant for final identification, but the message (e.g. a letter) first has to be routed to the destination by the postal system. For the postal system, a name on the envelope has little meaning. It requires a postal address.

The TCP/IP equivalent of this postal address is the IP address. As with the human postal address, this IP address does not belong to the node, but rather indicates its place of residence. For example, if an employee has a fixed IP address at work and he resigns, he will leave his IP address behind and his successor will 'inherit' it.

Since each host (which already has a MAC or hardware address) needs an IP address in order to communicate across the Internet, resolving host MAC addresses versus IP addressees is a mandatory function. This is performed by the **address resolution protocol** (ARP), which is to be discussed later on in this chapter.

6.2.3 IPv4 address notation

The IPv4 address consists of 32 bits, e.g.

11000000011001000110010000000001

Since this number is fine for computers but a little difficult for human beings, it is divided into four octets, which for ease of reference could be called a,b,c,d or w,x,y,z. Each octet is converted to its decimal equivalent.

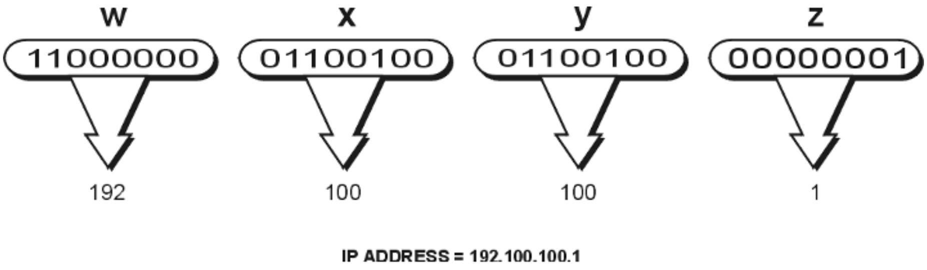


Figure 6.1
IP address structure

The result of the conversion is written as 192.100.100.1. This is known as the 'dotted decimal' or 'dotted quad' notation.

6.2.4 Network ID and host ID

Refer to the following postal address:

- 4 Kingsville Street

- Claremont 6010
- Perth WA
- Australia

The first part, viz. 4 Kingsville Street, enables the local postal deliveryman at the Australian post office in Claremont, Perth (zip code 6010) to deliver a letter to that specific residence. This assumes that the latter has already found its way to the local post office.

The second part (lines 2–4) enables the International Postal System to route the letter towards its destination post office from anywhere in the world.

In similar fashion, an IP address has two distinct parts. The first part, the network ID ('NetID') is a unique number identifying a specific network and allows the Internet routers to forward a packet towards its destination network from anywhere in the world. The second part, the host ID ('HostID') is a number allocated to a specific machine (host) on the destination network and allows the router servicing that host to deliver the packet directly to the host.

For example, in IP address 192.100.100.5 the computer or HostID would be 5, and it would be connected to network or NetID number 192.100.100.0.

6.2.5 Address classes

Originally, the intention was to allocate IP addresses in so-called address classes. Although the system proved to be problematic, and IP addresses are currently issued 'classless', the legacy of IP address classes remains and has to be understood.

To provide for flexibility in assigning addresses to networks, the interpretation of the address field was coded to specify either:

- A small number of networks with a large number of hosts (class A)
- A moderate number of networks with a moderate number of hosts (class B),
- A large number of networks with a small number of hosts (class C)

In addition, there was provision for extended addressing modes: class D was intended for multicasting whilst E was reserved for possible future use.

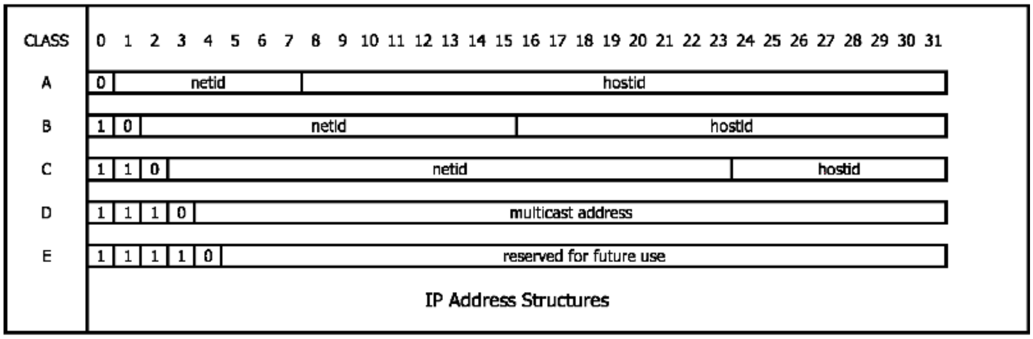


Figure 6.2
Address structure for IPv4

- For class A, the first bit is fixed as '0'
- For class B the first 2 bits are fixed as '10'
- For class C the first 3 bits are fixed as '110'

6.2.6 Determining the address class by inspection

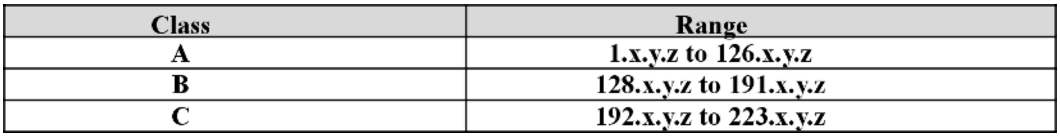
The NetID should normally not be all 0s as this indicates a local network. With this in mind, analyze the first octet (‘w’).

For class A, the first bit is fixed at 0. The binary values for ‘w’ can therefore only vary between 00000000_2 (0_{10}) and 01111111_2 (127_{10}). 0 is not allowed. However, 127 is also a reserved number, with 127.x.y.z reserved for loop-back testing. In particular, 127.0.0.1 is used to test that the TCP/IP protocol is properly configured by sending information in a loop back to the computer that originally sent the packet, without it traveling over the network. The values for ‘w’ can therefore only vary between 1 and 126, which allows for 126 possible class A NetIDs.

For class B, the first two bits are fixed at 10. The binary values for ‘w’ can therefore only vary between 10000000_2 (128_{10}) and 10111111_2 (191_{10}).

For class C, the first three bits are fixed at 110. The binary values for ‘w’ can therefore only vary between 11000000_2 (192_{10}) and 11011111_2 (223_{10}).

The relationship between ‘w’ and the address class can therefore be summarized as follows.



Class	Range
A	1.x.y.z to 126.x.y.z
B	128.x.y.z to 191.x.y.z
C	192.x.y.z to 223.x.y.z

Figure 6.3
IPv4 address range vs class

6.2.7 Number of networks and hosts per address class

Note that there are two reserved host numbers, irrespective of class. These are ‘all zeros’ or ‘all ones’ for HostID. An IP address with a host number of zero is used as the address of the whole network. For example, on a class C network with the NetID = 200.100.100, the IP address 200.100.100.0 indicates the whole network. If all the bits of the HostID are set to 1, for example 200.100.100.255, then a broadcast message will be sent to every host on that network.

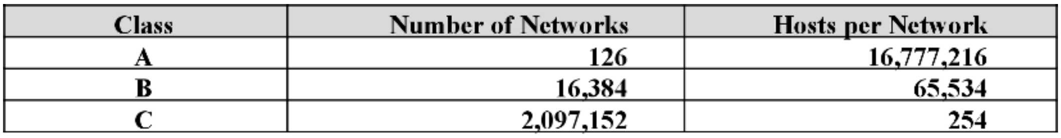
To summarize:

- HostID = ‘all zeros’ means ‘this network.’
- HostID = ‘all ones’ means ‘all hosts on this network’

For class A, the number of NetIDs is determined by octet ‘w’. Unfortunately, the first bit (fixed at 0) is used to indicate class A and hence cannot be used. This leaves seven usable bits. Seven bits allow 2^7 = 128 combinations, from 0 to 127. 0 and 127 are reserved; hence only 126 NetIDs are possible. The number of HostIDs, on the other hand, is determined by octets ‘x’, ‘y’ and ‘z’. From these 24 bits, 2^{24} = 16 777 218 combinations are available. All zeros and all ones are not permissible, which leaves 16 777 216 usable combinations.

For class B, the number of NetIDs is determined by octets ‘w’ and ‘x’. The first bits (10) are used to indicate class B and hence cannot be used. This leaves fourteen usable bits. Fourteen bits allow 2^{14} = 16 384 combinations. The number of HostIDs is determined by octets ‘y’ and ‘z’. From these 16 bits, 2^{16} = 65 536 combinations are available. All zeros and all ones are not permissible, which leaves 65 534 usable combinations.

For class C, the number of NetIDs is determined by octets 'w', 'x' and 'y'. The first three bits (110) are used to indicate class C and hence cannot be used. This leaves twenty-two usable bits. Twenty-two bits allow 2^{22} = 2 097 152 combinations. The number of HostIDs is determined by octet 'z'. From these 8 bits, 2^{8} = 256 combinations are available. Once again, all zeros and all ones are not permissible which leaves 254 usable combinations.



Class	Number of Networks	Hosts per Network
A	126	16,777,216
B	16,384	65,534
C	2,097,152	254

Figure 6.4
Hosts and subnets per class

6.2.8 Subnet masks

Strictly speaking, one should be referring to 'netmasks' in general, or to 'subnet masks' in the case of defining netmasks for the purposes of subnetting. Unfortunately, most people (including Microsoft) have confused the two issues and are referring to subnet masks in all cases.

For routing purposes it is necessary for a device to strip the HostID off an IP address, in order to ascertain whether or not the remaining NetID portion of the IP address matches the network address of that particular network.

Whilst it is easy for human beings, it is not the case for a computer and the latter has to be 'shown' which portion is NetID, and which is HostID. This is done by defining a netmask in which a '1' is entered for each bit which is part of NetID, and a '0' for each bit which is part of HostID. The computer takes care of the rest. The '1's start from the left and run in a contiguous block.

For example: A conventional class C IP address, 192.100.100.5, written in binary, would be represented in binary as 11000000 01100100 01100100 00000101. Since it is a class C address, the first 24 bits represent NetID and would therefore be masked by 1s. The subnet mask would therefore be:

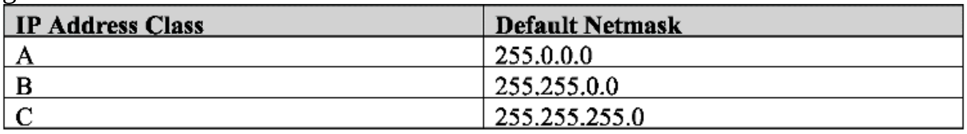
11111111 11111111 1111111 00000000.

To summarize:

- IP address: 01100100 01100100 01100100 00000101
- Subnet mask: 11111111 11111111 11111111 00000000

|< NetID >| |< HostID>|

The mask, written in decimal dotted notation, becomes 255.255.255.0. This is the so-called default netmask for class C. Default netmasks for classes A and B can be configured in the same manner.



IP Address Class	Default Netmask
A	255.0.0.0
B	255.255.0.0
C	255.255.255.0

Figure 6.5
Default netmasks

Currently IP addresses are issued classless, which means that it is not possible to determine the boundary between NetID and HostID by analyzing the IP address itself. This makes the use of a Subnet Mask even more necessary.

6.2.9 Subnetting

Although it is theoretically possible, one would never place all the hosts (for example, all 65 534 hosts on a class B address) on a single segment – the sheer volume of traffic would render the network useless. For this reason one might have to revert to subnetting.

Assume that a class C address of 192.100.100.0 has been allocated to a network. As shown earlier, a total of 254 hosts are possible. Now assume further that the company has four networks, connected by a router (or routers).

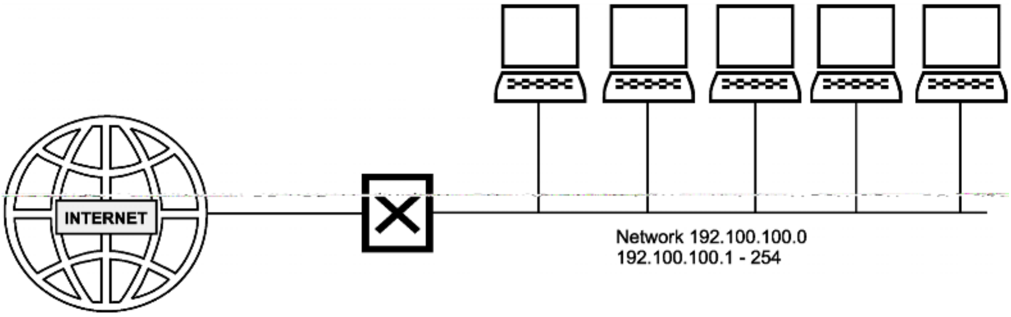


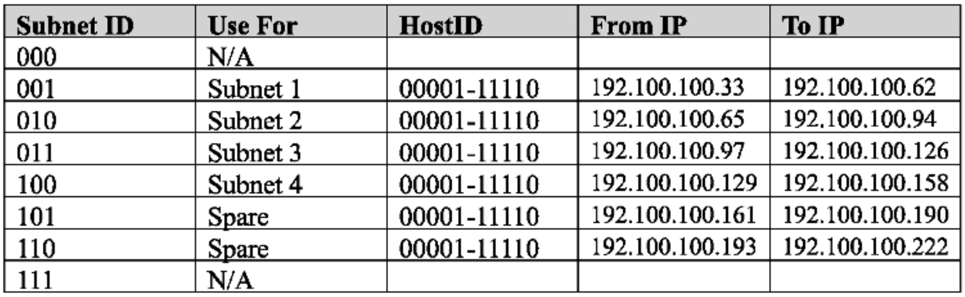
Figure 6.6
Before subnetting

Creating subnetworks under the 192.100.100.0 network address and assigning a different subnetwork number to each LAN segment could solve the problem.

To create a subnetwork, 'steal' some of the bits assigned to the HostID and use them for a subnetwork number, leaving fewer bits for HostID. Instead of NetID + HostID, the IP address will now represent NetID + SubnetID + HostID. To calculate the number of bits to be reassigned to the SubnetID, choose a number of bits 'n' so that (2_n)–2 is bigger than or equal to the number of subnets required. This is because two of the possible bit combinations of the new SubnetID, namely all 0s and all 1s, are not recommended. In this case, 4 subnets are required so 3 bits have to be 'stolen' from the HostID since (2_3)–2 = 6, which is sufficient in view of the 4 subnets we require.

Since only 5 bits are now available for HostID (3 of the 8 'stolen'), each subnetwork can now only have 30 HostIDs numbered 00001 (1_{10}) through 11110 (30_{10}), since neither 00000 nor 11111 is allowed. To be technically correct, each subnetwork will only have 29 computers (not 30) since one HostID will be allocated to the router on that subnetwork.

The 'z' of the IP address is calculated by concatenating the SubnetID and the HostID. For example, for HostID = 1 (00001) on SubnetID = 3 (011), z would be 011 appended to 00001 which gives 01100001 in binary or, 97_{10}.



Subnet ID	Use For	HostID	From IP	To IP
000	N/A			
001	Subnet 1	00001-11110	192.100.100.33	192.100.100.62
010	Subnet 2	00001-11110	192.100.100.65	192.100.100.94
011	Subnet 3	00001-11110	192.100.100.97	192.100.100.126
100	Subnet 4	00001-11110	192.100.100.129	192.100.100.158
101	Spare	00001-11110	192.100.100.161	192.100.100.190
110	Spare	00001-11110	192.100.100.193	192.100.100.222
111	N/A			

Figure 6.7
IPv4 address allocation – 6 subnets on class C address

Note that the total available number of HostIDs have dropped from 254 to 180.

In the preceding example, the first 3 bits of the HostID have been allocated as SubnetID, and have therefore effectively become part of the NetID. A default class C subnet mask would unfortunately obliterate these 3 bits, with the result that the routers would not be able to route messages between the subnets. For this reason the subnet mask has to be EXTENDED another 3 bits to the right, so that it becomes 11111111 11111111 11111111 ***111***00000. The extra bits have been typed in italics, for clarity. The subnet mask is now 255.255.255.224.

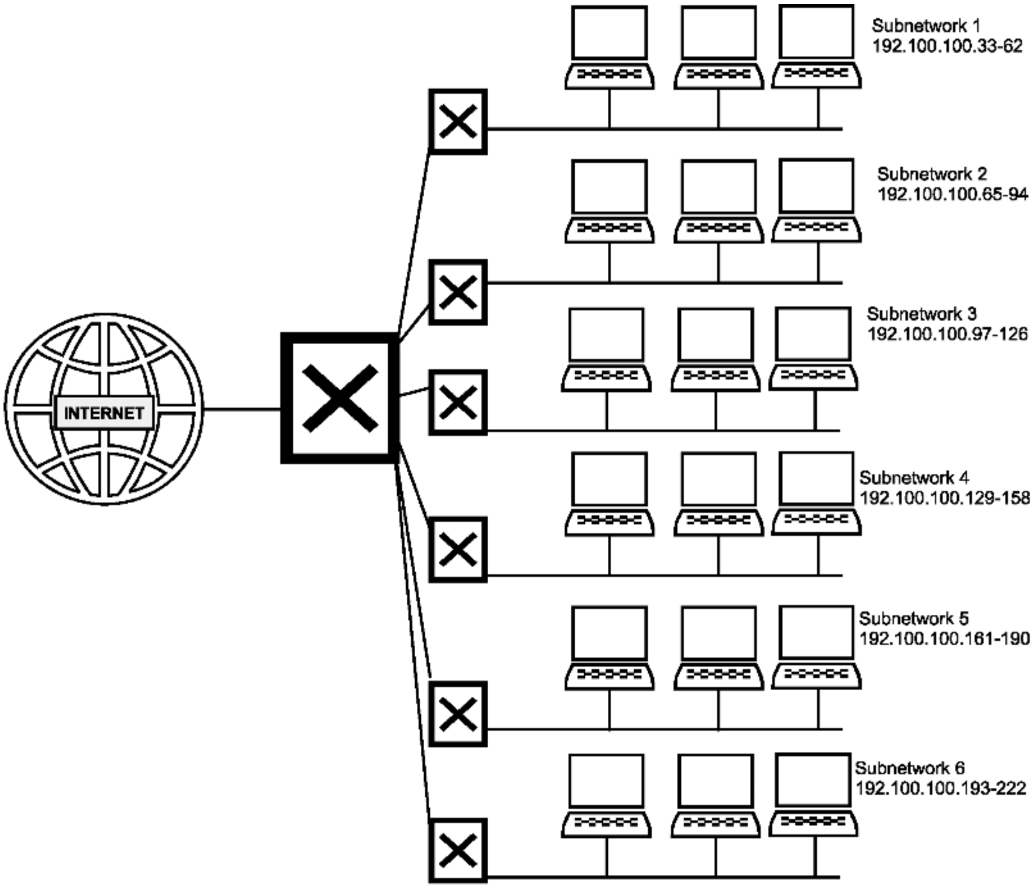


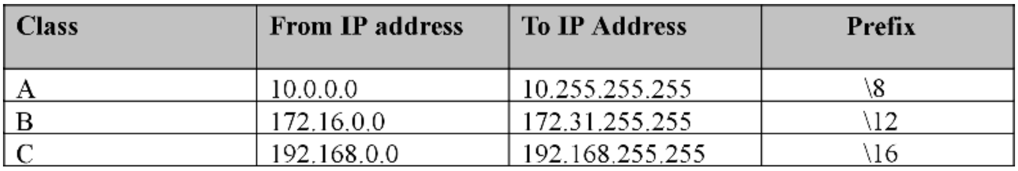
Figure 6.8
After subnetting

6.2.10 Private vs Internet-unique IP addresses

If it is certain that a network will never be connected to the Internet, any IP address can be used as long as the IP addressing rules are followed. To keep things simple, it is advisable to use class C addresses. Assign each LAN segment its own class C network number. Then it is possible to assign each host a complete IP address simply by appending the decimal host number to the decimal network number. With a unique class C network number for each LAN segment, there can be 254 hosts per segment.

If there is a possibility of connecting a network to the Internet, one should not use IP addresses that might result in address conflicts. In order to prevent such conflicts, either ask an ISP for Internet-unique IP addresses, or use IP addresses reserved for private works. The first method is the preferred one since none of the IP addresses will be used anywhere else on the Internet. The ISP may charge a fee for this privilege.

The second method of preventing IP address conflicts on the Internet is using addresses reserved for private networks. The IANA has reserved several blocks of IP addresses for this purpose as shown below:



Class	From IP address	To IP Address	Prefix
A	10.0.0.0	10.255.255.255	\8
B	172.16.0.0	172.31.255.255	\12
C	192.168.0.0	192.168.255.255	\16

Figure 6.9
Reserved IP addresses

Hosts on the Internet are not supposed to be assigned reserved IP addresses. Thus, if the network is eventually connected to the Internet, even if traffic from one of the hosts on the network somehow gets to the Internet, there should be no address conflicts. Furthermore, reserved IP addresses are not routed on the Internet because Internet routers are programmed not to forward messages sent to or from reserved IP addresses.

The disadvantage of using IP addresses reserved for private networks is that when a network does eventually get connected to the Internet, all the hosts on that network will need to be reconfigured. Each host will need to be reconfigured with an Internet-unique IP address, or one will have to configure the connecting gateway as a proxy to translate the reserved IP addresses into Internet-unique IP addresses that have been assigned by an ISP. For more information about IP addresses reserved for private networks, refer to RFC 1918.

6.2.11 Classless addressing

Initially, the IPv4 Internet addresses were only assigned in classes A, B and C. This approach turned out to be extremely wasteful, as large amounts of allocated addresses were not being used. Not only was the class D and E address space underutilized, but a company with 500 employees that was assigned a class B address would have 65,034 addresses that no-one else could use.

Presently, IPv4 addresses are considered classless. The issuing authorities simply hand down a block of contiguous addresses to ISPs, who can then issue them one by one, or break the large block up into smaller blocks for distribution to sub-ISPs, who will then repeat the process. Because of the fact that the 32 bit IPv4 addresses are no longer considered ‘classful’, the traditional distinction between class A, B and C addresses and the implied boundaries between the NetID and HostID can be ignored. Instead, whenever an IPv4 network address is assigned to an organization, it is done in the form of a 32-bit network address and a corresponding 32-bit mask. The 'ones' in the mask cover the NetID, and the 'zeros' cover the HostID. The 'ones' always run contiguously from the left and are called the **prefix**.

An address of 202.13.3.12 with a mask of 11111111111111111111111111000000 ('ones' in the first 26 positions) would therefore be said to have a prefix of 26 and would be written as **202.13.13.12/26.**

The subnet mask in this case would be 255.255.255.192.

Note that this address, in terms of the conventional classification, would have been regarded as a class C address and hence would have been assigned a prefix of /24 (subnet mask with 'ones' in the first 24 positions) by default.

6.2.12 Classless inter-domain routing (CIDR)

A second problem with the fashion in which the IP addresses were allocated by the **Network Information Center** (NIC), was the fact that it was done more or less at random and that each address had to be advertised individually in the Internet routing tables.

Consider, for example, the case of following 4 private ('traditional' class C) networks, each one with its own contiguous block of 256 (254 useable) addresses:

- **Network A:** 200.100.0.0 (IP addresses 200.100.0.1–200.100.0.255)
- **Network B:** 192.33.87.0 (IP addresses 192.33.87.1–192.33.87.255)
- **Network C:** 194.27.11.0 (IP addresses 194.27.11.1–194.27.11.255)
- **Network D:** 202.15.16.0 (IP addresses 202.15.16.1–202.15.16.255)

Assuming that there are no reserved addresses, then the concentrating router at the ISP would have to advertise 4 × 256 = 1024 separate network addresses. In a real life situation, the ISP's router would have to advertise tens of thousands of addresses. It would also be seeing hundreds of thousands, if not millions, of addresses advertised by the routers of other ISPs across the globe. In the early nineties the situation was so serious it was expected that, by 1994, the routers on the Internet would no longer be able to cope with the multitude of routing table entries.

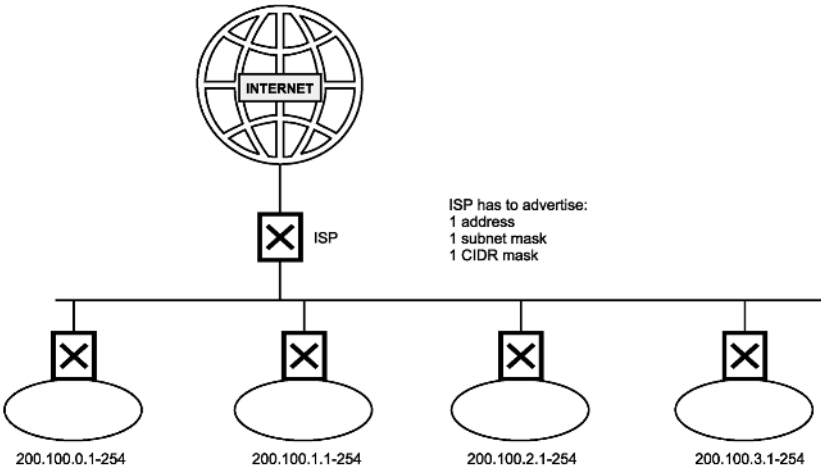


Figure 6.10
Network advertising with CIDR

To alleviate this problem, the concept of classless inter-domain routing (CIDR) was introduced. Basically, CIDR removes the imposition of the class A, B and C address masks and allows the owner of a network to 'supernet' multiple addresses together. It then allows the concentrating router to aggregate (or 'combine') these multiple contiguous network addresses into a single route advertisement on the Internet.

Take the same example as before, but this time allocates contiguous addresses. Note that 'w' can have any value between 1 and 255 since the address classes are no longer relevant.

	w x y z
Network A:	220.100.0. 0
Network B:	220.100.1. 0
Network C:	220.100.2. 0
Network D:	220.100.3. 0

CIDR now allows the router to advertise all 1000 computers under one advertisement, using the starting address of the block (220.100.0.0) and a CIDR (supernet mask) of 255.255.252.0. This is achieved as follows.

As with subnet masking, CIDR uses a mask, but it is less (shorter) than the network mask. Whereas the '1' s in the network mask indicate the bits that comprise the network ID, the '1's in the CIDR (supernet) mask indicates the bits in the IP address that do not change.

The total number of computers in this 'supernet' can be calculated as follows:

Number of '1's in network (subnet) mask = 24

Number of hosts per network = $(2^{(32-24)} - 2) = 2^8 - 2 = 254$

Number of '1's in CIDR mask = 22

X= (Number of '1's in network mask – Number of '1's in CIDR mask) = 2

Number of networks aggregated = 2 × X = 2 × 2 = 4

Total number of hosts = 4 × 254 = 1016

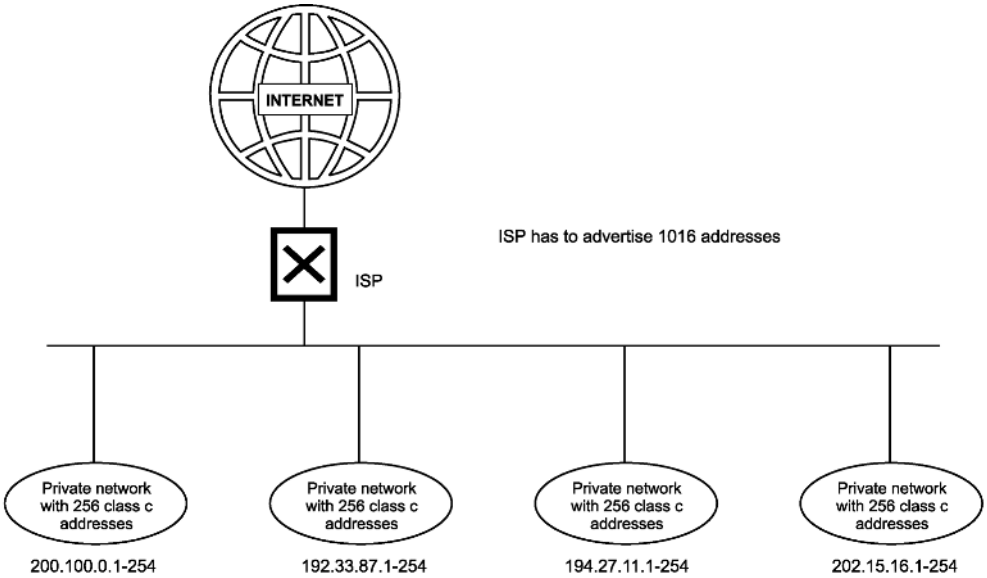
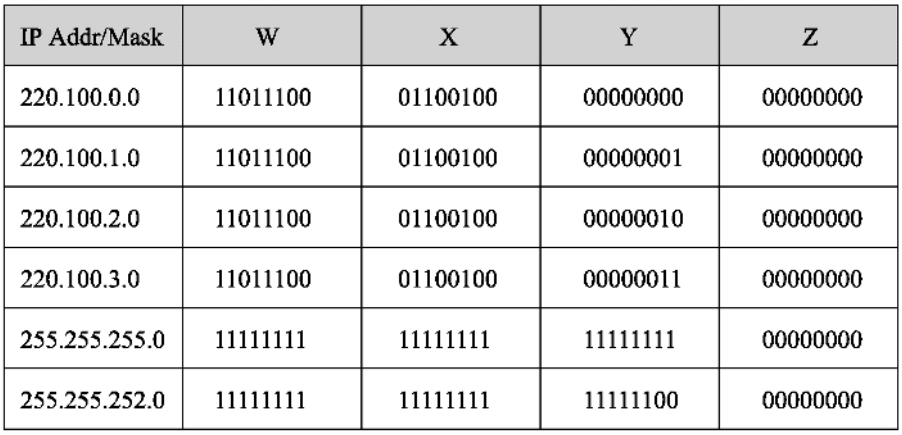

INTERNET
ISP has to advertise 1016 addresses
ISP
Private network with 256 class c addresses
Private network with 256 class c addresses
Private network with 256 class c addresses
Private network with 256 class c addresses
200.100.0.1-254
192.33.87.1-254
194.27.11.1-254
202.15.16.1-254

Figure 6.11
Network advertising without CIDR

The route advertisement of 220.100.0.0 255.255.252.0 implies a supernet comprising 4 networks, each with 254 possible hosts. The lowest IP address is 220.100.100.1 and the highest is 220.100.3.254. The first mask in the following table (255.255.255.0) is the subnet mask while the second mask (255.255.252.0) is the CIDR mask.



IP Addr/Mask	W	X	Y	Z
220.100.0.0	11011100	01100100	00000000	00000000
220.100.1.0	11011100	01100100	00000001	00000000
220.100.2.0	11011100	01100100	00000010	00000000
220.100.3.0	11011100	01100100	00000011	00000000
255.255.255.0	11111111	11111111	11111111	00000000
255.255.252.0	11111111	11111111	11111100	00000000

Figure 6.12
Binary equivalents of IP addresses and masks used in this example

CIDR and the concept of classless addressing go hand in hand since it is obvious that the concept can only work if the ISPs are allowed to exercise strict control over the issue and allocation of IP addresses. Before the advent of CIDR, clients could obtain IP addresses and regard it as their ‘property’. Under the new dispensation, the ISP needs to keep control over its allocated block(s) of IP addresses. A client can therefore only ‘rent’ IP addresses from ISP and the latter may insist on its return, should the client decide to change to another ISP.

6.2.13 IPv4 header structure

The IP header is appended to the data that IP accepts from higher-level protocols, before routing it around the network. The IP header consists of six 32-bit ‘long words’ and is made up as follows:

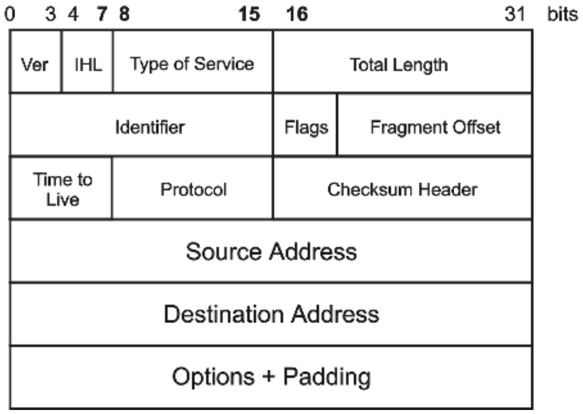


Figure 6.13
IPv4 header

Ver: 4 bits

The version field indicates the version of the IP protocol in use, and hence the format of the header. In this case it is 4.

IHL: 4 bits

The Internet header length is the length of the IP header in 32 bit 'long words', and thus points to the beginning of the data. This is necessary since the IP header can contain options and therefore has a variable length. The minimum value is 5, representing 5 × 4 = 20 bytes.

Type of service: 8 bits

The **type of service** (ToS) field is intended to provide an indication of the parameters of the quality of service desired. These parameters are used to guide the selection of the actual service parameters when transmitting a datagram through a particular network.

Some networks offer service precedence, which treats high precedence traffic as more important than other traffic (generally by accepting only traffic above a certain precedence at time of high load). The choice involved is a three-way trade-off between low delay, high reliability, and high throughput.

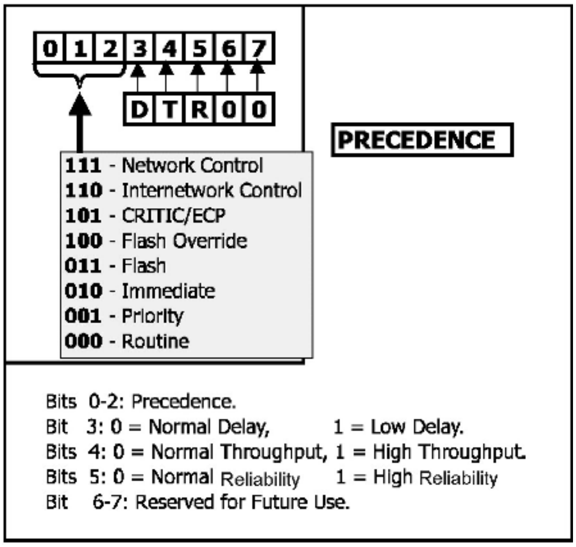

0 1 2 3 4 5 6 7
D T R 0 0
PRECEDENCE
111 - Network Control
110 - Internetwork Control
101 - CRITIC/ECP
100 - Flash Override
011 - Flash
010 - Immediate
001 - Priority
000 - Routine
Bits 0-2: Precedence.
Bit 3: 0 = Normal Delay, 1 = Low Delay.
Bits 4: 0 = Normal Throughput, 1 = High Throughput.
Bits 5: 0 = Normal Reliability 1 = High Reliability
Bit 6-7: Reserved for Future Use.

Figure 6.14

Type of service

The type of service (ToS) field is composed of a 3-bit precedence field (which is often ignored) and an unused (LSB) bit that must be 0. The remaining 4 bits may only be turned on one at a time, and are allocated as follows:

Bit 3: Minimize delay
Bit 4: Maximize throughput
Bit 5: Maximize reliability
Bit 6: Minimize monetary cost

RFC 1340 (corrected by RFC 1349) specifies how all these bits should be set for standard applications. Applications such as TELNET and RLOGIN need minimum delay since they transfer small amounts of data. FTP needs maximum throughput since it transfers large amounts of data. Network management (SNMP) requires maximum reliability and usenet news (NNTP) needs to minimize monetary cost.

Most TCP/IP implementations do not support the ToS feature, although some newer implementations of BSD and routing protocols such as OSPF and IS-IS can make routing decisions on it.

Total length: 16 bits

Total length is the length of the datagram, measured in bytes, *including* the header and data. Using this field and the header length, it can be determined where the data starts and ends. This field allows the length of a datagram to be up to 2^{16} = 65 536 bytes, the maximum size of the segment handed down to IP from the protocol above it.

Such long datagrams are, however, impractical for most hosts and networks. All hosts must at least be prepared to accept datagrams of up to 576 octets (whether they arrive whole or in fragments). It is recommended that hosts only send datagrams larger than 576 octets if they have the assurance that the destination is prepared to accept the larger datagrams.

The number 576 is selected to allow a reasonable sized data block to be transmitted in addition to the required header information. For example, this size allows a data block of 512 octets plus 64 header octets to fit in a datagram, which is the maximum size permitted by X.25. A typical IP header is 20 octets, allowing some space for headers of higher-level protocols.

Identification: 16 bits

This number uniquely identifies each datagram sent by a host. It is normally incremented by one for each datagram sent. In the case of fragmentation, it is appended to all fragments of the same datagram for the sake of reconstructing the datagram at the receiving end. It can be compared to the 'tracking' number of an item delivered by registered mail or UPS.

Flags: 3 bits

There are two flags:

- The DF (don't fragment) flag is set (=1) by the higher-level protocol (e.g. TCP) if IP is NOT allowed to fragment a datagram. If such a situation occurs, IP will not fragment and forward the datagram, but simply return an appropriate ICMP message to the sending host
- The MF (more flag) is used as follows. If fragmentation DOES occur, MF=1 will indicate that there are more fragments to follow, whilst MF=0 indicates that it is the last fragment

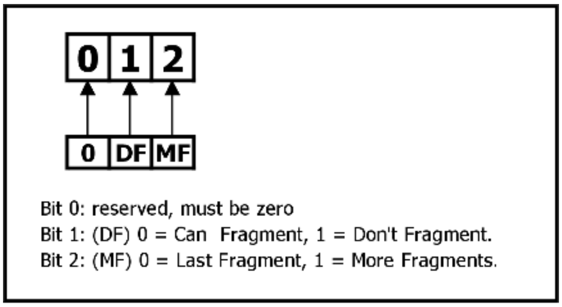

0
1
2
0
DF
MF
Bit 0: reserved, must be zero
Bit 1: (DF) 0 = Can Fragment, 1 = Don't Fragment.
Bit 2: (MF) 0 = Last Fragment, 1 = More Fragments.

Figure 6.15
Flag structure

Fragment offset: 13 bits

This field indicates where in the original datagram this fragment belongs. The fragment offset is measured in units of 8 bytes (64 bits). The first fragment has offset zero. In other words, the transmitted offset value is equal to the actual offset divided by eight. This constraint necessitates fragmentation in such a way that the offset is always exactly divisible by eight. The 13-bit offset also limits the maximum sized datagram that can be fragmented to 64 kb.

Time to live: 8 bits

The purpose of this field is to cause undeliverable datagrams to be discarded. Every router that processes a datagram must decrease the TTL by one and if this field contains the value zero, then the datagram must be destroyed.

The original design called for TTL to be decremented not only for the time it passed a datagram, but also for each second the datagram is held up at a router (hence the ‘time’ to live). Currently all routers simply decrement it every time they pass a datagram.

Protocol: 8 bits

This field indicates the next (higher) level protocol used in the data portion of the Internet datagram, in other words the protocol that resides above IP in the protocol stack and which has passed the datagram on to IP.

Typical values are 0x0806 for ARP and 0x8035 for RARP. (0x meaning ‘hex’.)

Header checksum: 16 bits

This is a checksum on the header only, referred to as a ‘standard Internet checksum’. Since some header fields change (e.g. TTL), this is recomputed and verified at each point that the IP header is processed. It is not necessary to cover the data portion of the datagram, as the protocols making use of IP, such as ICMP, IGMP, UDP and TCP, all have a checksum in their headers to cover their own header and data.

To calculate it, the header is divided up into 16-bit words. These words are then added together (normal binary addition with carry) one by one, and the interim sum stored in a 32-bit accumulator. When done, the upper 16 bits of the result is stripped off and added to the lower 16 bits. If, after this, there is a carry out to the 17th bit, it is carried back and added to bit 0. The result is then truncated to 16 bits.

Source and destination addresses: 32 bits each

These are the 32-bit IP addresses of both the origin and the destination of the datagram.

6.2.14 Packet fragmentation

It should be clear by now that IP might often have difficulty in sending packets across a network since, for example, Ethernet can only accommodate 1500 octets at a time and X.25 is limited to 576. This is where the fragmentation process comes into play. The relevant field here is ‘fragment offset’ (13 bits) while the relevant flags are DF (don’t fragment) and MF (more fragments).

Consider a datagram consisting of an IP header followed by 3500 bytes of data. This cannot be transported over an Ethernet network, so it has to be fragmented in order to ‘fit’. The datagram will be broken up into three separate datagrams, each with their own IP header, with the first two frames around 1500 bytes and the last fragment around 500 bytes. The three frames will travel to their destination independently, and will be recognized as fragments of the original datagram by virtue of the number in the identifier field. However, there is no guarantee that they will arrive in the correct order, and the receiver needs to reassemble them.

For this reason the fragment offset field indicates the distance or offset between the start of this particular fragment of data, and the starting point of the original frame. One problem though – since only 13 bits are available in the header for the fragment offset (instead of 16), this offset is divided by 8 before transmission, and again multiplied by 8 after reception, requiring the data size (i.e. the offset) to be a multiple of 8 – so an offset of 1500 won't do. 1480 will be OK since it is divisible by 8. The data will be transmitted as fragments of 1480, 1480 and finally the remainder of 540 bytes. The fragment offsets will be 0, 1480 and 2960 bytes respectively, or 0, 185 and 370 – after division by 8.

Incidentally, another reason why the data per fragment cannot exceed 1480 bytes for Ethernet, is that the IP header has to be included for each datagram (otherwise individual datagrams will not be routable) and hence 20 of the 1500 bytes have to be forfeited to the IP header.

The first frame will be transmitted with 1480 bytes of data, fragment offset = 0, and MF (more flag) = 1

The second frame will be transmitted with the next 1480 bytes of data, fragment offset = 185, and MF = 1

The last third frame will be transmitted with 540 bytes of data, fragment offset = 370, MF = 0.

Some protocol analyzers will indicate the offset in hexadecimal, hence it will be displayed as 0xb9 and 0x172, respectively.

For any given type of network the packet size cannot exceed the so-called MTU (maximum transmission unit) for that type of network. The following are some default values:

16 Mbps (IBM) token ring:	17 914 (bytes)
4 Mbps (IEEE802.5) token ring	4464
FDDI	4352
Ethernet/ IEEE802.3	1500
X.25	576
PPP (low delay)	296

The fragmentation mechanism can be checked by doing a 'ping' across a network, and setting the data (–l) parameter to exceed the MTU value for the network.

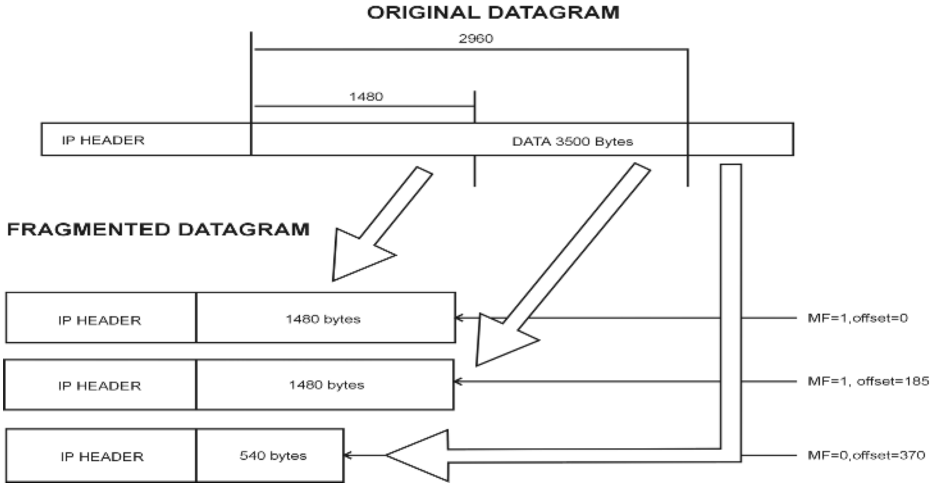


Figure 6.16
IPv4 fragmentation

6.3 Internet protocol version 6 (IPv6/IPng)

6.3.1 Introduction

IPng ('IP new generation'), as documented in RFC 1752, was approved by the Internet Engineering Steering Group in November 1994 and made a Proposed Standard. The formal name of this protocol is IPv6 ('IP version 6'). After extensive testing, IANA gave permission for its deployment in mid-1999.

IPv6 is an update of IPv4, to be installed as a 'backwards compatible' software upgrade, with no scheduled implementation dates. It runs well on high performance networks such as ATM, and at the same time remains efficient enough for low bandwidth networks such as wireless LANs. It also makes provision for Internet functions such as audio broadcasting and encryption.

Upgrading to and deployment of IPv6 can be achieved in stages. Individual IPv4 hosts and routers may be upgraded to IPv6 one at a time without affecting any other hosts or routers. New IPv6 hosts and routers can be installed one by one. There are no pre-requisites to upgrading routers, but in the case of upgrading hosts to IPv6 the DNS server must first be upgraded to handle IPv6 address records.

When existing IPv4 hosts or routers are upgraded to IPv6, they may continue to use their existing address. They do not need to be assigned new IPv6 addresses, neither do administrators have to draft new addressing plans.

The simplicity of the upgrade to IPv6 is brought about through the transition mechanisms built into IPv6. They include the following:

- The IPv6 addressing structure embeds IPv4 addresses within IPv6 addresses, and encodes other information used by the transition mechanisms
- All hosts and routers upgraded to IPv6 in the early transition phase will be 'dual' capable (i.e. implement complete IPv4 and IPv6 protocol stacks)
- Encapsulation of IPv6 packets within IPv4 headers will be used to carry them over segments of the end-to-end path where the routers have not yet been upgraded to IPv6

The IPv6 transition mechanisms ensure that IPv6 hosts can inter-operate with IPv4 hosts anywhere in the Internet up until the time when IPv4 addresses run out, and allows IPv6 and IPv4 hosts within a limited scope to inter-operate indefinitely after that. This feature protects the huge investment users have made in IPv4 and ensures that IPv6 does not render IPv4 obsolete. Hosts that need only a limited connectivity range (e.g., printers) need never be upgraded to IPv6.

6.3.2 IPv6 overview

The changes from IPv4 to IPv6 fall primarily into the following categories:

- **Expanded routing and addressing capabilities**

  IPv6 increases the IP address size from 32 bits to 128 bits, to support more levels of addressing hierarchy and a much greater number of addressable nodes, and simpler auto-configuration of addresses
- **Anycasting**

  A new type of address called an anycast address is defined; to identify sets of nodes where a packet sent to the group of anycast addresses is delivered to (only) one of the nodes. The use of anycast addresses in the IPv6 source route allows nodes to control the path that their traffic flows

- **Header format simplification**

  Some IPv4 header fields have been dropped or made optional, to reduce the effort involved in processing packets. The IPv6 header was also kept as small as possible despite the increased size of the addresses. Even though the IPv6 addresses are four times longer than the IPv4 addresses, the IPv6 header is only twice the size of the IPv4 header

- **Improved support for options**

  Changes in the way IP header options are encoded allows for more efficient forwarding, less stringent limits on the length of options, and greater flexibility for introducing new options in the future

- **Quality-of-service capabilities**

  A new capability is added to enable the labeling of packets belonging to particular traffic 'flows' for which the sender requests special handling, such as special 'quality of service' or 'real-time' service

- **Authentication and privacy capabilities**

  IPv6 includes extensions that provide support for authentication, data integrity, and confidentiality

6.3.3 IPv6 header format

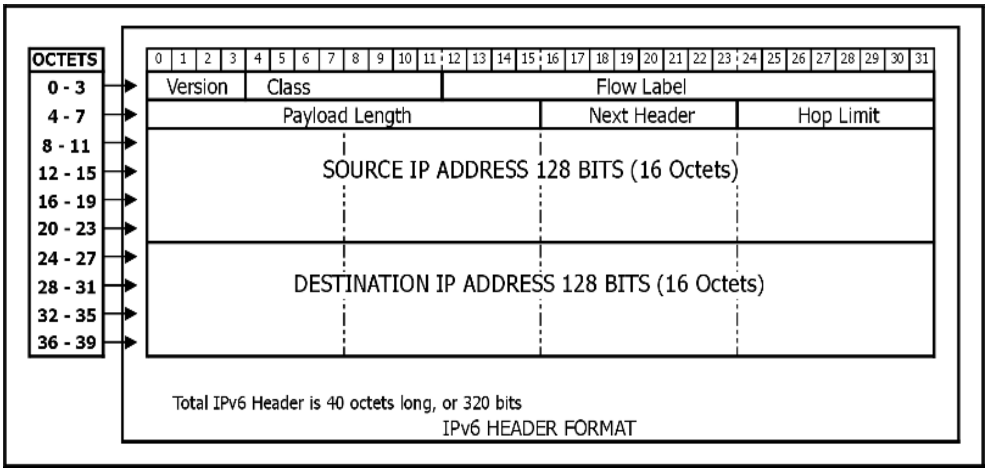


Figure 6.17

IPv6 header

The header contains the following fields:

Ver: 4 bits

The Internet protocol version number, viz. 6.

Class: 8 bits

Class value. This replaces the 4-bit priority value envisaged during the early stages of the design and is used in conjunction with the Flow label.

Flow label: 20 bits

A flow is a sequence of packets sent from a particular source to a particular (unicast or multicast) destination for which the source desires special handling by the intervening routers. This is an optional field to be used if specific non-standard ('non-default') handling is required to support applications that require some degree of consistent throughput in order to minimize delay and/or jitter. These types of applications are commonly described as 'multi-media' or 'real-time' applications.

The flow label will effect the way the packets are handled but will not influence the routing decisions.

Payload length: 16 bits

The payload is the rest of the packet following the IPv6 header, in octets. The maximum payload that can be carried behind a standard IPv6 header cannot exceed 65 536 bytes. With an extension header this is possible, the datagram is then referred to as a Jumbo datagram. Payload length differs slightly from the IPv4 in that the 'total length' field does not include the header.

Next hdr: 8 bits

This identifies the type of header immediately following the IPv6 header, using the same values as the IPv4 protocol field. Unlike IPv4, where this would typically point to TCP or UDP, this field could either point to the next protocol header (TCP) or to the next IPv6 extension header.

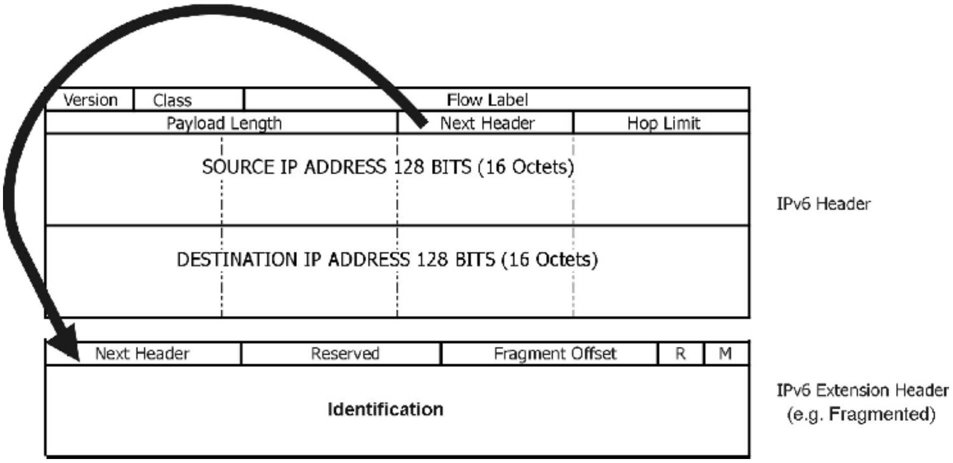


Figure 6.18
Header insertion and 'next header' field

Hop limit: 8 bits

This is an unsigned integer, similar to TTL in IPv4. It is decremented by 1 by each node that forwards the packet. The packet is discarded if hop limit is decremented to zero.

Source address: 128 bits

This is the address of the initial sender of the packet.

Destination address: 128 bits

This is address of the intended recipient of the packet, which is not necessarily the ultimate recipient, if an optional routing header is present.

6.3.4 IPv6 extensions

IPv6 includes an improved option mechanism over IPv4. Instead of placing extra options bytes within the main header, IPv6 options are placed in separate extension headers that are located between the IPv6 header and the transport layer header in a packet.

Most IPv6 extension headers are not examined or processed by routers along a packet's path until it arrives at its final destination. This leads to a major improvement in router performance for packets containing options. In IPv4 the presence of any options requires the router to examine all options.

IPv6 extension headers can be of arbitrary length and the total amount of options carried in a packet is not limited to 40 bytes as with IPv4. They are also not carried within the main header, as with IPv4, but are only used when needed, and are carried behind the main header. This feature plus the manner in which they are processed, permits IPv6 options to be used for functions, which were not practical in IPv4. Good examples of this are the IPv6 authentication and security encapsulation options.

In order to improve the performance when handling subsequent option headers and the transport protocol which follows, IPv6 options are always an integer multiple of 8 octets long, in order to retain this alignment for subsequent headers.

The IPv6 extension headers currently defined are:

- Routing header (for extended routing, similar to the IPv4 loose source route).
- Fragment header (for fragmentation and reassembly).
- Authentication header (for integrity and authentication).
- Encrypted security payload (for confidentiality).
- Hop-by-hop options header (for special options that require hop-by-hop processing).
- Destination options header (for optional information to be examined by the destination node).

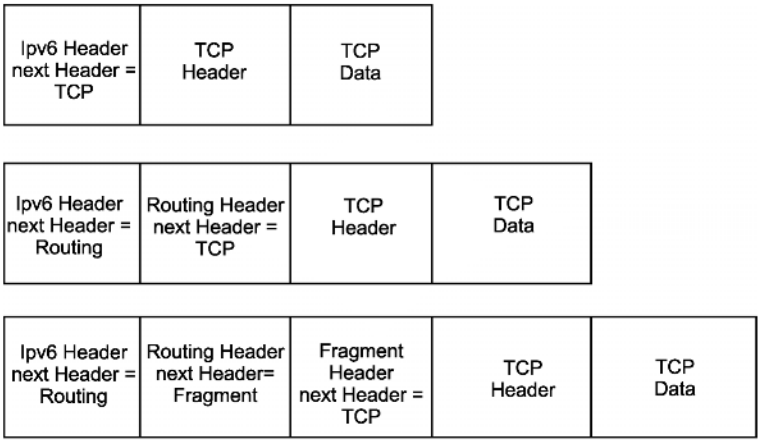

Ipv6 Header next Header = TCP
TCP Header
TCP Data
Ipv6 Header next Header = Routing
Routing Header next Header = TCP
TCP Header
TCP Data
Ipv6 Header next Header = Routing
Routing Header next Header= Fragment
Fragment Header next Header = TCP
TCP Header
TCP Data

Figure 6.19
Carrying IPv6 extension headers

6.3.5 IPv6 addresses

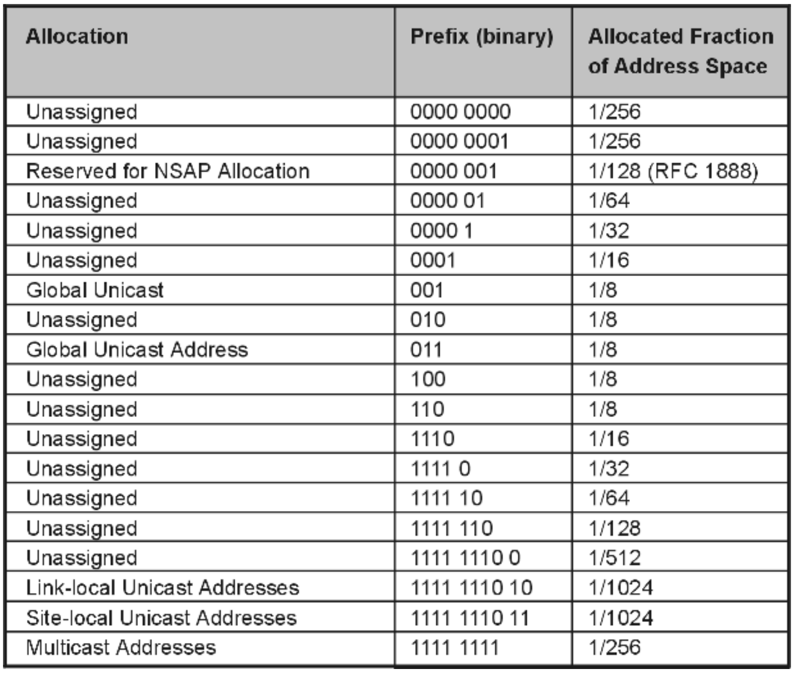
IPv6 addresses are 128 bits long and are identifiers for individual interfaces or sets of interfaces. IPv6 Addresses of all types are assigned to **interfaces** (i.e. network interface Cards) and NOT to nodes i.e. hosts. Since each interface belongs to a single node, any of that node's interface's unicast addresses may be used as an identifier for the node. A single interface may be assigned multiple IPv6 addresses of any type.

There are three types of IPv6 addresses. These are unicast, anycast, and multicast.

- **Unicast** addresses identify a single interface
- **Anycast addresses** identify a set of interfaces such that a packet sent to an anycast address will be delivered to one member of the set
- **Multicast** addresses identify a group of interfaces, such that a packet sent to a multicast address is delivered to all of the interfaces in the group. There are no broadcast addresses in IPv6, their function being superseded by multicast addresses

The IPv6 address is four times the length of IPv4 addresses (128 vs 32). This is 4 billion times 4 billion (2^{96}) times the size of the IPv4 address space (2^{32}). This works out to be 340 282 366 920 938 463 463 374 607 431 768 211 456. Theoretically this is approximately 665 570 793 348 866 943 898 599 addresses per square meter of the surface of the planet Earth (assuming the Earth surface is 511 263 971 197 990 square meters). In more practical terms, considering that the creation of addressing hierarchies, which reduces the efficiency of the usage of the address space, IPv6 is still expected to support between 8×10^{17} to 2×10^{33} nodes. Even the most pessimistic estimate provides around 1500 addresses per square meter of the surface of planet Earth.

The leading bits in the address indicate the specific type of IPv6 address. The variable-length field comprising these leading bits is called the format prefix (FP). The current allocation of these prefixes is as follows:



Allocation	Prefix (binary)	Allocated Fraction of Address Space
Unassigned	0000 0000	1/256
Unassigned	0000 0001	1/256
Reserved for NSAP Allocation	0000 001	1/128 (RFC 1888)
Unassigned	0000 01	1/64
Unassigned	0000 1	1/32
Unassigned	0001	1/16
Global Unicast	001	1/8
Unassigned	010	1/8
Global Unicast Address	011	1/8
Unassigned	100	1/8
Unassigned	110	1/8
Unassigned	1110	1/16
Unassigned	1111 0	1/32
Unassigned	1111 10	1/64
Unassigned	1111 110	1/128
Unassigned	1111 1110 0	1/512
Link-local Unicast Addresses	1111 1110 10	1/1024
Site-local Unicast Addresses	1111 1110 11	1/1024
Multicast Addresses	1111 1111	1/256

Figure 6.20

IPv6 address ranges

This allocation supports the direct allocation of global unicast addresses, local use addresses, and multicast addresses. Space is reserved for NSAP addresses, IPX addresses, and geographic-based unicast addresses. The remainder of the address space is unassigned for future use. This can be used for expansion of existing use (e.g., additional provider addresses, etc) or new uses (e.g., separate locators and identifiers). Note that anycast addresses are not shown here because they are allocated out of the unicast address space.

Approximately fifteen per cent of the address space is initially allocated. The remaining 85% is reserved for future use.

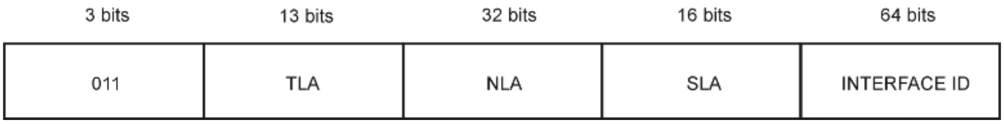
Unicast addresses

There are several forms of unicast address assignment in IPv6. These are:

- Global unicast addresses
- Unspecified addresses
- Loopback addresses
- IPv4-based addresses
- Site local addresses
- Link local addresses

Global unicast addresses

These addresses are used for global communication. They are similar in function to IPv4 addresses under CIDR. Their format is:



3 bits	13 bits	32 bits	16 bits	64 bits
011	TLA	NLA	SLA	INTERFACE ID

Figure 6.21
Address format: Global unicast address

The first 3 bits identify the address as a global unicast address.

The next, 13-bit, field (TLA) identifies the top level aggregator. This number will be used to identify the relevant Internet 'exchange point', or long-haul ('backbone') provider. These numbers (8192 of them) will be issued by IANA, to be further distributed via the three regional registries (ARIN, RIPE and APNIC), who could possibly further delegate the allocation of sub-ranges to national or regional registries such as the French NIC managed by INRIA for French networks.

The third, 32-bit, field (NLA) identifies the next level aggregator. This will probably be structured by long-haul providers to identify a second-tier provider by means of the first *n* bits, and to identify a subscriber to that second-tier provider by means of the remaining 32–*n* bits.

The fourth, 16-bit, field is the SLA or site local aggregator. This will be allocated to a link within a site, and is not associated with a registry or service provider. In other words, it will remain unchanged despite a change of service provider. Its closest equivalent in IPv4 would be the 'NetID'.

The last field is the 64-bit interface ID. This is the equivalent of the 'HostID' in IPv4. However, instead of an arbitrary number it would consist of the hardware address of the interface, e.g. the Ethernet MAC address.

- All identifiers will be 64 bits long even if there are only a few devices on the network
- Where possible these identifiers will be based on the IEEE EUI-64 format

Existing 48-bit MAC addresses are converted to EUI-64 format by splitting them in the middle and inserting the string FF-FE in between the two halves.

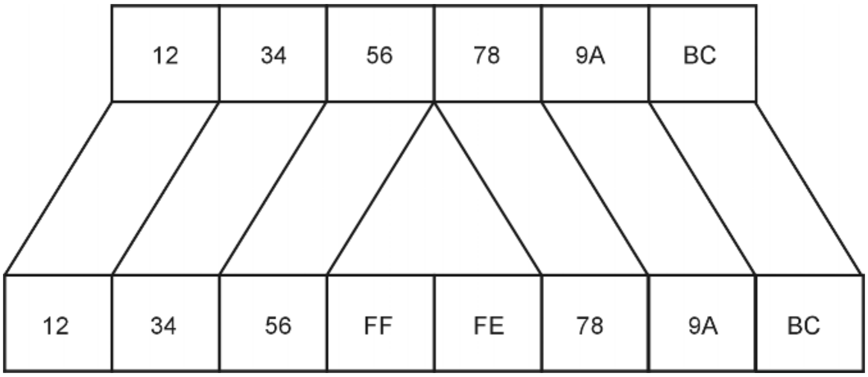


Figure 6.22
Converting a 48-bit MAC address to EUI-64 format

Unspecified addresses

This can be written as 0:0:0:0:0:0:0:0, or simply '::' (double colon). This address can be used as a source address by a station that has not yet been configured with an IP address. It can never be used as a destination address. This is similar to 0.0.0.0 in IPv4.

Loopback addresses

The loopback address 0:0:0:0:0:0:0:1 can be used by a node to send a datagram to itself. It is similar to the 127.0.0.1 of IPv4.

IPv4-based addresses

It is possible to construct an IPv6 address out of an existing IPv4 address. This is done by prepending 96 zero bits to a 32-bit IPv4 address. The result is written as 0:0:0:0:0:0:192.100.100.3, or simply ::192.100.100.3.

Site-local unicast addresses

Site local addresses are partially equivalent of the IPv4 private addresses. The site local addressing prefix 111 1110 11 has been reserved for this purpose. A typical site local address will consist of this prefix, a set of 38 zeros, a subnet ID, and the interface identifier. Site local addresses cannot be routed in the Internet, but only between two stations on a single site.

The last 80 bits of a site local address are identical to the last 80 bits of a global unicast address. This allows for easy renumbering where a site has to be connected to the Internet.

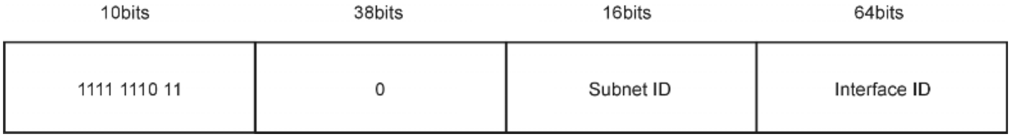


Figure 6.23
Site-local unicast addresses

Link-local unicast addresses

Stations that are not yet configured with either a provider-based address or a site local address may use link local addresses. Theses are composed of the link local prefix, 1111 1110 10, a set of 0s, and an interface identifier. These addresses can only be used by stations connected to the same local network and packets addressed in this way cannot traverse a router.

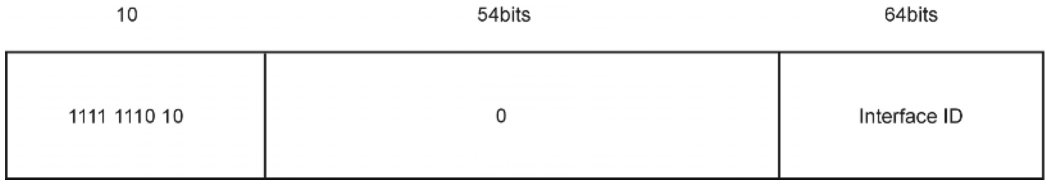


Figure 6.24
Link-local unicast addresses

Anycast addresses

An IPv6 anycast address is an address that is assigned to more than one interface (typically belonging to different nodes), with the property that a packet sent to an anycast address is routed to the 'nearest' interface having that address, according to the routing protocols' measure of distance. Anycast addresses, when used as part of a route sequence, permits a node to select which of several internet service providers it wants to carry its traffic. This capability is sometimes called 'source selected policies'. This would be implemented by configuring anycast addresses to identify the set of routers belonging to Internet service providers (e.g. one anycast address per Internet service provider). These anycast addresses can be used as intermediate addresses in an IPv6 routing header, to cause a packet to be delivered via a particular provider or sequence of providers.

Other possible uses of anycast addresses are to identify the set of routers attached to a particular subnet, or the set of routers providing entry into a particular routing domain. Anycast addresses are allocated from the unicast address space, using any of the defined unicast address formats. Thus, anycast addresses are syntactically indistinguishable from unicast addresses. When a unicast address is assigned to more than one interface, thus turning it into an anycast address, the nodes to which the address is assigned must be explicitly configured to know that it is an anycast address.

Multicast addresses

An IPv6 multicast address is an identifier for a group of interfaces. An interface may belong to any number of multicast groups. Multicast addresses have the following format:

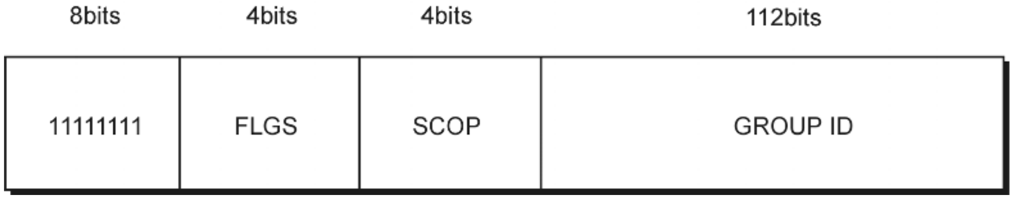


Figure 6.25
Address format: IPv6 multicast

The **11111111** (0xFF) at the start of the address identify the address as being a multicast address.

- **FLGS**. Four bits are reserved for flags. The first 3 bits are currently reserved, and set to 0. The last bit (the one on the right) is called T for 'transient'. T = 0 indicates a permanently assigned ('well-known') multicast address, assigned by IANA, while T = 1 indicates a non-permanently assigned ('transient') multicast address

- **SCOP** is a 4-bit multicast scope value used to limit the scope of the multicast group, for example to ensure that packets intended for a local videoconference are not spread across the Internet.

  The values are:

1	Interface-local scope
2	Link-local scope
3	Subnet-local scope
4	Admin-local scope
5	Site-local scope
8	Organization-local scope

- **GROUP ID** identifies the multicast group, either permanent or transient, within the given scope. Permanent group IDs are assigned by IANA.

The following example shows how it all fits together. The multicast address FF:08::43 points to all NTP servers in a given organization, in the following way:

- FF indicates that this is a multicast address
- 0 indicates that the T flag is set to 0, i.e. this is a permanently assigned multicast address
- 8 points to all interfaces in the same organization as the sender (see SCOPE options above)
- Group ID = 43 has been permanently assigned to network time protocol (NTP) servers

6.3.6 Flow labels

The 20-bit flow label field in the IPv6 header may be used by a source to label those packets for which it requests special handling by the IPv6 routers. Hosts or routers that do not support the functions of the flow label field are required to set the field to zero when originating a packet, pass the field on unchanged when forwarding a packet, and ignore the field when receiving a packet. The actual nature of that special handling might be conveyed to the routers by a control protocol, such as a resource reservation protocol (e.g. RSVP), or by information within the flow's packets themselves, e.g., in a hop-by-hop option. A flow is uniquely identified by the combination of a source IP address and a non-zero flow label.

A flow label is assigned to a flow by the flow's source node. Flow labels are chosen (pseudo-) randomly and uniformly from the range 0x1 to 0xFFFFFF. The purpose of the random allocation is to make any set of bits within the flow label field suitable for use as a hash key by routers, for looking up the state associated with the flow. All packets belonging to the same flow must be sent with the same source address, same destination address, and same (non-zero) flow label.

If any of those packets includes a hop-by-hop options header, then they all must be originated with the same hop-by-hop options header contents (excluding the next header field of the hop-by-hop options header). If any of those packets includes a routing header, then they all must be originated with the same contents in all extension headers up to and including the routing header (excluding the next header field in the routing header). The routers or destinations are permitted, but not required, to verify that these conditions are satisfied. If a violation is detected, it should be reported to the source by an ICMP parameter problem message, code 0, pointing to the high-order octet of the flow label field.

6.4 Address resolution protocol (ARP)

ARP is used with IPv4. Initially the designers of IPv6 assumed that it would use ARP as well, but subsequent work by the SIP, SIPP and IPv6 working groups led to the development of the IPv6 ‘neighbor discovery’ procedures that encompass ARP, as well as those of router discovery.

Some network technologies make address resolution difficult. Ethernet interface boards, for example, come with built-in 48-bit hardware addresses. This creates several difficulties:

- No simple correlation, applicable to the whole network, can be created between physical (MAC) addresses and Internet protocol (IP) addresses
- When the interface board fails and has to be replaced the Internet protocol (IP) address then has to be remapped to a different MAC address
- The MAC address is too long to be encoded into the 32-bit Internet protocol (IP) address

To overcome these problems in an efficient manner, and eliminate the need for applications to know about MAC addresses, the address resolution protocol (ARP) (RFC 826) resolves addresses dynamically.

When a host wishes to communicate with another host on the same physical network, it needs the destination MAC address in order to compose the basic level 2 frame. If it does not know what the destination MAC address is, but has its IP address, it broadcasts a special type of datagram in order to resolve the problem. This is called an address resolution protocol (ARP) request. This datagram requests the owner of the unresolved Internet protocol (IP) address to reply with its MAC address. All hosts on the network will receive the broadcast, but only the one that recognizes its own IP address will respond.

While the sender could, of course, just broadcast the original datagram to all hosts on the network, this would impose an unnecessary load on the network, especially if the datagram was large. A small address resolution protocol (ARP) request, followed by a small Address Resolution Protocol (ARP) reply, followed by a direct transmission of the original datagram, is a much more efficient way of resolving the problem.

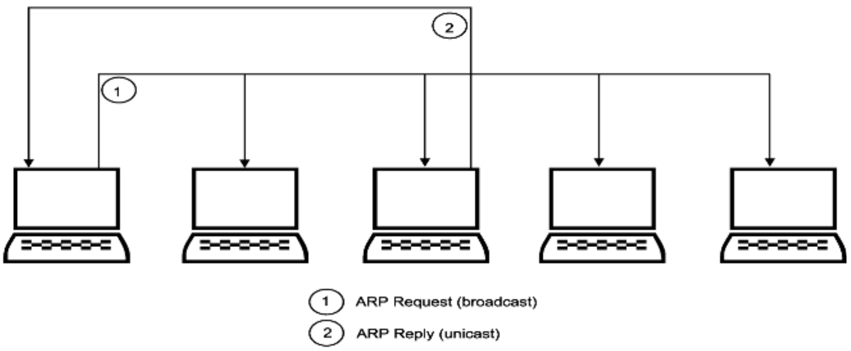


Figure 6.26
ARP operation

6.4.1 Address resolution cache

Because communication between two computers usually involves transfer of a succession of datagrams, it is prudent for the sender to ‘remember’ the MAC information it receives, at least for a while. Thus, when the sender receives an ARP reply, it stores the MAC address it receives as well as the corresponding IP address in its ARP cache. Before sending any message to a specific IP address it checks first to see if the relevant address binding is in the cache. This saves it from repeatedly broadcasting identical address resolution protocol (ARP) requests.

To further reduce communication overheads, when a host broadcasts an ARP request it includes its own IP address and MAC address, and these are stored in the ARP caches of all other hosts that receive the broadcast. When a new host is added to a network it can be made to send an ARP broadcast to inform all other hosts on that network of its address.

Some very small networks do not use ARP caches, but the continual traffic of ARP requests and replies on a larger network would have a serious negative impact on the network’s performance.

The ARP cache holds 4 fields of information for each device:

IF index – the number of the entry in the table

Physical address – the MAC address of the device

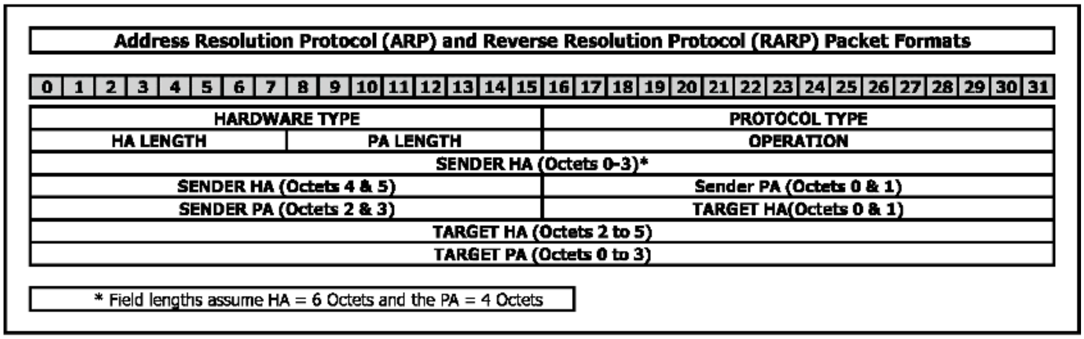
Internet protocol (IP) address – the corresponding IP address

Type – the type of entry in the ARP cache. There are 4 possible types:

4 = static – the entry will not change
3 = dynamic – the entry can change
2 = the entry is invalid
1 = none of the above

6.4.2 ARP header

The layout of an ARP datagram is as follows:



Address Resolution Protocol (ARP) and Reverse Resolution Protocol (RARP) Packet Formats

0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31

HARDWARE TYPE		PROTOCOL TYPE
HA LENGTH	PA LENGTH	OPERATION
SENDER HA (Octets 0-3)*		
SENDER HA (Octets 4 & 5)		Sender PA (Octets 0 & 1)
SENDER PA (Octets 2 & 3)		TARGET HA(Octets 0 & 1)
TARGET HA (Octets 2 to 5)		
TARGET PA (Octets 0 to 3)		

* Field lengths assume HA = 6 Octets and the PA = 4 Octets

Figure 6.27
ARP header

Hardware type: 16 bits

Specifies the hardware interface type of the target, e.g.:

1 = Ethernet
3 = X.25
4 = Token ring
6 = IEEE 802.x
7 = ARCnet

Protocol type: 16 bits

Specifies the type of high-level protocol address the sending device is using. For example,

2048_{10} (0x800): IP
2054_{10} (0x806): ARP
3282_{10} (0xcd2): RARP

HA length: 8 bits

The length, in bytes, of the hardware (MAC) address. For Ethernet it is 6.

PA length: 8 bits

The length, in bytes, of the internetwork protocol address. For IP it is 4.

Operation: 8 bits

Indicates the type of ARP datagram:

1 = ARP request
2 = ARP reply
3 = RARP request
4 = RARP reply

Sender HA: 48 bits

The hardware (MAC) address of the sender.

Sender PA: 32 bits

The (internetwork) protocol address of the sender.
Target HA: 48 bits
The hardware (MAC) address of the target host.

Target PA

The (internetwork) protocol address of the target host.

Because of the use of fields to indicate the lengths of the hardware and protocol addresses, the address fields can be used to carry a variety of address types, making ARP applicable to a number of different types of network.

The broadcasting of ARP requests presents some potential problems. Networks such as Ethernet employ connectionless delivery systems i.e. the sender does not receive any feedback as to whether datagrams it has transmitted were received by the target device. If the target is not available, the ARP request destined for it will be lost without trace and no ARP response will be generated. Thus the sender must be programmed to retransmit its ARP request after a certain time period, and must be able to store the datagram it is attempting to transmit in the interim. It must also remember what requests it has sent out so that it does not send out multiple ARP requests for the same address. If it does not receive an ARP reply it will eventually have to discard the outgoing datagrams.

Because it is possible for a machine's hardware address to change, as happens when an Ethernet interface fails and has to be replaced, entries in an ARP cache have a limited life span after which they are deleted. Every time a machine with an ARP cache receives an ARP message, it uses the information to update its own ARP cache. If the incoming address binding already exists it overwrites the existing entry with the fresh information and resets the timer for that entry.

The host trying to determine another machine's MAC address will send out an ARP request to that machine. In the datagram it will set operation = 1 (ARP request), and insert its own IP and MAC addresses as well as the destination machine's IP address in the header. The field for the destination machine's MAC address will be left zero.

It will then broadcast this message using all 'ones' in the destination address of the LLC frame so that all hosts on that subnet will 'see' the request.

If a machine is the target of an incoming ARP request, its own ARP software will reply. It swaps the target and sender address pairs in the ARP datagram (both HA and PA), inserts its own MAC address into the relevant field, changes the operation code to 2 (ARP reply), and sends it back to the requesting host.

6.4.3 Proxy ARP

Proxy ARP enables a router to answer ARP requests made to a destination node that is not on the same subnet as the requesting node. Assume that a router connects two subnets, A and B. If host A1 on subnet A tries to send an ARP request to host B1 on subnet B, this would normally not work as an ARP can only be performed between hosts on the same subnet (where all hosts can 'see' and respond to the FF:FF:FF:FF:FF:FF broadcast MAC address). The requesting host, A1, would therefore not get a response.

If proxy ARP has been enabled on the router, it will recognize this request and issue its own ARP request, on behalf of A1, to B1. Upon obtaining a response from B1, it would report back to A1 on behalf of B1. It must be understood that the MAC address returned to A1 will not be that of B1, but rather that of the router NIC connected to subnet A, as this is the physical address where A1 will send data destined for B1.

6.4.4 Gratuitous ARP

Gratuitous ARP occurs when a host sends out an ARP request looking for its own address. This is normally done at the time of boot-up. This can be used for two purposes.

Firstly, a host would not expect a response to the request. If a response does appear, it means that another host with a duplicate IP address exists on the network.

Secondly, any host observing an ARP request broadcast will automatically update its own ARP cache if the information pertaining to the destination node already exists in its cache. If a specific host is therefore powered down and the NIC replaced, all other hosts with the powered down host's IP address in their caches will update when the host in question is re-booted.

6.5 Reverse address resolution protocol (RARP)

As its name suggests, reverse address resolution protocol (RARP) (RFC 903) does the opposite to ARP. It is used to obtain an IP address when the physical address is known.

Usually, a machine holds its own IP address on its hard drive, where the operating system can find it on startup. However, a diskless workstation is only aware of its own hardware address and has to recover its IP address from an address file on a remote server at startup. It uses RARP to retrieve its IP address.

A diskless workstation broadcasts an RARP request on the local network using the same datagram format as an ARP request. It has, however, an opcode of 3 (RARP request), and identifies itself as both the sender and the target by placing its own physical address in both the sender hardware address field and the target hardware address field. Although the RARP request is broadcast, only a RARP server (i.e. a machine holding a table of addresses and programmed to provide RARP services) can generate a reply. There should be at least one RARP server on a network, often there are more.

The RARP server changes the opcode to 4 (RARP reply). It then inserts the missing address in the target IP address field, and sends the reply directly back to the requesting machine. The requesting machine then stores it in memory until next time it reboots.

All RARP servers on a network will reply to a RARP request, even though only one reply is required. The RARP software on the requesting machine sets a timer when sending a request and retransmits the request if the timer expires before a reply has been received.

On a best-effort local area network, such as Ethernet, the provision of more than one RARP server reduces the likelihood of RARP replies being lost or dropped because the server is down or overloaded. This is important because a diskless workstation often requires its own IP address before it can complete its bootstrap procedure. To avoid multiple and unnecessary RARP responses on a broadcast-type network such as Ethernet, each machine on the network is assigned a particular server, called its primary RARP server. When a machine broadcasts a RARP request, all servers will receive it and record its time of arrival, but only the primary server for that machine will reply. If the primary server is unable to reply for any reason, the sender's timer will expire, it will rebroadcast its request and all non-primary servers receiving the rebroadcast so soon after the initial broadcast will respond.

Alternatively, all RARP servers can be programmed to respond to the initial broadcast, with the primary server set to reply immediately, and all other servers set to respond after a random time delay. The retransmission of a request should be delayed for long enough for these delayed RARP replies to arrive.

RARP has several drawbacks. It has to be implemented as a server process. It is also prudent to have more than one server, since no diskless workstation can boot up if the single RARP server goes down. In addition to this, very little information (only an IP address) is returned. Finally, RARP uses a MAC address to obtain an IP address, hence it cannot be routed.

6.6 Internet control message protocol (ICMP)

Errors occur in all networks. These arise when destination nodes fail, or become temporarily unavailable, or when certain routes become overloaded with traffic. A message mechanism called the **Internet control message protocol** (ICMP) is incorporated into the TCP/IP protocol suite to report errors and other useful information about the performance and operation of the network.

6.6.1 ICMP message structure

ICMP communicates between the Internet layers on two nodes and is used by both gateways (routers) and individual hosts. Although ICMP is viewed as residing within the Internet layer, its messages travel across the network encapsulated in IP datagrams in the same way as higher layer protocol (such as TCP or UDP) datagrams. This is done with the protocol field in the IP header set to 0x1, indicating that an ICMP datagram is being carried. The reason for this approach is that, due to its simplicity, the ICMP header does not include any IP address information and is therefore in itself not routable. It therefore has little choice but to rely on IP for delivery. The ICMP message, consisting of an ICMP header and ICMP data, is encapsulated as 'data' within an IP datagram with the resultant structure indicated in the figure below.

The complete IP datagram, in turn, has to depend on the lower network interface layer (for example, Ethernet) and is thus contained as payload within the Ethernet data area.

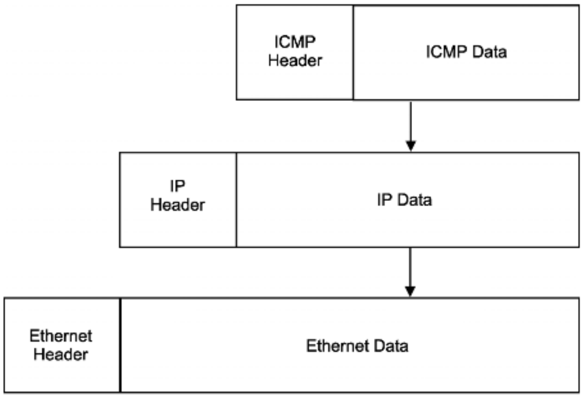


Figure 6.28
Encapsulation of the ICMP message

6.6.2 ICMP applications

The various uses for ICMP include:

- Exchanging messages between hosts to synchronize clocks
- Exchanging subnet mask information
- Informing a sending node that its message will be terminated due to an expired TTL
- Determining whether a node (either host or router) is reachable
- Advising routers of better routes
- Informing a sending host that its messages are arriving too fast and that it should back off

There are a variety of ICMP messages, each with a different format, yet the first 3 fields as contained in the first 4 bytes or 'long word' are the same for all.

The overall ICMP message structure is given in Figure 6.29.

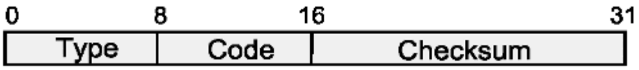


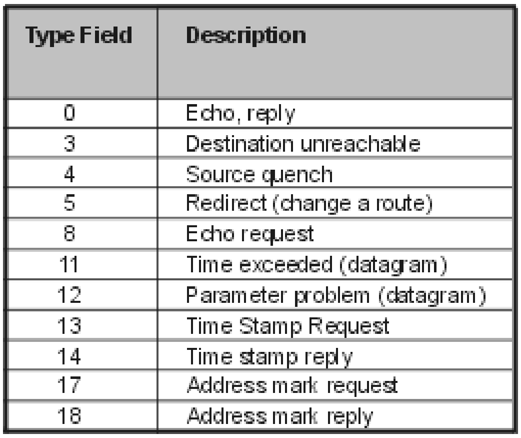
Figure 6.29
General ICMP message format

The three common fields are:

- **ICMP message type**

  A code that identifies the type of ICMP message
- **Code**

  A code in which interpretation depends on the type of ICMP message
- **Checksum**
  A 16-bit checksum that is calculated on the entire ICMP datagram



Type Field	Description
0	Echo, reply
3	Destination unreachable
4	Source quench
5	Redirect (change a route)
8	Echo request
11	Time exceeded (datagram)
12	Parameter problem (datagram)
13	Time Stamp Request
14	Time stamp reply
17	Address mark request
18	Address mark reply

Table 6.30
ICMP message types

ICMP messages can be further subdivided into two broad groups viz. ICMP error messages and ICMP query messages as follows.

ICMP error messages

- Destination unreachable
- Time exceeded
- Invalid parameters
- Source quench
- Redirect

ICMP query messages

- Echo request and reply messages
- Time-stamp request and reply messages
- Subnet mask request and reply messages

Too many ICMP error messages in the case of a network experiencing errors due to heavy traffic can exacerbate the problem, hence the following conditions apply:

- No ICMP messages are generated in response to ICMP messages
- No ICMP error messages are generated for multicast frames
- ICMP error messages are only generated for the first frame in a series of segments

Here follows a few examples of ICMP error messages.

6.6.3 Source quench

If a gateway (router) receives a high rate of datagrams from a particular source it will issue a source quench ICMP message for every datagram it discards. The source node will then slow down its rate of transmission until the source quench messages stop; at which stage it will gradually increase the rate again.

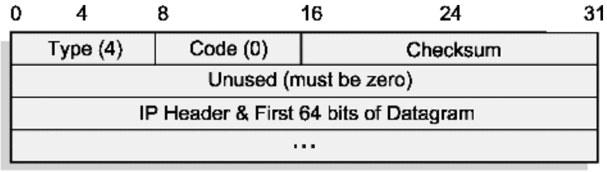


Figure 6.31
Source quench message format

Apart from the first 3 fields, already discussed, the header contains the following additional fields:

- **Original IP datagram header**

  The IP header of the datagram that led to the generation of this message
- **Original IP datagram data**

  The first 8 bytes of the data portion of the datagram that led to the generation of this message. This is for identification purposes

6.6.4 Redirection messages

When a gateway (router) detects that a source node is not using the best route in which to transmit its datagram, it sends a message to the node advising it of the better route.

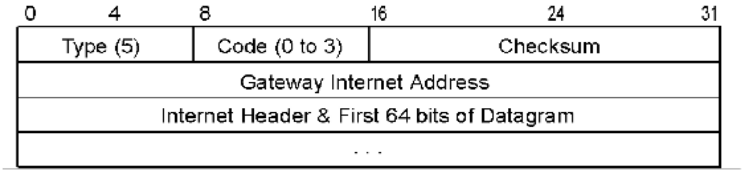


Figure 6.32
Redirect message format

Apart from the first 3 fields, already discussed, the header contains the following additional fields:

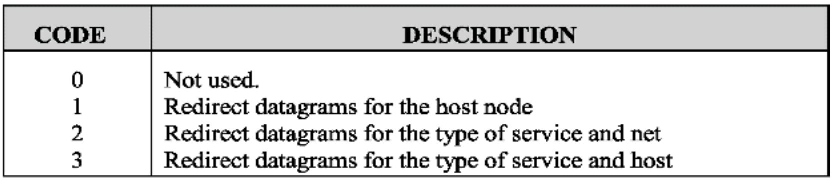
- **Router Internet address**

  The IP address of the router that needs to update its routing tables
- **Original IP datagram header**

  The IP header of the datagram that led to the generation of this message
- **Original IP datagram data**

  The first 8 bytes of the data portion of the datagram that led to the generation of this message. This is for identification purposes

The code values are as follows.



CODE	DESCRIPTION
0	Not used.
1	Redirect datagrams for the host node
2	Redirect datagrams for the type of service and net
3	Redirect datagrams for the type of service and host

Figure 6.33
Table of code values

6.6.5 Time exceeded messages

If a datagram has traversed too many routers, its TTL (time to live) counter will eventually reach a count of zero. The ICMP time exceeded message is then sent back to the source node. The time exceeded message will also be generated if one of the fragments of a fragmented datagram fails to arrive at the destination node within a given time period and as a result the datagram cannot be reconstructed.

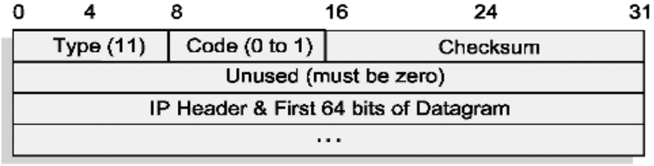


Figure 6.34
Time exceeded message structure

The code field is then as follows.

CODE	DESCRIPTION
0	Time to live count exceeded
1	Fragment re-assembly time exceeded

Figure 6.35
Table of code values

Code 1 refers to the situation where a gateway waits to reassemble a few fragments and a fragment of the datagram never arrives at the gateway.

6.6.6 Parameter problem messages

When there are problems with a particular datagram's contents, a parameter problem message is sent to the original source. The pointer field points to the problem bytes. (Code 1 is only used to indicate that a required option is missing – the pointer field is not used here.)

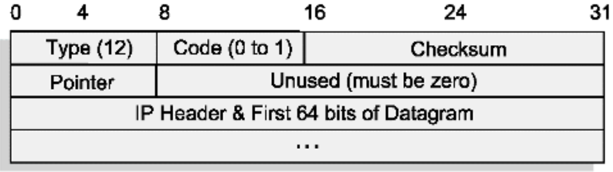


Figure 6.36
Parameter problem

6.6.7 Unreachable destination

When a gateway is unable to deliver a datagram, it responds with this message. The datagram is then 'dropped' (deleted).

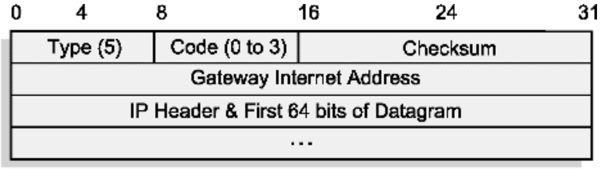
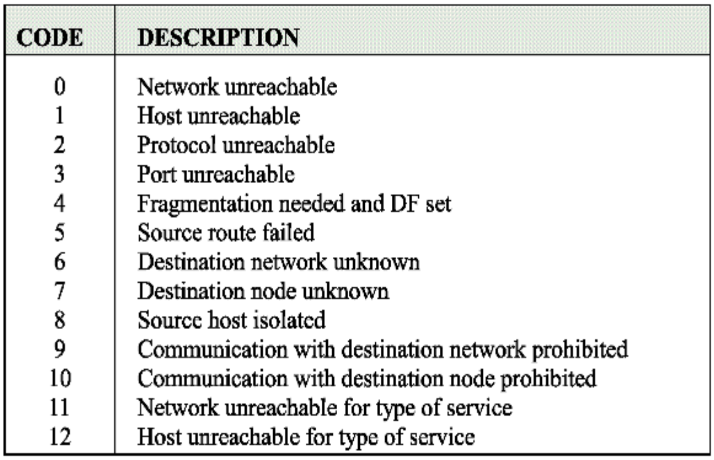


Figure 6.37
ICMP destination unreachable message

The values relating the code values in the above unreachable message are as follows.



CODE	DESCRIPTION
0	Network unreachable
1	Host unreachable
2	Protocol unreachable
3	Port unreachable
4	Fragmentation needed and DF set
5	Source route failed
6	Destination network unknown
7	Destination node unknown
8	Source host isolated
9	Communication with destination network prohibited
10	Communication with destination node prohibited
11	Network unreachable for type of service
12	Host unreachable for type of service

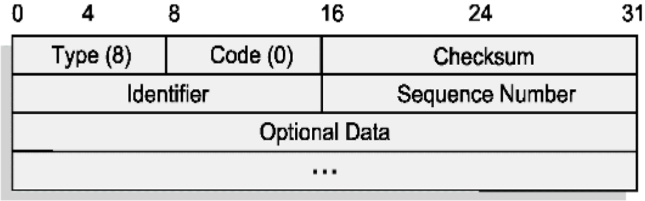
Figure 6.38
Typical code messages

6.6.8 ICMP query messages

In addition to the reports on errors and exceptional conditions, there is a set of ICMP messages to request information, and to reply to such request.

Echo request and reply

An echo request message is sent to the destination node. This message essentially enquires: ‘Are you alive?’ A reply indicates that the pathway (i.e. the network(s) in between, the gateways (routers)) and the destination node are all operating correctly. The structure of the request and reply are indicated below.



Type (8)	Code (0)	Checksum
Identifier		Sequence Number
Optional Data		
...		

Figure 6.39
ICMP echo request and reply

The first three fields have already been discussed. The additional fields are:

- **Type**

  8 for an echo request, and 0 for a reply
- **Identifier**

  A 16-bit random number, used to match a reply message with its associated request message
- **Sequence number**

  Used to identify each individual request or reply in a sequence of associated requests or replies with the same source and destination

- **Data**

  Generated by the sender and echoed back by the echoer. This field is variable in length; its length and contents are set by the echo request sender. It usually consists of the ASCII characters a, b, c, d, etc

Time-stamp request and replies

This can be used to estimate to synchronize the clock of a host with that of a timeserver.

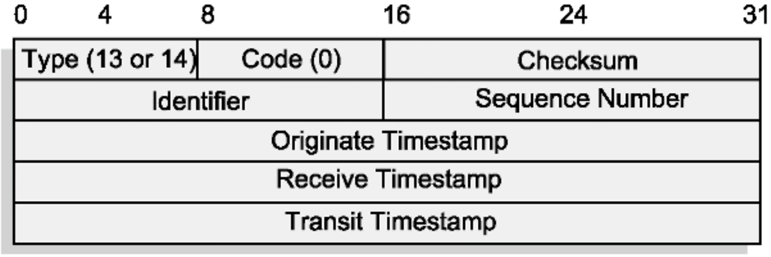


Figure 6.40
Structure of the time stamp request and reply

- **Type**

  13 for time-stamp request and 14 for time-stamp reply message
- **Originate time-stamp**

  Generated by sender and contains a time value identifying the time the initial time-stamp request was sent
- **Receive time-stamp**

  Generated by the echoer and contains the time the original time-stamp was received
- **Transmit time-stamp**

  Generated by the echoer and contains a value identifying the time the time-stamp reply message was sent.

The ICMP time-stamp request and reply enables a client to adjust its clock against an accurate server. The times referred to hereunder 32-bit integers, measured in milliseconds since midnight, **Co-ordinated Universal Time** (UCT). (Previously known as **Greenwich Mean Time** (GMT)).

The adjustment is initiated by the client inserting its current time in the ‘originate’ field, and sending the ICMP datagram off to the server. The server, upon receiving the message, then inserts the ‘received’ time in the appropriate field.

The server then inserts its current time in the ‘transmit’ field and returns the message. In practice, the ‘received’ and ‘transmit’ fields for the server are set to the same value.

The client, upon receiving the message back, records the ‘present’ time (albeit not within the header structure). It then deducts the ‘originate’ time from the ‘present’ time. Assuming negligible delays at the server, this is the time that the datagram took to travel to the server and back, or the round trip time (RTT). The time to the server is then one-half of this.

The correct time at the moment of originating the message at the client is now calculated by subtracting the RTT from the ‘transmit’ time-stamp created by the server. The client can now calculate its error by the relationship between the ‘originate’ time-stamp and the actual time, and adjust its clock accordingly. By repeated application of this procedure all hosts on a LAN can maintain their clocks to within less than a millisecond of each other.

Subnet mask request and reply

This is used to implement a simple client-server protocol that a host can use to obtain the correct subnet mask. Where implemented, one or more hosts in the internetwork are designated as subnet mask servers and run a process that replies to subnet mask request, this field is set to zero.

6.7 Routing protocols

6.7.1 Routing basics

Unlike the host-to-host layer protocols (e.g. TCP), which control end-to-end communications, the Internet layer protocol (IP) is rather 'short-sighted'. Any given IP node (host or router) is only concerned with routing (switching) the datagram to the *next* node, where the process is repeated. Very few routers have knowledge about the entire internetwork, and often the datagrams are forwarded based on default information without any knowledge of where the destination actually is.

Before discussing the individual routing protocols in any depth, the basic concepts of IP routing have to be clarified. This section will discuss the concepts and protocols involved in routing, while the routers themselves will be discussed in Chapter 10.

6.7.2 Direct vs indirect delivery

Refer to Figure 6.41. When the source host prepares to send a message to another host, a fundamental decision has to be made, namely: is the destination host also resident on the local network or not? If the NetID portions of the IP address match, the source host will assume that the destination host is resident on the same network, and will attempt to forward it locally. This is called direct delivery.

If not, the message will be forwarded to the local default gateway of a local router, which will forward it. This is called indirect delivery. The process will now be repeated. If the router can deliver it directly i.e. the host resides on a network directly connected to the router, it will. If not, it will consult its routing tables and forward it to the next appropriate router.

This process will repeat itself until the packet is delivered to its final destination.

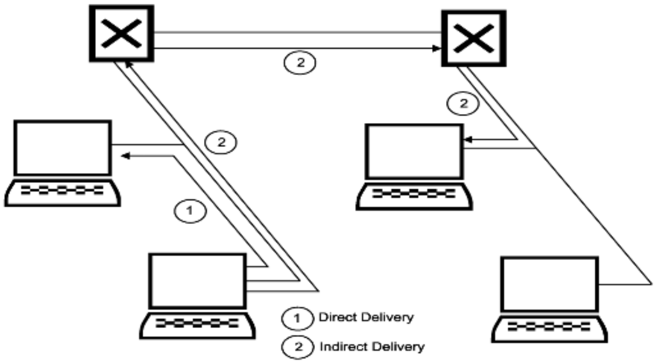


Figure 6.41
Direct vs indirect delivery

6.7.3 Static versus dynamic routing

Each router has a table with the following format:

```
Active routes for 207.194.66.100:
Network address    Netmask            Gateway address    Interface          Metric
127.0.0.0          255.0.0.0          127.0.0.1          127.0.0.1            1
207.194.66.0       255.255.255.224    207.194.66.100     207.194.66.100       1
207.194.66.0       255.255.255.255    127.0.0.1          127.0.0.1            1
207.194.66.255     255.255.255.255    207.194.66.100     207.194.66.100       1
224.0.0.0          224.0.0.0          207.194.66.100     207.194.66.100       1
255.255.255.255    255.255.255.255    207.194.66.100     0.0.0.0              1
C:\WINDOWS.000>
```

It basically reads as follows: 'If a packet is destined for network 207.194.66.0, with a Netmask of 255.255.255.224, then forward it to the router port: 207.194.66.100', etc. It is logical that a given router cannot contain the whereabouts of each and every network in the world in its routing tables; hence it will contain default routes as well. If a packet cannot be specifically routed, it will be forwarded on a default route, which should (it is hoped) move it closer to its intended destination.

These routing tables can be maintained in two ways. In most cases, the routing protocols will do this automatically. The routing protocols are implemented in software that runs on the routers, enabling them to communicate on a regular basis and allowing them to share their 'knowledge' about the network with each other. In this way they continuously 'learn' about the topology of the system, and upgrade their routing tables accordingly. This process is called *dynamic* routing. If, for example, a particular router is removed from the system, the routing tables of all routers containing a reference to that router will change. However, because of the interdependence of the routing tables, a change in any given table will initiate a change in many other routers and it will be a while before the tables stabilize. This process is known as **convergence**.

Dynamic routing can be further sub-classified as distance vector, link-state, or hybrid-depending on the method by which the routers calculate the optimum path.

In distance vector dynamic routing, the 'metric' or yardstick used for calculating the optimum routes is simply based on distance, i.e. which route results in the least number of 'hops' to the destination. Each router constructs a table, which indicates the number of hops to each known network. It then periodically passes copies of its tables to its immediate neighbors. Each recipient of the message then simply adjusts its own tables based on the information received from its neighbor.

The major problem with the distance vector algorithm is that it takes some time to converge to a new understanding of the network. The bandwidth and traffic requirements of this algorithm can also affect the performance of the network. The major advantage of the distance vector algorithm is that it is simple to configure and maintain as it only uses the distance to calculate the optimum route.

Link state routing protocols are also known as shortest path first protocols. This is based on the routers exchanging link state advertisements to the other routers. Link state advertisement messages contain information about error rates and traffic densities and are triggered by events rather than running periodically as with the distance routing algorithms.

Hybridized routing protocols use both the methods described above and are more accurate than the conventional distance vector protocols. They converge more rapidly to an understanding of the network than distance vector protocols and avoid the overheads of the link state updates. The best example of this one is the enhanced interior routing protocol (EIGRP).

It is also possible for a network administrator to make *static* entries into routing tables. These entries will not change, even if a router that they point to is not operational.

6.7.4 Autonomous systems

For the purpose of routing a TCP/IP-based internetwork can be divided into several autonomous systems (ASs) or domains. An autonomous system consists of hosts, routers and data links that form several physical networks that are administered by a single authority such as a service provider, university, corporation, or government agency.

Autonomous systems can be classified under one of three categories:

- **Stub AS**

  This is an AS that has only one connection to the 'outside world' and therefore does not carry any third-party traffic. This is typical of a smaller corporate network
- **Multi-homed non-transit AS**

  This is an AS that has two or more connections to the 'outside world' but is not setup to carry any third party traffic. This is typical of a larger corporate network
- **Transit AS**

  This is an AS with two or more connections to the outside world, and is set up to carry third party traffic. This is typical of an ISP network

Routing decisions that are made within an autonomous system (AS) are totally under the control of the administering organization. Any routing protocol, using any type of routing algorithm, can be used within an autonomous system since the routing between two hosts in the system is completely isolated from any routing that occurs in other Autonomous systems. Only if a host within one autonomous system communicates with a host outside the system, will another autonomous system (or systems) and possibly the Internet backbone be involved.

6.7.5 Interior, exterior and gateway to gateway protocols

There are three categories of TCP/IP gateway protocols, namely interior gateway protocols, exterior gateway protocols, and gateway-to-gateway protocols.

Two routers that communicate directly with one another and are both part of the same autonomous system are said to be interior neighbors and are called interior gateways. They communicate with each other using interior gateway protocols.

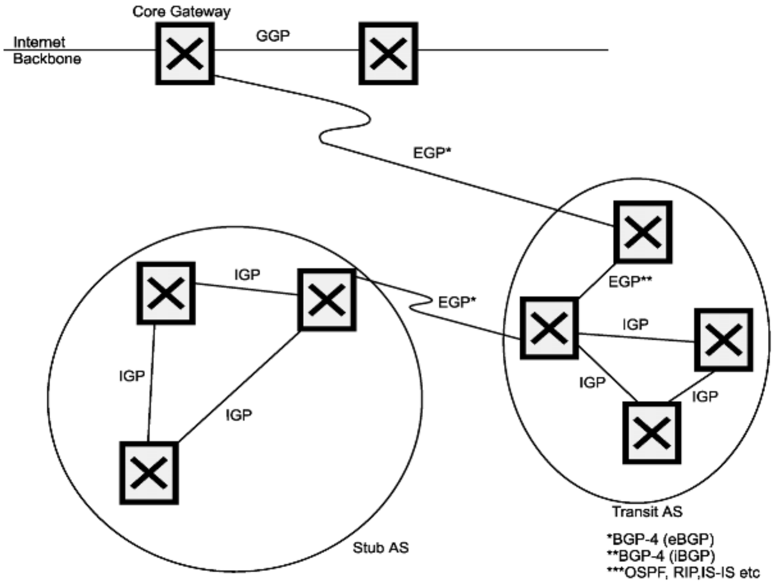


Figure 6.42
Application of routing protocols

In a simple AS consisting of only a few physical networks, the routing function provided by IP may be sufficient. In larger ASs, however, sophisticated routers using adaptive routing algorithms may be needed. These routers will communicate with each other using interior gateway protocols such as RIP, Hello, IS-IS or OSPF.

Routers in different ASs, however, cannot use IGPs for communication for more than one reason. Firstly, IGPs are not optimized for long-distance path determination. Secondly, the owners of ASs (particularly Internet service providers) would find it unacceptable for their routing metrics (which include sensitive information such as error rates and network traffic) to be visible to their competitors. For this reason routers that communicate with each other and are resident in different ASs communicate with each other using exterior gateway protocols.

The routers on the periphery, connected to other ASs, must be capable of handling both the appropriate IGPs and EGPs.

The most common exterior gateway protocol currently used in the TCP/IP environment is border gateway patrol (BGP), the current version being BGP-4.

A third type of routing protocol is used by the core routers (gateways) that connect users to the Internet backbone. They use gateway to gateway protocols (GGP) to communicate with each other.

6.8 Interior gateway protocols

The protocols that will be discussed are **RIPv2** (routing information protocol version 2), **EIGRP** (enhanced interior gateway routing protocol), and **OSPF** (open shortest path first).

RIPv2

RIPv2 originally saw the light as RIP (RFC 1058, 1388) and is one of the oldest routing protocols. The original RIP had a shortcoming in that it could not handle variable-length subnet masks, and hence could not support CIDR. This capability has been included with RIPv2.

RIPv2 is a distance vector routing protocol where each router, using a special packet to collect and share information about distances, keeps a routing table of its perspective of the network showing the number of hops required to reach each network. RIP uses as a metric (i.e. form of measurement) the hop counts.

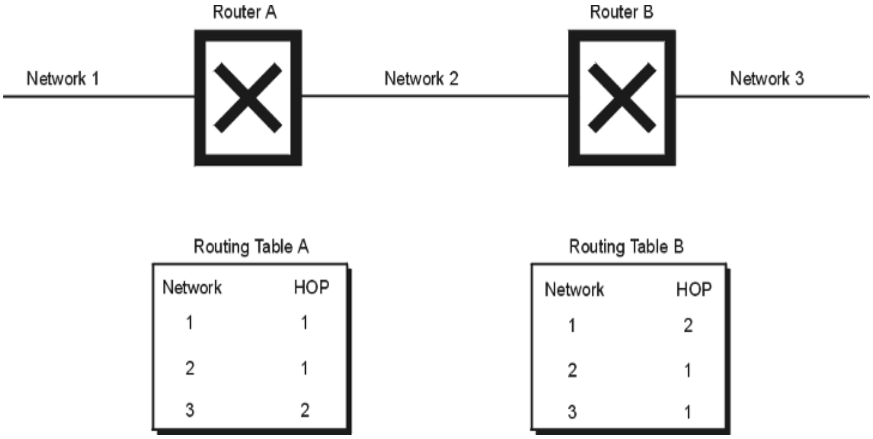
In order to maintain their individual perspective of the network, routers periodically pass copies of their routing tables to their immediate neighbors. Each recipient adds a distance vector to the table and forwards the table to its immediate neighbors. The hop count is incremented by one every time the packet passes through a router. RIP only records one route per destination (even if there are more).

The Figure 6.43 shows a sample network and the relevant routing tables.

The RIP routers have fixed update intervals and each router broadcasts its entire routing table to other routers at 30-second intervals (60 seconds for netware RIP). Each router takes the routing information from its neighbor, adds or subtracts one hop to the various routes to account for itself, and then broadcasts its updated table.

Every time a router entry is updated, the timeout value for the entry is reset. If an entry has not been updated within 180 seconds it is assumed suspect and the hop field set to 16 to mark the route as unreachable and it is later removed from the routing table.

One of the major problems with distance vector protocols like RIP is the **convergence time**, which is the time it takes for the routing information on all routers to settle in response to some change to the network. For a large network the convergence time can be long and there is a greater chance of frames being misrouted.



Routing Table A

Network	HOP
1	1
2	1
3	2

Routing Table B

Network	HOP
1	2
2	1
3	1

Figure 6.43
RIP tables

RIPv2 (RFC1723) also supports:

- **Authentication**

  This prevents a routing table from being corrupted with incorrect data from a bad source

- **Subnet masks**

  The IP address and its subnet mask enable the RIPv2 to identify the type of destination that the route leads to. This enables it to discern the network subnet from the host address

- **IP identification**

  This makes RIPv2 more effective than RIP as it prevents unnecessary hops. This is useful where multiple routing protocols are used simultaneously and some routes may never be identified. The IP address of the next hop router would be passed to neighboring routers via routing table updates. These routers would then force datagrams to use a specific route whether or not that route had been calculated to be the optimum route or not using least hop count

- **Multicasting of RIPv2 messages**

  This is a method of simultaneously advertising routing data to multiple RIP or RIPv2 devices. This is useful when multiple destinations must receive identical information

EIGRP

EIGRP is an enhancement of the original IGRP, a proprietary routing protocol developed by Cisco Systems for use on the Internet. IGRP is outdated since it cannot handle CIDR and variable-length subnet masks.

EIGRP is a distance vector routing protocol that uses a composite metric for route calculations. It allows for multipath routing, load balancing across 2, 3 or 4 links, and automatic recovery from a failed link. Since it does not only take hop count into consideration, it has better real-time appreciation of the link status between routers and is more flexible than RIP. Like RIP it broadcasts whole routing table updates, but at 90 second intervals.

Each of the metrics used in the calculation of the distance vectors has a weighting factor. The metrics used in the calculation are as follows:

- Hop count. Unlike RIP, EIGRP does not stop at 16 hops and can operate up to a maximum of 255
- Packet size (maximum transmission unit or MTU)
- Link bandwidth
- Delay
- Loading
- Reliability

The metric used is:

Metric = K1 * bandwidth + (K2 * bandwidth)/(256 – Load) + K3 * Delay

(K1, K2 and K3 are weighting factors.)

Reliability is also added in using the metric:

$$\textbf{Metric}_{\textbf{modified}} \textbf{ = Metric * K5/(reliability + K4)}$$

This modifies the existing metric calculated in the first equation above.

One of the key design parameters of EIGRP is complete independence from routed protocols. Hence EIGRP has implemented a modular approach to supporting routed protocols and can easily be retrofitted to support any other routed protocol.

OSPF

This was designed specifically as an IP routing protocol, hence it cannot transport IPX or Appletalk protocols. It is encapsulated directly in the IP protocol. OSPF can quickly detect topological changes by flooding link state advertisements to all the other neighbors with reasonably quick convergence.

OSPF is a link state routing or shortest path first (SPF) protocol detailed in RFCs 1131, 1247 and 1583. Here each router periodically uses a broadcast mechanism to transmit information to all other routers, about its own directly connected routers and the status of the data links to them. Based on the information received from all the other routers each router then constructs its own network routing tree using the shortest path algorithm.

These routers continually monitor the status of their links by sending packets to neighboring routers. When the status of a router or link changes, this information is broadcast to the other routers that then update their routing tables. This process is known as flooding and the packets sent are very small representing only the link state changes.

Using cost as the metric OSPF can support a much larger network than RIP, which is limited to 15 routers. A problem area can be in mixed RIP and OSPF environments if routers go from RIP to OSPF and back when hop counts are not incremented correctly.

6.9 Exterior gateway protocols (EGPs)

One of the earlier EGPs was, in fact called EGP! The current *de facto* Internet standard for inter-domain (AS) routing is border gateway patrol version 4, or simply BGP-4.

6.9.1 BGP-4

BGP-4, as detailed in RFC 1771, performs intelligent route selection based on the shortest autonomous system path. In other words, whereas interior gateway protocols such as RIP make decisions on the number of ROUTERS to a specific destination, BGP-4 bases its decisions on the number of AUTONOMOUS SYSTEMS to a specific destination. It is a so-called path vector protocol, and runs over TCP (port 179).

BGP routers in one autonomous system speak BGP to routers in other autonomous systems, where the ‘other’ autonomous system might be that of an Internet service provider, or another corporation. Companies with an international presence and a large, global WAN, may also opt to have a separate AS on each continent (running OSPF internally) and run BGP between them in order to create a clean separation.

GGP comes in two ‘flavors’ namely ‘internal’ BGP (iBGP) and ‘external BGP’ (eBGP). IBGP is used within an AS and eBGP between ASs. In order to ascertain which one is used between two adjacent routers, one should look at the AS number for each router. BGP uses a formally registered AS number for entities that will advertise their presence in the Internet. Therefore, if two routers share the same AS number, they are probably using iBGP and if they differ, the routers speak eBGP. Incidentally, BGP routers are referred to as ‘BGP speakers’, all BGP routers are ‘peers’, and two adjacent BGP speakers are ‘neighbors.’

The range of non-registered (i.e. private) AS numbers is 64512–65535 and ISP typically issues these to stub ASs i.e. those that do not carry third-party traffic.

As mentioned earlier, iBGP is the form of BGP that exchanges BGP updates within an AS. Before information is exchanged with an external AS, iBGP ensures that networks within the AS are reachable. This is done by a combination of ‘peering’ between BGP routers within the AS and by distributing BGP routing information to IGPs that run within the AS, such as EIGRP, IS-IS, RIP or OSPF. Note that, within the AS, BGP peers do not have to be directly connected as long as there is an IGP running between them. The routing information exchanged consists of a series of AS numbers that describe the full path to the destination network. This information is used by BGP to construct a loop-free map of the network.

In contrast with iBGP, eBGP handles traffic between routers located on **DIFFERENT ASs.** It can do load balancing in the case of multiple paths between two routers. It also has a synchronization function that, if enabled, will prevent a BGP router from forwarding remote traffic to a transit AS before it has been established that all internal non-BGP routers within that AS are aware of the correct routing information. This is to ensure that packets are not dropped in transit through the AS.

7

Host-to-host (transport) layer protocols

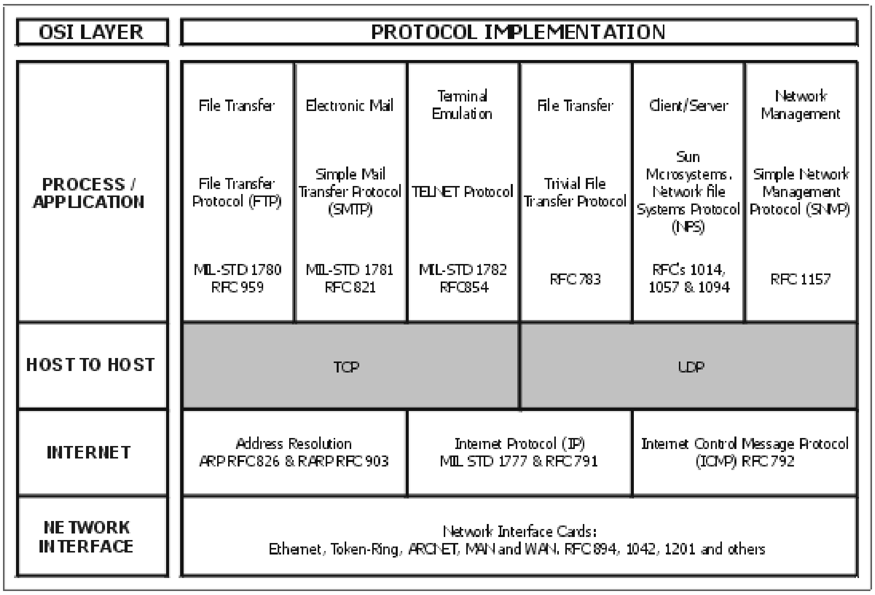
Objectives

When you have completed this chapter you should be able to:

- Explain the basic functions of the host-to-host layer
- Explain the basic operation of TCP and UDP
- Explain the fundamental differences between TCP and UDP
- Decide which protocol (TCP or UDP) to use for a particular application
- Explain the meaning of each field in the TCP and UDP headers

The host-to-host communications layer (also referred to as the service layer, or as the transport layer in terms of the OSI model) is primarily responsible for ensuring end-to-end delivery of packets transmitted by the **Internet protocol** (IP). This additional reliability is needed to compensate for the lack of reliability in IP.

There are only two relevant protocols residing in the host-to-host communications layer, namely **TCP** (transmission control protocol) and UDP (user datagram protocol). In addition to this, the host-to-host layer includes the APIs (application programming interfaces) used by programmers to gain access to these protocols from the process/ application layer.



OSI LAYER	PROTOCOL IMPLEMENTATION					
PROCESS / APPLICATION	File Transfer File Transfer Protocol (FTP) MIL-STD 1780 RFC 959	Electronic Mail Simple Mail Transfer Protocol (SMTP) MIL-STD 1781 RFC 821	Terminal Emulation TELNET Protocol MIL-STD 1782 RFC854	File Transfer Trivial File Transfer Protocol RFC 783	Client/Server Sun Microsystems, Network file Systems Protocol (NFS) RFCs 1014, 1057 & 1094	Network Management Simple Network Management Protocol (SNMP) RFC 1157
HOST TO HOST	TCP			UDP		
INTERNET	Address Resolution ARP RFC826 & RARP RFC 903		Internet Protocol (IP) MIL STD 1777 & RFC 791		Internet Control Message Protocol (ICMP) RFC 792	
NETWORK INTERFACE	Network Interface Cards: Ethernet, Token-Ring, ARCNET, MAN and WAN, RFC 894, 1042, 1201 and others					

Figure 7.1
TCP and UDP within the ARPA model

7.1 TCP (transmission control protocol)

7.1.1 Basic functions

TCP is a connection-oriented protocol and is therefore reliable, although this word is used in a data communications context and not in an everyday sense. TCP establishes a connection between two hosts before any data is transmitted. Because a connection is set up beforehand, it is possible to verify that all packets are received on the other end and to arrange re-transmission in the case of lost packets. Because of all these built-in functions, TCP involves significant additional overhead in terms of processing time and header size.

TCP includes the following functions:

- Fragmentation of large chunks of data into smaller segments that can be accommodated by IP. The word ‘segmentation’ is used here to differentiate it from the ‘fragmentation’ performed by IP
- Data stream reconstruction from packets received
- Receipt acknowledgment
- Socket services for providing multiple connections to ports on remote hosts
- Packet verification and error control
- Flow control
- Packet sequencing and reordering

In order to achieve its intended goals, TCP makes use of ports and sockets, connection oriented communication, sliding windows, and sequence numbers/acknowledgments.

7.1.2 Ports

Whereas IP can route the message to a particular machine on the basis of its IP address, TCP has to know for which process (i.e. software program) on that particular machine it is destined. This is done by means of port numbers ranging from 1 to 65 535.

Port numbers are controlled by IANA (the Internet Assigned Numbers Authority) and can be divided into three groups.

Well known ports, ranging from 1 to 1023, have been assigned by IANA and are globally known to all TCP users. For example, HTTP uses port 80.

Registered ports are registered by IANA in cases where the port number cannot be classified as well known, yet it is used by a significant number of users. Examples are port numbers registered for Microsoft Windows or for specific types of PLCs. These numbers range from 1024 to 49 151, the latter being 75% of 65 536.

A third class of port numbers is known as ephemeral ports. These range from 49 151 to 65 535 and can be used by anyone on an ad-hoc basis.

7.1.3 Sockets

In order to identify both the location and application to which a particular packet is to be sent, the IP address (location) and port number (process) is combined into a functional address called a socket. The IP address is contained in the IP header and the port number is contained in the TCP or UDP header.

In order for any data to be transferred under TCP, a socket must exist both at the source and at the destination. TCP is also capable of creating multiple sockets to the same port.

7.1.4 Sequence numbers

A fundamental notion in the TCP design is that every BYTE of data sent over the TCP connection has a unique 32-bit sequence number. Of course this number cannot be sent along with every byte, yet it is nevertheless implied. However, the sequence number of the FIRST byte in each segment is included in the accompanying TCP header, for each subsequent byte that number is simply incremented by the receiver in order to keep track of the bytes.

Before any data transmission takes place, both sender and receiver (e.g. client and server) have to agree on the initial sequence numbers (ISNs) to be used. This process is described under ‘establishing a connection’.

Since TCP supports full duplex operation, both client and server will decide on their initial sequence numbers for the connection, even though data may only flow in one direction for that specific connection.

The sequence number, for obvious reasons, cannot start at 0 every time, as it will create serious problems in the case of short-lived multiple sequential connections between two machines. A packet with a sequence number from an earlier connection could easily arrive late, during a subsequent connection. The receiver will have difficulty in deciding whether the packet belongs to a former or to the current connection. It is easy to visualize a similar problem in real life. Imagine tracking a parcel carried by UPS if all UPS agents started issuing tracking numbers beginning with 0 every morning.

The sequence number is generated by means of a 32-bit software counter that starts at 0 during boot-up and increments at a rate of about once every 4 microseconds (although this varies depending on the operating system being used). When TCP establishes a connection, the value of the counter is read and used as the initial sequence number. This creates an apparently random choice of the initial sequence number.

At some point during a connection the counter could rollover from 65 535 and start counting from 0 again. The TCP software takes care of this.

7.1.5 Acknowledgment numbers

TCP acknowledges data received on a PER SEGMENT basis, although several consecutive segments may be acknowledged at the same time.

The acknowledgment number returned to the sender to indicate successful delivery equals the number of the last byte received +1, hence it points to the next expected sequence number. For example: 10 bytes are sent, with sequence number 33. This means that the first byte is numbered 33 and the last byte is numbered 42. If received successfully, an acknowledgment number (ACK) of 43 will be returned. The sender now knows that the data has been received properly, as it agrees with that number.

TCP does not issue selective acknowledgments, so if a specific segment contains errors, the acknowledgement number returned to the sender will point to the first byte in the defective segment. This implies that the segment starting with that sequence number, and all subsequent segments (even though they may have been transmitted successfully) have to be retransmitted.

From the previous paragraph it should be clear that a duplicate acknowledgement received by the sender means that there was an error in the transmission of one or more bytes following that particular sequence number.

Please note that the sequence number and the acknowledgment number in one header are NOT related at all. The former relates to outgoing data, the latter refers to incoming data. During the connection establishment phase the sequence numbers for both hosts are setup independently, hence these two numbers will never bear any resemblance to each other.

7.1.6 Sliding windows

Obviously there is a need to get some sort of acknowledgment back to ensure that there is a guaranteed delivery. This technique, called positive acknowledgment with retransmission, requires the receiver to send back an acknowledgment message within a given time. The transmitter starts a timer so that if no response is received from the destination node within a given time, another copy of the message will be transmitted. An example of this situation is given in Figure 7.2.

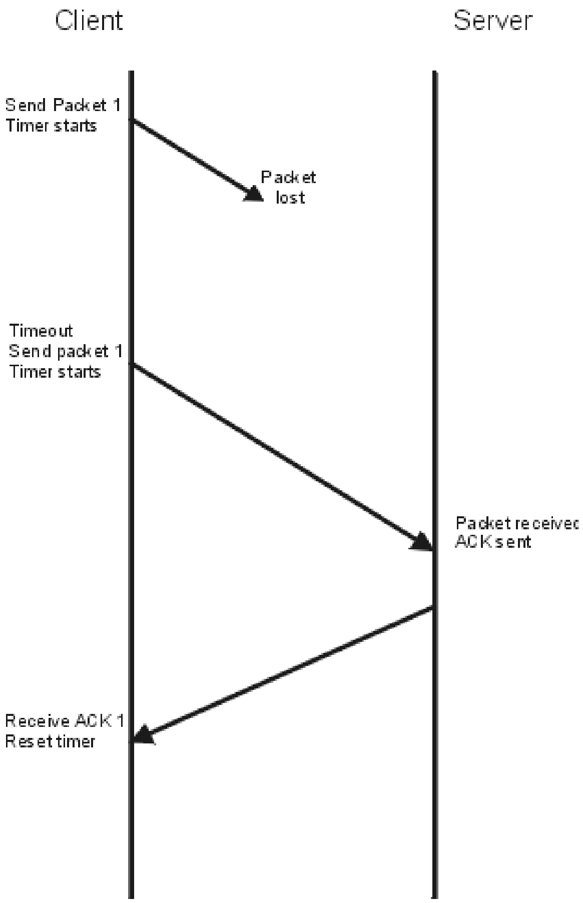


Figure 7.2
Positive acknowledgment philosophy

The sliding window form of positive acknowledgment is used by TCP, as it is very time consuming waiting for each individual acknowledgment to be returned for each packet transmitted. Hence the idea is that a number of packets (with cumulative number of bytes not exceeding the window size) are transmitted before the source may receive an acknowledgment to the first message (due to time delays, etc). As long as acknowledgments are received, the window slides along and the next packet is transmitted.

During the TCP connection phase each host will inform the other side of its permissible window size. For example, for Windows 95/98 this is typically 8K or around 8192 bytes. This means that, using Ethernet, 5 full data frames comprising 5 × 1460 = 7300 bytes can be sent without acknowledgment. At this stage the window size has shrunk to less than 1000 bytes, which means that unless an ACK is generated, the sender will have to pause its transmission.

7.1.7 Establishing a connection

A three-way SYN/ SYN_ACK/ACK handshake (as indicated in Figure 7.3) is used to establish a TCP connection. As this is a full duplex protocol it is possible (and necessary) for a connection to be established in both directions at the same time.

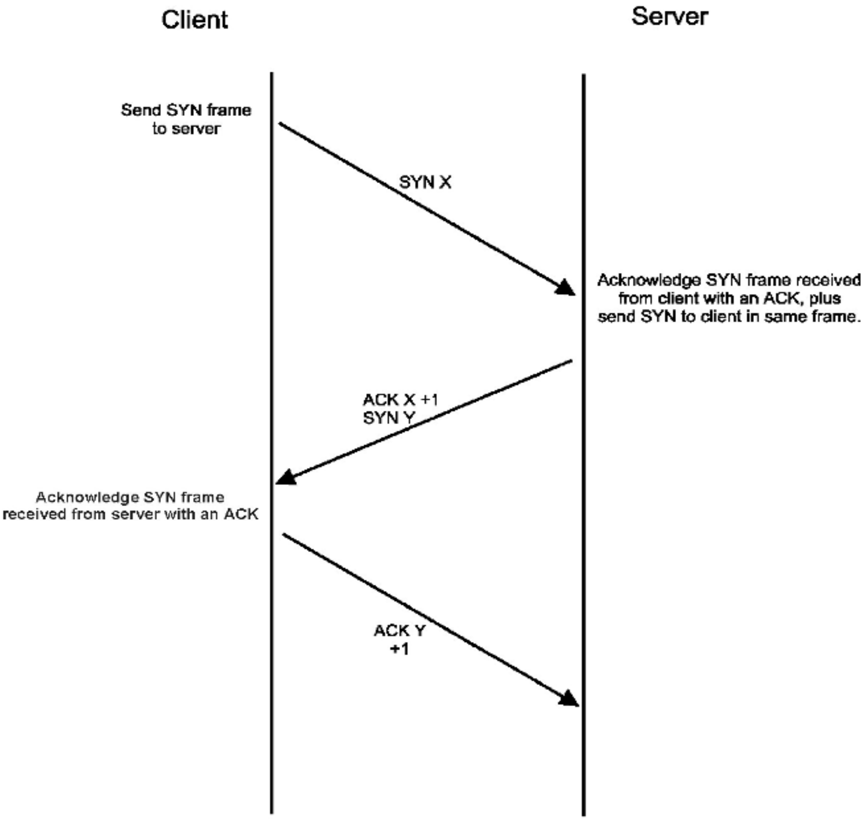

Client
Server
Send SYN frame to server
SYN X
Acknowledge SYN frame received from client with an ACK, plus send SYN to client in same frame.
ACK X +1
SYN Y
Acknowledge SYN frame received from server with an ACK
ACK Y +1

Figure 7.3
TCP connection establishment

As mentioned before, TCP generates pseudo-random sequence numbers by means of a 32-bit software counter that resets at boot-up and then increments every 4 microseconds. The host establishing the connection reads a value ‘x’ from the counter where x can vary between 0 and 2^{32} –1) and inserts it in the sequence number field. It then sets the SYN flag = 1 and transmits the header (no data yet) to the appropriate IP address and port number. Assuming that the chosen sequence number was 132, this action would then be abbreviated as SYN 132.

The receiving host (e.g. the server) acknowledges this by incrementing the received sequence number by one, and sending it back to the originator as an acknowledgment number. It also sets the ACK flag = 1 to indicate that this is an acknowledgment. This results in an ACK 133. The first byte expected would therefore be numbered 133. At the same time the server obtains its own sequence number (y), inserts it in the header, and also sets the SYN flag in order to establish a connection in the opposite direction. The header is then sent off to the originator (the client), conveying the message e.g. SYN 567. The composite ‘message’ contained within the header would thus be ACK 133, SYN 567.

The originator receives this, notes that its own request for a connection has been complied with, and also acknowledges the other node’s request with an ACK 568. Two-way communication is now established.

7.1.8 Closing a connection

An existing connection can be terminated in several ways.

Firstly, one of the hosts can request to close the connection by setting the FIN flag. The other host can acknowledge this with an ACK, but does not have to close immediately as it may need to transmit more data. This is known as a half-close. When the second host is also ready to close, it will send a FIN that is acknowledged with an ACK. The resulting situation is known as a full close.

Secondly, either of the nodes can terminate its connection with the issue of RST, resulting in the other node also relinquishing its connection and (although not necessarily) responding with an ACK.

Both situations are depicted in Figure 7.4.

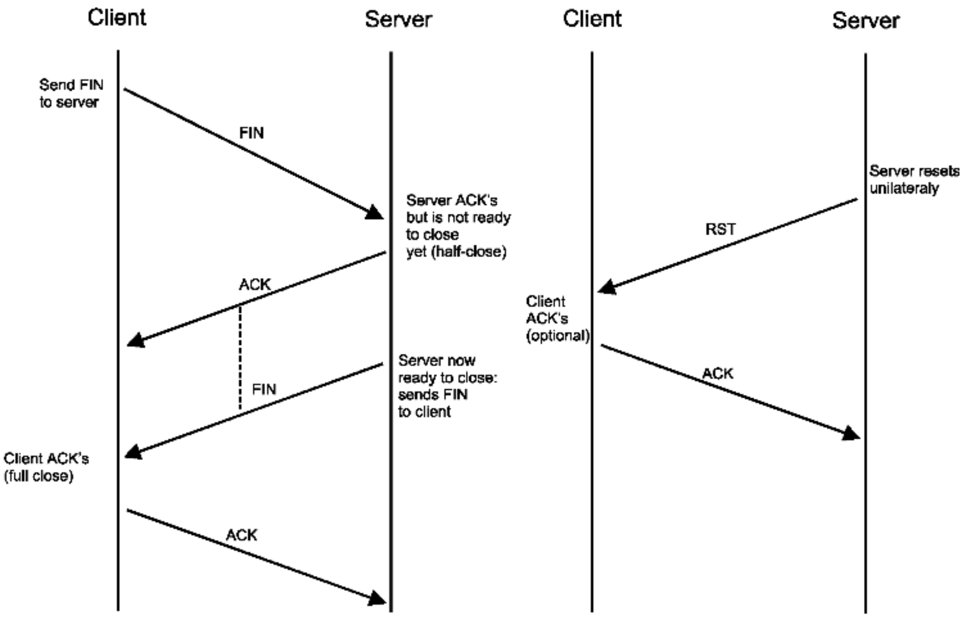

Client
Server
Client
Server
Send FIN to server
FIN
Server ACK's but is not ready to close yet (half-close)
ACK
FIN
Server now ready to close: sends FIN to client
Client ACK's (full close)
ACK
Server resets unilateraly
RST
Client ACK's (optional)
ACK

Figure 7.4
Closing a connection

7.1.9 The push operation

TCP normally breaks the data stream into what it regards are appropriately sized segments, based on some definition of efficiency. However, this may not be swift enough for an interactive keyboard application. Hence the push instruction (PSH bit in the code field) used by the application program forces delivery of bytes currently in the stream and the data will be immediately delivered to the process at the receiving end.

7.1.10 Maximum segment size

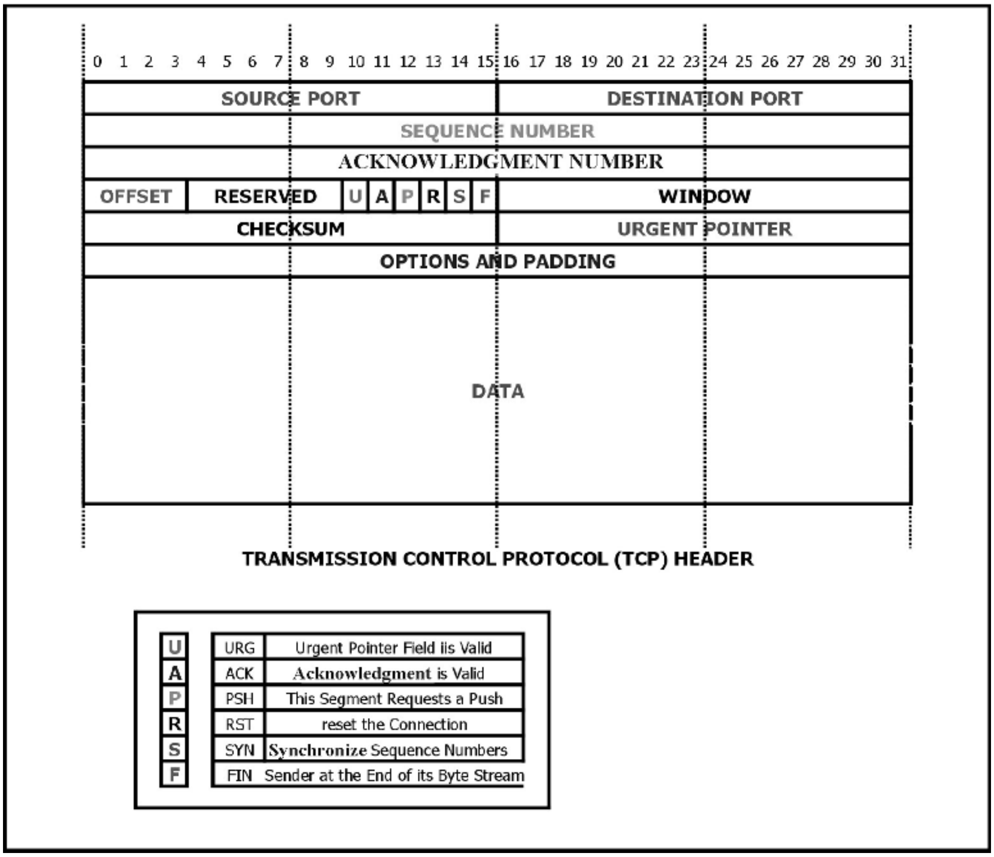
Both the transmitting and receiving nodes need to agree on the maximum size segments they will transfer. This is specified in the options field.

On the one hand TCP ‘prefers’ IP not to perform any fragmentation as this leads to a reduction in transmission speed due to the fragmentation process, and a higher probability of loss of a packet and the resultant retransmission of the entire packet.

On the other hand, there is an improvement in overall efficiency if the data packets are not too small and a maximum segment size is selected that fills the physical packets that are transmitted across the network. The current specification recommends a maximum segment size of 536 (this is the 576 byte default size of an X.25 frame minus 20 bytes each for the IP and TCP headers). If the size is not correctly specified, for example too small, the framing bytes (headers etc) consume most of the packet size resulting in considerable overhead. Refer to RFC 879 for a detailed discussion on this issue.

7.1.11 The TCP frame

The TCP Frame consists of a header plus data and is structured as follows:



U	URG	Urgent Pointer Field Iis Valid
A	ACK	Acknowledgment is Valid
P	PSH	This Segment Requests a Push
R	RST	reset the Connection
S	SYN	Synchronize Sequence Numbers
F	FIN	Sender at the End of its Byte Stream

Figure 7.5
TCP frame format

The various fields within the header are as follows:

Source port: 16 bits

The source port number.

Destination port: 16 bits

The destination port number.

Sequence number: 32 bits

The sequence number of the first data byte in the current segment, except when the SYN flag is set. If the SYN flag is set, a connection is still being established and the sequence number in the header is the initial sequence number (ISN). The first subsequent data byte is ISN+1.

Refer to the discussion on sequence numbers.

Acknowledgment number: 32 bits

If the ACK flag is set, this field contains the value of the next sequence number the sender of this message is expecting to receive. Once a connection is established this is always sent.

Refer to the discussion on acknowledgment numbers.

Data offset: 4 bits

The number of 32 bit words in the TCP header. (Similar to IHL in the IP header.) This indicates where the data begins. The TCP header (even one including options) is always an integral number of 32 bits long.

Reserved: 6 bits

Reserved for future use. Must be zero.

Control bits (flags): 6 bits

(From left to right)

URG: Urgent pointer field significant

ACK: Acknowledgment field significant

PSH: Push function

RST: Reset the connection

SYN: Synchronize sequence numbers

FIN: No more data from sender

Checksum: 16 bits

The checksum field is the 16-bit one's complement of the one's complement sum of all 16-bit words in the header and text. If a segment contains an odd number of header and text octets to be check-summed, the last octet is padded on the right with zeros to form a 16-bit word for checksum purposes. The pad is not transmitted as part of the segment. While computing the checksum, the checksum field itself is replaced with zeros.

This is known as the standard Internet checksum, and is the same as the one used for the IP header.

The checksum also covers a 96-bit 'pseudo header' conceptually appended to the TCP header. This pseudo header contains the source IP address, the destination IP address, the protocol number (06), and TCP length. It must be emphasized that this pseudo header is only used for computation purposes and is NOT transmitted. This gives TCP protection against misrouted segments.

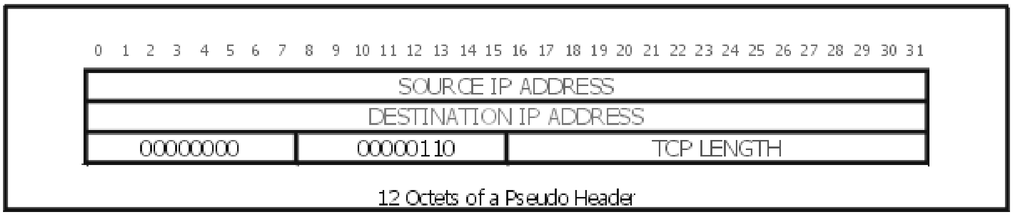


Figure 7.6
Pseudo TCP header format

Window: 16 bits

The number of data octets beginning with the one indicated in the acknowledgement field, which the sender of this segment is willing or able to accept.

Refer to the discussion on sliding windows.

Urgent pointer: Urgent data is placed in the beginning of a frame, and the urgent pointer points at the last byte of urgent data (relative to the sequence number i.e. the number of the first byte in the frame). This field is only being interpreted in segments with the URG control bit set.

Options: Options may occupy space at the end of the TCP header and are a multiple of 8 bits in length. All options are included in the checksum.

7.2 UDP (user datagram protocol)

7.2.1 Basic functions

The second protocol that occupies the host-to-host layer is UDP. As in the case of TCP, it makes use of the underlying IP protocol to deliver its datagrams.

UDP is a ‘connectionless’ or non-connection-oriented protocol and does not require a connection to be established between two machines prior to data transmission. It is therefore said to be an ‘unreliable’ protocol – the word ‘unreliable’ used here as opposed to ‘reliable’ in the case of TCP.

As in the case of TCP, packets are still delivered to sockets or ports. However, no connection is established beforehand and therefore UDP cannot guarantee that packets are retransmitted if faulty, received in the correct sequence, or even received at all. In view of this, one might doubt the desirability of such an unreliable protocol. There are, however, some good reasons for its existence.

Sending a UDP datagram involves very little overhead in that there are no synchronization parameters, no priority options, no sequence numbers, no retransmit timers, no delayed acknowledgement timers, and no retransmission of packets. The header is small; the protocol is quick, and streamlined functionally. The only major drawback is that delivery is not guaranteed. UDP is therefore used for communications that involve broadcasts, for general network announcements, or for real-time data. A particularly good application is with streaming video and streaming audio where low transmission overheads are a prerequisite, and where retransmission of lost packets is not only unnecessary but also definitely undesirable.

7.2.2 The UDP frame

The format of the UDP frame and the interpretation of its fields are described RFC 768.

The frame consists of a header plus data and contains the following fields:

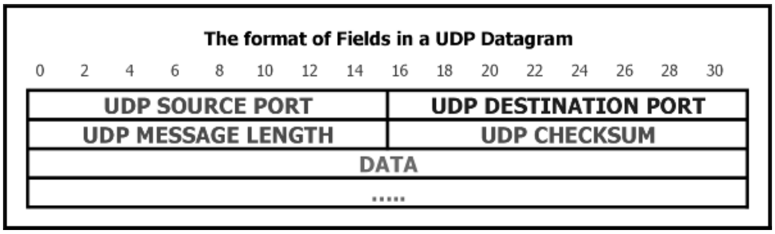


Figure 7.7
UDP frame format

Source port: 16 bits

This is an optional field. When meaningful, it indicates the port of the sending process, and may be assumed to be the port to which a reply should be addressed in the absence of any other information. If not used, a value of zero is inserted.

Destination port: 16 bits

As for source port

Message length: 16 bits

This is the length in bytes of this datagram including the header and the data. (This means the minimum value of the length is eight.)

Checksum: 16 bits

This is the 16-bit one’s complement of the one’s complement sum of a pseudo header of information from the IP header, the UDP header, and the data, padded with ‘0’ bytes at the end (if necessary) to make a multiple of two bytes.

The pseudo header, conceptually prefixed to the UDP header, contains the source address, the destination address, the protocol, and the UDP length. As in the case of TCP, this header is used for computational purposes only, and is NOT transmitted. This information gives protection against misrouted datagrams. This checksum procedure is the same as is used in TCP.

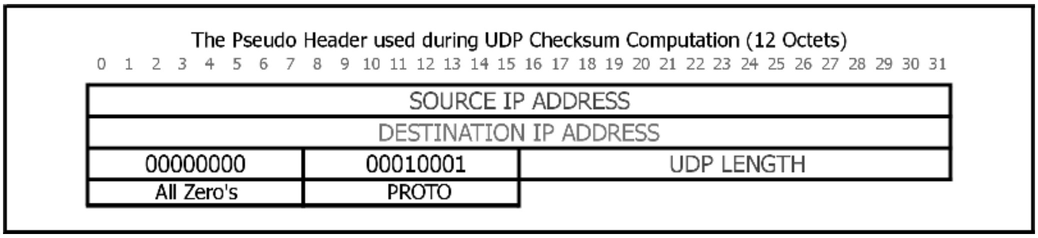


Figure 7.8
UDP pseudo header format

If the computed checksum is zero, it is transmitted as all ones (the equivalent in one’s complements arithmetic). An all zero transmitted checksum value means that the transmitter generated no checksum (for debugging or for higher level protocols that don’t care).

UDP is numbered protocol 17 (21 octal) when used with the Internet protocol.

8

Application layer protocols

Objectives

When you have completed study of this chapter you should have a basic understanding of the application and operation of the following application layer protocols:

- FTP
- TFTP
- TELNET
- RLOGIN
- NFS
- DNS
- WINS
- SNMP
- SMTP
- POP3
- HTTP
- BOOTP
- DHCP

8.1 Introduction

This chapter examines the process/application layer of the TCP/IP model. Protocols at this layer act as intermediaries between some user application (external to the TCP/IP communication stack) and the lower-level protocols such as TCP or UDP. An example is SMTP, which acts as an interface between an e-mail client or server and TCP.

Note that the list of protocols supplied here is by no means complete, as new protocols are developed all the time. Using a developer's toolkit such as WinSock, software developers can interface their own application protocols at this level to the TCP/IP protocol stack.

8.2 File transfer protocol (FTP)

File transfer requires a reliable transport mechanism, and therefore TCP connections are used. The FTP process running on the host that is making the file transfer request is called the FTP client, while the FTP process running on the host that is receiving the request is called the FTP server.

The process involved in requesting a file is as follows:

- The FTP client opens a control connection to port 21 of the server
- The FTP client forwards user name and password to the FTP server for authentication. The server indicates whether authentication was successful
- The FTP client sends commands indicating file name, data type, file type, transmission mode and direction of data flow (i.e. to or from the server) to the server. The server indicates whether the transfer options are acceptable
- The server establishes another connection for data flow, using port 20 on the server
- Data packages are now transferred utilizing the standard TCP flow control, error checking, and retransmission procedures. Data is transferred using the basic NVT format as defined by the TELNET network virtual terminal protocol (no option negotiation is provided for)
- When the file has been transferred, the sending FTP process closes the data connection, but retains the control connection

The control connection can now be used for another data transfer, or it can be closed

8.2.1 Internal FTP commands

These commands are exchanged between the FTP client and FTP server. Each internal protocol command comprises a four-character ASCII sequence terminated by a new-line (<CRLF>) character. Some commands also require parameters. The use of ASCII character sequences for commands allows the user to observe and understand the command flow, and aids the debugging process. The user can communicate directly with the server program by using these codes, but in general this is not advisable.

FTP commands can be divided into three categories, namely service commands, transfer parameter commands and access control commands. There is also a series of reply codes. Here follows a brief summary of the commands and reply codes.

Service commands

These commands define the operation required by the requester. The format of the pathname depends on the specific FTP server being used.

RETR<SP><pathname><CRLF>	Retrieve a copy of the file from the server
STOR<SP><pathname><CRLF>	Store data at the server
STOU<CRLF>	Store unique
APPE<SP><pathname><CRLF>	Append
ALLO<SP><decimal integer> [<SP>R<SP><decimal integer>]<CRLF>	Allocate storage
REST<SP><marker><SP>	Restart transfer at checkpoint
RNFR<SP><pathname><CRLF>	Rename from
RNTO<SP><pathname><CRLF>	Rename to
ABOR<CRLF>	Abort previous service command
DELE<SP><pathname><CRLF>	Delete file at server

RMD<SP><pathname><CRLF>	Remove directory
MKD<SP><pathname><CRLF>	Make directory
PWD<CRLF>	Print working directory
LIST<SP><pathname><CRLF>	List files or text
NLST<SP><pathname><CRLF>	Name list
SITE<SP><string><CRLF>	Site parameters
SYST<CRLF>	Determine operating system
STAT<SP><pathname><CRLF>	Status
HELP[<SP><string>]CRLF	Help information
NOOP<CRLF>	No operation

Transfer parameter commands

These commands are used to alter the default parameters used to transfer data on an FTP connection.

PORT<SP><host-port><CRLF>	Specifies the data port to be used.
PASV<CRLF>	Request server DTP to listen on a data port
TYPE<SP><type code><CRLF>	Representation type: ASCII, EBCDIC, image, or local.
STRU<SP><structure code><CRLF>	File structure: file, record or page.
MODE<SP><mode code><CRLF>	Transmission mode: stream, block or compressed

Access control commands

These commands are invoked by the server and determine which users may access a particular file.

USER<SP><username> <CRLF>	User name
PASS<SP><password><CRLF>	User password
ACCT<SP><acc. information><CRLF>	User account
CWD<SP><pathname><CRLF>	Change working directory
CDUP<CRLF>	Change to parent directory
SMNT<SP><pathname><CRLF>	Structure mount
REIN<CRLF>	Terminate user and re-initialize
QUIT<CRLF>	Logout
<SP>	Space character
<CRLF>	Carriage return, line feed characters

Reply codes

FTP uses a three-digit return code ‘xyz’ followed by a space to indicate transfer conditions. The first digit (value 1–5) indicates whether a response is good, bad or incomplete. The second and third digits are encoded to provide additional information about the reply. The values for the first digit are:

Value	**Description**
1yz	Action initiated. Expect another reply before sending a new command.
2yz	Action completed. Can send a new command.
3yz	Command accepted but on hold due to lack of information.
4yz	Command not accepted or completed. Temporary error condition exists. Command can be reissued.
5yz	Command not accepted or completed. Don’t reissue – reissuing the command will result in the same error.

The second digit provides more detail about the condition indicated by the first digit:

Value	**Description**
X0z	Syntax error or illegal command
X1z	Reply to request for information
X2z	Reply that refers to connection management
X3z	Reply for authentication command
X5z	Reply for status of server

The third digit of the reply code also provides further information about the condition, but the meanings vary between implementations.

8.2.2 FTP user commands

Although designed for use by applications, FTP software usually also provides interactive access to the user, with a range of commands that can be used to control the FTP session. There are several dozen commands available to the user, but for normal file transfer purposes very few of them ever need to be used.

Command	**Description**
ASCII	Switch to ASCII transfer mode
Binary	Switch to binary transfer mode
Cd	Change directory on the server
Cdup	Change remote working directory to parent directory
Close	Terminate the data connection
Del	Delete a file on the server
Dir	Display the server directory
Get	Get a file from the server
Help	Display help
Ls	List contents of remote directory
Lcd	Change directory on the client
Mget	Get several files from the server
Mput	Send several files to the server
Open	Connect to a server
Put	Send a file to the server
Pwd	Display the current server directory
Quote	Supply a file transfer protocol (FTP) command directly
Quit	Terminate the file transfer protocol (FTP) session
Trace	Display protocol codes
Verbose	Display all information

To execute a command, the user types the commands at the ftp prompt, e.g.

```
ftp>close
```

A list of available user commands can be viewed by typing help at the ftp prompt, e.g.

```
ftp> help close
```

After logging into another machine using FTP, the user is still logically connected to the (local) client machine. This is different to TELNET, where the user is logically connected to the (remote) server machine. References to directories and movements of files are relative to the client machine. For example, getting a file involves moving it from the server to the client; putting a file involves moving it from the client to the server.

It may be wise to create a special directory on the client computer just for the transfer of files into and out of the client's system. This helps guard against accidental file deletion, and allows easier screening of incoming files for viruses.

Many operating systems have a GUI-based FTP client such as NetManage's Chameleon NFS that displays the file systems of the local and the remote machines in two separate windows and allows file transfers from one machine to another by mouse movements on the screen.

Most UNIX machines act as FTP servers by default. A daemon process watches the TCP command port (21) continuously for the arrival of a request for a connection and calls the necessary FTP processes when one arrives.

Windows 95/98 does not include FTP server software, but it does provide an FTP client program. However, a number of third-party FTP packages have been written for use with Windows. Examples of such software are CuteFTP, an FTP client, and Serv-U-FTP server.

8.2.3 Anonymous FTP

Anonymous FTP access allows a client to access publicly available files using the login name 'anonymous' and the password 'guest'. Alternatively the password may be required to be a valid e-mail address. Public files are often placed in a separate directory on the server, and are commonly used by Internet sites such as Network Information Systems, Yellow Pages, etc.

8.3 Trivial file transfer protocol (TFTP)

8.3.1 Introduction

TFTP (RFC 1350) is a less sophisticated version of FTP, and caters for situations where the complexity of FTP and the reliability of TCP is neither desired nor required. TFTP does not log on to the remote machine; so it does not provide user access and file permission controls.

TFTP is used for simple file transfers and is typically placed in the read-only memory of diskless machines such as PLCs that use it for bootstrapping or to load applications.

The absence of authorization controls can be overcome by diligent system administration. For example, on a UNIX system, a file may only be transferred if it is accessible to all users on the remote machine (i.e. both read and write permissions are set).

TFTP does not monitor the progress of the file transfer so does not need the reliable stream transport service of TCP. Instead, it uses an unreliable packet delivery system such as UDP, using time-out and retransmission mechanisms to ensure data delivery. The UDP source and destination port fields are used to create the socket at each end, and TFTP transfer identifiers (TIDs) ranging between 0 and 65 535 are created by TFTP and passed to UDP to be placed in the UDP header field as a source port number. The destination (server) port number is set to the well-known port 69, which is reserved for TFTP.

The server returns an acknowledgment message, upon which the data transfer commences.

Data is then relayed in consecutively numbered blocks of 512 bytes. Each block must be acknowledged, using the block number in the message header, before the next block is transmitted. This system is known as a flip-flop protocol. A block of less than 512 bytes indicates the end of the file. A block is assumed lost and re-sent if an acknowledgment is not received within a certain time period. The receiving end of the connection also sets a timer and if the last block to be received was not the end of file block, on time-out the receiver will re-send the last acknowledgment message.

TFTP can fail for many reasons and almost any kind of error encountered during the transfer will cause complete failure of the operation. An error message sent either in place of a block of data or as an acknowledgment terminates the interaction between the client and the server.

8.3.2 Frame types

There are five TFTP package types, distinguished by an opcode field. They are:

Opcode	**Operation**
1	Read request (RRQ)
2	Write request (WRQ)
3	Data (DATA)
4	Acknowledgment (ACK)
5	Error (ERROR)

The frames for the respective operations are constructed as follows:

RRQ/WRQ frames

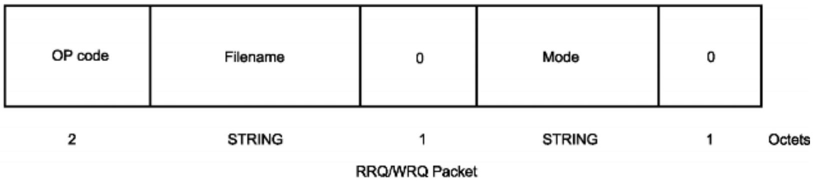


Figure 8.1
RRQ/WRQ frame format

The various fields are as follows:

- **Opcode: 2 bytes**
  1 for RRQ, 2 for WRQ
- **Filename: variable length**
  Written in Netascii, defined by ANSI X3.4-1968. Terminated by a 0 byte.
- **Mode: variable length**
  Indicates the type of transfer. Terminated by a 0 byte. The three available modes are:
  - Netascii
  - Byte – raw 8-bit bytes and binary information
  - Mail – indicates destination is a user not a file – information transferred as Netascii

DATA frames

The filename does not need to be included as the IP address and UDP protocol port number of the client are used as identification.

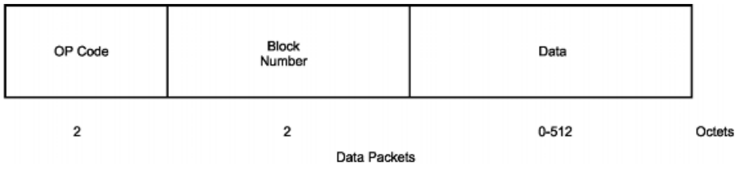


Figure 8.2
Data frame format

The fields are as follows:

- **Opcode: 2 bytes**
  3 indicates DATA
- **Block number: 2 bytes**
  The particular 512-byte block within a specific transfer (allocated sequentially)
- **Data: Variable, 1–512 bytes.**
  Data is transmitted as consecutive 512-byte blocks, a frame with less than 512 bytes means that it is the last block of a particular transfer

ACK frames

These frames are sent to acknowledge each block that arrives. TFTP uses a 'lock-step' method of acknowledgment, which requires each data packet to be acknowledged before the next can be sent.

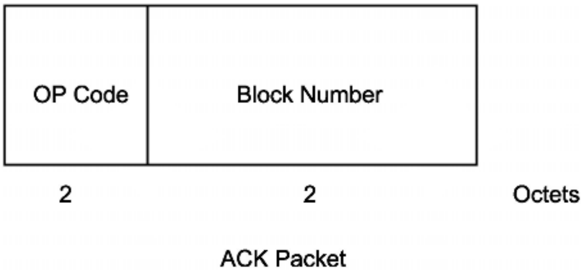


Figure 8.3
ACK frame format

The fields are as follows:

- **Opcode: 2 bytes**
  4 indicates acknowledgment
- **Block number: 2 bytes**
  The number of the block being acknowledged

Error frames

An error message causes termination of the operation.

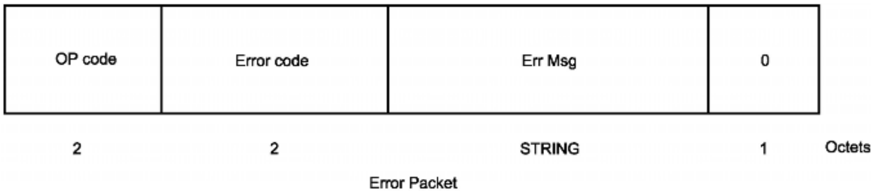


Figure 8.4
Error frame

The fields are:

- **Opcode: 2 bytes**
  5 indicates an error
- **Error code: 2 bytes**
  This field contains a code that describes the problem
  - 0 Not defined
  - 1 File not found
  - 2 Access violation
  - 3 Disk full/allocation exceeded

  - 4 Illegal operation
  - 5 Unknown transfer operation
  - 6 File already exists
  - 7 No such user
- **Error message: Variable length string**
  This is Netascii string, terminated by a 0 byte

8.4 TELNET (telecommunications network)

TELNET is a simple remote terminal protocol, included in the TCP/IP suite that enables virtual terminal capability across a network. That is, a user on machine A can log in to another machine B across a network without being aware that he is working across a network.

Once connected, the user's computer emulates the remote computer. When the user types in commands, they are executed on the remote computer. The user's monitor displays what is taking place on the remote computer during the TELNET session.

The procedure for connecting to a remote computer depends on how the user's Internet access is set up. The process is generally menu driven. Some remote machines require the user to have an account on the machine and will request a username and password. However, many information resources are available to the user without an account and password.

TELNET achieves a connection via the well known port number 23, using either the server's domain name or its IP address, and then passes keystrokes to the remote server and receives output back from it.

TELNET treats both ends of the connection similarly, so that software at either end of a connection can negotiate the parameters that will control their interaction. It provides a set of options, such as type of character set to be used (7-bit or 8-bit), type of carriage-return character to be recognized (e.g. CR or LF) etc, which can be negotiated to suit the client and the server. It is possible for a machine to act as both client and server simultaneously, enabling the user to log into other machines while other users log into his machine.

In the case of a server capable of managing multiple, concurrent connections, TELNET will listen for new requests and then create a new instantiation (or 'slave') to deal with each new connection.

The TELNET protocol uses the concept of a **network virtual terminal** (NVT) to define each end of a connection. NVT uses standard 7-bit US ASCII codes to represent printable characters and control codes such as 'move right one character', 'move down one line', etc. 8-bit bytes with the high order bit set are used for command sequences. Each end has a virtual keyboard that can generate characters (it could represent the user's keyboard or some other input stream such as a file) and a logical printer that can display characters (usually a terminal screen). The TELNET programs at either end handle the translation from virtual terminal to physical device. As long as this translation is possible, TELNET can interconnect any type of device. When the connection is first established and the virtual terminals are setup, they are provided with codes that indicate which operations the relevant physical devices can support.

An operating system usually reserves certain ASCII keystroke sequences for use as control functions. For example, an application running on UNIX operating systems will not receive the Ctrl-C keystroke sequence as input if it has been reserved for interrupting the currently executing program. TELNET must therefore define such control functions so that they are interpreted correctly at both ends of the connection. In this case, Ctrl-C would be translated into the TELNET IP command code.

TELNET does not use ASCII sequences to represent command codes. Rather, it encodes them using an escape sequence. This uses a reserved octet, called the ‘interpret as command’ (IAC) octet, to indicate that the following octet contains a control code. The actual control code can be represented as a decimal number, as follows:

Command	**Decimal Value**	**Meaning**
EOR	239	End of record
SE	240	End of option sub-negotiation
NOP	241	No operation
DMARK	242	Data mark – the data stream part of a SYNCH (always marked by TCP as urgent)
BRK	243	Break
IP	244	Interrupt process – interrupts or terminates the active process
AO	245	Abort output – allows the process to run until completion, but does not send the end of record command
AYT	246	Are you there – used to check that an application is functioning at the other end
EC	247	Erases a character in the output stream
EL	248	Erases a line in the output stream
GA	249	Go ahead – indicates permission to proceed when using half-duplex (no echo) communications
SB	250	Start of option sub-negotiation
WILL	251	Agreement to perform the specified option or confirmation that the specified option is now being performed
WON’T	252	Refusal to perform the specified option or confirmation that the specified option will no longer be performed
DO	253	Asks for the other end to perform the specified option, or acknowledges that the other end will perform the specified option
DON’T	254	Demand that the other end stops performing the specified option, or confirmation that the other end is no longer performing the specified option
IAC	255	Interpret as command – interpret the next octet as a command. When the IAC octet appears as data the 2-octet sequence that is sent will be IAC-IAC

The IAC character to have the above meanings must precede the control code. For example, the two-octet sequence IAC-IP (or 255-244) would induce the server to abort the currently executing program.

The following command options are used by TELNET:

Option Code	**Meaning**
0	Transmit binary – change transmission to 8-bit binary
1	Echo

2	Reconnection
3	Suppress go ahead – i.e. no longer send go-ahead signal after data
4	Approximate message size negotiation
5	Status request – used to obtain the status of a TELNET option from the remote machine.
6	Request timing mark – used to synchronize the two ends of a connection
7	Remote controlled transmission and echo
8	Output line width
9	Output page length
10	Output carriage-return action
11	Output horizontal tab stop setting
12	Output horizontal tab stop action
13	Output form feed action
14	Output vertical tab stop setting
15	Output vertical tab stop action
16	Output line feed action
17	Extend ASCII characters
18	Logout
24	Terminal type – used to exchange information about the make and model of a terminal being used
25	End of record – sent at end of data
28	Terminal location number
31	Window size
34	Line-mode – uses local editing and sends complete lines instead of individual characters.

The two-octet sequence may be followed by a third octet containing optional parameters.

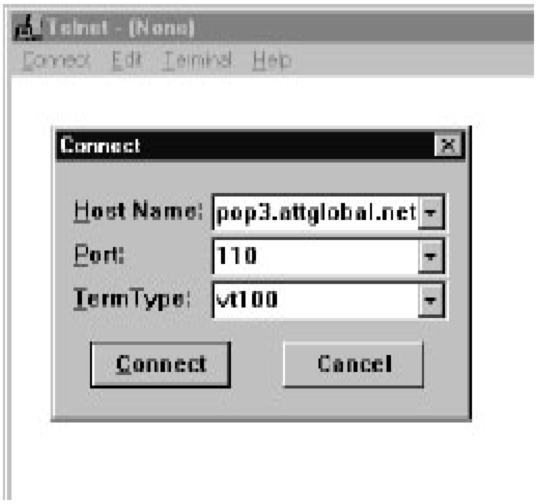
An optional code of 1 indicates 'ECHO'; therefore, the three octets sequence 255-251-1 means 'WILL ECHO' and instructs the other end to begin echoing back the characters that it receives.

A command sequence of 255-252-1 indicates that the sender either will not echo back characters or wants to stop echoing.

The negotiation of options allows clients and servers to optimize their interaction. It is also possible for newer versions of TELNET software that provide more options to work with older versions, as only the options that are recognized by both ends are negotiated.

If the server application malfunctions and stops reading data from the TCP connection, the operating system buffers will fill up until TCP eventually indicates to the client system a window size of zero, thus preventing further data flow from the client. In such a situation TELNET control codes will not be read and therefore will have no effect. To bypass the normal flow control mechanism, TELNET uses an 'out of band' signal. Whenever it places a control signal in the data stream, it also sends a SYNCH command and appends a data mark octet. This induces TCP to send a segment with the URGENT DATA flag set, which reaches the server directly and causes it to read and discard all data until it finds the data mark in the data stream, after which it returns to normal processing.

TELNET programs are freely available and can be downloaded through the Internet. Windows 95/98 includes a simple TELNET program called Microsoft TELNET 1.0.



Host Name: pop3.attglobal.net
Port: 110
TermType: vt100

Figure 8.5
TELNET login (courtesy of Microsoft Corporation)

8.5 RLOGIN (remote login)

The Rlogin service is related to TELNET but is typically used in a UNIX environment. In the case of TELNET, a user at any type of TCP/IP host can log into any other type of TCP/IP host. The local host and remote host may be running totally different operating systems. Rlogin, on the other hand, is normally used when a user at a local UNIX host wants to log into a remote UNIX host.

Rlogin is somewhat easier to use than TELNET and provides a few additional services. For example, it allows the user to maintain a list of hosts in a **rhosts** file so the user does not have to enter a user name and password at the time of each login.

8.6 NFS (network file system)

NFS was originally created by SUN Microsystems to share resources (files and directories) among hosts running UNIX with a local host in such a way that all resources seem to be resident on the local host. Because of its popularity implementations have been created on other operating systems such as UNIX, OS/2, Microsoft Windows and NetWare.

Say that a company stores all of the company sales reports on computer **sales1.** Users from the marketing department wish to access those reports from their computer, **market1,** without having to log in into **sales1,** or copy everything from one machine to the other.

Both computers are connected together on the same TCP/IP network and both have NFS installed and running. The reports are contained in the sales/reports directory on the **sales1** computer.

To share the sales reports, the administrator from **sales1** types the following command:

```
share -F nfs/sales/reports
```

This command makes the sales/reports directory available to any other computer on the network that can access **sales1**. The -F option identifies the resource being shared as an NFS file system. Other options could be used to restrict access to certain computers and to allow read/write or read-only access.

On the computer named **market1**, the system administrator decides to connect the reports directory to the directory called **usr2/salesrpt** locally using this command:

```
Mount -F nfs sales1:/sales/reports/usr2/salesrpt
```

Once the directory is mounted users can:

- Change to the /usr2/salesrpt directory, list the content, open the files and use any standard commands to access and work with the local files
- Move down and search the subdirectories beneath the mount point on the remote system
- Run applications stored on remote file system so that they run as any other application on the local system
- Access the files and directories based on standard UNIX file system permissions

8.7 DNS (domain name system)

8.7.1 Name resolution using hosts files

In small TCP/IP internetworks hosts are typically given simple names such as **computer1**. The mapping between these host names and their associated IP addresses is then maintained as a ‘flat’ database in a local file (the hosts file) on each host. The resolver process on each host translates host names into IP addresses by a simple lookup procedure.

In a large network the maintenance of the hosts files, which have to be identical for all hosts and continuously updated in order to reflect additions and changes, can become quite a tedious task. On the Internet, with millions of names, this becomes impossible.

8.7.2 Name resolution using DNS

The **domain name system** (DNS) provides a network-wide (and in the case of the Internet – a world-wide) directory service that maps host names against IP addresses. For most users this is a transparent process and not relevant whether the resolution takes place via a hosts file or via DNS.

When the IP address of a specific destination host has to be resolved, the DNS resolver on the source host contacts a DNS server somewhere on the internetwork. There is usually more than one DNS server, and the database may be distributed among them. Where an individual DNS server does not have access to the entire database, the host’s name resolver may have to contact more than one DNS server, or the DNS servers may exchange information amongst themselves in order to resolve the query.

Each DNS name server maintains a tree-structured directory database. The collective database stored on all the DNS servers forms a global namespace of all the hosts that can be referenced anywhere on the internetwork.

The Internet naming scheme hierarchical namespace

The original Internet namespace was ‘flat’ i.e. it had no hierarchical structure. At this stage it was still administered by the **Network Information Center** (NIC). The task eventually became too large because of the rapidly increasing number of hosts and a hierarchical (tree-structured) namespace was adopted. At present, the ultimate responsibility for the maintenance of this namespace is vested in the Internet Assigned Names Authority (IANA).

In a domain name, the most local domain is written first and the most global domain is written last. The domain name purdue.edu might identify Purdue University. This domain name is registered against a specific IP address. The administrator of this domain name may now create sub-domains such as, say, cs.purdue.edu for the computer science department at Purdue University. The administrator of the computer science department, in turn, may assign a **fully qualified domain name** (FQDN) to an individual host as follows:

computer1.cs.purdue.edu

If a user is referring to a specific host within a local network, a FAQN is not needed, as the DNS resolver will automatically supply the missing high-level domain name qualifier.

The following commands are therefore equivalent when the ftp client and ftp server are located on the same network:

- ftp computer1.purdue.edu
- ftp computer1

Standard domain names

The original namespace contained a set of standard top-level domains without any reference to a specific country. Since the original Internet was not envisaged to exist beyond the borders of the United States, the absence of any reference to a country implies an organization within the USA.

The following are some of the common top-level domains administered by IANA. More detailed information can be obtained from www.iana.org.

• **.com**	Commercial organizations
• **.net**	Major network support centers
• **.edu**	Educational institutions
• **.gov**	Government institutions (United States government only)
• **.mil**	Military groups (United States military only)
• **.int**	Organizations established by international treaties between governments, or Internet infrastructure databases
• **.org**	Organizations other than the above

Domain names for the .com, .net and .org domains can be obtained from the following registrar web sites:

- CORE
- Melbourne IT
- Network Solutions (a.k.a. NetSol)
- Oleane (France Telekom)
- Register.com

Domain names for the .EDU domain are registered only through Network Solutions.

Country codes

As the Internet backbone was extended into countries other than the USA, the top-level domain names were extended with a two-letter country code as per ISO 3166 (e.g. uk for the United Kingdom, au for Australia, za for South Africa, ca for Canada). The complete list of all Country Code Top-Level Domains (CCTLDs) can be obtained from the IANA website (www.iana.org). This site also contains the basic information for each CCTLD such as the governing agency, administrative and technical contact names telephone and fax numbers, and server information. This information can also be obtained from the Network Solutions web site (www.netsol.com).

DNS clients and servers

Each host on a network that uses the DNS system runs the DNS client software, which implements the resolver function. Designated servers, in turn, implement the DNS nameserver functions. In processing a command that uses a domain name instead of an IP address, the TCP/IP software automatically invokes the DNS resolver function. The resolver then accesses one or more nameservers in order to obtain the relevant IP address.

On a small network, one nameserver may be sufficient and the nameserver software may run on a machine already used for other server purposes (such as a Windows NT server acting as a file server). On large networks it is prudent to run at least two nameservers for reasons of availability, viz. a primary and a secondary nameserver. On large internetworks it is also common to use multiple nameservers, each of which contains a portion of the namespace. It is also possible to replicate portions of the namespace across several servers in order to increase availability.

A network connected to the Internet needs access to at least one primary nameserver and one secondary nameserver, both capable of performing naming operations for the registered domain names on the Internet. In this case, the number of domain names is so large that the namespace is distributed across multiple servers, called authoritative servers, in different countries. For example, all the co.za domain names (i.e. South African Company names) may be hosted on one or more nameserver(s) located in South Africa.

Name resolution

A resolver on a host in Canada requiring the IP address for www.hp.co.za in South Africa, will contact its designated DNS server (wherever it may be), which in turn will contact the relevant authoritative server(s) located in South Africa in order to obtain the IP address. There are two methods by which the interaction between DNS resolver and nameserver can take place.

With recursive resolution, the DNS client makes the initial request. The burden of the processing is then borne by the server, who may have to contact other servers before eventually passing the result back to the client. This is typical for smaller hosts such as PCs and laptops.

With iterative recursion, the resolver contacts a server that either provides the answer, or refers the resolver to another nameserver. This process is repeated until the resolution process is completed. The computational burden is shared between resolver and nameservers. This is typical for larger computers and mainframes.

The DNS client resolver software can implement a caching function by storing the results from the name resolution operation. In this way the resolver can resolve a future query by looking up the cache rather than actually contacting the nameserver. Cache entries are given a time to live so that they are purged from the cache after a given amount of time.

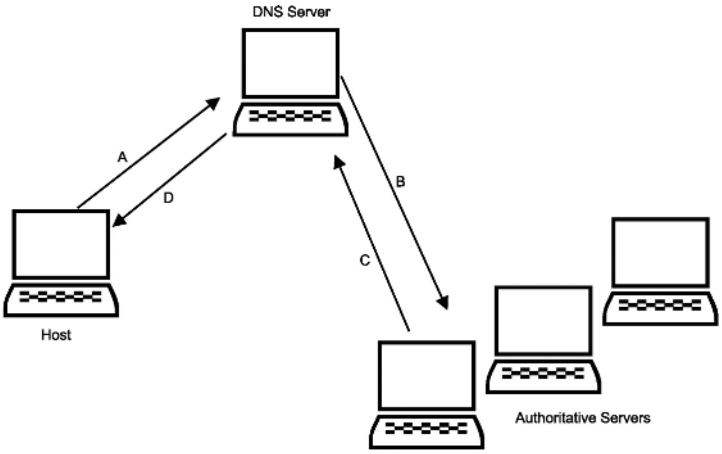


Figure 8.6
DNS name resolution

DNS frame format

The message format for DNS messages is as follows.

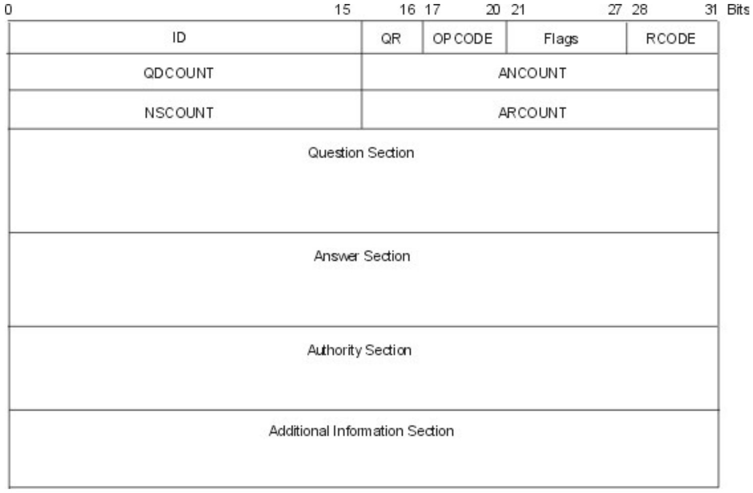


Figure 8.7
DNS message format

- **ID** (IDENTIFICATION), a tracking number (16 bits) used to correlate queries and responses
- **QR,** a one-bit flag that identifies the message as a query (QR=0) or a response (QR=1)
- **OPCODE.** This 4-bit field further defines a query as follows:
  - 0 = Standard query
  - 1= Inverse query
  - 2 = Server status request
  - The other opcodes (3–15) are not used
- **Flags,** used to describe the message further. They are, from right to left:
  - Authoritative answer (AA)
  - Truncation (TC)
  - Recursion desired (RD)
  - Recursion available (RA)

- **RCODE,** the last field in the first long-word is used for response codes with the following meanings:
    - 0 = No error
    - 1 = Format error
    - 2 = Server error
    - 3 = Name error
    - 4 = Not used
    - 5 = Refused
- **Four COUNT** fields indicate the length of the fields to follow:
    - **QDCOUNT** gives the number of question entries
    - **ANCOUNT** gives the number of resource records in the answer section
    - **NSCOUNT** refers to the number of name server resource records in the Authority section
    - **ARCOUNT** refers to the number of resource records in the additional records section
- **Question section**
  Contains queries in the format shown below. A query consists of a query domain name field containing the FQDN about which information is required, a query type field specifying the type of information required, and a query class field identifying the protocol suite with which the name is associated
- **Answer section**
  Contains information returned in response to a query in the format shown below. The resource domain name, type, and class fields are from the original query. The time to live field specifies how long this information can be used if it is cached at the local host. The format of the resource data field depends on the type of information required
- **Authority section**
  Identifies the server that actually provided the information if a nameserver has to contact another nameserver for a response. The format for this field is the same as for the answer section
- **Additional query information**
  Contains additional information related to the name in query; (e.g. the IP address of the host that is the mail exchanger, in response to a MX query)

The DNS message contains a query type field, since the nameserver database consists of many types of information. The following list shows some of the types:

A	Host IP Address
CNAME	Canonical domain name for an alias
MINFO	Information about a mail box or mail list
MX	Name of a host that acts as mail exchanger for a domain
NS	Name of authoritative server for a domain
PTR	Domain name
SOA	Multiple fields that specify which parts of the naming hierarchy a server implements

8.8 WINS

8.8.1 Introduction

WINS is not a general TCP/IP application layer protocol, but rather a Microsoft Windows-specific utility with the primary role of NetBIOS name registration and resolution on TCP/IP. In many respects WINS is like DNS. However, while DNS resolves TCP/IP host names to static IP addresses, WINS resolves NetBIOS names on TCP/IP to dynamic addresses assigned by DHCP.

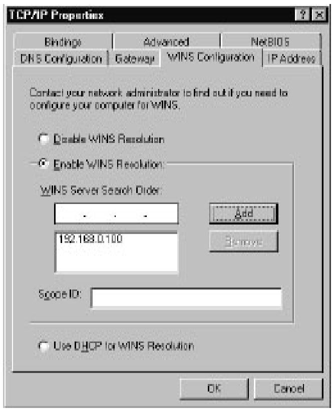
A WIN maintains a database on the WINS server. This database provides a computer name to IP address mapping, allowing computers on the network to interconnect on the basis of machine names.

WINS features the following:

- It resolves NetBIOS names to IP addresses, supporting dynamic IP address mapping (i.e. IP addresses issued by DHCP)
- It prevents two machines from registering the same name
- With traditional NetBIOS name resolution techniques that relied on broadcast, it was not possible to browse across an IP router. WINS overcome this problem by providing name resolution regardless of location on the network
- WINS reduce the number of the broadcast packets, which are normally used to resolve NetBIOS names. This reduction in broadcast packets can improve the network performance

A WIN, like DHCP, is a client/server application. In order to run it on a network, at least one WINS server is needed. The WINS server must have a statically assigned IP address, which is entered into the TCP/IP configuration information for all machines on the network that want to take advantage of the WINS server for name resolution and name registration.

The following figure shows how WINS is configured on the host computer. This is done by selecting Control Panel-> Network, selecting TCP/IP for the LAN interface card, clicking Properties, and then selecting WINS Configuration. The Scope ID (not entered here) defines a group of computers that require a registered NetBIOS name. Computers with the same scope ID will be able to recognize each other's NetBIOS traffic or messages.



Contact your network administrator to find out if you need to configure your computer for WINS.

○ Disable WINS Resolution

● Enable WINS Resolution:

WINS Server Search Order:

192.168.0.100

Scope ID:

○ Use DHCP for WINS Resolution

Figure 8.8
WINS Configuration (courtesy of Microsoft Corporation)

8.8.2 WINS name registration

When a WINS client is turned on for the first time it tries to register its NetBIOS name and IP address with the WINS server by sending a name registration request via a direct UDP packet. When the WINS server receives the request it checks its database to make sure the requested NetBIOS name is in use on the network. If the name registration is successful, than the server sends a name registration acknowledgment back to the client. This acknowledgment includes the time to live for the name registration. The TTL indicates how long the WINS server will keep the name registration before cancelling it. It is the responsibility of the WINS client to send a name refresh request to the WINS server before the name expires in order to keep the name.

If the client tries to register a name that is already in use, the WINS server sends a denial message back to the client. The client than displays a message telling the user that the computer's name is already in use on the network.

When a WINS client shuts down it sends a name release request to the WINS server, releasing its name from the WINS database.

8.8.3 WINS name resolution

When a WINS-enabled client needs to resolve the NetBIOS name to IP address, it uses a resolution method called h-node name resolution, which includes the following procedures:

- It checks to make sure that the name request doesn't point to itself
- It looks in its name resolution cache for a match. Names remain in the cache for about 10 minutes
- It sends a direct name lookup to the WINS server. If the WINS server can match the name to an IP address, the WINS server sends a response to the client
- If the WINS server cannot do the match, the client broadcasts to the network.
- If there is still no response the client will look into its own local LMHOSTS file
- Finally the client will look into the local HOSTS file, or by asking the DNS if it has a matching host name. This is only done if the client is configured to use the DNS for NetBIOS name resolution

8.8.4 WINS proxy agents

WINS proxy agents are used to allow non-WINS-enabled clients to interact with a WINS service. A WINS proxy agent listens to the local network for clients trying to use broadcast to resolve NetBIOS names. The WINS proxy agent picks these requests off the network and forwards them to the WINS server, which responds with the resolved IP address. The WINS proxy agent then provides this information to the client requesting the name resolution.

The advantage of this system is that there is no need to make any changes to the existing non-WINS-enabled clients, and in fact they are completely unaware that the name resolution has been provided by the WINS service.

8.9 SNMP (simple network management protocol)

The **simple network management protocol** (SNMP) is an application-layer protocol that facilitates the exchange of management information between network devices. It enables network administrators to manage network performance, find and solve network problems, and plan for network growth.

Two current versions of SNMP exist: SNMP Version 1 (SNMPv1) and SNMP Version 2 (SNMPv2). Both have a number of features in common, but SNMPv2 offers enhancements, such as additional protocol operations. Standardization of SNMPv3 is pending.

8.9.1 SNMP basic components

An SNMP managed network consists of three key components namely managed devices, agents, and network-management systems:

- A managed device is a network node that contains an SNMP agent and resides on a managed network. These devices collect and store management information and make this information available to **network-management systems** (NMSs) using SNMP. Managed devices can be routers, access servers, switches, bridges, hubs, computer hosts or printers
- An agent is a network-management software module that resides in a managed device. It has local knowledge of management information and translates that information into a form compatible with SNMP
- A network-managed system 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

8.9.2 SNMP basic commands

Managed devices are monitored and controlled using four basic SNMP commands namely read, write, trap, and traversal operations:

- The read command is used by an NMS to monitor managed devices. The NMS examines different variables that are maintained by managed devices
- The write command is used by an NMS to control managed devices. The NMS changes the values of variables stored within managed devices
- The trap command is used by managed devices to asynchronously report the 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

8.9.3 SNMP management information base (MIB)

A **management information base** (MIB) is a collection of information that is organized hierarchically. MIBs are accessed using a network-management protocol such as SNMP. They are comprised of managed objects and are identified by object identifiers.

A managed object (sometimes called an MIB object, an object, or an MIB) is one of any number of specific characteristics of a managed device. Managed objects are comprised of one or more ‘object instances’, which are essentially variables.

Two types of managed objects exist: scalar and tabular. Scalar objects define a single object instance. Tabular objects define multiple related object instances that are grouped together in MIB tables. An example of a managed object is at Input, which is a scalar object that contains a single object instance, the integer value that indicates the total number of input Novell Netware packets on a router interface. An object identifier (or object ID) uniquely identifies a managed object in the MIB hierarchy. The MIB hierarchy can be depicted as a tree with a nameless root, the levels of which are assigned by different organizations.

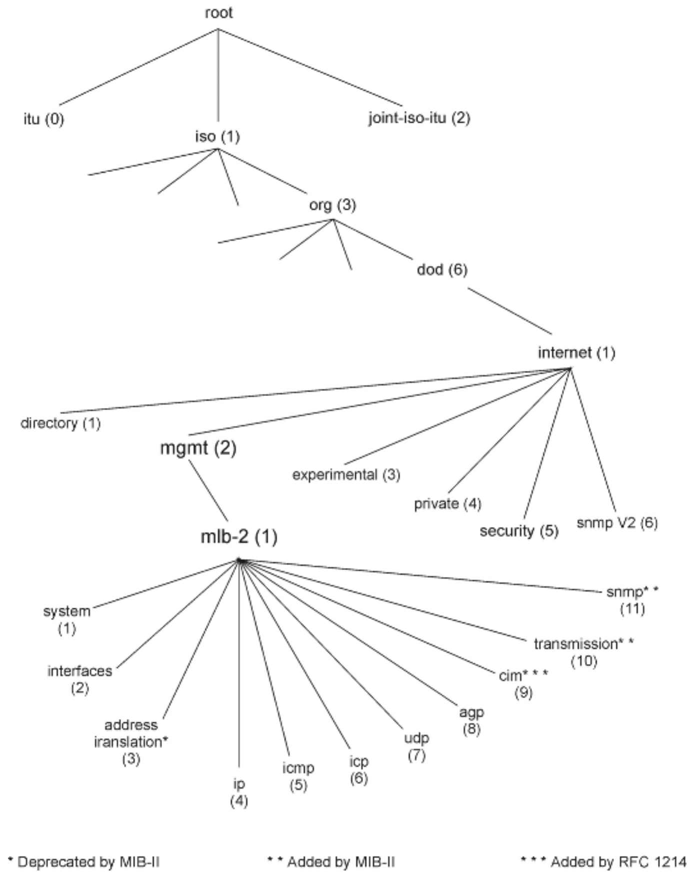


Figure 8.9
MIB tree

The top-level MIB object Ids belong to different standards organizations, while lower-level object Ids are allocated by associated organizations. Vendors can define private branches that include managed objects for their own products. MIBs that have not been standardized typically are positioned in the experimental branch. The managed object at Input can be uniquely identified either by the object name – iso.identified-organization. dod.internet. private.enterprise.cisco.temporary variables.Novell.atInput – or by the equivalent object descriptor: 1.3.6.1.4.1.9.3.4.1.

8.9.4 SNMPv2 protocol operations

SNMP is a simple request–response protocol. The network-management system issues a request, and managed devices return responses. This behavior is implemented by using one of four protocol operations: Get, GetNext, Set, and Trap:

- The Get operation is used by the NMS to retrieve the value of one or more object instances from an agent. If the agent responding to the Get operation cannot provide values for all the object instances in a list, it does not provide any values

- The GetNext operation is used by the NMS to retrieve the value of the next object instance in a table or list within an agent
- The Set operation is used by the NMS to set the values of object instances within an agent
- The Trap operation is used by agents to asynchronously inform the NMS of a significant event

The Get, GetNext, and Set operations used in SNMPv2 are exactly the same as that used in SNMPv1. SNMPv2, however, adds and enhances some operations. The SNMPv2 Trap operation, for example, serves the same function as that used in SNMPv1. It, however, uses a different message format and is designed to replace the SNMPv1 Trap.

SNMPv2 also defines two additional protocol operations: GetBulk and Inform:

- The GetBulk operation is used by the NMS to efficiently retrieve large blocks of data, such as multiple rows in a table. GetBulk fills a response message with as much of the requested data as will fit
- The Inform operation allows one NMS to send trap information to another NMS and receive a response. In SNMPv2, if the agent responding to GetBulk operations cannot provide values for all the variables in a list, it provides partial results

8.9.5 SNMP management

SNMP is a distributed-management protocol. A system can operate exclusively as either an NMS or an agent, or it can perform the functions of both. When a system operates as both an NMS and an agent, another NMS might require that the system query managed devices and provide a summary of the information learned, or that it report locally stored management information.

8.9.6 SNMP security

SNMP lacks any authentication capabilities, which results in vulnerability to a variety of security threats.

These include masquerading, modification of information, message sequence and timing modifications, and disclosure:

- Masquerading consists of an unauthorized entity attempting to perform management operations by assuming the identity of an authorized management entity
- Modification of information involves an unauthorized entity attempting to alter a message generated by an authorized entity so that the message results in unauthorized accounting management or configuration management operations
- Message sequence and timing modifications occur when an unauthorized entity reorders, delays, or copies and later replays a message generated by an authorized entity
- Disclosure results when an unauthorized entity extracts values stored in managed objects, or learns of notifiable events by monitoring exchanges between managers and agents

Because SNMP does not implement authentication, many vendors do not implement Set operations, thereby reducing SNMP to a monitoring facility.

8.9.7 SNMP interoperability

SNMPv2 is incompatible with SNMPv1 in two areas: message formats and protocol operations. SNMPv2 messages use different header and **protocol data unit** (PDU) formats than SNMPv1 messages. SNMPv2 also uses two protocol operations that are not specified in SNMPv1. Furthermore, RFC 1908 defines two possible SNMPv1/v2 coexistence strategies: proxy agents and 'bilingual' network-management systems.

Proxy agents

An SNMPv2 agent can act as a proxy agent on behalf of SNMPv1 managed devices, as follows:

- An SNMPv2 NMS issues a command intended for an SNMPv1 agent
- The NMS sends the SNMP message to the SNMPv2 proxy agent
- The proxy agent forwards Get, GetNext, and Set messages to the SNMPv1 agent unchanged
- GetBulk messages are converted by the proxy agent to GetNext messages and then are forwarded to the SNMPv1 agent
- The proxy agent maps SNMPv1 Trap messages to SNMPv2 Trap messages and then forwards them to the NMS

Bilingual network-management system

Bilingual SNMPv2 network-management systems support both SNMPv1 and SNMPv2. To support this dual-management environment, a management application in the bilingual NMS must contact an agent. The NMS then examines information stored in a local database to determine whether the agent supports SNMPv1 or SNMPv2. Based on the information in the database, the NMS communicates with the agent using the appropriate version of SNMP.

8.10 SMTP (simple mail transfer protocol)

TCP/IP defines an electronic messaging protocol named SMTP. SMTP is used by E-mail programs such as Outlook Express or Eudora to send E-mail messages and files from a user on a local network to a user on a remote network.

SMTP defines the interchange between two SMTP processes. It does NOT define how the mail is to be passed from the sender to SMTP, or how the mail is to be passed from SMTP to the recipient. The SMTP process with mail to send is called the SMTP client, while the receiving SMTP process is called the SMTP server.

Mail is forwarded to an SMTP client in one of two ways. The user can either use TELNET to enter the information manually, or the application such as Outlook Express can invoke the client.

The first step in the transmission of the data is the connection setup, whereby the SMTP client opens a TCP connection to the remote SMTP server at port 25. The client then sends a 'Hello' command containing the name of the sending user. The SMTP server now sends a reply indicating its ability to receive mail. TELNET users will have to enter the IP address or the domain name of the server (e.g. smtp-01.ny.us.ibm.net), the relevant port number (25) and a terminal type (e.g. VT-100). The first two items are necessary for TCP to create the necessary socket.

The second step in the process involves the actual mail transfer. Mail transfer begins with a ‘Mail From’ command containing the name of the sender, followed by a ‘Receipt’ command indicating the recipient. A ‘Data’ command is followed by the actual message. SMTP can be considered a reliable delivery service in that the underlying TCP protocol ensures correct delivery to the SMTP server. SMTP, however, does not guarantee nor offer mechanisms for reliable delivery from the SMTP server to the user.

When the message transfer is complete another message can be sent, the direction of transfer changed, or the connection closed.

Closing the connection, involves the SMTP client issuing a ‘Quit’ command. Both sides then execute a TCP close operation in order to release the connection.

SMTP commands begin with a four-octet command code (in ASCII), which can be followed by an argument. The SMTP replies use the same format as FTP, i.e. a 3-digit numeric value followed by a text string. Here follows some SMTP commands.

```
HELO
MAIL FROM: sender-e-mail-address
VRFY recipient-mail-address
RCPT TO:  recipient-e-mail-address
EXPN alias-name
DATA
HELP command-name
RSET
NOOP
QUIT
```

In the following example a simple ASCII string is sent to a recipient via a TELNET connection.

```
220 prserv.net – Maillenium ESMTP/ MULTIBOX out4 #30
MAIL FROM: john@hotmail.com
250 ok
RCPT TO: deb@iinet.net.au
250 ok; forward to <deb@iinet.net.au)
DATA
354 ok
‘This is only a test message.’
.
250 ok, id = 2000042800030723903cb80fe
QUIT
```

8.11 POP (post office protocol)

The current version of the post office protocol is POP3. POP3 uses the well known port number 110. Like SMTP, it involves a client running on a local machine and a server running on a remote machine. POP3 is very much the opposite of SMTP in that its function is to retrieve mail from a remote POP3 server to a local POP3 client.

It was developed to ease the load on mail servers. Instead of multiple clients logging in to a remote mail server, the remote POP3 server makes a quick connection to the actual mail server, retrieves and removes the mail from the mail server, and then downloads it to the local POP3 client. As in the case of SMTP, it uses a TCP connection for this purpose. Unlike SMTP, proper authentication with a user name and a password is required.

POP3 commands include the following.

```
STAT
```

```
LIST message-number
RETR message-number
DELE message-number
NOOP
RSET
QUIT
TOP message-number number-of-lines
```

The following example shows the interaction with a POP3 server via a TELNET connection.

```
+OK POP Server Version 1.0 at pop1a.prserv.net
USER auinet.deb
+OK Password required for deb
PASS geronimo
+OK deb has 2 messages (750 octets)_
LIST
+OK 2 messages (75 octets)
1 374
2 376
TOP 1 10
+OK 274 octets
Received from out4.prserv.net [32.97.166.34] by in2.prserv.net id 956880005.59882-1:
Fri, 28 Apr 2000- 00:00:00 +0000
Received: from <unknown domain> ([129:37:1675:208] by prserv.net (out4) with SMTP
Id <2000042723591123901fj001e; Thu, 27 Apr 2000 23:59:48: +0000
'This is only a test message.'
QUIT
```

8.12 BOOTP (bootstrap protocol)

The **bootstrap protocol** BOOTP (RFC 951) is an alternative to RARP. When a diskless workstation (for example a PLC) is powered up, it broadcasts a BOOTP request on the network.

A BOOTP server hears the request, looks up the requesting client's MAC address in its BOOTP file, and responds by telling the requesting client machine:

- The server's IP address and NetBIOS name
- The fully qualified name of the file that is to be loaded into memory of the requesting machine, and executed at boot-up

Although BOOTP is an alternative to RARP, it operates in an entirely different way. RARP operates at the data link layer and the RARP packets are contained within the local network (e.g. Ethernet) frames; hence it cannot cross any routers. With BOOTP the information is carried by UDP via IP, hence it can operate on an internetwork across routers and the server can be several hops away from the client and facilitate address resolution across routers. Although BOOTP uses IP and UDP, it is still small enough to fit within a bootstrap ROM on a client workstation.

Figure 8.10 depicts the BOOTP message format.

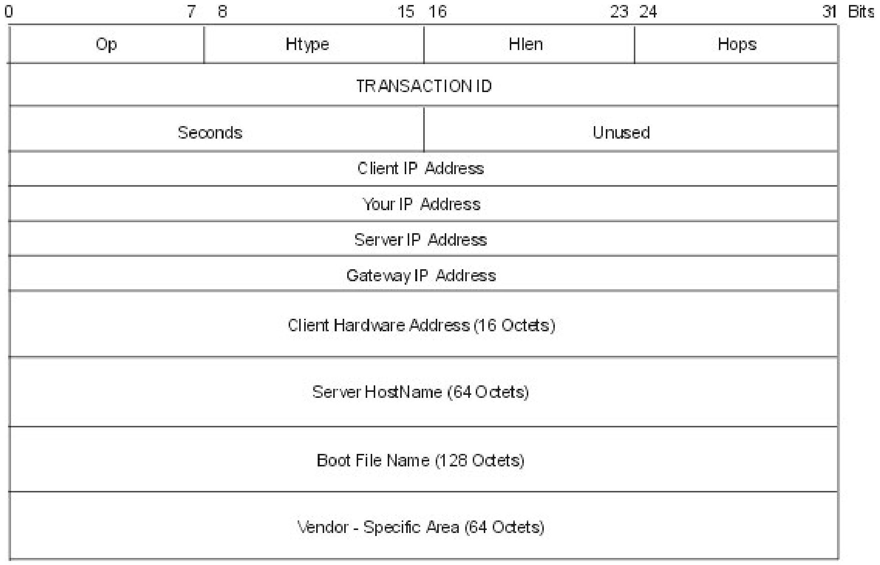


Figure 8.10
BootP frame

- **Op: 8 bits**
  The message type, 1 = BOOTREQUEST, 2 = BOOTREPLY
- **Htype: 8 bits**
  Same as for ARP/RARP
- **Hlen: 8 bits**
  Same as for ARP/RARP
- **Hops: 8 bits**
  Used by relay agents when booting via a relay agent. A client sets this field to 0.
- **Transaction ID: 32 bits**
  (Also called XID). A random tracking number as for the IP and ICMP protocols
- **Seconds: 16 bits**
  The seconds elapsed since the client started to boot
- **Client IP address: 32 bits**
  Set by the client to its IP address, or initially to zero
- **Your IP address: 32 bits**
  Set by the server to the correct IP address for the client, if the client advertises its IP address as 0
- **Server IP address: 32 bits**
  Server IP address, set by the server
- **Gateway IP address: 32 bits**
  The Gateway (router) address, set by the relay agent
- **Client hardware address: 16 bytes**
  The client MAC address, set by itself
- **Server host name: 64 bytes**
  An optional server name, e.g. Garfield or Computer10
- **Boot file name: 128 bytes**
  Used by the server to return a fully qualified directory path name to the

client, e.g. c:\windows\bootfiles\startup.exe. This is the location on the server from which the boot file has to be downloaded

- **Vendor-specific area: 64 bytes**
  DHCP options as per RFC 1531

RFC 1532 and RFC 1533 contain subsequent clarifications and extensions to BOOTP.

8.13 DHCP (dynamic host configuration protocol)

DHCP, as defined by RFC 1533, 1534, 1541, and 1542 was developed out of BOOTP in an effort to centralize and streamline the allocation of IP addresses. DHCP's purpose is to centrally control IP-related information and eliminate the need to manually keep track of where individual IP addresses are allocated.

When TCP/IP starts up on a DHCP-enabled host, a special message is sent out requesting an IP address and a subnet mask from a DHCP server. The contacted server checks its internal database, then replies with an offer message comprising the information the client requested. DHCP can also respond with a default gateway address, DNS address(es), or a NetBIOS name server, such as WINS. When the IP offer is accepted, it is then extended to the client for a specified period of time, called a lease. If the DHCP server runs out of IP addresses, no IP addressing information can be offered to the clients, causing TCP/IP initialization to fail.

DHCP's advantages include the following:

It's inexpensive. The server software comes built into many operating systems, and the manual effort involved in managing large numbers of IP addresses is reduced.

IP configuration information is entered electronically by another system, eliminating the possibility of human error.

IP becomes a 'plug and play operation'.

It does, however, have some drawbacks such as:

A new user may randomly (delinquently!) enter a fixed IP address on his computer in order to gain immediate access to the network. Later, that number may be assigned by DHCP to a different user and show up as a duplicate.

Because the initial input for IP addresses, subnet masks, gateways, DNS addresses, and NetBIOS name server address is done by a human on a PC, it can easily be entered incorrectly.

Exclusive reliance on the DHCP server during the TCP/IP initialization phase could result in an initialization failure if that server is down, or otherwise unavailable.

Certain applications of TCP/IP, like logging in to a remote network through a firewall, require the use of a specific IP address. DHCP allows for exclusions that prevent certain IP address ranges from being used. If the specific IP address that is needed for remote login has been excluded, the user has a problem.

There is an extensive amount of incredibly tedious work involved in maintaining an accurate roster of both used and free IP addresses.

8.13.1 DHCP operation

IP lease request

This is the first step in obtaining an IP address under DHCP. It is initiated by a host with TCP/IP, configured to obtain an IP address automatically, if booted up. Since the requesting client is not aware of its own IP address, or that belonging to the DHCP server, it will use 0.0.0.0 and 255.255.255.255, respectively. This is known as a DHCP discover message. The broadcast is created with UDP ports 67 (BOOTP client) and 68 (BOOTP server). This message contains the MAC address and NetBIOS name for the client system to be used in the next phase of sending a lease offer. If no DHCP server responds to the initial broadcast, the request is repeated three more times at 9, 13, and 16-second intervals, plus a random event occurring in the period between 0 and 1000 milliseconds. If still no response is received, a broadcast message is made every five minutes until it is finally answered. If no DHCP server ever becomes available, no TCP/IP communications will be possible.

IP lease offer

The second phase involves the actual information given by all DHCP servers that have valid addressing information to offer. Their offers consist of an IP address, subnet mask, lease period (in seconds), and the IP address of the proposing DHCP server. These offers are sent to the requesting client's MAC address. The pending IP address offer is reserved temporarily to prevent it from being taken simultaneously by other machines, which would otherwise create chaos. Since multiple DHCP servers can be configured, it also adds a degree of fault tolerance, should one of the DHCP servers go down.

IP lease selection

During this phase, the client machine selects the first IP address offer it receives. The client replies by broadcasting an acceptance message, requesting to lease IP information. Just as in stage one, this message will be broadcast as a DHCP request, but this time, it will additionally include the IP address of the DHCP server whose offer was accepted. All other DHCP servers will then revoke their offers.

IP lease acknowledgment

The accepted DHCP server proceeds to assign an IP address to the client, then sends an acknowledgement message, called a DHCPACK, back to the client. Occasionally, a negative acknowledgment, called a DHCPNACK, is returned. This type of message is most often generated if the client is attempting to lease its old IP address, which has since been reassigned elsewhere. Negative acceptance messages can also mean that the requesting client has an inaccurate IP address, resulting from physically changing locations to an alternate subnet.

After this final phase has been successfully completed, the client machine integrates the new IP information into its TCP/IP configuration. It is then usable with all utilities, as if it has been manually entered into the client host.

Lease renewal:

Regardless of the length of time an IP address is leased, the leasing client will send a DHCPREQUEST to the DHCP server when its lease period has elapsed by 50%. If the DHCP server is available, and there are no reasons for rejecting the request, a DHCP acknowledge message is sent to the client, updating the configuration and resetting the lease time. If the server is unavailable, the client will receive an 'eviction' notice stating that it had not been renewed. In this event, that client would still have a remaining 50% lease time, and would be allowed full usage privileges for its duration. The rejected client would react by sending out an additional lease renewal attempt when 87.5% of its lease time had elapsed. Any available DHCP server could respond to this DHCPREQUEST message with a DHCPACK, and renew the lease. However, if the client received a DHCPNACK (negative) message, it would have to stop using the IP address immediately, and start the leasing process over, from the beginning.

Lease release

If the client elects to cancel the lease, or is unable to contact the DHCP server before the lease elapses, the lease is automatically released.

Note that DHCP leases are not automatically released at system shutdown. A system that has lost its lease will attempt to re-lease the same address that it had previously used.

8.13.2 DHCP message format

The DHCP message format is based on the BOOTP format, and is illustrated in Fig 8.11.

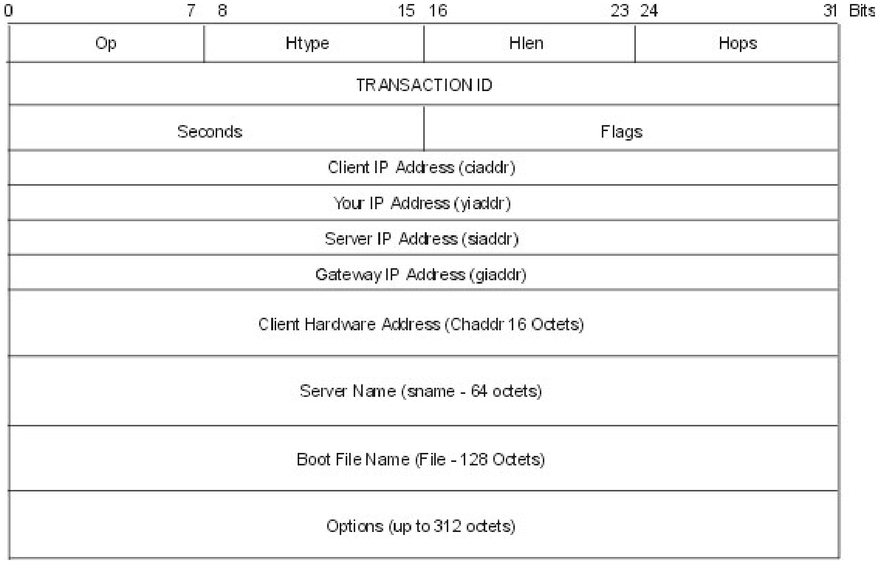

0
7
8
15
16
23
24
31
Bits
Op
Htype
Hlen
Hops
TRANSACTION ID
Seconds
Flags
Client IP Address (ciaddr)
Your IP Address (yiaddr)
Server IP Address (siaddr)
Gateway IP Address (giaddr)
Client Hardware Address (Chaddr 16 Octets)
Server Name (sname - 64 octets)
Boot File Name (File - 128 Octets)
Options (up to 312 octets)

Figure 8.11
DHCP message format

The fields are as follows:

- **Op: 8 bits**
  The message type, 1 = BOOTREQUEST, 2 = BOOTREPLY
- **Htype: 8 bits**
  Same as for ARP/RARP
- **Hlen: 8 bits**
  Same as for ARP/RARP
- **Hops: 8 bits**
  Used by relay agents when booting via a relay agent. A client sets this field to 0
- **Transaction ID: 32 bits**
  (Also called XID). A random tracking number as for the IP and ICMP protocols
- **Seconds: 16 bits**
  The seconds elapsed since the client started to boot
- **Flags: 16 bits**
  This field contains a 1-bit broadcast flag, as described in RFC 1531
- **Client IP address (ciaddr): 32 bits**
  Set by the client to its IP address, or initially to zero

- **Your IP address (yiaddr): 32 bits**
  Set by the server to the correct IP address for the client, if the client advertises its IP address as 0
- **Server IP address (siaddr): 32 bits**
  Server IP address, set by the server
- **Gateway IP address (giaddr): 32 bits**
  The gateway (router) address, set by the relay agent
- **Client hardware address (chaddr): 16 bytes**
  The clients MAC address set by itself
- **Server name (sname): 64 bytes**
  An optional server name, e.g. Garfield or Computer10
- **Boot file name: 128 bytes**
  Used by the server to return a fully qualified directory path name to the client, e.g. c:\windows\bootfiles\startup.exe. This is the location on the server from which the boot file has to be downloaded
- **Options: Up to 312 bytes**
  DHCP options as per RFC 1531

9

TCP/IP utilities

Objectives

When you have completed study of this chapter you should able to apply the following utilities:

- Ping
- ARP
- NETSTAT
- NBTSTAT
- IPCONFIG
- WINIPCFG
- tracert
- ROUTE

9.1 Introduction

The TCP/IP utilities are discussed throughout the book. This section is designed to bring them all together in one section for ease of reference, as they are very important in network management and troubleshooting.

Most of the older utilities are DOS-based. However, more and more Windows-based utilities are becoming available, many of them as freeware or shareware.

9.2 Ping (packet Internet groper)

'Pinging' is one of the easiest ways to test connectivity across the network and confirm that an IP address is reachable. The DOS ping utility (ping.exe) uses ICMP to forward an echo request packet to the destination address. The destination then responds with an ICMP echo response packet. Although the test seems trivial at first sight, it is a powerful diagnostic tool and can demonstrate correct operation between the Internet layers of two hosts across a WAN regardless of the distance and number of intermediate routers involved.

Technically speaking, the ping utility can only ‘ping’ an IP address. This is due to the fact that the ICMP messages are carried within IP datagrams, which require the source and destination IP addresses in the header. Without this feature, it would have been impossible to ‘ping’ across a router. If, therefore, the user does not know the IP address, the name resolver on the local host system has to look it up e.g. via the domain name system or in the hosts file.

The IP datagram, in turn, is transported by means of a network interface layer frame (e.g. Ethernet), which requires, in its header, the MAC, addresses of the source and destination nodes on the local network. If this is not to be found in the ARP cache, the ARP protocol is invoked in order to obtain the MAC address. The result of this action (the mapping of MAC address against IP address) is then stored in the ARP cache. The easiest way to get an overall impression of the process is to capture the events described here by means of a protocol analyzer.

If the IP address is known, the following format can be used:

- **ping <IP Address>** e.g. ping 192.100.100.4
  Ping 192.100.100.255 will cause all hosts on network 192.100.100.0 to respond and will cause unnecessary traffic

If the IP address is unknown, one of the following ways can be used to define the target machine:

- **ping <host name>** e.g. ping computer1
  This can be done provided computer1’s IP address has already been resolved by NetBIOS
- **ping <own machine>** e.g. ping 127.0.0.1
  This is a reserved IP address for loopback testing
- **ping <own machine>** e.g. ping localhost
  This is a reserved name for loopback testing
- **ping <domain name>** e.g. ping www.idc-online.com
  This will be resolved by the domain name system

There are several options available under the ping command, as shown below:

C:\WINDOWS.000>ping

Usage: ping [-t] [-a] [-n count] [-l size] [-f] [-i TTL] [-v TOS] [-r count] [-s count] [[-j host-list] |
[-k host-list]]
[-w timeout] destination-list

Options

• t	Ping the specified host until stopped To see statistics and continue – type Control-Break To stop – type Control-C
• a	Resolve addresses to hostnames
• n count	Number of echo requests to send
• l size	Send buffer size
• f	Set don’t fragment flag in packet
• i TTL	Time to live

• v TOS	Type of service
• r count	Record route for count hops
• s count	Time-stamp for count hops
• j host-list	Loose source route along host-list
• k host-list	Strict source route along host-list
• w timeout	Timeout in milliseconds to wait for each reply

```
C:\WINDOWS.000>
```

The following examples show how some of the ping options can be applied:

- **Ping 193.2.45.66** -t will ‘ping’ the specified IP address repetitively until stopped by typing Ctrl-C
- **Ping 193.2.45.66** -n 10 will ‘ping’ the specified IP address 10 times instead of the default of 4
- **Ping 193.2.45.66** -l 3500 will ‘ping’ the specified IP address with 3500 bytes of data instead of the default of 32 bytes

Here are some examples of what could be learned by using the ping command.

Example 1: A host with IP address 207.194.66.100 is being ‘pinged’ by another host on the same subnet, i.e. with the same NetID. In this example both addresses are conventional class C addresses. Note that the screen display differs between operating systems, even between Windows95 and Windows98, although the basic parameters are the same.

The following response is obtained:

```
C:\WINDOWS.000>ping 207.194.66.100
Pinging 207.194.66.100 with 32 bytes of data:
Reply from 207.194.66.100: bytes=32 time<10ms TTL=128
Reply from 207.194.66.100: bytes=32 time=1ms TTL=128
Reply from 207.194.66.100: bytes=32 time=1ms TTL=128
Reply from 207.194.66.100: bytes=32 time=1ms TTL=128
Ping statistics for 207.194.66.100:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milliseconds:
Minimum = 0ms, Maximum =  1ms, Average =  0ms
C:\WINDOWS.000>
```

From the result, the following can be observed:

- The ICMP message contained 32 bytes
- The average RTT (round trip time) to the target host and back is in the vicinity of 1 millisecond
- The TTL (time to live) remaining in the IP header after its return is 128. Since TTL is normally set at an initial value representing 2^5 (i.e. 32), 2^6 (i.e. 64) or 2^7 (i.e. 128), it can be safely assumed that the TTL value was not altered, and hence there are no routers between the source and destination hosts

Example 2: A host with IP address 207.194.66.101 now ‘pinged’. Although this host is, in fact, nonexistent, it seems ‘legitimate’ since the NetIDs match. The originating host will therefore attempt a ping, but a timeout will occur.

```
C:\WINDOWS.000>ping 207.194.66.101
Pinging 207.194.66.101 with 32 bytes of data:
```

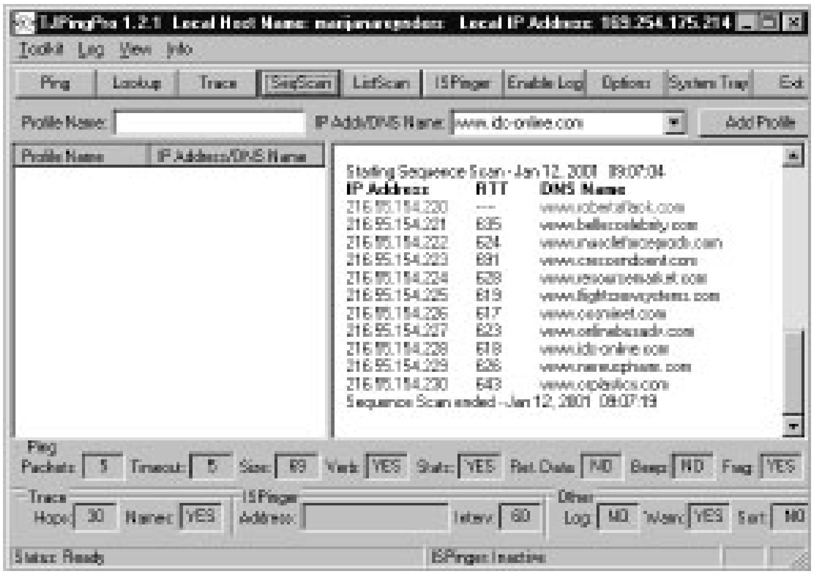
```
Request timed out.
Request timed out.
Request timed out.
Request timed out.
Ping statistics for 207.194.66.101:
Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
Approximate round trip times in milliseconds:
Minimum = 0ms, Maximum =  0ms, Average =  0ms
C:\WINDOWS.000>
```

Example 3. As before, but this time the NetID differs i.e. the target host is assumed to reside on a different network. Since, in this case, no default gateway has not been specified, the originating host does not even attempt to issue an ICMP message, and immediately issues a 'host unreachable' response.

```
C:\WINDOWS.000>ping 208.194.66.100
Pinging 208.194.66.100 with 32 bytes of data:
Destination host unreachable.
Destination host unreachable.
Destination host unreachable.
Destination host unreachable.
Ping statistics for 208.194.66.100:
Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
Approximate round trip times in milliseconds:
Minimum = 0ms, Maximum =  0ms, Average =  0ms
C:\WINDOWS.000>
```

The DOS ping command is not particularly 'user friendly'. It is, for example, not possible to ping a large number of hosts sequentially. There are, however, several Windows-based Ping utilities available as freeware or shareware, of which TJPingPro is an example.

The following example shows how a block of contiguous IP addresses can be pinged with a single 'click', after setting up 'start' and 'end' IP addresses on the options screen.



Profile Name:

IP Addr/DNS Name: www.idc-online.com

```
Starting Sequence Scan - Jan 12, 2001  09:07:04
IP Address        RTT     DNS Name
216.95.154.220    ---     www.robertaflack.com
216.95.154.221    635     www.bellacoolebaby.com
216.95.154.222    624     [illegible]
216.95.154.223    691     www.cascoendoent.com
216.95.154.224    628     www.resourcemarket.com
216.95.154.225    619     www.flightsurveysystems.com
216.95.154.226    617     www.casnieet.com
216.95.154.227    623     www.onlinebazadr.com
216.95.154.228    618     www.idc-online.com
216.95.154.229    626     www.nanuspharm.com
216.95.154.230    643     www.cplastics.com
Sequence Scan ended - Jan 12, 2001  09:07:19
```

Packets: 5 Timeout: 5 Size: 69 Verb: YES Stats: YES Ret. Data: NO Beep: NO Frag: YES

Hops: 30 Names: YES Address: Interv: 60 Log: NO Warn: YES Sort: NO

Status: Ready | ISPinger: Inactive

Figure 9.1
TJPingPro sequential scan (courtesy of Top Jimmy Software)

9.3 ARP

The arp utility (arp.exe) is used to display the arp cache which holds the IP to MAC address translation of hosts on the local subnet. This utility is not to be confused with the ARP (address resolution protocol) that actually determines the IP to MAC address translation. The ARP utility can also be used to manually add entries to the cache, using the -s option.

```
C:\WINDOWS.000>arp
Displays and modifies the IP-to-physical address translation tables used by address
resolution protocol (ARP).
ARP -s              inet_addr eth_addr [if_addr]
                    ARP -d inet_addr [if_addr]
                    ARP -a [inet_addr] [-N if_addr]
-a                  Displays current ARP entries by interrogating the current
                    protocol data.
                    If inet_addr is specified, the IP and physical addresses for
                    only the specified computer are displayed.  If more than one
                    network interface uses ARP, entries for each ARP table are
                    displayed.
-g                  Same as -a.
inet_addr           Specifies an Internet address.
-N if_addr          Displays the ARP entries for the network interface specified
                    by if_addr.
-d                  Deletes the host specified by inet_addr.
-s                  Adds the host and associates the Internet address inet_addr
                    with the physical address eth_addr. The physical address is
                    given as 6 hexadecimal bytes separated by hyphens. The
                    entry is permanent.
eth_addr            Specifies a physical address.
if_addr             If present, this specifies the Internet address of the interface
                    whose address translation table should be modified.
                    If not present, the first applicable interface will be used.
Example:
> arp -s 157.55.85.212   00-aa-00-62-c6-09  ....  Adds a  static entry.
> arp -a                                    ....  Displays the arp table.
```

The following shows a typical display in response to the arp -a command. Note the third column, which indicates type. Entries in the arp cache can be entered manually as static entries, but that poses a problem as IP addresses can be changed and physical Network cards (and hence MAC addresses) can be swapped, rendering the stored IP to MAC address mapping useless unless updated. For this reason the ARP protocol (not to be confused with the utility by the same name) binds IP addresses and physical (MAC) addresses in a temporary (dynamic) way. Dynamic entries are deleted from the cache after a few minutes, if not used.

```
C:\WINDOWS.000>arp -a
Interface: 0.0.0.0 on Interface 0x1000002
Internet Address          Physical Address          Type
192.100.100.7             00-00-c6-f6-34-43         static
192.100.100.99            00-00-fe-c6-57-a8         dynamic
C:\WINDOWS.000>
```

9.4 NETSTAT

This is used for obtaining protocol statistics and current active connections utilizing TCP/IP. Nowadays there are many Windows-based utilities that can do much more; yet in an emergency netstat is certainly better than nothing at all. Here follows the netstat options.

```
C:\WINDOWS.000>netstat /?
Displays protocol statistics and current TCP/IP network connections.
NETSTAT [-a] [-e] [-n] [-s] [-p proto] [-r] [interval]
-a              Displays all connections and listening ports.
-e              Displays Ethernet statistics. This may be combined with the
                -s option.
-n              Displays addresses and port numbers in numerical form.
-p proto        Shows connections for the protocol specified by proto; proto
                may be TCP or UDP.  If used with the -s option to display
                per-protocol statistics, proto may be TCP, UDP, or IP.
-r              Displays the routing table.
-s              Displays per-protocol statistics. By default, statistics are
                shown for TCP, UDP and IP; the -p option may be used to
                specify a subset of the default.
interval        Re-displays selected statistics, pausing interval seconds
                between each display.
                Press CTRL+C to stop re-displaying statistics.  If omitted,
                netstat will print the current configuration information once.
C:\WINDOWS.000>
```

In response to the netstat -e command the following packet and protocol statistics are displayed. This is a summary of events on the network since the last re-boot.

```
C:\WINDOWS.000>netstat -e
Interface Statistics
                        Received            Sent
Bytes                   2442301             1000682
Unicast packets         4769                3776
Non-unicast packets     113                 4566
Discards                0                   0
Errors                  0                   0
Unknown protocols       19
C:\WINDOWS.000>
```

9.5 NBTSTAT

This provides protocol statistics and current TCP/IP connections using NBT (NetBIOS over TCP/IP). This is relevant with Windows 95/98 etc, which uses NetBIOS for the upper layers of the OSI model.

```
C:\WINDOWS.000>nbtstat /?
Displays protocol statistics and current TCP/IP connections using NBT (NetBIOS over
TCP/IP).
NBTSTAT [-a RemoteName] [-A IP address] [-c] [-n] [-r]
[-R] [-s] [S] [interval] ]
-a              (adapter status) Lists the remote machine's name table given
                its name
```

```
-A                  (Adapter status) Lists the remote machine's name table given
                    its IP address.
-c                  (cache) Lists the remote name cache including the IP
                    addresses
-n                  (names) Lists local NetBIOS names.
-r                  (resolved) Lists names resolved by broadcast and via WINS
-R                  (Reload) Purges and reloads the remote cache name table
-S                  (Sessions) Lists sessions table with the destination IP
                    addresses
-s                  (sessions) Lists sessions table converting destination IP
                    addresses to host names via the hosts file.
RemoteName          Remote host machine name.
IP address          Dotted decimal representation of the IP address.
Interval            Re-displays selected statistics, pausing interval seconds
                    between each display. Press Ctrl+C to stop re-displaying
                    statistics.
C:\WINDOWS.000>
```

9.6 IPCONFIG

This shows the entire TCP/IP configuration present in a host. It also has the additional versatility of interfacing with a DHCP server to renew a leased IP address.

Ipconfig will return, amongst other things, the host's IP address, its subnet mask and default gateway.

```
C:\WINDOWS.000>ipconfig /?
Windows 98 IP Configuration
Command line options:
/All - Display detailed information.
/Batch [file]       - Write to file or ./WINIPCFG.OUT
/renew_all          - Renew all adapters.
/release_all        - Release all adapters.
/renew   N          - Renew adapter N.
/release N          - Release adapter N.
C:\WINDOWS.000>
```

An options often used is 'ipconfig /all'. In the case of a multi-homed host, i.e. one with more than one network interface card (including dial-up modem) 'ipconfig /all' will display the details of each card.

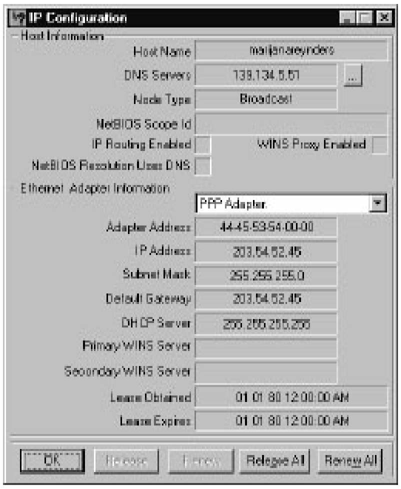
Note that ipconfig will list the generic name of the adapter. Therefore, a 3010 3Com US Robotics 56K modem is simply listed as a PPP adapter, while a Linksys Ethernet 10BaseT/10Base2 Combo PCMCIA card is listed as a generic Novell 2000 adapter, which it emulates.

```
C:\WINDOWS.000>ipconfig /all
Windows 98 IP Configuration
        Host Name . . . . . . . . . : COMPUTER100
        DNS Servers . . . . . . . . :
        Node Type . . . . . . . . . : Broadcast
        NetBIOS Scope ID. . . . . . :
        IP Routing Enabled. . . . . : No
        WINS Proxy Enabled. . . . . : No
        NetBIOS Resolution Uses DNS : No
```

```
0 Ethernet adapter :
            Description . . . . . . . . : PPP Adapter.
            Physical Address. . . . . . : 44-45-53-54-00-00
            DHCP Enabled. . . . . . . . : Yes
            IP Address. . . . . . . . . : 0.0.0.0
            Subnet Mask . . . . . . . . : 0.0.0.0
            Default Gateway . . . . . . :
            DHCP Server . . . . . . . . : 255.255.255.255
            Primary WINS Server . . . . :
            Secondary WINS Server . . . :
            Lease Obtained. . . . . . . :
            Lease Expires . . . . . . . :
1 Ethernet adapter :
            Description . . . . . . . . : Novell 2000 Adapter.
            Physical Address. . . . . . : 00-E0-98-71-57-AF
            DHCP Enabled. . . . . . . . : No
            IP Address. . . . . . . . . : 207.194.66.100
            Subnet Mask . . . . . . . . : 255.255.255.224
            Default Gateway . . . . . . :
            Primary WINS Server . . . . :
            Secondary WINS Server . . . :
            Lease Obtained. . . . . . . :
            Lease Expires . . . . . . . :
C:\WINDOWS.000>
```

9.7 WINIPCFG

Winipcfg (Windows IP Configuration) provides the same information as 'ipconfig /all', but in a Windows format. Like ipconfig, it is capable to force a DHCP server into releasing and reissuing leased IP addresses.



IP Configuration

Host Information

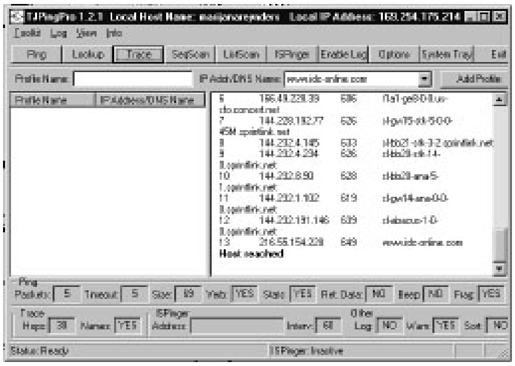
Host Name: marijanareynders
DNS Servers: 139.134.5.51
Node Type: Broadcast
NetBIOS Scope Id:
IP Routing Enabled ☐ WINS Proxy Enabled ☐
NetBIOS Resolution Uses DNS ☐

Ethernet Adapter Information

PPP Adapter
Adapter Address: 44-45-53-54-00-00
IP Address: 203.54.52.45
Subnet Mask: 255.255.255.0
Default Gateway: 203.54.52.45
DHCP Server: 255.255.255.255
Primary WINS Server:
Secondary WINS Server:
Lease Obtained: 01 01 80 12:00:00 AM
Lease Expires: 01 01 80 12:00:00 AM

Figure 9.2
Windows IP configuration

It can be invoked from the DOS prompt, or from the Windows 'run' command. Click the more details tab for an expanded view.



Profile Name:
IP Addr/DNS Name: www.idc-online.com

```
6       166.48.228.39     606   fla1-ge8-0-0.us-
dfo.concert.net
7       144.228.182.77    626   sl-gw15-stk-5-0-0-
45M.sprintlink.net
8       144.232.4.145     633   sl-bb21-stk-3-2.sprintlink.net
9       144.232.4.234     626   sl-bb20-stk-14-
0.sprintlink.net
10      144.232.8.90      628   sl-bb20-ana-5-
1.sprintlink.net
11      144.232.1.102     619   sl-gw14-ana-0-0-
0.sprintlink.net
12      144.232.191.146   639   sl-abacus-1-0-
0.sprintlink.net
13      216.55.154.228    649   www.idc-online.com
Host reached
```

Ping: Packets: 5 Timeout: 5 Size: 69 Verb: YES Stats: YES Ret. Data: NO Beep: NO Flag: YES

Trace: Hops: 30 Names: YES ISPinger: Address: Interv: 60 Other: Log: NO Warn: YES Sort: NO

Status: Ready ISPinger: Inactive

Figure 9.3
Winipcfg display (courtesy of Microsoft Corporation)

9.8 TRACE RouTe

This is often used to trace failures along a TCP/IP communications path. The spelling of the command varies slightly. For UNIX it is **traceroute**, for Windows it is **tracert**.

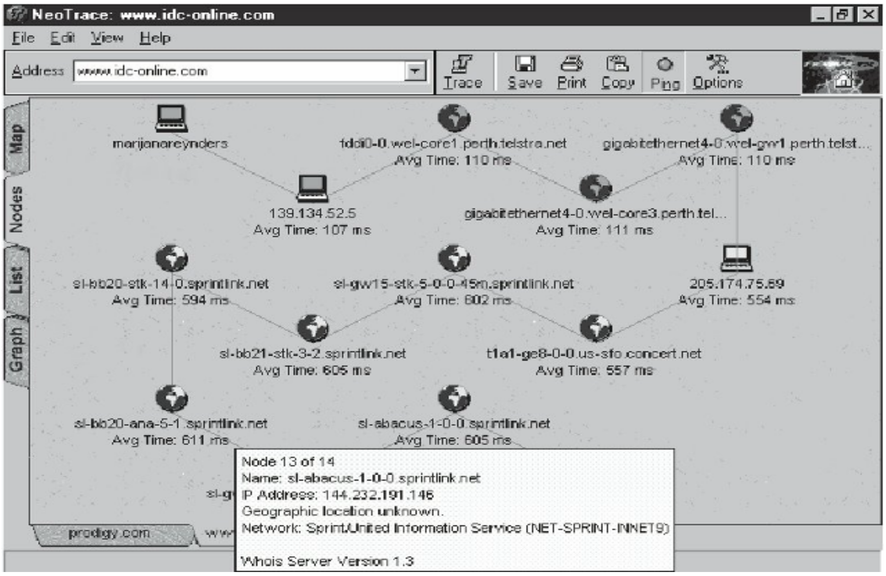
The following figure shows the tracert options.

```
C:\WINDOWS.000>tracert
Usage: tracert [-d] [-h maximum_hops] [-j host-list] [-w timeout] target_name
Options:
-d                    Do not resolve addresses to hostnames.
-h maximum_hops       Maximum number of hops to search for target.
-j host-list          Loose source route along host-list.
-w timeout            Wait timeout milliseconds for each reply.
C:\WINDOWS.000>
```

Here follows a route trace from Perth, Australia, to a server in the USA.

```
C:\WINDOWS.000>tracert www.idc-online.com
Tracing route to www.idc-online.com [216.55.154.228] over a maximum of 30 hops:
1    169ms   160ms   174ms   slip202-135-15-3-0.sy.au.ibm.net [202.135.15.30]
2    213ms   297ms   296ms   152.158.248.250
3    624ms   589ms   533ms   sfra1sr1-2-0-0-5.ca.us.prserv.net [165.87.225.46]
4    545ms   535ms   628ms   sfra1sr2-101-0.ca.us.prserv.net [165.87.33.185]
5    564ms   562ms   573ms   165.87.160.193
6    558ms   564ms   573ms   114.ATM3-0.XR1.SFO1.ALTER.NET
                             [146.188.148.210]
7    574ms   701ms   555ms   187.at-2-10.TR1.SAC1.ALTER.NET
                             [152.63.50.230]
8    491ms   480ms   500ms   127.at-6-10.TR1.LAX9.ALTER.NET
                             [152.63.5.101]
9    504ms   534ms   511ms   297.ATM7-0.XR1.LAX2.ALTER.NET
                             [152.63.112.149]
10   500ms   478ms   491ms   195.ATM9-0-0.GW2.SDG1.ALTER.NET
                             [146.188.249.81]
11   491ms   564ms   584ms   anet-gw.customer.ALTER.NET [157.130.224.154]
12   575ms   554ms   613ms   www.idc-online.com [216.55.154.228]
Trace complete.
C:\WINDOWS.000>
```

As is often the case, the DOS approach is not the user-friendliest option. Notice the result when the same trace is done with TJPingPro. The same TCP/IP protocols viz. ARP and ICMP are still used, but now they are accessed through a third-party application program (TJPingPro) which accesses the TCP/IP stack through a WinSock interface.

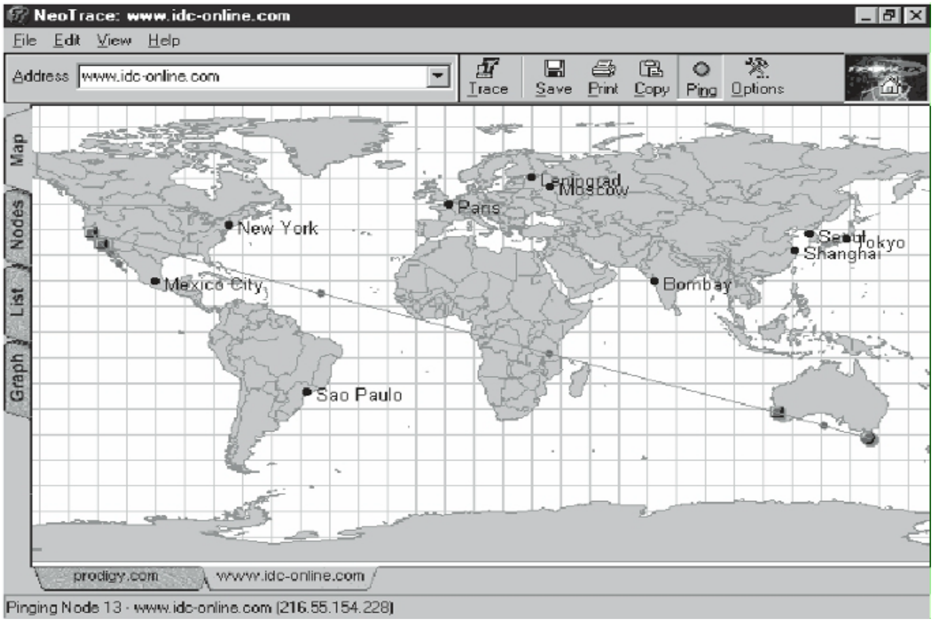

NeoTrace: www.idc-online.com
File Edit View Help
Address www.idc-online.com
Trace Save Print Copy Ping Options
Map
Nodes
List
Graph
marijanareynders
fddi0-0.wel-core1.perth.telstra.net
Avg Time: 110 ms
gigabitethernet4-0.wel-gw1.perth.telst...
Avg Time: 110 ms
139.134.52.5
Avg Time: 107 ms
gigabitethernet4-0.wel-core3.perth.tel...
Avg Time: 111 ms
sl-bb20-stk-14-0.sprintlink.net
Avg Time: 594 ms
sl-gw15-stk-5-0-0-45m.sprintlink.net
Avg Time: 602 ms
205.174.75.69
Avg Time: 554 ms
sl-bb21-stk-3-2.sprintlink.net
Avg Time: 605 ms
t1a1-ge8-0-0.us-sfo.concert.net
Avg Time: 557 ms
sl-bb20-ana-5-1.sprintlink.net
Avg Time: 611 ms
sl-abacus-1-0-0.sprintlink.net
Avg Time: 605 ms
sl-g
prodigy.com
ww

Node 13 of 14
Name: sl-abacus-1-0-0.sprintlink.net
IP Address: 144.232.191.146
Geographic location unknown.
Network: Sprint/United Information Service (NET-SPRINT-INNET9)

Whois Server Version 1.3

Figure 9.4
TJPingPro trace (courtesy of Top Jimmy Software)

The most comprehensive tracing is, however, done via application programs such as Neotrace. The following figures give some of the results of a trace to the same location used for the previous two examples.



Pinging Node 13 - www.idc-online.com (216.55.154.228]

Figure 9.5
NeoTrace display (courtesy NeoWorx Inc)

9.9 ROUTE

The route command is used to configure network routing tables. This may be a tedious task but is sometimes necessary for reasons of security or because a specific route has to be added.

The following shows the route options.

```
C:\WINDOWS.000>route /?
Manipulates network routing tables.
ROUTE [-f] [command [destination] [MASK netmask] [gateway] [METRIC metric]]
-f              Clears the routing tables of all gateway entries.  If this is used
                in conjunction with one of the commands, the tables are
                cleared prior to running the command.
command         Must be one of four:
PRINT           Prints a route
ADD             Adds a route
DELETE          Deletes a route
CHANGE          Modifies an existing route
destination     Specifies the destination host.
MASK            Specifies that the next parameter is the 'netmask' value.
netmask         Specifies a subnet mask value to be associated with this route
                entry. If not specified, it defaults to 255.255.255.255.
METRIC          Specifies that the next parameter 'metric' is the cost for this
                destination
All symbolic names used for destination are looked up in the network database file
NETWORKS. The symbolic names for gateway are looked up in the host name database
file HOSTS.
If the command is PRINT or DELETE, wildcards may be used for the destination and
gateway, or the gateway argument may be omitted.
Diagnostic notes:
Invalid MASK generates an error, that is when (DEST & MASK) != DEST.
Example> route ADD 255.0.0.0  157.0.0.0 MASK 155.0.0.0
         157.55.80.1
The route addition failed: 87
Examples:
> route PRINT
> route    ADD 157.0.0.0  MASK 255.0.0.0       157.55.80.1 METRIC 3
                ^destination    ^mask          ^gateway       ^metric
> route PRINT
> route DELETE 157.0.0.0
> route PRINT
C:\WINDOWS.000>
```

The route table exists on both hosts and routers. An individual entry is read from left to right as follows: 'If a message is destined for network 192.100.100.0, with subnet mask 255.255.255.0, then route it through to the gateway address 192.100.100.1'. Remember that a HostID equal to 0, as used here, does not refer to a specific host but rather to the network as a whole.

Routes can also be added with the route add and route delete commands.

Route add 192.100.100.0 mask 255.255.255.0 192.100.100.1 will add a route and Route delete 192.100.100.0 will delete a particular route. Manual adding of routes are sometimes necessary, for example in the case where the installation of dial-up proxy server software on a given host sometimes overwrites the existing default gateway setting on that host in order to ‘point’ to the Internet service provider’s default gateway. This makes it impossible for the host to reach an existing adjacent network across the intermediate router, unless a manual entry is made. If said entry ‘does the job’ but disappears when the host is re-booted, the appropriate route command needs to be included in the autoexec.bat file.

The following response was obtained from the route print command.

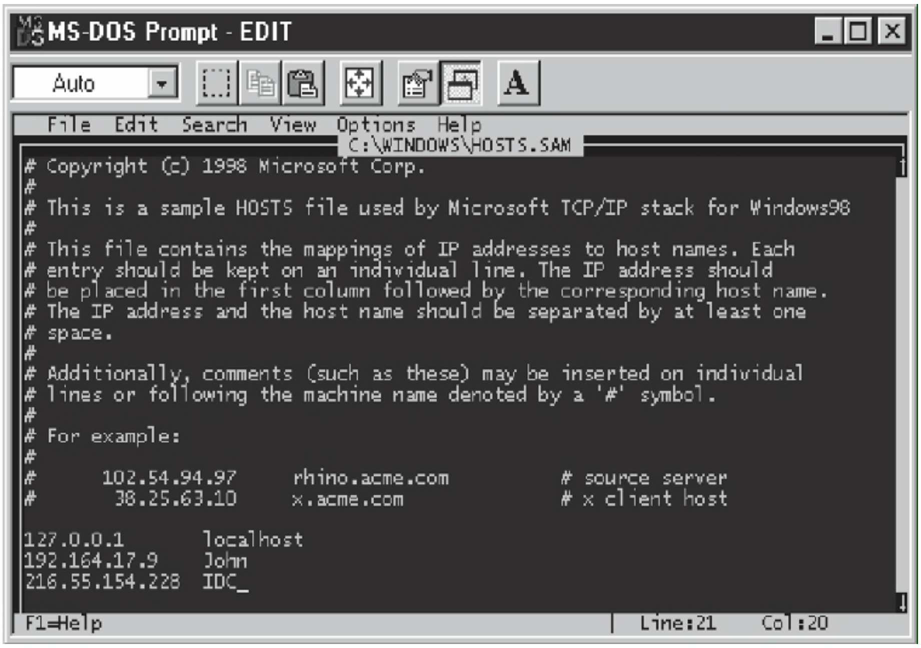
```
Active routes:
Network Address   Netmask           Gateway Address     Interface         Metric
127.0.0.0         255.0.0.0         127.0.0.1 1         27.0.0.1          1
207.194.66.96     255.255.255.224   207.194.66.100      207.194.66.100    1
207.194.66.100    255.255.255.255   127.0.0.1           127.0.0.1         1
207.194.66.255    255.255.255.255   207.194.66.100      207.194.66.100    1
224.0.0.0         224.0.0.0         207.194.66.100      207.194.66.100    1
255.255.255.255   255.255.255.255   207.194.66.100      0.0.0.0           1
C:\WINDOWS.000>
```

9.10 The HOSTS file

The hosts file is used on UNIX and Windows systems to resolve the mapping of a ‘name’ (any given name) to an IP address.

The following is an example of a typical Windows hosts file. This file is saved in the same directory as Windows itself as c:\windows\hosts. If a user is uncertain about the correct format of the entries, a sample file can be found at c:\windows\hosts.sam. Note that, as a matter of convenience, the hosts sample file can be edited as in the accompanying example, but it MUST then be saved as hosts only, i.e. without the. same extension.

In the example, host 192.100.100.2 can simply be interrogated by typing ping john.



```
# Copyright (c) 1998 Microsoft Corp.
#
# This is a sample HOSTS file used by Microsoft TCP/IP stack for Windows98
#
# This file contains the mappings of IP addresses to host names. Each
# entry should be kept on an individual line. The IP address should
# be placed in the first column followed by the corresponding host name.
# The IP address and the host name should be separated by at least one
# space.
#
# Additionally, comments (such as these) may be inserted on individual
# lines or following the machine name denoted by a '#' symbol.
#
# For example:
#
#      102.54.94.97     rhino.acme.com          # source server
#       38.25.63.10     x.acme.com              # x client host

127.0.0.1       localhost
192.164.17.9    John
216.55.154.228  IDC_
```

Figure 9.6
The Hosts file (courtesy of Microsoft Corporation)

10

LAN system components

Objectives

When you have completed this chapter you should be able to:

- Explain the basic function of each of the devices listed under 10.1
- Explain the fundamental differences between the operation and application of switches (layer 2 and 3), bridges and routers

10.1 Introduction

In the design of an Ethernet system there are a number of different components that can be used. These include:

- Repeaters
- Media converters
- Bridges
- Hubs
- Switches
- Routers
- Gateways
- Print servers
- Terminal servers
- Remote access servers
- Time servers
- Thin servers

The lengths of LAN segments are limited due to physical and collision domain constraints and there is often a need to increase this range. This can be achieved by means of a number of interconnecting devices, ranging from repeaters to gateways. It may also be necessary to partition an existing network into separate networks for reasons of security or traffic overload.

In modern network devices the functions mentioned above are often mixed:

- A shared 10BaseT hub is, in fact, a multi-port repeater
- A layer II switch is essentially a multi-port bridge
- Segmentable and dual-speed shared hubs make use of internal bridges
- Switches can function as bridges, a two-port switch being none other than a bridge
- Layer III switches function as routers

These examples are not meant to confuse the reader, but serve to emphasize the fact that the functions should be understood, rather than the 'boxes' in which they are packaged.

10.2 Repeaters

A repeater operates at the physical layer of the OSI model (layer 1) and simply retransmits incoming electrical signals. This involves amplifying and re-timing the signals received on one segment onto all other segments, without considering any possible collisions. All segments need to operate with the same media access mechanism and the repeater is unconcerned with the meaning of the individual bits in the packets. Collisions, truncated packets or electrical noise on one segment are transmitted onto all other segments.

10.2.1 Packaging

Repeaters are packaged either as stand-alone units (i.e. desktop models or small cigarette package-sized units) or 19" rack-mount units. Some of these can link two segments only, while larger rack-mount modular units (called Concentrators) are used for linking multiple segments. Regardless of packaging, repeaters can be classified either as local repeaters (for linking network segments that are physically in close proximity), or as remote repeaters for linking segments that are some distance apart.

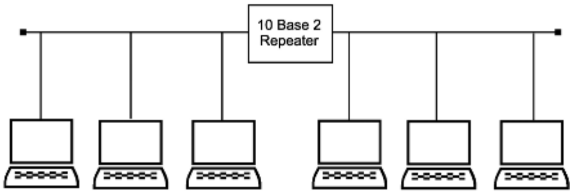


Figure 10.1
Repeater application

10.2.2 Local Ethernet repeaters

Several options are available:

- Two-port local repeaters offer most combinations of 10Base5, 10Base2, 10BaseT and 10Base-FL such as 10Base5/10Base5, 10Base2/10Base2, 10Base5/10Base2, 10Base2/10BaseT, 10BaseT/10BaseT and 10Base-FL/10Base-FL. By using such devices (often called boosters or extenders) one can, for example, extend the distance between a computer and a 10BaseT hub by up to 100 m, or extend a 10Base-FL link between two devices (such as bridges) by up to 2 km

- Multi-port local repeaters offer several ports of the same type (e.g. 4× 10Base2 or 8× 10Base5) in one unit, often with one additional connector of a different type (e.g. 10Base2 for a 10Base5 repeater). In the case of 10BaseT the cheapest solution is to use an off-the-shelf 10BaseT shared hub, which is effectively a multi-port repeater
- Multi-port local repeaters are also available as chassis-type units; i.e. as frames with common back planes and removable units. An advantage of this approach is that 10Base2, 10Base5, 10BaseT and 10Base-FL can be mixed in one unit, with an option of SNMP management for the overall unit. These are also referred to as Concentrators

10.2.3 Remote repeaters

Remote repeaters, on the other hand, have to be used in pairs with one repeater connected to each network segment and a fiber-optic link between the repeaters. On the network side they typically offer 10Base5, 10Base2 and 10BaseT. On the interconnecting side the choices include 'single pair Ethernet', using telephone cable up to 457 m in length, or single mode/multimode optic fiber, with various connector options. With 10Base-FL (backwards compatible with the old FOIRL standard), this distance can be up to 1.6 km.

In conclusion it must be emphasized that although repeaters are probably the cheapest way to extend a network, they do so without separating the collision domains, or network traffic. They simply extend the physical size of the network. All segments joined by repeaters therefore share the same bandwidth and collision domain.

10.3 Media converters

Media converters are essentially repeaters, but interconnect mixed media viz. copper and fiber. An example would be 10BaseT/10Base-FL. As in the case of repeaters, they are available in single and multi-port options, and in stand-alone or chassis type configurations. The latter option often features remote management via SNMP.

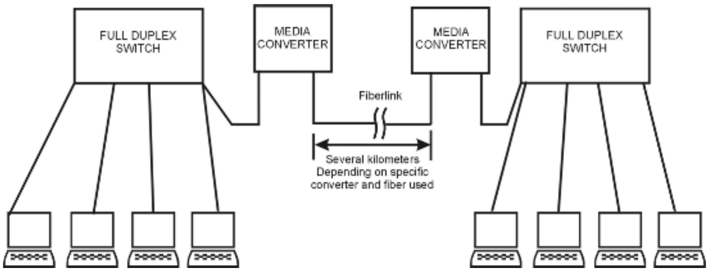


Figure 10.2
Media converter application

Models may vary between manufacturers, but generally Ethernet media converters support:

- 10 Mbps (10Base2, 10BaseT, 10Base-FL – single and multi-mode)
- 100 Mbps (fast) Ethernet (100Base-TX, 100Base-FX – single and multimode)
- 1000 Mbps (gigabit) Ethernet (single and multimode)

An added advantage of the fast and gigabit Ethernet media converters is that they support full-duplex operation that effectively doubles the available bandwidth.

10.4 Bridges

Bridges operate at the data link layer of the OSI model (layer 2) and are used to connect two separate networks to form a single large continuous LAN. The overall network, however, still remains one network with a single network ID (NetID). The bridge only divides the network up into two segments, each with its own collision domain and each retaining its full (say, 10 Mbps) bandwidth. Broadcast transmissions are seen by all nodes, on both sides of the bridge.

The bridge exists as a node on each network and passes only valid messages across to destination addresses on the other network. The decision as to whether or not a frame should be passed across the bridge is based on the layer 2 address, i.e. the media (MAC) address. The bridge stores the frame from one network and examines its destination MAC address to determine whether it should be forwarded across the bridge.

Bridges can be classified as either MAC or LLC bridges, the MAC sublayer being the lower half of the data link layer and the LLC sublayer being the upper half. For MAC bridges the media access control mechanism on both sides must be identical; thus it can bridge only Ethernet to Ethernet, token ring to token ring and so on. For LLC bridges, the data link protocol must be identical on both sides of the bridge (e.g. IEEE 802.2 LLC); however, the physical layers or MAC sublayers do not necessarily have to be the same. Thus the bridge isolates the media access mechanisms of the networks. Data can therefore be transferred, for example, between Ethernet and token ring LANs. In this case, collisions on the Ethernet system do not cross the bridge nor do the tokens.

Bridges can be used to extend the length of a network (as with repeaters) but in addition they improve network performance. For example, if a network is demonstrating fairly slow response times, the nodes that mainly communicate with each other can be grouped together on one segment and the remaining nodes can be grouped together in another segment. The busy segment may not see much improvement in response rates (as it is already quite busy) but the lower activity segment may see quite an improvement in response times. Bridges should be designed so that 80% or more of the traffic is within the LAN and only 20% cross the bridge. Stations generating excessive traffic should be identified by a protocol analyzer and relocated to another LAN.

10.4.1 Intelligent bridges

Intelligent bridges (also referred to as transparent or spanning-tree bridges) are the most commonly used bridges because they are very efficient in operation and do not need to be taught the network topology. A transparent bridge learns and maintains two address lists corresponding to each network it is connected to. When a frame arrives from the one Ethernet network, its source address is added to the list of source addresses for that network. The destination address is then compared to that of the two lists of addresses for each network and a decision made whether to transmit the frame onto the other network. If no corresponding address to the destination node is recorded in either of these two lists the message is retransmitted to all other bridge outputs (flooding), to ensure the message is delivered to the correct network. Over a period of time, the bridge learns all the addresses on each network and thus avoids unnecessary traffic on the other network. The bridge also maintains time out data for each entry to ensure the table is kept up to date and old entries purged.

Transparent bridges cannot have loops that could cause endless circulation of packets. If the network contains bridges that could form a loop as shown in Figure 10.3, one of the bridges (C) needs to be made redundant and deactivated.

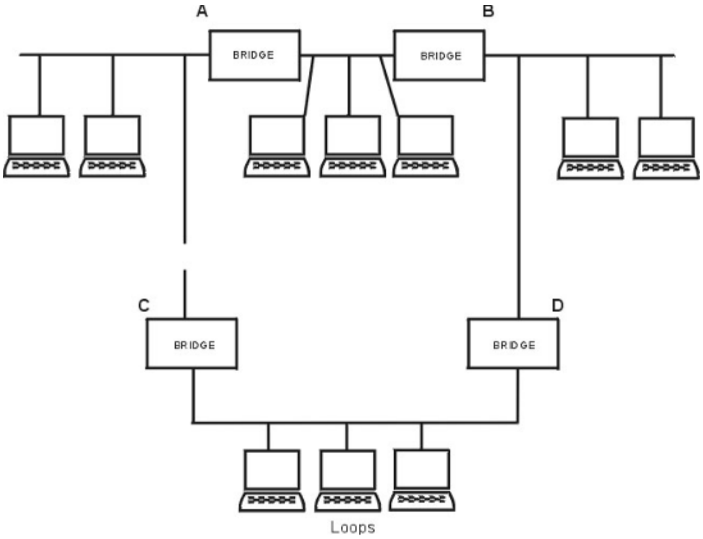


Figure 10.3
Avoidance of loops in bridge networks

The spanning tree algorithm (IEEE 802.1d) is used to manage paths between segments having redundant bridges. This algorithm designates one bridge in the spanning tree as the root and all other bridges transmit frames towards the root using a least cost metric. Redundant bridges can be reactivated if the network topology changes.

10.4.2 Source-routing bridges

Source-routing (SR) bridges are popular for IBM token ring networks. In these networks, the sender must determine the best path to the destination. This is done by sending a discovery frame that circulates the network and arrives at the destination with a record of the path token. These frames are returned to the sender who can then select the best path. Once the path has been discovered, the source updates its routing table and includes the path details in the routing information field in the transmitted frame.

10.4.3 SRT and translational bridges

When connecting Ethernet networks to token ring networks, either **source-routing transparent** (SRT) bridges or translational bridges are used. SRT bridges are a combination of a transparent and source-routing bridge, and are used to interconnect Ethernet (IEEE802.3) and token ring (IEE802.5) networks. It uses source routing of the data frame if it contains routing information; otherwise it reverts to transparent bridging. Translational bridges, on the other hand, translate the routing information to allow source-routing networks to bridge to transparent networks. The IBM 8209 is an example of this type of bridge.

10.4.4 Local vs remote bridges

Local bridges are devices that have two network ports and hence interconnect two adjacent networks at one point. This function is currently often performed by switches, being essentially intelligent multi-port bridges.

A very useful type of local bridge is a 10/100 Mbps Ethernet bridge, which allows interconnection of 10BaseT, 100Base-TX and 100Base-FX networks, thereby performing the required speed translation. These bridges typically provide full-duplex operation on

100Base-TX and 100Base-FX, and employ internal buffers to prevent saturation of the 10BaseT port.

Remote bridges, on the other hand, operate in pairs with some form of interconnection between them. This interconnection can be with or without modems, and include RS-232/V.24, V.35, RS-422, RS-530, X.21, 4-wire, or fiber (both single and multi-mode). The distance between bridges can typically be up to 1.6 km.

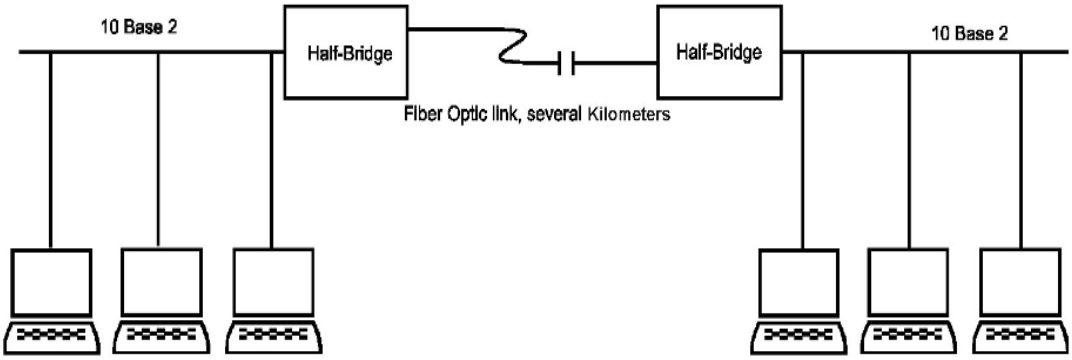


Figure 10.4
Remote bridge application

10.5 Hubs

Hubs are used to interconnect hosts in a physical star configuration. This section will deal with Ethernet hubs, which are of the 10/100/100BaseT variety. They are available in many configurations, some of which will be discussed below.

10.5.1 Desktop vs stackable hubs

Smaller desktop units are intended for stand-alone applications, and typically have 5 to 8 ports. Some 10BaseT desktop models have an additional 10Base2 port. These devices are often called workgroup hubs.

Stackable hubs, on the other hand, typically have up to 24 ports and can be physically stacked and interconnected to act as one large hub without any repeater count restrictions. These stacks are often mounted in 19-inch cabinets.

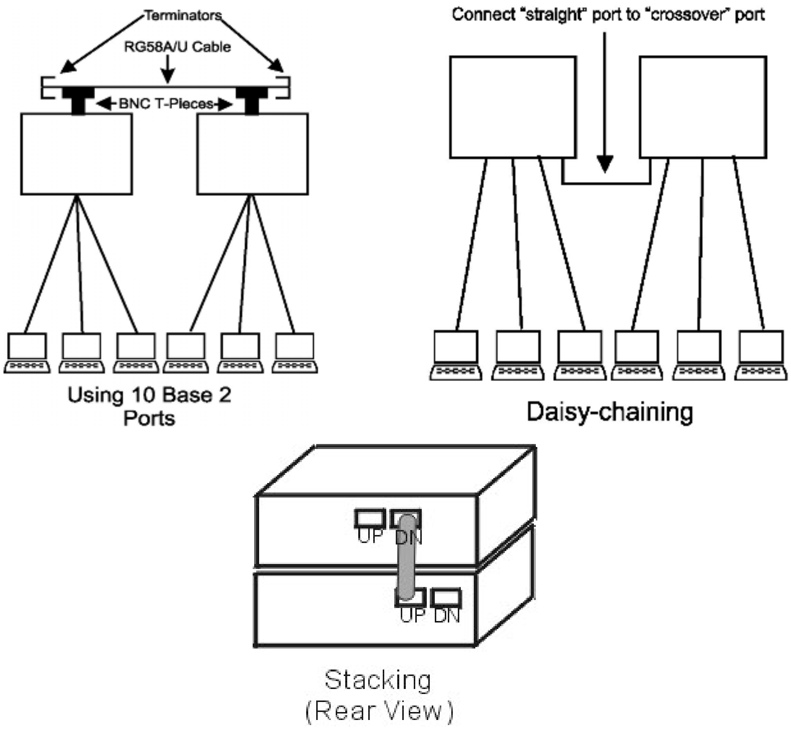


Figure 10.5
10BaseT hub interconnection

10.5.2 Shared vs switched hubs

Shared hubs interconnect all ports on the hub in order to form a logical bus. This is typical of the cheaper workgroup hubs. All hosts connected to the hub share the available bandwidth since they all form part of the same collision domain.

Although they physically look alike, switched hubs (better known as switches) allow each port to retain and share its full bandwidth only with the hosts connected to that port. Each port (and the segment connected to that port) functions as a separate collision domain. This attribute will be discussed in more detail in the section on switches.

10.5.3 Managed hubs

Managed hubs have an on-board processor with its own MAC and IP address. Once the hub has been set up via a PC on the hub's serial (COM) port, it can be monitored and controlled via the network using SNMP or RMON. The user can perform activities such as enabling/disabling individual ports, performing segmentation (see next section), monitoring the traffic on a given port, or setting alarm conditions for a given port.

10.5.4 Segmentable hubs

On a non-segmentable (i.e. shared) hub, all hosts share the same bandwidth. On a segmentable hub, however, the ports can be grouped, under software control, into several shared groups. All hosts on each segment then share the full bandwidth on that segment, which means that a 24-port 10BaseT hub segmented into 4 groups effectively supports 40 Mbps. The configured segments are internally connected via bridges, so that all ports can still communicate with each other if needed.

10.5.5 Dual-speed hubs

Some hubs offer dual-speed ports, e.g. 10BaseT/100Base-T. These ports are auto-configured, i.e. each port senses the speed of the NIC connected to it, and adjusts its own speed accordingly. All the 10BaseT ports connect to a common low-speed internal segment, while all the 100BaseT ports connect to a common high-speed internal segment. The two internal segments are interconnected via a speed-matching bridge.

10.5.6 Modular hubs

Some stackable hubs are modular, allowing the user to configure the hub by plugging in a separate module for each port. Ethernet options typically include both 10 and 100 Mbps, with either copper or fiber. These hubs are sometimes referred to as chassis hubs.

10.5.7 Hub interconnection

Stackable hubs are best interconnected by means of special stacking cables attached to the appropriate connectors on the back of the chassis.

An alternative method for non-stackable hubs is by ‘daisy-chaining’ an interconnecting port on each hub by means of a UTP patch cord. Care has to be taken not to connect the transmit pins on the ports together (and, for that matter, the receive pins) – it simply will not work. This is similar to interconnecting two COM ports with a ‘straight’ cable i.e. without a null modem. Connect transmit to receive and vice versa by (a) using a crossover cable and interconnecting two ‘normal’ ports, or (b) using a normal (‘straight’) cable and utilizing a crossover port on one of the hubs. Some hubs have a dedicated uplink (crossover) port while others have a port that can be manually switched into crossover mode.

A third method that can be used on hubs with a 10Base2 port is to create a backbone. Attach a BNC T-piece to each hub, and interconnect the T-pieces with RG 58 coax cable. The open connections on the extreme ends of the backbone obviously have to be terminated.

Fast Ethernet hubs need to be deployed with caution because the inherent propagation delay of the hub is significant in terms of the 5.12 microsecond collision domain size. Fast Ethernet hubs are classified as class I, II or II+, and the class dictates the number of hubs that can be interconnected. For example, class II dictates that there may be no more than two hubs between any given pair of nodes, that the maximum distance between the two hubs shall not exceed 5 m, and that the maximum distance between any two nodes shall not exceed 205 m. The safest approach, however, is to follow the guidelines of each manufacturer.

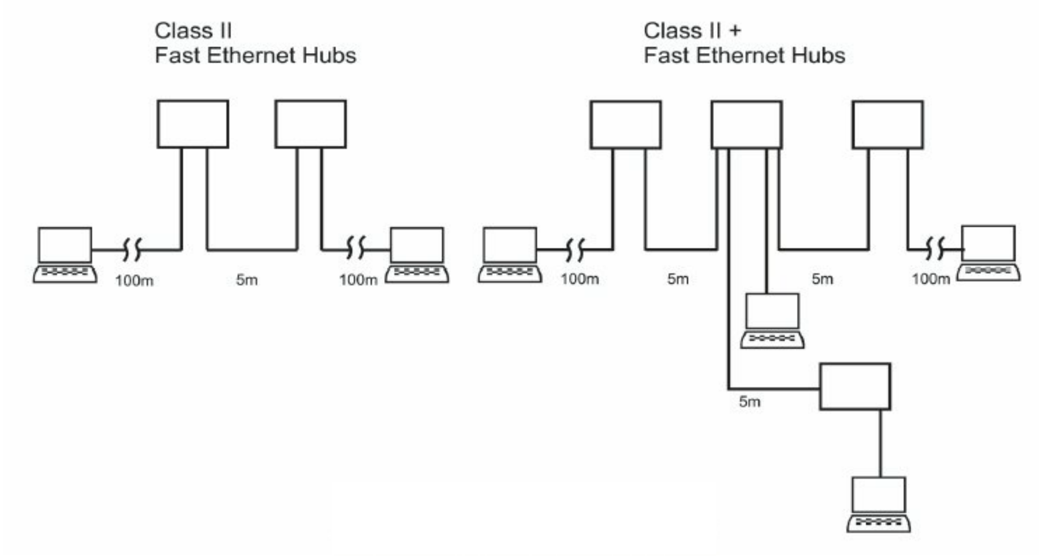


Figure 10.6
Fast Ethernet hub interconnection

10.6 Switches

Ethernet switches are an expansion of the concept of bridging and are, in fact, intelligent (self-learning) multi-port bridges. They enable frame transfers to be accomplished between any pair of devices on a network, on a per-frame basis. Only the two ports involved 'see' the specific frame. Illustrated below is an example of an 8 port switch, with 8 hosts attached. This comprises a physical star configuration, but it does not operate as a logical bus as an ordinary hub does. Since each port on the switch represents a separate segment with its own collision domain, it means that there are only 2 devices on each segment, namely the host and the switch port. Hence, in this particular case, there can be no collisions on any segment!

In the sketch below hosts 1 & 7, 3 & 5 and 4 & 8 need to communicate at a given moment, and are connected directly for the duration of the frame transfer. For example, host 7 sends a packet to the switch, which determines the destination address, and directs the package to port 1 at 10 Mbps.

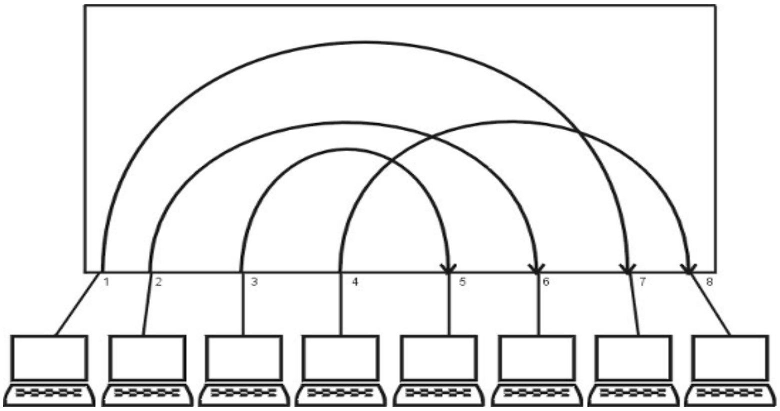


Figure 10.7
8-Port Ethernet switch

If host 3 wishes to communicate with host 5, the same procedure is repeated. Provided that there are no conflicting destinations, a 16-port switch could allow 8 concurrent frame exchanges at 10 Mbps, rendering an effective bandwidth of 80 Mbps. On top of this, the switch could allow full-duplex operation, which would double this figure.

10.6.1 Cut-through vs store-and-forward

Switches have two basic architectures, cut-through and store-and-forward. In the past, cut-through switches were faster because they examined the packet destination address only before forwarding the frame to the destination segment. A store-and-forward switch, on the other hand, accepts and analyzes the entire packet before forwarding it to its destination. It takes more time to examine the entire packet, but it allows the switch to catch certain packet errors and keep them from propagating through the network. The speed of modern store-and-forward switches has caught up with cut-through switches so that the speed difference between the two is minimal. There are also a number of hybrid designs that mix the two architectures.

Since a store-and-forward switch buffers the frame, it can delay forwarding the frame if there is traffic on the destination segment, thereby adhering to the CSMA/CD protocol. In the case of a cut-through switch this is a problem, since a busy destination segment means that the frame cannot be forwarded, yet it cannot be stored either. The solution is to force a collision on the source segment, thereby enticing the source host to retransmit the frame.

10.6.2 Layer 2 switches vs layer 3 switches

Layer 2 switches operate at the data link layer of the OSI model and derive their addressing information from the destination MAC address in the Ethernet header. Layer 3 switches, on the other hand, obtain addressing information from the network layer, namely from the destination IP address in the IP header. Layer 3 switches are used to replace routers in LANs as they can do basic IP routing (supporting protocols such as RIP and RIPv2) at almost 'wire-speed'; hence they are significantly faster than routers.

10.6.3 Full-duplex switches

An additional advancement is full-duplex Ethernet where a device can simultaneously transmit AND receive data over one Ethernet connection. This requires a different Ethernet NIC in the host, as well as a switch that supports full-duplex. This enables two devices to transmit and receive simultaneously via a switch. The node automatically negotiates with the switch and uses full-duplex if both devices can support it.

Full-duplex is useful in situations where large amounts of data are to be moved around quickly, for example between graphics workstations and file servers.

10.6.4 Switch applications

High-speed aggregation

Switches are very efficient in providing a high-speed aggregated connection to a server or backbone. Apart from the normal lower-speed (say, 10BaseT) ports, switches have a high-speed uplink port (100Base-TX). This port is simply another port on the switch, accessible by all the other ports, but features a speed conversion from 10 Mbps to 100 Mbps.

Assume that the uplink port was connected to a file server. If all the other ports (say, eight times 10BaseT) wanted to access the server concurrently, this would necessitate a bandwidth of 80 Mbps in order to avoid a bottleneck and subsequent delays. With a 10BaseT uplink port this would create a serious problem. However, with a 100Base-TX uplink there is still 20 Mbps of bandwidth to spare.

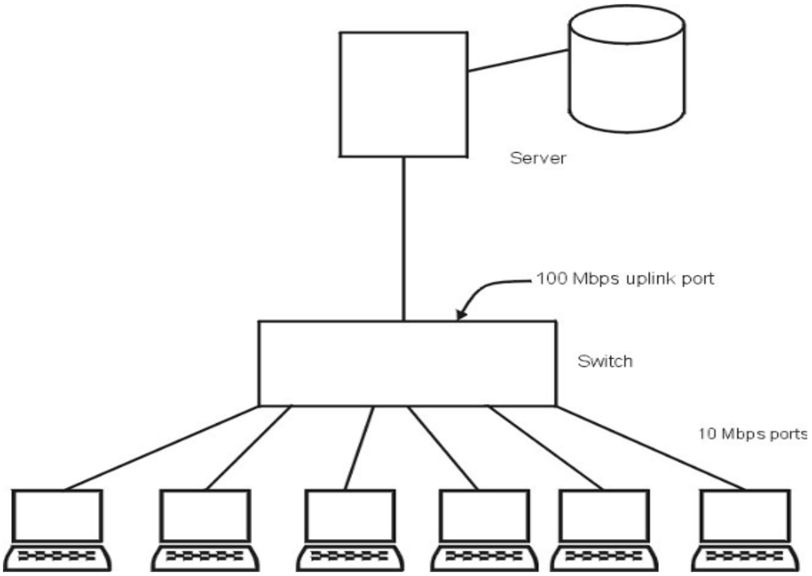


Figure 10.8
Using a switch to connect users to a server

Backbones

Switches are very effective in backbone applications, linking several LANs together as one, yet segregating the collision domains. An example could be a switch located in the basement of a building, linking the networks on different floors of the building. Since the actual ‘backbone’ is contained within the switch, it is known in this application as a ‘collapsed backbone’.

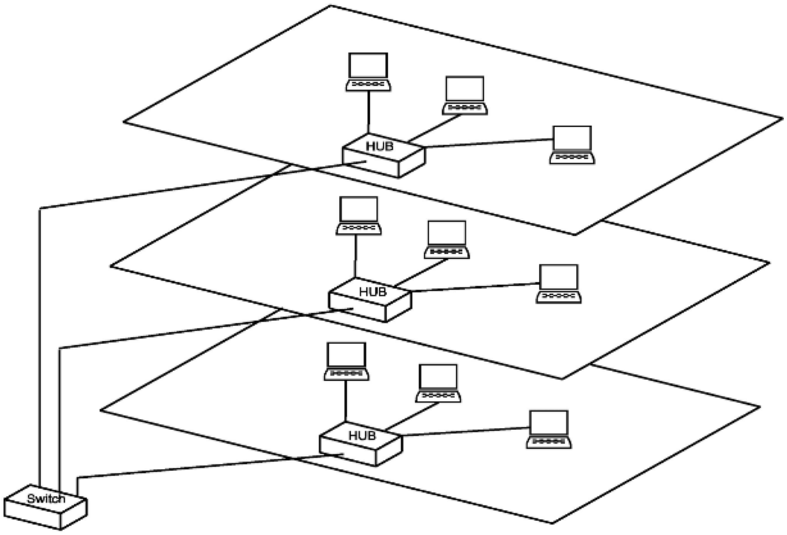


Figure 10.9
Using a switch as a backbone

VLANs and deterministic Ethernet

Provided that a LAN is constructed around switches that support VLANs, individual hosts on the physical LAN can be grouped into smaller Virtual LANs (VLANs), totally invisible to their fellow hosts. Unfortunately, the 'standard' Ethernet/ IEEE802.3 header does not contain sufficient information to identify members of each VLAN; hence, the frame had to be modified by the insertion of a 'tag', between the Source MAC address and the type/length fields. This modified frame is known as an Ethernet 802.1Q tagged frame and is used for communication between the switches.

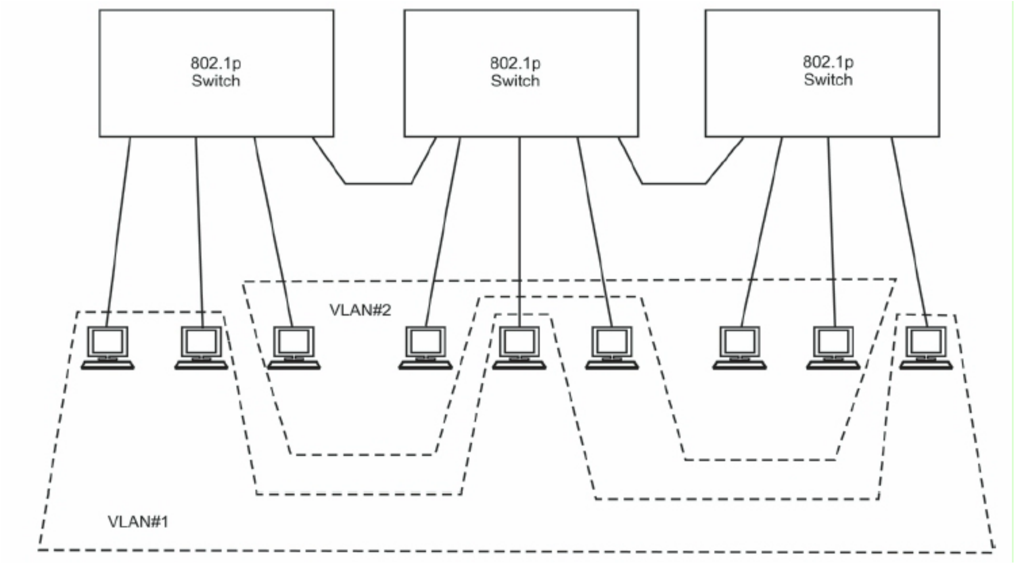


Figure 10.10
Virtual LANs using switches

The IEEE 802.1p committee has defined a standard for packet-based LANs that supports layer 2 traffic prioritization in a switched LAN environment. IEEE 802.1p is part of a larger initiative (IEEE 802.1p/Q) that adds more information to the Ethernet header (as shown in Fig 10.11) to allow networks to support VLANs and traffic prioritization.

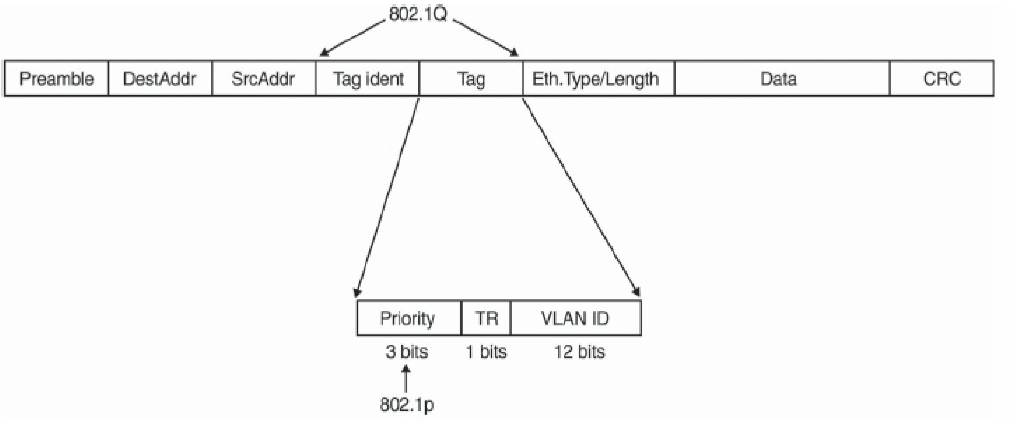


Figure 10.11
IEEE 802.1p/Q modified Ethernet header

802.1p/Q adds 16 bits to the header, of which three are for a priority tag and twelve for a VLAN ID number. This allows for eight discrete priority layers from 0 (high) to 7 (low) that support different kinds of traffic in terms of their delay-sensitivity. Since IEEE 802.1p/Q operates at layer II, it supports prioritization for all traffic on the VLAN, both IP and non-IP. This introduction of priority layers enables so-called deterministic Ethernet where, instead of contending for access to a bus, a source node can pass a frame directly to a destination node on the basis of its priority, and without risk of any collisions.

10.7 Routers

Unlike bridges and layer 2 switches, routers operate at layer 3 of the OSI model, namely at the network layer (or, the Internet layer of the DOD model). They therefore ignore address information contained within the data link layer (the MAC addresses) and rather delve deeper into each frame and extract the address information contained in the network layer. For TCP/IP this is the IP address.

Like bridges or switches, routers appear as hosts on each network that it is connected to. They are connected to each participating network through an NIC, each with a MAC address as well as an IP address. Each NIC has to be assigned an IP address with the same NetID as the network it is connected to. This IP address allocated to each network is known as the default gateway for that network and each host on the internetwork requires at least one default gateway (but could have more). The default gateway is the IP address to which any host must forward a packet if it finds that the NetID of the destination and the local NetID do not match, which implies remote delivery of the packet.

A second major difference between routers and bridges or switches is that routers will not act autonomously but rather have to be GIVEN the frames that need to be forwarded. A host to the designated default gateway forwards such frames.

Protocol dependency

Because routers operate at the network layer, they are used to transfer data between two networks that have the same Internet layer protocols (such as IP) but not necessarily the same physical or data link protocols. Routers are therefore said to be protocol dependent, and have to be able to handle all the Internet layer protocols present on a particular network. A network utilizing Novell Netware therefore requires routers that can accommodate IPX (Internet packet exchange) – the network layer component of SPX/IPX. If this network has to handle Internet access as well, it can only do this via IP, and hence the routers will need to be upgraded to models that can handle both IPX and IP.

Routers maintain tables of the networks that they are connected to and of the optimum path to reach a particular network. They then redirect the message to the next router along that path.

10.7.1 Two-port vs multi-port routers

Multi-port routers are chassis-based devices with modular construction. They can interconnect several networks. The most common type of router is, however, a 2-port router. Since these are invariably used to implement WANs, they connect LANs to a ‘communications cloud’; the one port will be a local LAN port e.g. 10BaseT, but the second port will be a WAN port such as X.25.

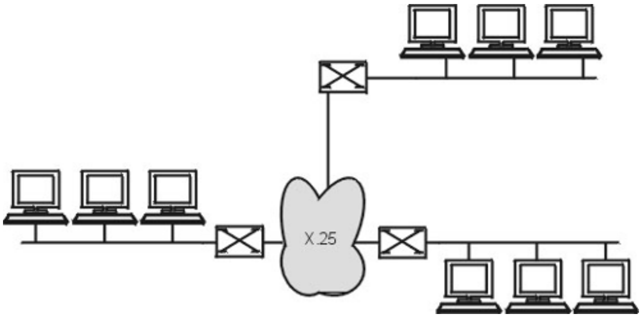


Figure 10.12
Implementing a WAN with 2-port routers (gateways)

10.7.2 Access routers

Access routers are 2-port routers that use dial-up access rather than a permanent (e.g. X.25) connection to connect a LAN to an ISP and hence to the ‘communications cloud’ of the Internet. Typical options are ISDN or dial-up over telephone lines, using either the V.34 (ITU 33.6 kbps) or V.90 (ITU 56 kbps) standard. Some models allow multiple phone lines to be used, using multilink PPP, and will automatically dial up a line when needed or redial when a line is dropped, thereby creating a ‘virtual leased line’.

10.7.3 Border routers

Routers within an autonomous system normally communicate with each other using an interior gateway protocol such as RIP. However, routers within an autonomous system that also communicate with remote autonomous systems need to do that via an exterior gateway protocol such as BGP-4. Whilst doing this, they still have to communicate with other routers within their own autonomous system, e.g. via RIP. These routers are referred to as border routers.

10.7.4 Routing vs bridging

It sometimes happens that a router is confronted with a layer 3 (network layer) address it does not understand. In the case of an IP router, this may be a Novell IPX address. A similar situation will arise in the case of NetBIOS/NetBEUI, which is non-routable. A ‘brouter’ (bridging router) will revert to a bridge if it cannot understand the layer 3 protocol, and in this way forward the packet towards its destination. Most modern routers have this function built in.

10.8 Gateways

Gateways are network interconnection devices, not to be confused with default gateways which are the IP addresses to which packets are forwarded for subsequent routing (indirect delivery).

A gateway is designed to connect dissimilar networks and could operate anywhere from layer 4 to layer 7 of the OSI model. In a worst case scenario, a gateway may be required to decode and re-encode all seven layers of two dissimilar networks connected to either side, for example when connecting an Ethernet network to an IBM SNA network. Gateways thus have the highest overhead and the lowest performance of all the internetworking devices. The gateway translates from one protocol to the other and handles differences in physical signals, data format, and speed.

Since gateways are, per definition, protocol converters, it so happens that a 2-port (WAN) router could also be classified as a gateway since it has to convert both layer 1 and layer 2 on the LAN side (say, Ethernet) to layer 1 and layer 2 on the WAN side (say, X.25). This leads to the confusing practice of referring to (WAN) routers as gateways.

10.9 Print servers

Print servers are devices, attached to the network, through which printers can be made available to all users. Typical print servers cater for both serial and parallel printers. Some also provide concurrent multi-protocol support, which means that they support multiple protocols and will execute print jobs on a first-come first-served basis regardless of the protocol used. Protocols supported could include SPX/IPX, TCP/IP, AppleTalk/EtherTalk, NetBIOS/NetBEUI, or DEC LAT.

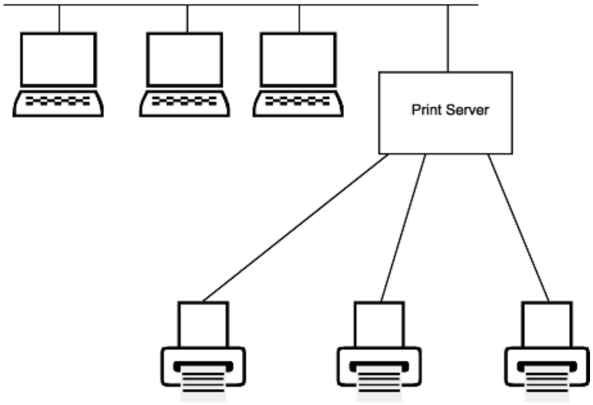


Figure 10.13
Print server applications

10.10 Terminal servers

Terminal servers connect multiple (typically up to 32) serial (RS-232) devices such as system consoles, data entry terminals, bar code readers, scanners, and serial printers to a network. They support multiple protocols such as TCP/IP, SPX/IPX, NetBIOS/NetBEUI, AppleTalk and DEC LAT, which means that they not only can handle devices which support different protocols, but that they can also provide protocol translation between ports.

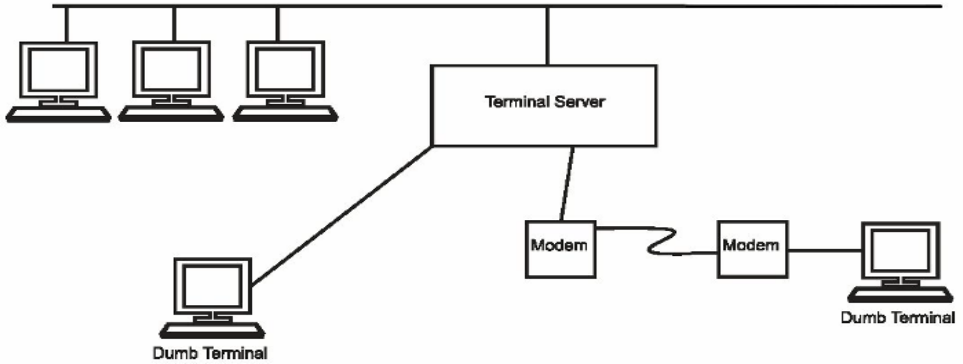


Figure 10.14
Terminal server applications

10.11 Thin servers

Thin servers are essentially single-channel terminal servers. They provide connectivity between Ethernet (10BaseT/100Base-TX) and any serial devices with RS-232 or RS-485 ports. They implement the bottom 4 layers of the OSI model with Ethernet and layer 3/4 protocols such as TCP/IP, SPX/IPX and DEC LAT.

A special version, the industrial thin server, is mounted in a rugged DIN rail package. It can be configured over one of its serial ports, and managed via TELNET or SNMP. A software redirector package enables a user to remove a serial device such as a weighbridge from its controlling computer, locate it elsewhere, then connect it via a thin server to an Ethernet network through the nearest available hub. All this is done without modifying any software. A software package called a port redirector makes the computer 'think' that it is still communicating via the weighbridge via the COM port while, in fact, the data and control messages to the device are routed via the network.

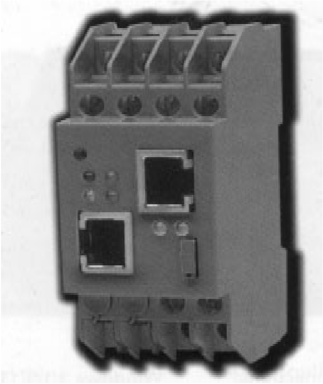


Figure 10.15
Industrial thin server (courtesy of Lantronix)

10.12 Remote access servers

Remote access servers are devices that allow users to dial into a network via analog telephone or ISDN. Typical remote access servers support between 1 and 32 dial-in users via PPP or SLIP. User authentication can be done via Radius, Kerberos or SecurID. Some offer dial-back facilities whereby the user authenticates to the server's internal table, after which the server dials back the user so that the cost of the connection is carried by the network and not the remote user.

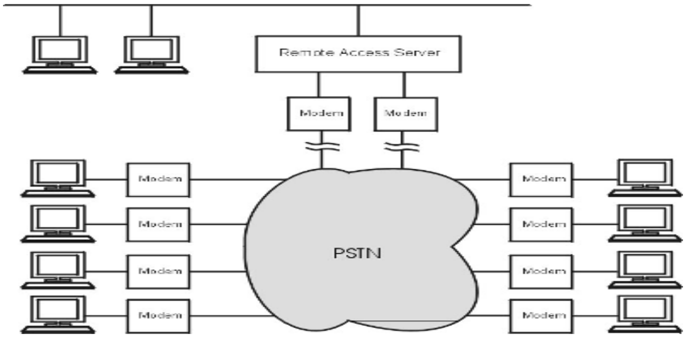


Figure 10.16
Remote access server application (courtesy of Lantronix)

10.13 Network timeservers

Network time-servers are stand-alone devices that compute the correct local time by means of a global positioning system (GPS) receiver, and then distribute it across the network by means of the network time protocol (NTP).

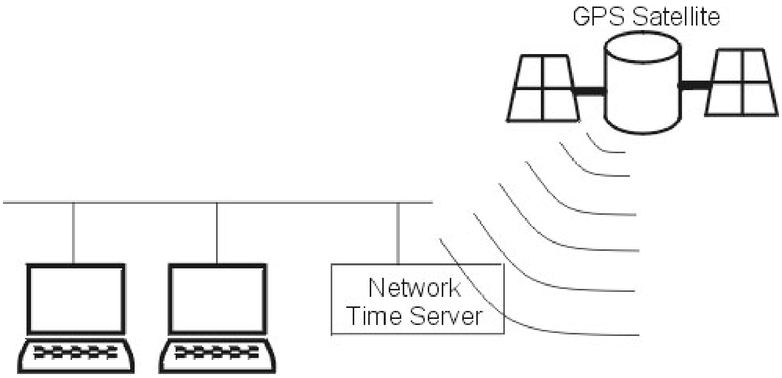


Figure 10.17
Network timeserver application

11

The Internet

Objectives

When you have completed study of this chapter you should be able to:

- Describe briefly the origins of the Internet
- Describe the various organizations associated with the Internet
- Describe the World Wide Web and the associated tools used with it

11.1 The Internet and internet

Finally, a brief explanation of the words 'Internet and internet'.

When referred to in lowercase, as 'internet', this alludes to a physical collection of packet switching networks interconnected by gateways along with protocols that enable the system to exist as a virtual network to exist.

If the word is used as 'Internet', using a capital 'I'; this indicates a collection of networks and gateways that use the TCP/IP suite and operates as a single cooperative virtual network worldwide.

11.2 The objectives, background and history of TCP/IP

11.2.1 The origin of TCP and IP

The Internet was originally known as the **Advanced Research Projects Agency Network** (ARPANET)) and was built by Bolt, Beranek, and Newman Inc. (BBN). This system operated from 1969 through to 1990 and was the template, or design base for TCP/IP, using packet switching over leased lines.

11.2.2 The history and background of TCP/IP

In the early 1960s The American **Department of Defense** (DoD) indicated the need for a wide-area, cross platform communication system. To accommodate this the ARPA system was renamed the [United States] Defense Advanced Research Projects Agency (DARPA), and it used the **Xerox Networking System** (XNS) protocol. However, this particular protocol was found to be inadequate, and as a result the TCP/IP protocol suite was developed.

In 1967 the Stanford Research Institute was contracted to develop this new suite of protocols, with the resulting timetable of development occurring:

1970: Commencement of the development.

1972: Approx. 40 sites connected and TCP/IP support commenced.

1973: The first international connection made.

1974: TCP/IP released to the public.

Initially TCP/IP was used to interconnect government; military and educational sites together, slowly connecting to commercial companies as time progressed.

In actual fact TCP/IP was developed by the US Government to build a heterogeneous (supporting multiple platforms) network across a wide area, the United States.

11.3 The Internet organizational structure

11.3.1 Internet Configuration and Control Board (ICCB)/ Internet Activities Board (IAB)

Originally in 1980, the group formed to develop standards for the Internet was referred to as the **Internet Configuration and Control Board** (ICCB); however in 1983 the name was changed to the Internet Activities Board (IAB). The task of these early groups was to design, engineer and manage the Internet.

Each member of the IAB chaired an Internet Task Force whose purpose was to investigate relevant issues and concerns of the Internet. There were approximately ten task forces, and they looked at various topics relating to the Internet. The IAB met a few times each year to hear from the task forces, check technical directions and focus, discuss policy and exchange information with various other agencies and groups such as ARPA and the **National Science Foundation** (NSF).

Most of these early pioneers of the Internet and the engineers and volunteers who made up the task force groups were largely motivated by the desire to make the Internet work efficiently, and the desire to contribute to the Internet structure. They often worked completely voluntarily and were not on any Internet payroll.

11.3.2 The Internet Engineering Task Force (IETF)/Internet Research Task Force (IRTF)

In 1986 the IAB formed two subsidiary groups to handle two distinct areas of Internet activity. The **Internet Engineering Task Force** (IETF) was formed for the purpose of developing Internet standards. The task of long-term research was given to another group called the **Internet Research Task Force** (IRTF).

The IETF concentrates on short and medium-term engineering problems but due to the large participation in the IETF, the IAB split the IETF into approximately a dozen areas, each with its own manager. The IETF now refers to the entire body including the chairman, area managers and working groups. A steering committee was formed to include the chairman and the managers of each of the working groups. This steering committee has been named the **Internet Engineering Steering Group** (IESG). The IRTF is the research component of the IAB. In a similar vein to the IETF, the IRTF also has a smaller body of people who make up the **Internet Research Steering Group** (IRSG).

11.3.3 The Internet society

In 1992 the IAB was renamed the Internet Architecture Board, and a society was formed to help people use and join the Internet around the world – The Internet Society.

11.3.4 The Internet Architecture Board (IAB)

The structure of the IAB is illustrated below in Figure 11.1.

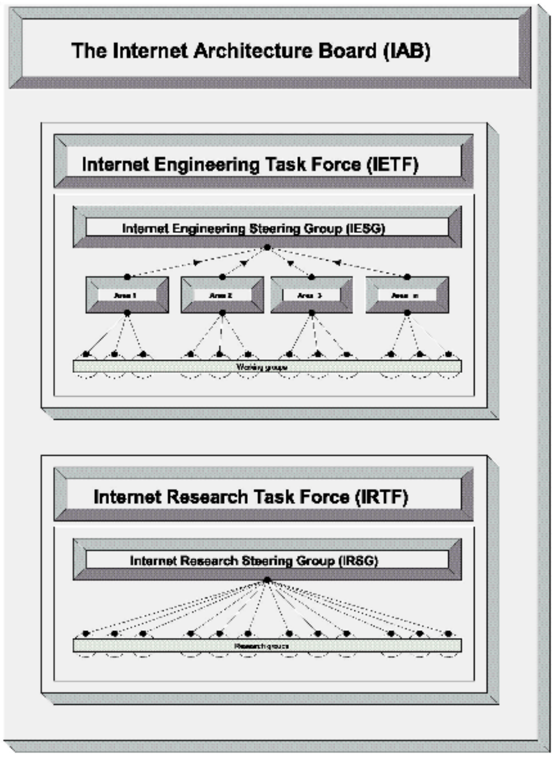

The Internet Architecture Board (IAB)
Internet Engineering Task Force (IETF)
Internet Engineering Steering Group (IESG)
Area 1
Area 2
Area 3
Area n
Working groups
Internet Research Task Force (IRTF)
Internet Research Steering Group (IRSG)
Research groups

Figure 11.1
The Internet Architecture Board

11.4 The World Wide Web

There are people who confuse the Internet with the World Wide Web, also known as WWW, www or W3. Whereas the Internet provides the infrastructure, which allows computers across the globe to interconnect, the Web is software that ‘lives’ on the Internet, providing a graphical interface or ‘doorway’ to the Internet. The web server runs on a host computer, in a similar way as a mail or print server.

By the late 1980s there was still no common user-friendly interface to the Internet. In 1989 Tim Berners-Lee, a scientist working at the **European Organization for Nuclear Research** (CERN) in Switzerland, conceived the idea of the WWW for the purpose of aiding research, collaboration and communication amongst colleagues within CERN. The rest is history. The result proved to be so popular that the Web gained world-wide acceptance.

A web browser allows web pages (which are, in fact, files) resident on any web server to be selected and viewed as requested by a remote user. The original Web browser was fairly unsophisticated and was driven by command line keyboard inputs. Subsequent mouse based browsers were developed and graphics support was added. Information is accessed by pointing and clicking on hyperlinks – images or words that enable access to new information.

There are two types of hyperlinks, namely hypertext and hypermedia.

Hypertext is the most commonly found hyperlink. Whether using a browser, such as Netscape Navigator and Microsoft Internet Explorer, or a Word Processor such as Corel WordPerfect 8 and Word 97 (and subsequent releases thereof) the hyperlinks can be shown in different colors and styles in order to make them more visible. Clicking on the hyperlink establishes a connection to the particular web page.

Hypermedia is another type of hyperlink technology used extensively today. Originally hypermedia meant that one could click on, say, a picture in order to access a particular web page. Nowadays it also means that different types of media (images, sound, animation) can actually be linked to information.

Web server software is available from many vendors. Refer to the last section in this chapter for more information on freeware versions such as Apache and OmniHTTPd.

11.5 An introduction to HTML

All web pages are created using a special language known as hypertext markup language (HTML), which allows one to organize text, graphics, animation and sound into documents that a browser can understand. HTML is the ‘glue’ that holds the Web together; it is the language that makes hypertext and hypermedia possible.

Although HTML is indeed a language, it is not the type of programming language typically associated with computers and software development (such as Pascal or C++). Instead, HTML is a user-friendly markup language that practically everyone can begin using within a day or two.

Markup languages define a formal set of rules and procedures for preparing text to be electronically interpreted and presented. With HTML, one surrounds text and references to files with special directives known as tags. Tags are used to specify how the text or files are supposed to appear when viewed with a web browser; they are used to ‘mark up’ the document in a way that the web browser understands how to deal with. Using tags to mark up a document for electronic publication is easy. One can take a standard word processor document, add some HTML, thereby creating a Web page. The whole process can take less than 15 minutes when creating simple pages.

What really makes HTML powerful is its ability to organize any number of files onto a single page. Files appearing on a page may be physically located on the same computer as the page itself, or anywhere else on the Web. Each file is stored independent of the pages in which they appear; that is, files are not stored inside of the web pages that display them. Instead, HTML merely references, or points to, these files, telling the browser exactly where they are located so it can go out and get them when the time comes for the page to be displayed. A web page is nothing more than a text file that may contain references to any number of image, animation, and sound files that the browser will retrieve, assemble and display when that page is accessed.

11.6 HTTP

HTTP (hypertext transfer protocol) is the protocol that enables the connection between a web server and a client. By using a browser one could, for example, access IDC’s web site at www.idc-online.com by using the browser’s ‘go to’ command and entering http://www.idc-online.com. Typing www.idc-online.com is usually sufficient since most browsers would by default use the http protocol to access the web site.

The first web page displayed would be the home page or top level web page. From here on one would navigate to other associated pages by clicking on hyperlinks.

It is not imperative to use the http protocol in order to display the contents of a web page. One could simply dial up a TELNET connection to the web server, for example by invoking TELNET and connecting to www.idc-online.com at port 80. (Port 80 is used since web servers, by default, listen out for connection on port 80.) Alternatively, one could type >telnet www.idc-online.com 80 under the DOS command prompt. The only problem when not using http is that the page would not be interpreted and displayed as a typical web page as we know it, but as a listing of the html code only.

At its most basic level, the HTTP protocol consists of a single connection and a single command line delivered to a web server residing at a specific IP address. A problem with the real-life situation is that a single web server could hold several hundred web sites, each one theoretically needing its own IP address. In addition to this, each web site could have several dozens of web pages, each page requiring a separate connection with the client. To overcome this problem the HTTP 1.1 specification (and upwards) allows the administrator to assign a virtual host, which allows the web site to appear to the outside world as a single entity with only one IP address.

11.7 Java

Web pages made with HTML are, unfortunately, static. Java programs, called applets, bring pages to life with animation, sound and other forms of executable content. Unfortunately Java applets are usually ‘Plug and Play’ since they cannot easily be modified. There are several reasons for this.

They are:

- The Java language is rather complicated and before one can write or modify an applet, the language first has to be mastered
- It is not possible to view the source code of the applet
- Applets cannot be (or rather should not be) downloaded without permission of the author
- Even if the end user is capable of writing applets, an existing applet cannot be modified unless all parameters have been provided

Once an applet has been developed, it can be woven into the existing HTML code by placing it between the <APPLET> and </APPLET> tags.

Java resources are available from four different sources. These are:

- **Repositories**
  These contain ‘bunches’ of Java applets and links to other Java sites
- **Electronic magazines** (Java e-zines or Javazines)
  These are targeted at Java developers and high-end users
- **Support areas**
  These are web sites aimed at Java developers
- **Search engines**
  If the previous three sources cannot come up with a suitable applet, for example, then a search engine such as Alta Vista can be used to search the web

11.8 CGI

CGI (common gateway interface) looks like HTML and can accomplish some of the things that Java applets and JavaScript can, but it has distinct shortcomings. Java and JavaScript are therefore expected to make CGI obsolete.

- Compared to Java, CGI is difficult to learn
- CGI seems to be user interactive, but it is NOT
- It is mainly used for entering alphanumeric text (e.g. parameters for search engines or credit card numbers for on-line purchases). It does, however, not really interact with the user but rather submits the entered information to the web server for further processing.

11.9 Scripting: JavaScript

Scripting languages have been around since the inception of programming languages and computers, and are commonly known as macros. Macros outline a list of predetermined steps that a spreadsheet performs when that macro is invoked – making macros little more than special purpose scripts. A macro in a spreadsheet is therefore a form of scripting language.

JavaScript is a scripting language for the World Wide Web, developed by Netscape Communications Corp and Sun Microsystems, and is not to be confused with Java itself. Whereas Java is a full-blown programming language meant to be used by experienced software developers, JavaScript is a scripting language for the less experienced and consists of easy-to-understand English-type language. In terms of difficulty, scripting languages fall somewhere between markup languages, such as HTML, and full-blown programming languages, such as Java. Scripting languages provide much more than the ability to prepare documents for electronic publication, yet are not nearly as powerful as true programming languages. Scripting languages are, in essence, mini-programming languages for the average person.

JavaScript differs from Java in that it is not only easier to understand, but the code can be viewed by using, for example, the View->Document Source command under Netscape. It can therefore be customized easily.

Scripting languages fill a void left by programming languages. Whereas programming languages are used to create software products (such as word processors, spreadsheets, web browsers, applets), scripting language lets the end user control such programs. In fact, a scripting language is defined as a relatively easy-to-use programming language that allows the end user to control existing programs. A software engineer creates a program using a programming language like Java, and the end user gets to control the program using a scripting language like JavaScript.

JavaScript information can be found from the same repositories, e-zines and support areas as used for Java applet development. Once the script has been developed, it can be inserted into HTML code between <SCRIPT> and </SCRIPT> tags.

11.10 XML

XML stands for eXtensible Markup Language, and is a data format for structured document interchange on the Web. Like HTML, it is a markup language derived from SGML. It differs from HTML in that it is best suited for organizing data, whereas HTML which was created to allow cross-platform formatting of information for display. Stated in another way; while HTML specifies how a document should be displayed, it does not describe what kind of information the document contains, or how it is organized. XML allows document authors to organize information in a standard way. It is said that 'XML does for data what HTML does for display'.

The development of XML is a public project headed by the World Wide Web Consortium and is not owned by a specific company. The group is only open to members of W3C member companies, but their work can be followed by viewing the w3c web site.

11.11 Server side includes

Most HTML documents are static – that is, the server just sends the client the requested file with no changes. Unless, of course, the file contains Java or JavaScript applets. Sometimes, however, the user might want the server to modify the file every time it is accessed.

This might be desirable in, for example, the following cases:

- Updating a counter each time a file is accessed, and forwarding this value with the file
- Including additional text files in a document
- Including the 'date last modified' in a file, or the current date and tie
- Including the output of a CGI program

This can be done using server side includes. The server processes the file (this is called parsing) and then sends the result to the client. Special commands are included in the following form: <!-#command tag1='value1' tag2='value2'->. The server needs to know that the file includes 'server side includes' to be parsed, and this can be done by using the extension .html instead of .html.

11.12 Perl

Perl (practical extraction and report language) is a text processing programming language created, written, developed and maintained by Larry Wall. It is claimed to have sophisticated pattern matching capabilities and flexible syntax, and is used for applications such as input/output, file processing, file management, process management and system administration tasks.

12

Internet access

Objectives

When you have completed this chapter you should know, in principle, how to:

- Connect your home PC to the Internet using dial-up facilities
- Connect your home PC to the office LAN using a PPP server
- Connect your LAN (small or large) to the Internet using either a proxy server, NAT machine, IP sharer, Unix/NT gateway, or dedicated IP router

12.1 Connecting a single host to the Internet

Connection to the Internet backbone is supplied by 'primary' **Internet service providers** (ISPs) such as AOL (**America On-Line**), CompuServe and Internet Africa. ISPs outside of the USA are connected to the US Internet backbone as well as to ISPs on other continents through high-speed undersea (fiber optic) and satellite connections with a bandwidth of several tens or even hundreds of Megabits per second. These ISPs also own the servers needed for functions such as user authentication, mail (POP3 and SMTP) and **domain name system** (DNS) services. Users can subscribe to, and directly access these ISPs.

There is also a proliferation of 'secondary' ISPs differing from the others in that they do not own their own international access, but lease it from the primary ISPs such as those mentioned above. The 'secondary' ISPs are geographically dispersed and connect to the main ISPs via high speed public or private switched network links, (for example X.25 and E1/T1).

The ISPs supply the points through which the Internet can be accessed (the so-called **points of presence** or **PoP**) either on a regional or national level, e.g. Ozemail (ozemail.com) in Australia or Internet Africa (iafrica.com) in South Africa, or on a global level e.g. IBM Global Network (ibm.net). The disadvantage of a regional ISP as opposed to global ISP lies in that the former has points of presence (PoP) only within one country or region, whereas the latter, e.g. ibm.net, has PoPs in most major cities across the globe (approximately 2500 in this particular case); thus simplifying life for a traveling person in possession of a laptop or notebook computer. With a global ISP it is possible for a traveler to connect at airports before and after a transcontinental flight, and possibly even during the flight, just by selecting the nearest PoP on the dialing program.

The ISP's equipment at the point of presence consists of:

- A router (or routers) which route traffic to other ISPs and to the Internet backbone
- A **point-to-point protocol** (PPP) server to provide Internet connectivity with multiple Internet users (subscribers) across serial telephone lines. Some ISPs also offer SLIP (serial link interface protocol) but SLIP has largely been superseded by PPP
- Analog (dial-up or leased-line) modems and ISDN connections as required for user access. The modems are connected to the local POTS exchange through dedicated telephone lines, one per modem, with a so-called 'hunting line' at the exchange so that all modems can be accessed via the same telephone number

Until recently these routers, modems and PPP servers were installed as discrete units. The current trend is to purchase them as integrated access servers, with the routing, dial-up server and modem functions in one box. The typical number of modems per access server is around 30 but this number can vary, and the number of ports can simply be increased by stacking additional units.

Users can access the ISP through several means. In all cases, the user pays the ISP for the Internet access, as well as the telephone supplier for the connection to the ISP. Usually the connection can be accomplished as a 'local' call. Access methods include:

Dial-up modem over a normal telephone connection

This is by far the most cost effective method for a single user or a small group of users but a serious drawback is lack of speed, not so much due to the bandwidth limitation of the user's telephone line or modem, but by the total demand imposed on the access server by all the users and the capacity of the link between the secondary and primary ISPs. Experienced 'web surfers' know that the best time to access the Internet is during the early hours of the morning when most other users are asleep! Even a 56 kbps modem can often not accomplish a connection at higher than 24 kbps and even then the user can be fortunate to achieve a data download rate of more than a few kbps during peak hours.

ISDN connection

This is also a dial-up service, but the communication is digital and the bandwidth between subscriber and ISP is substantially higher. The typical '2B + D' connection offers a 128 kbps bandwidth, and additional channels can be dialed up if more bandwidth is required. Because of the higher performance, the charges for this service are substantially higher.

Leased lines

These provide permanent connection to the ISP and are divided into two categories: analog and digital. Analog leased line modems use the same technology and therefore have the same speed limitations. At present analog leased line modems operate at typically 33.6 kbps to 56 kbps. Distance and noise are limiting factors, and analog leased lines are often only half-duplex, which means that traffic can only travel in one direction at a time. Digital leased lines (e.g. X.25) are faster, more reliable, and not limited by distance.

Cellular (mobile) phone

Laptop computers can link up with a suitably equipped ISP without using a traditional telephone-type connection. Apart from the cellular phone rates usually being higher than normal dial-up rates, this connectivity solution may necessitate the purchase of a dedicated PCMCIA (also known as **CardBus** or PC-Card) interface in order to connect to the laptop, or a new infrared compatible cellular phone!

Older cellular phones such as the Nokia 2110 have an external communications connector but need a special PCMCIA interface for a laptop. Newer models such as the Ericsson SH 888 and Nokia 6110 come equipped with a built-in PCMCIA interface and can communicate with the laptop either via infrared link or RS-232.

12.2 Connecting remote hosts to corporate LAN

Larger organizations often have an existing in-house LAN with permanent access to the Internet. Over and above the need for Internet access, users may still rather want to log in to the corporate network as opposed to an ISP for the following reasons:

- They may wish to access corporate databases and file servers from home or whilst on the road
- Remote customer and vendor access to restricted corporate information such as order status or purchasing data
- Remote diagnostic and maintenance activities by system administrators

The solution is the installation of a communication server (also called a PPP server) supporting at least the IP (preferably also IPX, for Novell Netware users) protocol families. This enables workstations to dial in over standard telephone lines using modems. The communication server answers the phone, authenticates the user, and attaches the remote workstation to the LAN. Subject to security constraints, the remote user can then access all IP (and IPX) LAN based resources including databases, file servers, web servers and routers. Depending on the specific model, a communication server typically supports between 1 and 32 hosts. Such servers are manufactured, for example, by TECHSMITH Corporation, CABLETRON, CITRIX and MICRONET.

12.3 Connecting multiple hosts to the Internet

12.3.1 Connection via proxy server

This approach is ideal for a LAN with only a few hosts on it, for example a small office LAN or 2–3 networked PCs at home, which all need access to the Internet at the same time.

In general, a 'proxy' stands-in for something, or somebody. A paid-up member of an organization, unable to attend the AGM, could hand a proxy to another member to vote on her behalf. In the case of a network the proxy server is the machine with the connection to the Internet (e.g. via dial-up modem). The server runs special proxy software such as Wingate or Win Proxy, which allows any other client computer on the network to forward its request, for something like a web page, to be handled on its behalf by the proxy server. The proxy server, in turn, downloads the web page and passes it back to the client in a manner, which is transparent to the user.

Proxy servers can usually handle only one protocol and are generally aimed at occasional dial-up Internet connection for small organizations. They are not intended for organizations where they would be key connections to the Internet.

The only machine with a valid IP address is the proxy server, which obtains it via a DHCP server at the ISP. This IP address is allocated to the dial-up adapter in the proxy server and NOT to the Ethernet adapter, which is used to link the proxy server to the other machines on the LAN. The question now arises: how do the machines on the LAN communicate? What do we do to allocate IP addresses to the individual machines? The solution is simple: any fixed IP address will do, as long as they are all on the same subnet. Nobody will be inconvenienced, since these IP addresses will not be seen beyond the proxy server. If we want to be technically 100% correct, we should choose our IP addresses to conform to the range of IP addresses reserved for private TCP/IP networking, as explained in Chapter 6.

No special configuration for the client machines are normally necessary, apart from informing Internet Explorer during setup that there is indeed a proxy server, what its IP address is, and at what port number it runs. Information regarding the latter will be obtained from the proxy server’s documentation.

12.3.2 Connection via NAT server (IP masquerading)

NAT, or network address translation (also referred to as IP masquerading) is intended for a permanent, ‘heavy duty’ connection to the Internet. Whereas this solution physically looks the same as proxy serving, it operates on a totally different principle.

Its operation is entirely transparent to the rest of the network. Client computers on the network can use virtually any protocol; there is no special software and very little configuration required for them, apart from the normal TCP/IP setup. The only problem is that from the Internet point of view, there will be only one IP address and hence only one host visible on the network, namely the machine configured as the NAT server.

The client machines are configured to view the NAT machine as the default gateway (router), which is indeed what it is. The NAT server receives a packet from a client, replaces the IP address in the frame with its own, and forwards it onto the Internet. When a return message reaches the NAT gateway, it replaces the destination address with that of the client computer or forwards it on to its own subnet. Besides just translating addresses, NAT must also translate header information and packet checksums.

12.3.3 Connection via IP sharer

An Internet IP sharer such as Micronet’s SP86X is a hardware device that comes pre-programmed with a set of valid IP addresses. It acts as a DHCP server, automatically allocating IP addresses to each active station on the LAN.

It provides a firewall function and will automatically dial-up and disconnect depending on usage. Connection with the ISP is achieved via 56 kbps dial-up modems or 128 kbps ISDN. Depending on the model being used, 1, 2 or 4 modems can be connected in parallel, individual modems being activated or deactivated according to bandwidth requirement.

12.3.4 Connection via UNIX or NT gateway

This is one of the easiest solutions for a large company wishing to give Internet access to all its members. A UNIX or NT host is set up as a gateway to the Internet. This solution requires at least a set of 254 Class C IP addresses, either permanently allocated to hosts, or dynamically allocated via a DHCP server, which could run on the same machine.

The UNIX/ NT machine is set up with two network adapters, i.e. as a 'multi-homed' host. The first adapter is connected to the internal LAN, the second to the Internet. This implies that the second adapter should have the necessary connectivity e.g. X.25 built-in. Each card will need its own permanent IP address, and each card will be configured in such a way that the 'other' card's IP address will be given as its default gateway. In this way, each card will pass on a received message to the other card.

12.3.5 Connection via dedicated router

The simplest way of connecting a network to the Internet is via a dedicated 2-port IP router (also referred to as an Internet router). One port of the router will be, for example, an Ethernet port, to be connected to the local area network. The other port could be an X.25 WAN port, which will be connected to a public packet switching network. The X.25 link provides the connection to the ISP.

As in the previous case, a set of IP addresses is required, and these are allocated either permanently or via a DHCP server.

13

The Internet for communications

Objectives

When you have completed study of this chapter you should be able to:

- Briefly explain the speed/bandwidth issues
- Briefly explain the various options for e-mail
- Briefly describe the use of voice over IP
- Indicate briefly how voice mail is performed using TCP/IP
- Briefly indicate how video conferencing is performed using the Internet

13.1 Introduction

The following chapter gives an overview of the current (as of mid 2001) state-of-the-art in Internet communications. This particular area of technology is advancing so rapidly that the only way to keep track of developments is by regularly browsing the Internet. The list of products mentioned in this section is by no means complete, but represents a fair cross-section of what is currently available.

A very interesting and significant global development is the changeover from traditional MAN/WAN architectures, used for linking company resources over large geographical areas, to Internet communication because of (a) the much lower cost involved and (b) simplicity of interconnection imposed by the necessity of standardizing on TCP/IP. This tendency is not only manifesting itself in the so-called **information technology** (IT) environment, but also in commerce and industry, particularly in the manufacturing and process control environments. It is therefore only logical that the Internet will also be used for telecommunications (voice, fax, video etc) on an ever-increasing scale.

13.2 Hardware and software issues

The advantage of the current generation of Internet communications products lies therein that they coexist on the already established Internet, PSTN (public switched telephone network) and PBX (private branch exchange) infrastructure. Internet communications products are predominantly software-based and in many cases, they are available either as freeware or shareware.

For the top-end products it may be necessary to purchase dedicated Internet interfaces for telephones or fax machines, but in these cases the end-users are typically medium to large enterprises and the capital outlay can be justified in terms of cost savings.

13.3 Speed/bandwidth issues

As far as long-distance communication over the Internet is concerned, transmission speed can be a problem. For a modem-connected dial-up user the fastest modem in the world will not improve things much since the bottleneck is imposed by the bandwidth made available to the I**nternet service provider** (ISP) and the number of simultaneous users competing for access to the ISP. In some cases the bandwidth of this data 'pipe' feeding the ISP is as little as 64 kbps; with 100 users connected at a given time this translates to only 640 bps per user!

When it comes to increasing the available bandwidth within a LAN, however, there are several possibilities open to the LAN owner. Increasing the data transmission speed of the LAN (say, by upgrading from 10BaseT to 100BaseT) is one option, but not the only option.

Additional (and in some cases less costly) options are:

- Careful segmentation of a large flat network, with bridges, switches and routers. This reduces traffic interference as well as collisions
- Cutting down on unnecessary broadcast packets (there are ways to accomplish this
- Minimizing the number of routers between a given workstation and the point where the LAN attaches to the WAN leads to fewer 'hops' across routers and thus reduces latency (time delays) which adversely affect voice and video transmissions
- Tasks which result in heavy network traffic, such as backups and large file transfers can be scheduled for off-peak periods in order to minimize interference with voice/video/fax traffic, which normally takes place during working hours

For transmitting voice and video across a WAN there are several options as discussed in the following sections. Some are purely software based; some need hardware; some are hybrids. All approaches have advantages as well as disadvantages, and the prospective user will have to weigh them up against each other.

13.4 Legal issues

In certain countries the use of IP services to carry voice traffic over the Internet (referred to as voice over IP, or VoIP) is illegal. Such is the case in South Africa, where the Telecommunications Bill specifically prohibits it.

This is in sharp contrast with more enlightened telecommunications regulators, such as Deutsche Telekom, who openly embraced the new Internet telecommunications trend by investing substantially in companies involved in 'voice over IP' technology development, and in the process attracting business that would have been lost to competition.

In the USA, ACTA (America’s Carriers Telecommunications Association) tried to ban Internet telephony software but the American **FCC** (Federal Communications Commission) refuses to have Internet telephony regulated.

The fact that there may be a law prohibiting VoIP can hardly deter people from using it. It is difficult to enforce such a law because once a voice call is in progress it is practically impossible to detect it or its origin. Once voice or video has been digitized into packets it is no different from any other data!

It is also a fact that Internet telephony is here to stay and the telecommunications regulatory bodies will simply have to find a way to accommodate this reality.

13.5 E-mail

13.5.1 POP and SMTP servers

In order to gain a better understanding of the operation of an e-mail program, one has to look at the mechanism of reception and transmission of electronic mail, with reference to the TCP/IP model. Whereas the actual transmission of the mail (in digital format) from end to end is handled by the Internet and host-to-host layers of the TCP/IP (DoD) model, the interface with the user is handled by the **SMTP** (simple mail transport protocol) and POP (post office protocol) which reside in the process/application layer. As a practical exercise, one can actually connect into POP3 and SMTP servers, using TELNET, and manipulate the contents of the mailboxes as well as send and receive messages. It will soon become evident that although physically possible, this is a very tedious and user-unfriendly process hence a more elegant user friendly interface is required between the actual user on the one hand, and POP3/SMTP on the other hand. This brings us to the popular e-mail programs as discussed in the subsequent paragraphs.

13.5.2 E-mail software residing on the local host (workstation)

In this category, we look at popular e-mail programs such as Eudora Lite, Netscape Communicator and Microsoft Internet Mail. The software resides on a PC or Laptop, and enables the user to compose messages, add files as attachments, send, receive and forward messages as well as print mail at leisure.

For dial-up users, this category of program offers a particular cost advantage in that mail can be prepared off-line. A connection can then be made, messages sent and received and the user can once again log off resulting in relatively low telephone bills.

13.5.3 E-mail software residing on a remote server

This category comprises e-mail programs such Hotmail and Eudora Web mail. Although these services are ostensibly free of charge, the user still needs access e.g. via a dial-up connection to the Internet in order to access mail. To complicate matters, the user needs to remain connected while composing mail and browsing the contents of the mailbox, which becomes costly for dial-up users. The latter problem can be alleviated by composing messages off-line beforehand and then cutting and pasting them into the message setup window before transmitting, minimizing connection time.

An advantage of this system is that the user can retain his user ID regardless of where he lives, and can also access E-mail from any site in the world where there is Internet access. For a traveling user without a laptop, a cyber café (Internet café) can fulfill this function. It is also a solution for the corporate user who struggles to get an e-mail user account from a system administrator! It is a particularly ideal solution for university and college students with access to the institutional Internet service since it costs nothing and they retain their e-mail addresses when they leave the institution.

13.5.4 Voice retrieval of e-mail

Eudora Webmail is now offering a voice retrieval service whereby a traveling user not having access to a PC or Internet café, can actually dial into the mail server using a regular phone and retrieve mail by voice. At this point in time, this technology is still in its infancy and costly for users not residing in the United States.

13.6 Internet telephony

13.6.1 Introduction

This section describes a novel and fast developing aspect of the Internet, namely the transmission of voice over IP. Antagonists of this concept are quick to point out that the Internet was designed to transport data and not voice, video, or other low latency applications. They also insist that the additional voice transmissions will place an unacceptable burden on the existing infrastructure. Fortunately, this has prompted companies such as QualComm to produce ‘smart’ voice compression techniques which utilize less than 10% of the bandwidth of conventional **pulse code modulation** (PCM) systems, yet achieving a subjectively perceived voice quality that actually exceeds that delivered by normal public telephones!

In May 1996, a forum called **voice over IP** (VoIP) was formed to try maintaining and monitoring an Internet telephony standard. Current members of VoIP include 3Com, Cisco, Microsoft, VocalTec, Netspeak, Intel, IBM and US Robotics. This collaboration should supply substantial impetus to the development of Internet telephony hardware and software.

An indication of the legitimacy of Internet telephony is Deutsche Telekom’s $48 million investment in VocalTec, developers of Internet Phone and one of the forerunners in Internet voice systems development. Deutsche Telekom has also committed to using at least $30 million of VocalTec’s products and services in the future, thereby gaining access to VocalTec’s Telephony Gateway, and through that the Internet users and revenue they would otherwise have forfeited to competitors.

13.6.2 PC to PC

FreeTel

A typical product in this category is FreeTel, which provides real-time voice communication via the Internet. Best of all, this product is free and the only cost involved is the connection to the local ISP.

The primary advantage in using the Internet is that one does not incur any long distance telephone charges. Another advantage is the ability to transmit data while talking. On the downside, the Internet introduces a delay of typical 1/2 to one second, similar to the delay present in transcontinental satellite telephone connections. There is also no connectivity with the existing telephone system, so that one can only communicate with other Internet users connected at the time of the call.

Internet phone release 5

Internet Phone by VocalTec enables PC users to make regular full-duplex telephone calls over the Internet to any other Internet connected PC in the world, providing the caller signs up with an **internet telephony provider** (ITSP) which supports the VocalTec system. As is the case with FreeTel, functions include some of the amenities of a full-featured telephone such as caller ID, call waiting, muting and blocking. In addition to this, it is possible to introduce live motion video, which enables the recipient to actually see the caller without additional hardware required. It also supports audio conferencing with up to 100 people, 'white boarding' which enables the sharing of documents, photos and drawings with other users, and text chatting via the PC keyboard.

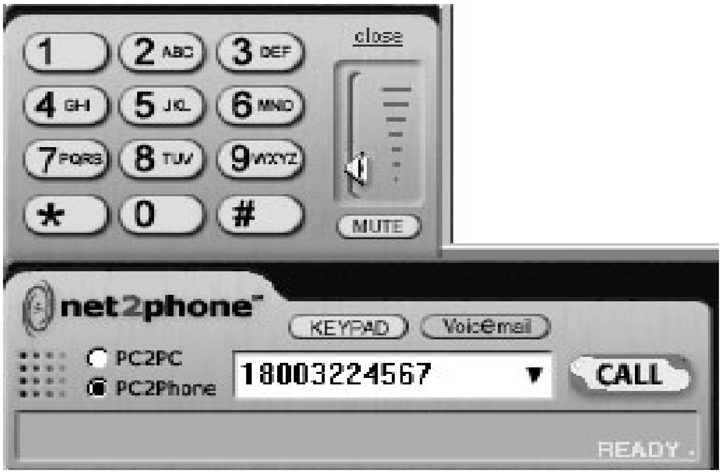
Netspeak web phone

Like Internet phone, Netspeak Webphone offers voice, video and data communications over the Internet and any other TCP/IP-based networks with all the functions of a top-of-the range conventional telephone as well as videophone support using the H.263 standard. WebPhone uses TrueSpeech G.723.1, G.711 GSM voice compression (as used in cellular phones) to overcome the bandwidth limitation inherent to Internet communication.

13.6.3 PC to phone

Net2Phone

Net2Phone is an innovative software product system from IDT. Net2Phone is a service that allows the user of an Internet connected multimedia PC or laptop to place calls from anywhere in the world to any regular phone in the world using a dedicated Internet exchange in the USA. Users are not limited to PC–PC technology which requires both users to have access to multimedia PCs and to be connected to the Internet at the same time, either by coincidence or prearrangement. Only the caller needs a PC with an Internet connection and can use any ISP in order to make the call. The cost of calls varies, but is on average less than 10% of a normal phone call. In addition to this, calls to any 1–800 number in the United States are free.



○ PC2PC
● PC2Phone

18003224567

READY .

Figure 13.1
Net2phone (courtesy of Net2Phone)

13.6.4 Phone to phone

Aplio

Aplio offers a Voice over IP service by means of a small stand-alone box that interconnects regular telephones across the Internet, enabling the calling party to talk over the phone using the Internet instead of the traditional long distance carrier. This approach requires no computer, no Internet phone software, no specific carriers, and no gateways/routers. Connection is accomplished by simply plugging the phone into the Aplio box and connecting the latter to a regular telephone wall jack. Once contact is established with the other party, the sender presses a button on the Aplio phone and hangs up. In less than 45 seconds, the sender's phone will ring after which he can pick it up and continue with the conversation. The only difference is that this part of the communication is via the Internet, hence a slight (half-second) delay.

A disadvantage of this approach is the cost of the interface unit (around $250, mid 2001) as well as the fact that an Aplio interface is required on both sides. This could, however, soon be offset against a reduction in long distance telephone bills, especially for parties who have to conduct regular long-distance calls between two specific locations.

13.6.5 Mixed PC/phone to mixed PC/phone with Intranet PBX

NetPhone IPBX

NetPhone IPBX is a scalable intranet based PBX (private branch exchange) featuring amongst other things automatic call distribution, call accounting, and least-cost routing. Businesses can use their existing PBX resources for voice communication between corporate locations, bypassing the PSTN by using the Internet to provide a cost-effective means of communication. The NetPhone IPBX supports up to 96 extensions and offers a fall-back feature which routes all calls back to the PSTN during an Internet failure which means that the telephone communication system would always be up and running.

This system enables users:

- To identify callers before picking up the phone
- To access, in real-time, applications containing information regarding the caller when the phone rings (such as the location of a specific parcel in the case of a freight shipping company)
- To return phone calls without having to look up numbers
- To return calls to prospective customers who left no messages
- To view, prioritize and sort through voice mail on the PC screen
- To transfer, park or forward calls under control of a mouse

This system is built around dedicated hardware such as IP telephony gateways and communication servers, which means that it would typically be deployed in larger organizations because of the capital outlay involved. This expenditure can, however, be justified in terms of cost savings in telephone bills.

13.6.6 Incoming calls with call waiting

By mid 2001, VocalTec plans to introduce Internet phone call waiting. With the rise in new Internet-based business applications, users are spending more and more time connected to the Internet. These on-line connections are tying up phone lines, preventing employees from receiving business related telephone calls. This might lead to limiting the revenue, which service providers can generate through call termination. VocalTec Internet phone call waiting will solve the problem by enabling Internet service providers to provide a virtual second line service to Internet connected customers. This will alert on-line users to incoming calls (including caller ID information) from regular phones. Using Internet Phone Lite software running on a multimedia PC, the subscriber can then accept the call without disconnecting from the Internet.

13.6.7 Outgoing calls with regular phone through PC

Net2Phone Pro, a product by the developers of Net2Phone, enables the user to plug a standard telephone into a multimedia PC and thereby conducting conversations over the Internet as if through a normal POTS connection.

13.6.8 Live voice communication with customers via web page

Click2Talk from IDT is a product, which adds on to the Net2Phone software, described above, and enables a prospective client, browsing a web page, to gain real-time access to the service provider by phone, by simply clicking the Click2Talk button on the screen. This of course implies that the prospective customer is browsing via a multimedia PC.

13.6.9 Secure phones

PGPfone (PGP = pretty good privacy) permits computer owners who have modems or connections to the Internet to use their computers as secure phones. PGP software is available from Massachusetts Institute of Technology by WWW only, or via PGP Inc. Unfortunately the legal distribution of PGP is restricted to US citizens or persons resident in the US only.

13.7 Paging

13.7.1 SMS (short message service)

Whereas the following two examples are by no means representative of the whole range of products, they nevertheless give a good idea of what can be achieved.

The first example of such a service is SMS via e-mail offered by Mobile Data Systems. SMS is a 116-character that can be sent to and from a cellular phone. Mobile originating SMS messages (MO) are messages generated from a cellular phone, whilst mobile terminating SMS messages (MT) are messages sent to a cellular phone.

In many instances, an SMS message can be a better method of communication than voice for the following reasons:

- The message does not disturb the recipient
- Messages can be stored for later use
- The message will be delivered as soon as the recipient is connected to the cellular network
- The sender can receive e-mail notification whether the message was received or not (recipient out of range, phone off)

There are many applications for SMS, including but not limited to the following:

- Messages can be generated and disseminated automatically, for example in the case of network or process alarms

- Product information (price lists etc) can be sent to sales staff
- Notification messages can be used to notify users of voice mail, e-mail, staff meetings etc
- Alarm alerts can be sent directly to a support technician's phone and because notification of the delivery of the message is provided, problems can be escalated to other technicians' or managers' phones if the alarms are not cleared timeously
- Customer care related information e.g. reminding customers of account balances, credit limits surpassed, product launches
- Messages sent by operator based messaging centers to staff phones informing them to contact clients (the contact phone numbers can also be forwarded in these messages)

For security PGP (**pretty good privacy**) is used to authenticate messages submitted. Users must be able to sign e-mail with a PGP signature and will be required to submit their PGP public key on registration.

13.7.2 DAJDsock

Another interesting messaging program is a powerful TCP/IP testing tool, DAJDsock, which is available as freeware. DAJDsock is setup with the socket names (that is IP addresses and associated port numbers of specific critical processes running on servers). At regular predefined intervals it interrogates these sockets, and if a proper response is not received it alerts the system administrator either by sending an e-mail message or by automatically placing a message (similar to SMS) on his beeper.

13.8 Voice mail

13.8.1 PC to PC

There are several commercially available software packages which perform this function. A voice message is recorded on a multimedia PC in a digital format and transmitted as an e-mail message to another machine.

Internet VoiceMail

Internet VoiceMail is a program, which allows transmission of voice mail over the Internet to anyone with an e-mail address. The software includes a voice mail player so that even recipients without voicemail software on their computer may still hear the message. The software is fairly inexpensive and an evaluation version can be downloaded free of charge.

QualComm PureVoice/SmartRate

The technology used here is a standard in CDMA cellular technology and will also be included in upcoming versions of Eudora Software.

PureVoice and SmartRate are voice-coding technologies developed for use with wireless telephony applications, and in particular digital cellular telephones. However, they also work well with e-mail. The resultant compressed file is ten times smaller than a corresponding .wav files. For example, a 1 Megabyte .wav file would take 7 minutes to transmit using a 28.8 kbps modem. By comparison, a PureVoice file (.qcp) would be less than 100 kilobytes and take less than 45 seconds to transmit, using the same modem speed. Using the SmartRate technology will reduce the storage to 60 kilobytes and transmit time to 30 seconds!

Despite this severe compression, PureVoice and SmartRate compare favorably with conventional pulse code modulation systems. In comparing these new technologies with established standard voice encoding systems viz. PCM and ADPCM by means of subjective mean opinion scoring (MOS) techniques, SmartRate and PureVoice either equaled or outperformed their rivals despite running at about one-eighth the bit rate of their rivals!

The virtues of PureVoice and SmartRate are being extolled here, not in order to promote these particular products, but to highlight the tremendous strides that have been made in curtailing bandwidth requirements in VoIP applications and yet maintaining a high-perceived voice quality.

13.9 Fax

13.9.1 Fax machine/e-mail to fax machine/e-mail

VocalTec markets a product by the name of PASSaFAX, a real time Internet faxing device.

According to Communications News 1998, the average Fortune 500 company spends over $15-million yearly on fax transmissions. To reduce this financial burden, IP (Internet protocol) provides a cost effective way to route faxes by using existing data networks. The system includes SNMP based management and configuration. It uses store and forward IP faxing based on the ITU and IETF standards, which allows fax-to-fax, fax to e-mail and e-mail to fax transmissions. This particular system requires hardware in the form of a PASSaFAX module with built-in modem(s) and PBX emulation. Fax machines connect either directly onto the PASSaFAX unit, or to the PASSaFAX unit via the PSTN or PBX. The PASSaFAX unit, in turn, connects onto the Internet.

13.9.2 Fax machine to fax machine

Net2Fax uses a conventional fax machine plugged into the phone connector of the user’s modem and the Net2Phone software-dialing interface. Once the connection has been established, the fax machine’s start button is depressed and the transmission takes place in the normal way.

13.10 Video conferencing

The traditional approach has been a ‘brute force’ methodology using high-speed switched data networks such as ISDN, T1/E1 and Switched 56. ISDN, for example, provides a 64 kbps bandwidth per ‘B’ channel and several channels can be utilized simultaneously in order to obtain the required bandwidth. Modern technology, however, allows systems like those described hereunder to send full motion video across a LAN without bogging down other data traffic.

13.10.1 Video/audio/document conferencing

VidCall is a product, which delivers video, voice and document conferencing to people around the globe via the Internet. Not only can users participate in shared workspace activities on the computer screen, but they can see live, scalable video of the person(s) with whom they are working. It is possible to incorporate live action color video and voice with shared workspace on the PC screen over the Internet.

VidCall uses anyone of over 25 inexpensive video capture boards (including digital plug and play cameras) to transmit still and motion video.

Depending on the speed of the computer modem, VGA display and LAN/WAN, virtual motion of up to 10 frames per second is achievable. Multi-point video and document conferencing is available over LAN/WAN accommodating up to 10 participants who can be located across the globe.

VidCall also supplies additional freeware for registered VidCall users on the Internet. WhoIsThere enables individuals to setup their own user groups and to be informed as soon as members of their group are logging in to the video conference.

13.10.2 Video networking

Another company providing video networking and video conferencing software is BitField. BitField claims to be the first company to have provided a complete H320 compatible video codec on a single PC video adapter board. Their video communication products turn PCs into video communication workstations, transferring full motion video between standard PCs via ISDN, LANs, and other communication networks. Since the products utilize existing PC and networking technology, they can be applied in many areas where traditional video equipment is too expensive and flexible.

13.11 News

13.11.1 News push

The original approach to news via the Internet has been to search the World Wide Web and ‘pull down’ relevant news items. This is not only time consuming but also costly for an individual who has to connect through an ISP. On the other hand, products such as PointCast aggregate news from more than 700 sources, process this in a central broadcast facility and then broadcast it world-wide via the Internet. Only the news categories selected by the user is PUSHED down for collection every morning.

PointCast delivers national, international, business, industry and company news, stock quotes, sports scores, weather reports, entertainment news and more. Currently, it acquires information from sources such as CNN, CNNfn, *NY Times*, *Wall Street Journal*, Reuters, Business Wire, PR News Wire, Standard and Poor’s ComStock, Sports Ticker and AccuWeather.

PointCast network will work with a dial-up connection, as long as the dial-up connection assigns a valid IP Address. The PointCast network offers support for viewers using CompuServe, Shiva, FTP and Windows 95 Internet dialers. As for the on-line services, it depends on how they implement the Internet access. Currently, PointCast will work with CompuServe, America On-line and MicroSoft Network.

From an enterprise point of view, PointCast allows management to selectively broadcast news to all employees using the existing network infrastructure, keeping in mind that PointCast is completely free thanks to commercial advertisers.

To help the system administrator customize the PointCast Network, PointCast offers a suite of free tools, the so-called Intranet Broadcast Solution which allows the following:

- It ensures that important company news is widely seen and read by broadcasting it directly to employees desktops via a private Intranet channel on PointCast

- It allows the management to effectively communicate time sensitive messages through special windows on employees' desktops
- It empowers 'knowledge workers' by supplying all the news they need to be competitive including news from customers, suppliers, competitors and industry

13.11.2 News pull

Individual publications

The conventional pull services are still widely available with many private newspapers maintaining their own webster. A particular case in point is the South African based *East London Daily Dispatch*, which is widely read by Australian-based South African expatriates living in Australia.

Collated news

IBM Internet Connection Services offers a news service to subscribers available in 10 different languages and utilizing a news search engine that pulls stories from over 250 news sources from over the globe. Topics include top stories in technology, politics, business, culture news, CAN world news, TechWeb and USA today. It is also possible to select world news on a regional basis.

13.11.3 News groups: USENET

While web (www) sites have received most of the attention from software developers and the press, the Internet's news and conferencing service, USENET, represents another major Internet resource. USENET is based on news groups, such as comp.client.server or misc.education.adult, that contain articles similar to e-mail messages. News groups exist for virtually every conceivable professional and personal interest. Some news groups serve as problem solving forums and as the 'help desks' of the Internet. Most news groups are public and can be viewed anywhere the Internet reaches. Anyone with a news client and an Internet connection can submit an article. Some news group's subject articles to a screening prior to publication, others automatically distribute all submissions. World wide, over 300 000 articles are posted each day to over 50 000 news groups. Because of the massive storage requirements, articles begin to disappear within a few days of publication. Stand-alone news client software is available from several sources. News clients are also built into Netscape Navigator and Microsoft Internet Explorer. As users of these web browsers become more experienced, they are able to 'graduate' to the world of USENET news.

There are several packages available for dealing with USENET and many of them are available as freeware. We will now deal with some of them:

13.11.4 Search agents

The question may well be asked: Why on earth would a company care about USENET postings? Here are a few reasons:

- Find out who is mentioning your products
- Find out who is using or misusing your trademarks
- Discover people and companies looking for solutions that your organization can provide

- Find out what people are saying about your competitors' products and services
- Detect imposters forging messages that appear to originate from your organization

A good example of a USENET search agent is NewsMonger by TechSmith Corporation. NewsMonger constructs the search query for you, or allows you to create the query yourself. It searches public USENET groups for each of your active queries using Digital's Alta Vista engine. It removes duplicate articles and identifies new submissions. It then notifies you via e-mail when articles matching your criteria are discovered.

A similar product is OUI (off-line user interface), which has the ability to locate, retrieve, download and post information to and from news groups.

Upload utilities

AutoPost is a program, which is capable UUEncoding files, and automatically uploading them to a new server, i.e. it automates the process of posting large volumes of files to news services.

Retrieval programs

There are several downloading programs for USENET, available as either shareware or freeware. Programs worth noting are:

- Agent
- Free Agent
- News XPress
- Pluckit 3
- SBNews; NewsRobot

Functions of these packages include on-line/off-line news reading, e-mail functionality, built-in viewers for graphics files, and the ability to launch URLs, firewall protection, spam elimination, and automatic encryption to protect sensitive images. The prospective user will have to peruse the specifications and make a decision as to the most suitable package for a particular application.

13.12 Additional information

Additional details about products mentioned in this chapter can be obtained from the following web sites.

13.12.1 Internet telephony

PGPfone (Pretty Good Privacy Phone): http://web.mit.edu
Net2Phone: http://www.net2phone.com
FreeTel: http://www.freetel.com
Internet Phone Release 5: http://www.vocaltec.com
Internet Phone Call Waiting: http://www.vocaltec.com
Aplio (Voice over IP): http://www.voiceoverip.sitehosting.net
NetPhone IPBX: http://www.netphone.com
WebPhone: http://www.netspeak.com
Net2Phonepro: http:/www.net2phonepro.com

13.12.2 Video conferencing

VidCall: http://www.powernethk.com
PictureTel: http://picturetel.com
BitField: http://www.bitfield.fi

VidCall has a ‘continuously running’ demo system with IP Address 205.157.131.91. First download and register the demo program, then ping the demo system to make sure it is running, then follow the steps supplied on the screen.

13.12.3 Paging

SMS (Small Message Services): http://www.mobiledata.co.za

13.12.4 Fax

Net2Fax: http://www.net2phone.com
VocalTec PASSaFAX: http://www.vocaltec.com

13.12.5 Voice communication via web page link

Click2Talk: http://www.net2phone.com
Click2CallMe: http://www.net2phone.com
Mini WebPhone: http://www.netspeak.com

13.12.6 Voice mail

Internet Voice Mail: http://www.vocaltec.com
QualComm: http://eudora.qualcomm.com
BitWare: http://www.cheyenne.com

13.12.7 News services

NewsMonger: http://www.techsmith.com
PointCast: http://www.pointcast.com
Commercial News Services on the Internet (listing) : http://www.jou.ufl.edu
WebGate: http://ngw.webgate.net
CNN Interactive: http://www.ibm.net
UseNet News Readers: http://tucows.netactive.co.za
East London Daily Dispatch: http://www.dispatch.co.za

13.12.8 PPP servers

Foray PPP Remote Access Server: http://www.techsmith.com

13.12.9 E-mail

Hotmail: http://www.hotmail.com
Eudora Webmail: http://www.eudoramail.com
Eudora Lite: http://www.eudora.com
Voice e-mail: http://eudora.qualcom.com

14

Security considerations

Objectives

When you have completed study of this chapter you should be able to:

- Explain the security problem
- Define the ways of controlling access to a network

14.1 The security problem

Although people tend to refer to the 'Internet' as one global entity, there are in fact three clearly defined subsets of this global network. Four, in fact, if one wishes to include the so-called 'community network'. It just depends on where the conceptual boundaries are drawn.

- In the center is the in-house corporate 'intranet', primarily for the benefit of the people within the organization
- The intranet is surrounded by the 'extranet', exterior to the organization yet restricted to access by business partners, customers and preferred suppliers
- Third, and this is optional, there can be a 'community' layer around the extranet. This space is shared with a particular community of interest, e.g. industry associations
- Finally, these three layers are surrounded by the global Internet as we know it, which is shared by prospective clients/customers and the rest of the world

This expansion of the Internet into organizations, in fact right down to the factory floor, has opened the door to incredible opportunities. Unfortunately it has also opened the door to pirates and hackers. Therefore, as the use of the Internet, intranets, and extranets has grown, so has the need for security. The TCP/IP protocols and network technologies are inherently designed to be open in order to allow interoperability. Therefore, unless proper precautions are taken, data can readily be intercepted and altered – often without either the sending or the receiving party being aware of the security breach. Because dedicated links between the parties in a communication are often not established in advance, it is easy for one party to impersonate another party.

There is a misconception that attacks on a network will always take place from the outside. This is as true of networks as it is true of governments. In recent times the growth in network size and complexity has increased the potential points of attack both from outside and from within.

Without going into too much detail, the following list attempts to give an idea of the magnitude of the threat experienced by intranets and extranets:

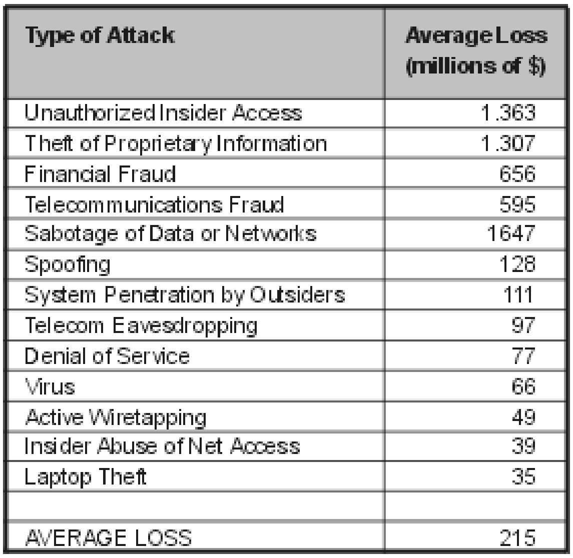
- Unauthorized access by contractors or visitors to a company's computer system
- Access by authorized users (employees or suppliers) to unauthorized databases. For example, an engineer might break into the Human Resources database to obtain confidential salary information
- Confidential information might be intercepted as it is being sent to an authorized user. A hacker might attach a network-sniffing device (probe) to the network, or use sniffing software on his computer. While sniffers are normally used for network diagnostics, they can also be used to intercept data coming over the network medium
- Users may share documents between geographically separated offices over the Internet or extranet, or 'telecommuters' users accessing the corporate intranet from their home computer via a dial-up connection can expose sensitive data as it is sent over the medium
- Electronic mail can be intercepted in transit, or hackers can break into the mail server

Here follows a list of some additional threats:

- SYN flood attacks
- Fat ping attacks (ping of death)
- IP spoofing
- Malformed packet attacks (TCP and UDP)
- ACK storms
- Forged source address packets
- Packet fragmentation attacks
- Session hijacking
- Log overflow attacks
- SNMP attacks
- Log manipulation
- ICMP broadcast flooding
- Source routed packets
- Land attack
- ARP attacks
- Ghost routing attacks
- Sequence number prediction
- FTP bounce or port call attack
- Buffer overflows
- ICMP protocol tunneling
- VPN key generation attacks
- Authentication race attacks

These are not merely theoretical concerns. While computer hackers breaking into corporate computer systems over the Internet have received a great deal of press in recent years, in reality, insiders such as employees, former employees, contractors working onsite, and other suppliers are far more likely to attack their own company's computer systems over an intranet. In a 1998 survey of 520 security practitioners in US corporations and other institutions conducted by the **Computer Security Institute** (CSI) with the participation of the FBI, 44 per cent reported unauthorized access by employees compared with 24 per cent reporting system penetration from the outside!

Such insider security breaches are likely to result in greater losses than attacks from the outside. Of the organizations that were able to quantify their losses, the CSI survey found that the most serious financial losses occurred through unauthorized access by insiders, with 18 companies reporting total losses of $51 million as compared with $86 million for the remaining 223 companies. The following list gives the average losses from various types of attacks as per the CSI/FBI 1998 Survey of Computer Security:



Type of Attack	Average Loss (millions of $)
Unauthorized Insider Access	1.363
Theft of Proprietary Information	1.307
Financial Fraud	656
Telecommunications Fraud	595
Sabotage of Data or Networks	1647
Spoofing	128
System Penetration by Outsiders	111
Telecom Eavesdropping	97
Denial of Service	77
Virus	66
Active Wiretapping	49
Insider Abuse of Net Access	39
Laptop Theft	35
AVERAGE LOSS	215

Fortunately technology has kept up with the problem, and the rest of this chapter will deal with possible solutions to the threat. Keep in mind that securing a network is a continuous process, not a one-time prescription drug that can be bought over the counter.

Also, remember that the most sensible approach is a defense-in-depth ('belt-and-braces') approach as used by the nuclear industry. In other words, one should not rely on a single approach, but rather a combination of measures with varying levels of complexity and cost.

14.2 Controlling access to the network

There are several ways of addressing the problem. These include:

- Authentication
- Routers
- Firewalls
- Intrusion detection systems
- Encryption

14.2.1 Authentication

A company whose LAN (or intranet) is not routed to the Internet mainly has to face internal threats to its network. In order to allow access only to authorized personnel, authentication is often performed by means of passwords. A password, however, is mainly used to 'keep the good guys out' since it is usually very easy to figure out someone's password, or to capture the password with a sniffer (protocol analyzer) as it travels across the network.

To provide proper authentication, two or three items from the following list are required.

- Something the user knows. These can be a password or a PIN number, and by itself it is not very secure
- Something the user has. This can be a SecurID tag, or similar. The SecurID system has a server on the network, generating a 6-bit pseudo-random code every 60 seconds. The user has a credit-card size card or a key fob with a 6-digit LCD display. After initialization at the server, the code on the user's card follows the code on the server. After entering a PIN number, the prospective user enters the 6-digit code. Even if someone manages to obtain the code, it will be useless in less than a minute
- Something the user is. This can be done with an iris or fingerprint scan. The hardware for this purpose is readily available

14.2.2 Routers

A router can be used as a simple firewall that connects the intranet to the 'outside world'. Despite the fact that its primary purpose is to route packets, it can also be used to protect the intranet.

In comparison to firewalls, routers are extremely simple devices and are clearly not as effective as firewalls in properly securing a network perimeter access point. However, despite their lack of sophistication, there is much that can be done with routers to improve security on a network. In many cases these changes involve little administrative overhead.

There are two broad objectives in securing a router, namely:

- Protecting the router itself
- Using the router to protect the rest of the network

Protecting the routers

The following approaches can be taken:

- Keep the router software current. This could be a formidable task, especially for managers maintaining a large routed network and are likely to be faced with the prospect of updating code on hundreds of devices. It is, however, essential since operating routers on current code is a substantial step toward protecting them from attack and properly maintaining security on a network. In addition, new updated software revisions often provide improved performance, offering more leeway to address security concerns without bringing network traffic to a halt
- It is imperative for network managers to keep current on release notes and vendor bulletins. Release notes are a good source of information and enable network managers to determine whether or not a fix is applicable to their organization. In the case of a detected vulnerability in the software for a

particular router, CERT advisories and vendor bulletins often provide workarounds to minimize risk until a solution to the problem has been found

- Verify that the network manager’s password is strong and make sure the password is changed periodically and distributed as safely and minimally as possible. More important, verify that all non-supervisory level accounts are password protected, to prevent unauthorized users from reading the router’s configuration information
- Allow TELNET access to the router only from specific IP addresses
- Authenticate any routing protocol possible
- From a security perspective, SNMP is a poor protocol to use. However, it does aid in managing the network. Defining a limited set of authorized SNMP management stations is always prudent

Protecting the network

- **Logging**
  Logging the actions of the router can assist in completing the overall picture of the condition of the network. The ideal solution is to keep one copy of the log on the router, as well as one on a remote logging facility, such as syslog, since an attacker could potentially fill the router’s limited internal log storage to erase details of the attack. With only remote storage, though, the attacker need only disrupt the logging service to prevent events from being recorded

- **Access control lists (ACLs)**
  ACLs allow the router to reject or pass packets based on TCP port number, IP source address or IP destination address. Traffic control can be accomplished on the basis of (a) implicit permission, which means only traffic not specifically prohibited will be passed through, or (b) implicit denial which means that all traffic not specifically allowed will be denied

14.2.3 Firewalls

Routers can be used to block unwanted traffic and therefore act as a first line of defense against unwanted network traffic, thereby performing basic firewall functions. It must, however, be kept in mind that they were developed for a different purpose, namely routing, and that their ability to assist in protecting the network is just an additional advantage. Routers, however sophisticated, generally do not make particularly intricate decisions about the content or source of a data packet. For this reason network managers have to revert to dedicated firewalls.

Firewalls are designed to sit on the boundary between an intranet and the rest of the world, monitoring both incoming and outgoing traffic, allowing only specific incoming and outgoing packets to pass and rejecting all other packets. This is not such an impossible task, since all TCP/IP communications is based on a port number contained in the TCP header. On the basis of the port number, a firewall can be instructed about who can transmit data, to what port they can transmit, and what sort of incoming connections are allowed on the network. One firewall is usually sufficient, but since a firewall only guards against attacks ‘from the other side’ and not from within, several of them might have to be deployed internally within an intranet if information on a particular part or ‘region’ has to be secured against other parts of the organization.

Firewalls are implemented in two ways – hardware-based and software-based. Hardware-based firewalls are dedicated self-contained ‘firewalls in boxes’, and are generally faster albeit more expensive. On the other hand, software firewalls are implemented with firewall software on individual hosts. This solution is generally less costly,

but slower.

Apart from the way they are implemented (i.e. hardware or software), firewalls can also be divided into two distinct types. The two most common types are packet filtering firewalls (also referred to as network layer firewalls) and application layer firewalls.

Network layer firewalls

Network layer firewalls deal mostly with routing rules. In other words, when a packet of data arrives at the firewall, it checks to see where the packet came from, where it is going, what it is used for, and then decides whether or not it is authorized. It monitors the actual content of data streams and the services exchanging these streams, while also checking for IP or DNS (domain name service) spoofing. The most distinguishing feature of a network layer firewall is its ability to allow IP traffic to pass through it. Network layer firewalls are almost completely transparent and anyone using the intranet will, generally, not even be aware of its presence. Unfortunately, this means that the intranet is probably going to need an assigned IP address block which can be difficult to obtain.

These routers employ several advanced techniques, including dynamic IP address allocation, sequence number scrambling, DMZ (de-militarized zoning) and ‘strikeback’. These techniques will now be discussed briefly in order to facilitate a better understanding of how these devices operate.

Dynamic IP address allocation

This is also known as **natural address translation** or NAT.

With NAT, the private IP addresses of machines inside the network are hidden from the outside world. They therefore need not be registered, and can be assigned by the system administrator. The firewall, on the other hand, has a built-in set of legitimate IP addresses, which are typically contained within one class C address.

An outward-bound packet sent by a host inside the Intranet follows a default route to the inside interface of the firewall. Upon receipt of the outbound packet, the firewall extracts the host’s source addresses (MAC and IP) and replaces it with its own MAC address and a globally unique IP number from the firewall’s pool of available IP addresses. The packet therefore seems to originate from the firewall. Since the difference between the original and translated versions of the packet are known, the checksums are updated with a simple adjustment rather than complete recalculation, which saves time.

Since it seems, to the outside world, as if the message has originated from the firewall, any returned messages would be routed back to the firewall. The firewall inspects returning packets, and once it is satisfied with their legitimacy, it strips the allocated IP address, returns it to the available pool of IP addresses, and restores the IP and MAC addresses of the original sender before sending it off to the originating host.

After a user-configurable timeout period during which there have been no returned packets for a particular address mapping, the firewall removes the entry, freeing the global address for use by another inside host. This is done so that a particular IP address will not be tied up indefinitely in the case of a packet getting lost along the way.

TCP sequence number randomization

Dynamic IP address allocation, while secure, is not port-specific and relies on a simple configuration table to track removed addresses. As a result, it does not provide absolute security because a spoofer could, theoretically, initiate a packet from outside the network that travels with a signal coming back through the configuration table; thus obtaining all addresses.

To remove this potential weakness of dynamic IP address allocation, firewalls can track the TCP sequence numbers and port numbers of originating TCP/IP connections. In order for spoofers to penetrate the firewall to reach an end server, they would need not only the IP address, but the port number and TCP sequence numbers as well.

To minimize the possibility of unauthorized network penetration, some firewalls also support sequence number randomization, a process that prevents potential IP address spoofing attacks, as described in a Security Advisory (CA-95:01) from the Computer Emergency Response Team (CERT). Essentially, this advisory proposes to randomize TCP sequence numbers in order to prevent spoofers from deciphering these numbers and then hijacking sessions. By using a randomizing algorithm to generate TCP sequence numbers, the firewall then makes this spoofing process extremely difficult, if not impossible. In fact, the only accesses that can occur through this type of firewall are those made from designated servers, which network administrators configure with a dedicated 'conduit' through the firewall to a specific server – and that server alone.

DMZs (de-militarized zones)

Most firewalls have two ports, one connected to the intranet and the other to the outside world. The problem arises: on which side does one place a particular (e.g. WWW, FTP or any other application) server? On either side of the firewall the server is exposed to attacks, either from insiders or from outsiders.

In order to address this problem, some firewalls have a third port, protected from both the other ports, leading to a so-called DMZ or de-militarized zone. A server attached to this port is protected from attacks, both from inside and outside.

Strike back intruder response

Some firewalls have a so-called intruder response function. If an attack is detected or an alarm is triggered, it collects data on the attackers, their source, and the route they are using to attack the system. They can also be programmed to automatically print these results, e-mail them to the designated person, or initiate a real-time response via SNAP or a pager.

Some firewalls will even send out a global distress call to all its peers (from the same manufacturer) and inform them of the origin of the attack. Although the actual attacker may be incognito, the router of his ISP is not, and can easily be traced. All the firewalls then start pinging the ISP's router 'to death' to slow it down or disable it.

Application layer firewalls

Application layer firewalls generally are hosts running proxy servers, and perform basically the same function as network layer firewalls, although in a slightly different way. Basically, an application layer firewall acts as an ambassador for a LAN or intranet connected to the Internet. Proxies tend to perform elaborate logging and auditing of all the network traffic intended to pass between the LAN and the outside world, and can cache (store) information such as web pages so that the client accesses it internally rather than directly from the Web.

A proxy server or application layer firewall will be the only Internet connected machine on the LAN. The rest of the machines on the LAN have to connect to the Internet via the proxy server, and for them Internet connectivity is just simulated.

Because no other machines on the network are connected to the Internet, a valid IP address is not needed for every machine. Application layer firewalls are very effective for small office environments that are connected with a leased line and do not have allocated

IP address blocks. They can even perform a dial-up connection on behalf of a LAN, and manage e-mail and any other Internet requests.

They do, however, have some drawbacks. Since all hosts on the network have to access the outside world via the proxy, any machine on the network that requires Internet access usually needs to be configured for the proxy. A proxy server hardly ever functions at a level completely transparent to the users. Furthermore, a proxy has to provide all the services that a user on the LAN uses, which means that there is a lot of server type software running for each request. This results in a slower performance than that of a network layer firewall.

Other types of firewalls

Stateful inspection firewalls are becoming very popular. They are software firewalls running on individual hosts and monitor the state of any active network connection on that host, and based on this information determines what packets to accept or reject. This is an active process that does not rely on any static rules. Generally speaking this is one of the easiest firewalls to configure or use.

14.3 Intrusion detection systems (IDS)

Intrusion detection is a new technology that enables network and security administrators to detect patterns of misuse within the context of their network traffic. IDS is a growing field and there are several excellent intrusion detection systems available today, not just traffic monitoring devices.

These systems are capable of centralized configuration management, alarm reporting, and attack info logging from many remote IDS sensors. IDS systems are intended to be used in conjunction with firewalls and other filtering devices, not as the only defence against attacks.

There are two ways that intrusion detection is implemented in the industry today: host-based systems and network-based systems.

14.3.1 Host-based IDS

Host-based intrusion detection systems use information from the operating system audit records to watch all operations occurring on the host on which the intrusion detection software has been installed. These operations are then compared with a pre-defined security policy. This analysis of the audit trail, however, imposes potentially significant overhead requirements on the system because of the increased amount of processing power required by the intrusion detection software. Depending on the size of the audit trail and the processing power of the system, the review of audit data could result in the loss of a real-time analysis capability.

14.3.2 Network-based IDS

Network-based intrusion detection, on the other hand, is performed by dedicated devices (probes) that are attached to the network at several points and passively monitor network activity for indications of attacks. Network monitoring offers several advantages over host-based intrusion detection systems. Because intrusions might occur at many possible points over a network, this technique is an excellent method of detecting attacks which may be missed by host-based intrusion detection mechanisms.

The greatest advantage of network monitoring mechanisms is their independence from reliance on audit data (logs). Because these methods do not require input from any operating system's audit trail they can use standard network protocols to monitor heterogeneous sets of operating systems and hosts.

Independence from audit trails also frees network-monitoring systems from possessing an inherent weakness caused by the vulnerability of the audit trail to attack. Intruder actions, which interfere with audit functions or which modify audit data can lead to the prevention of intrusion detection or the inability to identify the nature of an attack. Network monitors are able to avoid attracting the attention of intruders by passively observing network activity and reporting unusual occurrences.

Another significant advantage of detecting intrusions without relying on audit data is the improvement of system performance, which results from the removal of the overhead imposed by the analysis of audit trails. In addition, techniques, which move the audit data across network connections, reduce the bandwidth available to other functions.

14.4 Security management

14.4.1 Certification

Certification is the process of proving that the performance of a particular piece of equipment conforms to the laid-down policies and specifications. Whereas this is easy in the case of electrical wiring and wall sockets, where Underwriters' Laboratory can certify the product, it is a different case with networks where no official bodies and/or guidelines exist.

If one needs a certified network security solution, there are only two options viz:

- Trusting someone else's assumptions about one's network
- Certifying it oneself

It is possible to certify a network by oneself. This exercise will demand some time but will leave the certifier with a deeper knowledge of how the system operates.

The following are needed for self-certification:

- A company policy that favors security
- A security policy (see next section)
- Some basic knowledge of TCP/IP networking
- Access to the Web
- Time

To simplify this discussion, we will assume we are certifying a firewall configuration. Let us look at each individually.

A company policy that favors security

One of the biggest weaknesses in security practice is the large number of cases in which a formal vulnerability analysis finds a hole that simply cannot be fixed. Often the causes are a combination of existing network conditions, office politics, budgetary constraints, or lack of management support. Regardless of who is doing the analysis, management needs to clear up the political or budgetary obstacles that might prevent implementation of security.

Security policy

In this case, 'policy' means the access control rules that the network security product is intended to enforce. In the case of the firewall, the policy should list:

- The core services that are being permitted back and forth.
- The systems to which those services are permitted
- The necessary controls on the service, either technical or behavioral
- The security impact of the service
- Assumptions that the service places on destination systems

Basic TCP/IP knowledge

Many firewalls expose details of TCP/IP application behavior to the end user. Unfortunately, there have been cases where individuals bought firewalls and took advantage of the firewall's easy 'point and click' interface, believing they were safe because they had a firewall. One needs to understand how each service to be allowed in and out operates, in order to make an informed decision about whether or not to permit it.

Access to the Web

When starting to certify components of a system, one will need to research existing holes in the version of the components to be deployed. The Web, and its search engines, are an invaluable tool for finding vendor-provided information about vulnerabilities, hacker-provided information about vulnerabilities, and wild rumors that are totally inaccurate. Once the certification process has been deployed, researching the components will be a periodic maintenance effort.

Time

Research takes time, and management needs to support this and to invest the time necessary to do the job right. Depending on the size/complexity of the security system in question, one could be looking at anything between a day's work and several weeks.

14.4.2 Information security policies

The ultimate reason for having security policies is to save money.
This is accomplished by:

- Minimizing cost of security incidents; accelerating development of new application systems
- Justifying additional amounts for information security budgets
- Establishing definitive reference points for audits

In the process of developing a corporate security consciousness, one will, amongst other things, have to:

- Educate and train staff to become more security conscious
- Generate credibility and visibility of the information security effort by visibly driving the process from a top management level
- Assure consistent product selection and implementation
- Coordinate the activities of internal decentralized groups

The corporate security policies are not only limited to minimize the possibility on internal and external intrusions, but also to:

- Maintain trade secret protection for information assets
- Arrange contractual obligations needed for legal action
- Establish a basis for disciplinary actions
- Demonstrate quality control processes for example ISO 9000 compliance

The topics covered in the security policy document should, for example, include:

- Web pages
- Firewalls
- Electronic commerce
- Computer viruses
- Contingency planning
- Internet usage
- Computer emergency response teams
- Local area networks
- Electronic mail
- Telecommuting
- Portable computers
- Privacy issues
- Outsourcing security functions
- Employee surveillance
- Digital signatures
- Encryption
- Logging controls
- Intranets
- Microcomputers
- Password selection
- Data classification
- Telephone systems
- User training

In the process of implementing security policies, one need not re-invent the wheel. Products such as *Information Security Policies Made Easy* are available in a hardcopy book and CD-ROM. By using a word processing package, one can generate or update a professional policy statement in a couple of days.

14.4.3 Security advisory services

There are several security advisory services available to the systems administrator. This section will deal with only three of them, as examples.

Microsoft

All software vendors issue security advisories from time to time, warning users about possible vulnerabilities in their software. A particular case in point is Microsoft's advisory regarding the Word97 template security, which was issued on 19 January 1999. This weakness was exploited by a devious party who subsequently devised the Melissa virus. See Section 14.6 for a Web address.

CERT

The CERT (Computer Emergency Response Team) co-ordination center is based at the Carnegie Mellon Software Engineering Institute and offers a security advisory service on the Internet. Their services include:

- CERT advisories
- Incident notes
- Vulnerability notes
- Security improvement modules

The latter include topics such as:

- Detecting signs of intrusions
- Security for public web sites
- Security for information technology service contracts
- Securing desktop stations
- Preparing to detect signs of intrusion
- Responding to intrusions
- Securing network services

These modules can be downloaded from the Internet in PDF or PostScript versions and are written for system and network administrators within an organization. These are the people whose day-to-day activities include installation, configuration and maintenance of the computers and networks.

Once again, a particular case in point is the CERT/CC CA-99-04-MELISSA-MICRO-VIRUS.HTML dated March 27, 1999 which deals with the Melissa virus which was first reported at approximately 2:00 pm GMT-5 on Friday, 26 March 1999. This example indicates the swiftness with which organizations such as CERT react to threats.

CSI

CSI (The Computer Security Institute) is a membership organization specifically dedicated to serving and training the information computer and network security professionals. CSI sponsors two conferences and exhibitions each year: NetSec in June and the CSI Annual in November. CSI also hosts seminars on encryption, intrusion, management, firewalls and awareness. They also publish surveys and reports on topics such as computer crime and information security program assessment.

14.5 The public-key infrastructure (PKI)

14.5.1 Introduction to cryptography

The concept of securing messages through cryptography has a long history. Indeed, Julius Caesar is credited with creating one of the earliest cryptographic systems to send military messages to his generals.

Throughout history, however, there has been one central problem limiting widespread use of cryptography. That problem is key management. In cryptographic systems, the term key refers to a numerical value used by an algorithm to alter information, making that information secure and visible only to individuals who have the corresponding key to recover the information. Consequently, the term key management refers to the secure administration of keys to provide them to users where and when they are required.

Historically, encryption systems used what is known as symmetric cryptography. Symmetric cryptography uses the same key for both encryption and decryption. Using symmetric cryptography, it is safe to send encrypted messages without fear of interception, because an interceptor is unlikely to be able to decipher the message. However, there always remains the difficult problem of how to securely transfer the key to the recipients of a message so that they can decrypt the message.

A major advance in cryptography occurred with the invention of public-key cryptography. The primary feature of public-key cryptography is that it removes the need to use the same key for encryption and decryption. With public-key cryptography, keys come in pairs of matched ‘public’ and ‘private’ keys. The public portion of the key pair can be distributed in a public manner without compromising the private portion, which must be kept secret by its owner. Encryption done with the public key can only be undone with the corresponding private key.

Prior to the invention of public-key cryptography, it was essentially impossible to provide key management for large-scale networks. With symmetric cryptography, as the number of users increases on a network, the number of keys required to provide secure communications among those users increases rapidly. For example, a network of 100 users would require almost 5000 keys if it used only symmetric cryptography. Doubling such a network to 200 users increases the number of keys to almost 20 000. Thus, when only using symmetric cryptography, key management quickly becomes unwieldy even for relatively small-scale networks.

The invention of public-key cryptography was of central importance to the field of cryptography and provided answers to many key management problems for large-scale networks. For all its benefits, however, public-key cryptography did not provide a comprehensive solution to the key management problem.

Indeed, the possibilities brought forth by public-key cryptography heightened the need for sophisticated key management systems to answer questions such as the following:

- The encryption of a file once for a number of different people using public-key cryptography
- The decryption of all files that were encrypted with a specific key in case the key gets lost
- The certainty that a public key apparently originated from a specific individual is genuine and has not been forged by an imposter
- The assurance that a public key is still trustworthy

The next section provides an introduction to the mechanics of encryption and digital signatures.

14.5.2 Encryption and digital signature explained

To better understand how cryptography is used to secure electronic communications, a good everyday analogy is the process of writing and sending a cheque to a bank.

Remember that both the client and the bank are in possession of matching private key/public key sets. The private keys need to be guarded closely, but the public keys can be safely transmitted across the Internet since all it can do is unlock a message locked (encrypted) with its matching private key. Apart from that it is pretty useless to anybody else.

Securing the electronic equivalent of the cheque

The simplest electronic version of the cheque can be a text file, created with a word processor, asking a bank to pay someone a specific sum. However, sending this cheque over an electronic network poses several security problems:

Privacy

Enabling only the intended recipient to view an encrypted message. Since anyone could intercept and read the file, confidentiality is needed.

Authentication

Ensuring that entities sending the messages, receiving messages, or accessing systems are who they say they are, and have the privilege to undertake such actions. Since someone else could create a similar counterfeit file, the bank needs to authenticate that it was actually you who created the file.

Non-repudiation

Establishing the source of a message so that the sender cannot later claim that they did not send the message. Since the sender could deny creating the file, the bank needs non-repudiation.

Content integrity

Guaranteeing that messages have not been altered by another party since they were sent. Since someone could alter the file, both the sender and the bank need data integrity.

Ease of use

Ensuring that security systems can be consistently and thoroughly implemented for a wide variety of applications without unduly restricting the ability of individuals or organizations to go about their daily business.

To overcome these issues, the verification software performs a number of steps hidden behind a simple user interface. The first step is to ‘sign’ the cheque with a digital signature.

Digital signature

The process of digitally signing starts by taking a mathematical summary (called a hash code) of the cheque. This hash code is a uniquely identifying digital fingerprint of the cheque. If even a single bit of the cheque changes, the hash code will dramatically change.

The next step in creating a digital signature is to sign the hash code with the sender’s private key. This signed hash code is then appended to the cheque.

How is this a signature? Well, the recipient (in this case the bank) can verify the hash code sent to it, using the sender’s public key. At the same time, a new hash code can be created from the received check and compared with the original signed hash code. If the hash codes match, then the bank has verified that the cheque has not been altered. The bank also knows that only the genuine originator could have sent the cheque because only he has the private key that signed the original hash code.

Confidentiality and encryption

Once the electronic cheque is digitally signed, it can be encrypted using a high-speed mathematical transformation with a key that will be used later to decrypt the document. This is often referred to as a symmetric key system because the same key is used at both ends of the process.

As the cheque is sent over the network, it is unreadable without the key, and hence cannot be intercepted. The next challenge is to securely deliver the symmetric key to the bank.

Public-key cryptography for delivery symmetric keys

Public-key encryption is used to solve the problem of delivering the symmetric encryption key to the bank in a secure manner. To do so, the sender would encrypt the symmetric key using the bank's public key. Since only the bank has the corresponding private key, only the bank will be able to recover the symmetric key and decrypt the cheque.

Why use this combination of public-key and symmetric cryptography? The reason is simple. Public-key cryptography is relatively slow and is only suitable for encrypting small amounts of information – such as symmetric keys. Symmetric cryptography is much faster and is suitable for encrypting large amounts of information such as files.

Organizations must not only develop sound security measures, they must also find a way to ensure consistent compliance with them. If users find security measures cumbersome and time consuming to use, they are likely to find ways to circumvent them – thereby putting the company's Intranet at risk.

Organizations can ensure the consistent compliance to their security policy through:

- **Systematic application**
  The system should automatically enforce the security policy so that security is maintained at all times
- **Ease of end-user deployment**
  The more transparent the system is, the easier it is for end-users to use – and the more likely they are to use it. Ideally, security policies should be built into the system, eliminating the need for users to read detailed manuals and follow elaborate procedures
- **Wide acceptance across multiple applications**
  The same security system should work for all applications a user is likely to employ. For example, it should be possible to use the same security system whether one wants to secure e-mail, e-commerce, server access via a browser, or remote communications over a virtual private network

14.5.3 PKI definition (public-key infrastructure)

Imagine a company that wants to conduct business electronically, exchanging quotes and purchase orders with business partners over the Internet.

Parties exchanging sensitive information over the Internet should always digitally sign communications so that:

- The sender can securely identify themselves – assuring business partners that the purchase order really came from the party claiming to have sent it (providing a source authentication service)
- An entrusted third party cannot alter the purchase orders to request hypodermic needles instead of sewing needles (data integrity)

If a company is concerned about keeping the nature of particulars of their business private, they may also choose to encrypt these communications (confidentiality).

The most convenient way to secure communications on the Internet is to employ public-key cryptography techniques. But before doing so, the user will need to find and verify the public keys of the party with whom he or she wishes to communicate. This is where a public-key infrastructure comes in.

14.5.4 PKI functions

A successful public-key infrastructure needs to perform the following:

- Certify public keys (by means of certification authorities)
- Store and distribute public keys
- Revoke public keys
- Verify public keys

Let us now look at each of these in turn.

Certification authorities

Deploying a successful public-key infrastructure requires looking beyond technology. As one might imagine, when deploying a full scale PKI system, there may be dozens or hundreds of servers and routers, as well as thousands or tens of thousands of users with certificates. These certificates form the basis of trust and interoperability for the entire network. As a result, the quality, integrity, and trustworthiness of a public-key infrastructure depend on the technology, infrastructure, and practices of the certificate authority that issues and manages these certificates.

Certificate authorities (CA) have several important duties. First and foremost, they must determine the policies and procedures, which govern the use of certificates throughout the system.

The CA is a ‘trusted third party’, similar to a passport office, and its duties include:

- Registering and accepting applications for certificates from end users and other entities
- Validating entities’ identities and their rights to receive certificates
- Issuing certificates
- Revoking, renewing, and performing other life cycle services on certificates
- Publishing directories of valid certificates
- Publishing lists of revoked certificates
- Maintaining the strictest possible security for the CA’s private key
- Ensure that the CA’s own certificate is widely distributed
- Establishing trust among the members of the infrastructure
- Providing risk management

Since the quality, efficiency and integrity of any PKI depends on the CA, the trustworthiness of the CA must be beyond reproach.

On the one end of the spectrum, certain users prefer one centralized CA, which controls all certificates. Whilst this would be the ideal case, the actual implementation would be a mammoth task.

At the other end of the spectrum, some parties elect not to employ a central authority for signing certificates. With no CAs, the individual parties are responsible for signing each other’s certificates. If a certificate is signed by the user or by another party trusted by the user, then the certificate can be considered valid. This is sometimes called a ‘web of trust’ certification model. This is the model popularized by the PGP (pretty good privacy) encryption product.

Somewhere in the middle ground lies a hybrid approach that relies upon both independent CAs and peer-to-peer certification. In such an approach, businesses may act as their own CA, issuing certificates for its employees and trading partners. Alternatively, trading partners may agree to honor certificates signed by trusted third party CAs. This decentralized model most closely mimics today's typical business relationships, and it is likely the way PKIs will mature.

Building a public-key infrastructure is not an easy task. There are a lot of technical details to address – but the concept behind an effective PKI is quite simple: a PKI provides the support elements necessary to enable the use of public-key cryptography. One thing is certain: the public-key infrastructure will eventually – whether directly or indirectly – reach every Internet user.

Storage and distribution of public keys

E-commerce transactions don't always involve parties who share a previously established relationship. For this reason, a PKI provides a means for retrieving certificates. If provided with the identity of the person of interest, the PKI's directory service will provide the certificate. If the validity of a certificate needs to be verified, the PKI's certificate directory can also provide the means for obtaining the signer's certificate.

Revocation of public keys

Occasionally, certificates must be taken out of circulation, or revoked. After a period of time, a certificate will expire. In other cases, an employee may leave the company or a person may suspect that his or her private key has been compromised. In such circumstances, simply waiting for a certificate to expire is not the best option, but it is nearly impossible to physically recall all possible copies of a certificate already in circulation. To address this problem, CAs publish certificate revocation lists (CRLs) and compromised key lists (KRLs).

Verification of public keys

The true value of a PKI is that it provides all the pieces necessary to verify certificates. The certification process links public keys to individual entities, directories supply certificates as needed, and revocation mechanisms help ensure that expired or untrustworthy certificates are not used.

Certificates are verified where they are used, placing responsibility on all PKI elements to keep current copies of all relevant CRLs and KRLs. In an emerging standard, **on-line certificate status protocol** (OCSP) servers may take on CRL/KRL tracking responsibilities and perform verification duties when asked.

14.6 References

14.6.1 Internet/extranet/intranet security

General Index of Sources
http://www-ns.rutgers.edu
CERIAS
Centre for Education and Research in Information Assurance and Security
http://www.cerias.com
Notes on hijack detection
http://www.netsys.com
Network monitoring (network flight recorder)
http://www.nfr.com

COAST (Computer Operations, Audit and Security Technology)
http://www.cs-purdue.edu
ISS (Information System Support, Inc.)
http://www.iss-md.com
CSI (Computer Security Institute)
http://www.gocsi.com
Network Security Policies
http://www.baselinesoft.com
CERT Coordination Center (Carnegie Mellon Software Engineering Institute)
http://www.cert.org
Internet Security Magazine
http://www.securecomputing.com
An example of specific product updates e.g. Microsoft Office
http://officeupdate.microsoft.com

14.6.2 Encryption

Secure computing
http://www.sctc.com
VeriSign
http://www.verisign.com
PGP
http://pgp5.mit.edu
Entrust
http://www.entrust.com

14.6.3 Firewalls, proxy servers etc

CISCO systems
http://www.cisco.com
SECURE computing
http://securecomputing.com

- Security operating systems
- Virtual private networking

Firewall Report Overview
http://www.outlink.com
Secure Zone
http://www.sctc.com
WinGate
http://www.wingate.com

15

Process automation

Objectives

When you have completed study of this chapter you should be able to:

- Explain legacy architectures and the factory of the future
- Indicate the key elements of the modern Ethernet and TCP/IP architecture

15.1 Background

In the past, **supervisory control and data acquisition** (SCADA) functions were primarily performed by dedicated computer-based SCADA systems. Whereas these systems still do exist and are widely used in industry, the SCADA functions can increasingly be performed by TCP/IP/Ethernet-based systems. The advantage of the latter approach is that the system is open, hence hardware and software components from various vendors can be seamlessly and easily integrated to perform control and data acquisition functions.

One of the most far reaching implications of the Internet type approach, is that plants can be controlled and monitored from anywhere on the globe using the technologies that will be discussed in this chapter.

Stand-alone SCADA systems are still being marketed. However, SCADA vendors such as WIZNET are now also manufacturing Internet compatible SCADA systems that can easily be integrated into an existing TCP/IP/Ethernet plant automation system.

15.2 Legacy automation architectures

Traditionally, automation systems have implemented networking in a hierarchical fashion, with different techniques used for the so-called enterprise, device and 'fieldbus' layers. 'Fieldbus' is used here in a generic sense, and is printed in quotation marks in order to differentiate it from Foundation Fieldbus.

- The enterprise layer is found at the top of the network hierarchy. It provides communication between conventional computers which are used for applications such as e-mail and database applications. Users with browsers on their PCs could have access to this network. This network could also be connected via a firewall to the Internet in order to facilitate global access

- The device layer is found at the bottom of the hierarchy and is used to allow control systems such as PLCs access to the remote input/output (I/O). Devices at this level include PLCs and robots, and the buses are high performance cyclic buses
- The 'fieldbus' is found at the middle level and comprises networks with different levels of versatility and performance. This level in particular is highly fragmented, with strong proponents for each variant (ProfiBus, FIP, DeviceNet, ControlNet, Modbus Plus) and very little interoperability

The interfaces between 'layers' require intricate data collection and application gateway techniques. The task of configuring these devices, in addition to configuring the PLCs and enterprise layer computers, provides much scope for confusion and delays. In addition to this, the need for different network hardware and maintenance techniques in the three levels complicates spares holding and technician training. In order to overcome this problem, there is a growing tendency to use a single set of networking techniques (such as Ethernet and TCP/IP), to communicate at and between all three levels.

At the enterprise layer, this networking infrastructure is primarily used to transfer large units of information, on an irregular basis. Examples are sending electronic mail messages, downloading web pages, making ad-hoc SQL queries, printing documents, and fetching computer programs from file servers.

A particular problem area is the 'fieldbus' level. At this level, there is an attempt to mix routine scanning of data values with on-demand signaling of alarm conditions, along with transfer of large items such as control device programs, batch reports and process recipes. Unfortunately, there are many networks used at this level, such as ProfiBus, FIP, Modbus Plus, DeviceNet, Fieldbus Foundation H-1. Even worse, the design characteristics of each are sufficiently different to make seamless interconnection very difficult. In particular, all these networks have their own techniques for addressing, error checking, statistics gathering, and configuration. This imposes complications even when the underlying data itself is handled in a consistent way.

One technique commonly used to offset this problem is to divide the information available at each layer into 'domains', and have the devices which interconnect these domains be responsible for 'translating' requests for information. As an example, the PLC might use its device bus to scan raw input values, and then make a subset of them available as 'data points' on the 'fieldbus'. Similarly, a cell control computer or operator station might scan data points from its various 'fieldbus' segments, and make available selected data available in response to queries on the enterprise network.

Although these techniques can be made to work, they have a number of significant disadvantages:

- The intermediate 'boxes', known as gateways or data collectors, need to be configured to handle any data, which is processed through them. This means that if a PLC program is updated, it is necessary to update any HMI or cell controller programs to reflect the changes, otherwise the information reflected to the user level will be incomplete or inconsistent. Often this must be done with little automatic support from the device vendors, who jealously guard the 'features' of their data items and resist the attempt to 'dumb them down' by conforming to standard naming and attribute conventions

- Although devices like PLCs are designed to be extremely reliable, HMI and cell controllers are typically general-purpose computer systems, and will have a higher incidence of failures due to hardware or software problems. When such failures occur (and they will, even if care is taken in hardware design), it is important to be able to configure a replacement system and get it running as rapidly as possible. Many users today experience downtime of many hours if a single gateway or HMI goes down, because of the difficulty of getting a replacement device to the same state as one which failed

Typical MTBF (mean time between failures) of general-purpose computer systems are 50 000 hours for hardware and 14 000 hours for software. Typical MTBF for PLC systems are 100 000 hours. At these rates, a plant with 100 PLCs or computers would expect to experience about one failure requiring hardware replacement PER MONTH. Losing a number of hours' production each month due to hardware problems is an untenable situation. That is why automation vendors consider the ability to reinstall and restart a PLC or control system from virgin hardware in a rapid and reliable way to be mandatory.

15.3 The factory of the future

It is widely recognized nowadays that the traditional hierarchical structure of factory automation systems can be replaced with a single network, using the Internet as a model. In this model, all stations can conceivably intercommunicate and all stations can also communicate to the outside world via a firewall. Such a network would obviously be segmented for performance and security. The traditional computer-based gateways separating the three layers (enterprise, device, fieldbus) can now be replaced with dedicated bridges, switches and routers which feature a high degree of reliability.

One of the challenges in designing such a network-based solution, is the choice of a common interconnect. This should ideally be universal and vendor neutral and inexpensive to deploy in terms of hardware, cabling and training time. It should facilitate integration of all equipment within the plant, it should be simple to understand and configure, and it should be scalable in performance to support future growth of the network.

Five specific areas have to be addressed in order to enable the implementation of fully open (or 'transparent') control systems architecture for modern day factories. They are:

- The networking protocol stack
- Application layer data structures
- The use of embedded web servers
- The replacement of computer-based gateways with dedicated routers and switches
- Network access

15.3.1 The networking protocol stack

The ideal choice here is a TCP/IP. This enables integration with the Internet, enabling access to the plant on a global basis.

TCP/IP was originally designed for end-to-end connection oriented control over long-haul networks. It is tolerant of wide speed variations. It is compatible with firewalls and proxy servers, which are required for network security. It takes advantage of switching architectures for Internet and ATM, and is easy to troubleshoot with tools such as TELNET.

Considering all the advantages of TCP/IP, the overhead of 40 bytes per packet is a small price to pay.

15.3.2 Application layer data structures

An ideal solution for the implementation of the application layer is the MODBUS, a vendor neutral data representation protocol. MODBUS is referred to as ‘every vendor’s second choice but every integrator’s first choice’. Reasons for this include the fact that the specification is open and published and that the minimal implementation involves only two messages. It is easy to adapt existing serial interface software for MODBUS and also very easy to perform automatic protocol translation.

Although most implementations of the MODBUS protocol are used on low-speed point-to-point serial links or twisted pair multidrop networks, it has been adapted successfully for radio, microwave, public switch telephone, infrared, and almost any other communication mechanism conceivable. Despite the simplicity of MODBUS, it performs its intended function extremely well.

15.3.3 Embedded web servers

Once all computers and control devices are connected via a seamless Internet-compatible network, it becomes possible to use web servers to make plant information available to operators. This can be done in two different ways.

Firstly, a control device can incorporate its own local web server. This means that information, accessible only to that device, can be reported in a legible form as a set of web pages, and therefore displayed on any computer on the intranet, extranet, or Internet by means of a web browser.

Alternatively, a general-purpose computer can act as a web server, and gather data for individual web page requests by generating the native MODBUS requests used by the control devices. These requests can then be sent out over the MODBUS/TCP network, and will interrogate either the control devices directly (if they have TCP/IP interfaces) or via simple protocol converters (such as a MODBUS gateway) to convert requests into a form that legacy equipment would understand.

In both cases, there are two specific obstacles, namely that of reconfiguring a computer that has replaced a defective one, and maintaining the data directory.

In order to solve the problem of reconfiguring a computer after replacing a defective one, a network computer can be used as a web server, which means that the latter would be self-configuring on installation. A network computer installs itself from a server elsewhere on the network when it is powered up. This means that if such a computer were ever to fail, a new computer could be installed in its place, powered up, and it would immediately take on the same identity as its predecessor.

Another problem is how to present and maintain the directory, which stores and maintains the attributes of all data items. Despite a variety of proprietary solutions to this problem, there is an emerging standard called LDAP (lightweight directory access protocol), which was originally intended for keeping a registry of e-mail addresses for an organization. Under this scheme, LDAP maintains a hierarchical ‘picture’ of plant points within machines, machines within locations, and areas within an organization. LDAP makes it easy to reorganize the directory if the organization of the physical machines and data points need to be modified.

Each plant point could have attributes such as:

- Tag name
- Data type, scale, input limits, units
- Reference number, size, orientation
- Physical machine name

In addition to this, each physical machine has attributes such as:

- Network address
- Node number

This concept of enabling a device with a web server, and then controlling/supervising it with a browser, was dealt with in more detail in Chapter 9.

15.3.4 Routers and switches

The advantage of using Ethernet for the enterprise, device and 'fieldbus' networks, is that these levels can be interlinked by means of standard Ethernet compatible products such as routers and switches.

Switches

The use of switching hubs is the key to high performance coupling between the different plant network layers since it becomes easy to intermix stations of different speeds. Inserting switches between sub-networks requires no change to hardware or software and effectively isolates the traffic on the two network layers joined by it (i.e. the traffic on the subnets connected via the switch does not 'leak' across the switch). In order to preserve bandwidth, it is imperative not to use broadcast techniques.

Routers

In terms of inter-layer connection, routers can augment the speed adaptation function by being deployed in series with a switching hub. A throttling router, connected in series with the switch, can impose delays in order to achieve flow control in a situation where the destination network cannot cope with the data flow.

15.3.5 Network access

Ethernet is becoming the *de facto* standard for the implementation of the network access layer because of its scalability and low cost. The following paragraphs will briefly deal with the factors that until recently have been Ethernet shortcomings namely throughput, determinism and redundancy.

Throughput concerns

The entry-level Ethernet standard is10BaseT (IEEE 802.3) but this can be upgraded with little effort to 100BaseT (IEEE 802.3u) and even 1000BaseT (IEEE 802.3z) providing the network wiring has been done with CAT 5 UTP as per specifications. It is therefore one of the fastest network standards today.

Determinism (response time)

Until recently, it has been argued that Ethernet does not possess sufficient determinism. This problem had been solved by IEEE 802.1p – 'traffic class expediting' or 'message prioritization'. This specification addresses the need to deliver time critical messages in a deterministic fashion. Initially designed for multimedia applications, it directly impacts

Ethernet as a control network by allowing system designers to prioritize messages, guaranteeing the delivery of time critical data with deterministic response times. This ability has been used by companies such as HOST engineering and think & do software to produce an Ethernet bus that can provide deterministic scan time in the 2 to 3 millisecond range for one I/O rack with 128 points.

Redundancy

The IEEE 802.12d standard provides the ability to add redundant links to a network device. This facilitates automatic recovery of network connectivity when there is a link or repeater failure anywhere in the network path. This standard obviates the need for custom solutions when redundancy is required as part of the control solution.

15.3.6 Thin servers

Universal thin servers

A universal thin server is an appliance that network-enables any serial device such as a printer or weighbridge, which has an RS-232 port. In addition to the operating system and protocol independence of general thin servers, a universal thin server is application independent by virtue of its ability to network any serial device.

The universal thin server is a product developed primarily for environments in which machinery, instruments, sensors and other discrete 'devices' generate data that was previously inaccessible through enterprise networks. They allow nearly any device to be connected, managed and controlled over a network or the Internet.

Thin server applications

One of the pioneers in the field of universal thin servers is the US-based company Lantronix, that manufactures the MSS family of thin servers. In general, thin servers can be used for data acquisition, factory floor automation, security systems, scanning devices and medical devices.

One of the more unusual thin server applications regulates cattle feed in stock yards. Cattle wear radio frequency ID tags in their ears that relay data over a TCP/IP network as they step on to a scale. By the time the cattle put their heads in a trough to eat, the system has distributed the proper mix of feed.

Thin servers control video cameras used by the California Department of Transportation to monitor highway traffic. The US Border Patrol uses similar cameras to spot illegal border crossings. Food processing companies use the technology to track inventory in a warehouse, or the weight of consumable items rolling off an assembly line.

Here is another list of devices, which are being connected to Ethernet LANs via thin servers:

- Blood analyzers
- LAN security devices
- PBX accounting systems
- Card readers in debit systems
- Remote power management controllers
- Telecommunications equipment
- Displays in call centers
- Security alarms
- Time and attendance clocks and terminals
- Badge access control

- Customer traffic measurement
- UPS management devices
- High-end fax machines
- Electronic key systems
- Radiation equipment
- Marine equipment aboard ships
- Video cameras for ATM surveillance
- Vending machines
- Data loggers
- Static control boards
- Postal equipment
- CNC machines in machine shops
- Electronic signboards
- Temperature monitoring devices
- Chemical and gas chromatography instrumentation
- Oil rig monitors
- Satellite receivers
- Serial devices on wireless station adapters
- ATM machines on cruise ships
- Warehouse inventory tracking devices
- Unix workstations console port
- Refrigeration and heating controls
- Bar code scanners
- Heart monitors
- Electronic maps
- Power meter measurement devices
- Oil and gas automation
- Battery monitors
- Robotic controls
- Chemical monitors in pools
- Modems (character-mode)
- Weather stations
- Rocket launch pad telemetry equipment

15.3.7 Network capable application processors (NCAPs)

With the recently approved IEEE 1451.2 network independent standard, sensors and actuators can be easily interfaced onto control networks. By allowing the sensor to communicate directly on the network, complete distributed control can be achieved. This IEEE specification will act as a catalyst to sensor manufacturers who otherwise would have to support multiple protocols, hereby driving their costs up.

The IEEE 1451.2 activity is only one-half of the overall IEEE 1451 activity. IEEE 1451 is actually composed of two components, each of which is managed by its own working group. P1451.1 targets the interface between the smart device and the network, while the 1451.2 focuses on the interface between the sensor/transducer and the on-board microprocessor within the smart device.

IEEE 1451.1 defines a ‘Network Capable Application Processor (NCAP) Information Model’ that allows smart sensors and actuators to interface to many networks including

Ethernet. The standard strives to achieve this goal by means of a common network object model and use of a standard API, but it specifically does not define device algorithms or message content.

IEEE 1451.2 is concerned with 'Transducer To Microprocessor Communication Protocols and Transducer Electronic Data Sheet Formats'. This standard provides an interface that sensor and actuator suppliers can use to connect transducers to microprocessors within their smart device without worrying about what kind of microprocessor is on-board. A second part of the P1451.2 activity is the specification of electronic data sheets and their formats. These electronic data sheets, which amount to physically placing the device descriptions inside of the smart sensor, provide a standard means for describing smart devices to other systems. These transducer electronic data sheets, dubbed TEDS, also allow for self-identification of the device on the network.

One of the early adopters of this technology is Hewlett Packard. Using the standard and combining it with microprocessor-based technology with embedded Java software, HP's Ventera product line allows for seamless integration of any compatible IEEE 1451.2 sensors directly onto Ethernet. Using standard Web browser technology, users can obtain or modify sensor information by communicating with the sensor as if it were an URL address on the Web.

15.3.8 Ethernet compatible PLCs

There are several models available, one of the more popular ones being the series manufactured by KOYO in China, and marketed by companies such as Siemens, PLC Direct and Allen-Bradley under their own brand names.

An example of such a PLC is Allen-Bradley's PLC-5 Ethernet compatible PLC. This is a modular PLC, which accepts either a 10BaseT or a 10BaseF (fiber) communications module. The fiber option enables the PLC to operate in very (electrically) noisy environments, yet still retain their Ethernet connectivity.

The PLC-5 processors have TCP/IP and SNMP (simple network management protocol) built in, which enables them to be managed via the network using commercially available network management software.

15.3.9 Ethernet compatible SCADA systems

One of the industry's first Java-based SCADA systems is WIZNET, manufactured by Conlab. WIZNET allows a PLC to be integrated with the Plant Intranet (typically at the device layer level) and includes a web server. This allows operators and managers to monitor and control the plant through a standard Web browser, and view both factory data and corporate information through a common interface and from any desktop or mobile computer.

The web server provides security by allowing user access according to IP address or via selected web pages only.

15.4 References

15.4.1 Automation trends

ARC (Automation Research Corporation): http://www.arc.com

15.4.2 TCP/IP based factory automation

Schneider Automation: http://www.transparentfactory.com
Richard Hirschmann Gmbh: http://www.hirschmann.de
Allen-Bradley: http://www.ab.com

15.4.3 Thin servers

LANTronix, http://www.lantronix.com

15.4.4 Web compatible SCADA systems

WizNet, http://www.conlab.com.au

15.4.5 Java

RCS-7 Java, http://www.auspex-inc.com

16

Installing and troubleshooting Ethernet systems

Objectives

When you have completed study of this chapter you should be able to:

- Define the functions of various types of network driver software
- Describe the parameters which need to be set for a network card to function correctly
- Determine how network cards are configured under the plug and play and PCMCIA architectures
- Specify the uses for a protocol analyzer

16.1 Network drivers

16.1.1 Network drivers

The network driver is a program used to provide an interface between the network card and the higher-level protocols. It spans the data link and network layers of the OSI model as shown in Figure 16.1.

The network driver needs to match the specific hardware configuration of the network card such as its addresses of I/O ports, control and status registers, etc. Ethernet cards use the same IEEE 802.3 protocol so they can communicate with one another but each needs a unique driver because of the different vendor hardware implementations.

16.1.2 Compatibility and usage

The network driver must also be compatible with the appropriate network operating system protocols in the network, transport and session layers that are used to send data across the network. In early systems changing from one network protocol to another, e.g. TCP/IP to SPX/IPX, generally necessitated changing the network driver. The network operating systems communication protocols and their relationship to the OSI model are shown in Figure 16.1.

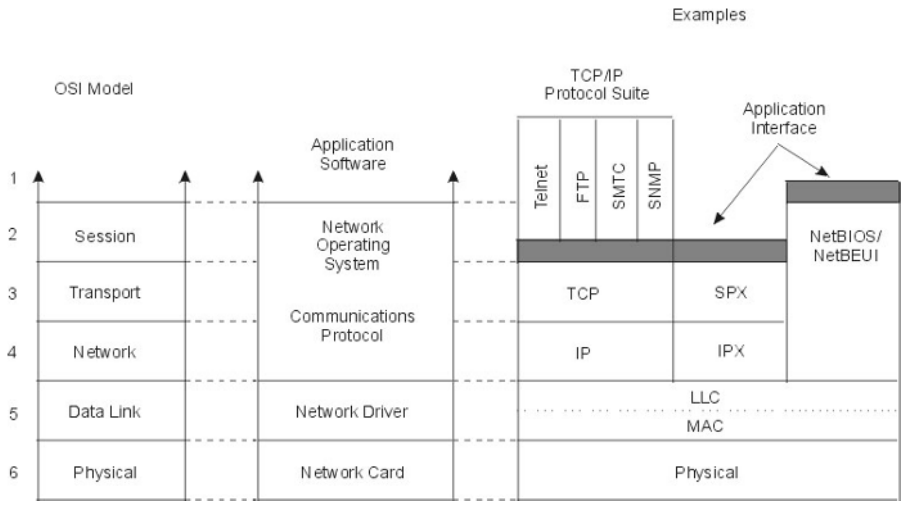


Figure 16.1
Network operating system drivers/protocols

16.2 Network card/driver configuration

The configuration of the network card must be set at installation to avoid conflict with the other devices installed on your computer. The manufacturers usually provide an installation guide and/or configuration software to help you set the correct options. The main parameters that may need to be set are as follows:

IRQ channel

The IRQ channel sets a unique interrupt request (IRQ) vector for when the network card needs attention. This must not conflict with existing devices. It needs to be checked in the normal operating environment. For example, with Windows applications the network card configuration software should be run from within the Windows shell.

DMA/shared memory

The Network card communicates its data to computer using either direct memory access (DMA) or use of a block of shared memory. The base address of the shared memory (usually 64 kilobytes) is defined here. Network cards are designed to be flexibly reconfigurable to accommodate other applications on your computer.

RAM base address

RAM base address defines the beginning of the address space (usually 16 kilobytes) used by the network card. Other devices must not use such address space. For example, extended memory manager software should be set to exclude such memory to avoid conflicts.

I/O base address

I/O base address defines the beginning of the address space used to communicate with the internal registers on the network card. Avoid conflicts with existing devices.

ROM base address and size

This is used for systems with an auto-boot ROM to configure diskless workstations. These load their operating systems over the network. Match the base address and EPROM size to the supplied boot ROM.

The network card configuration is set on the card using switches or links and/or stored in flash memory (EEPROM) on the card by the configuration software.

16.3 Network driver interface specification (NDIS)

This is a software specification developed by 3COM and Microsoft in 1988 for use mainly in DOS and OS/2 operating systems. This defines a standard interface for communication between the MAC layer and any compatible protocols. This means that the MAC driver in any vendor's NDIS compatible network cards can pass data to any NDIS compatible protocol. This standardization enables products from various vendors to be interconnected and to provide simultaneous support for multiple protocols. These NDIS drivers can be unloaded from memory to conserve DOS RAM space or allow changes to other drivers.

16.4 Open data link interface (ODI)

The ODI architecture provides an alternative to the OSI layering structure that can allow a number of network cards to simultaneously support different protocol stacks such as TCP/IP, SPX/IPX, etc.

The ODI architecture makes use of a link support layer (LSL) and a multiple link interface driver (MLID) as shown in Figure 16.2. The MLID corresponds to part of the data link layer and interfaces to the LSL, which covers part of both the data link and network layers. This provides a standard, hardware independent, virtual interface for the network cards. The LSL switches multiple protocol packets to the correct MLID or the correct protocol stack as required.

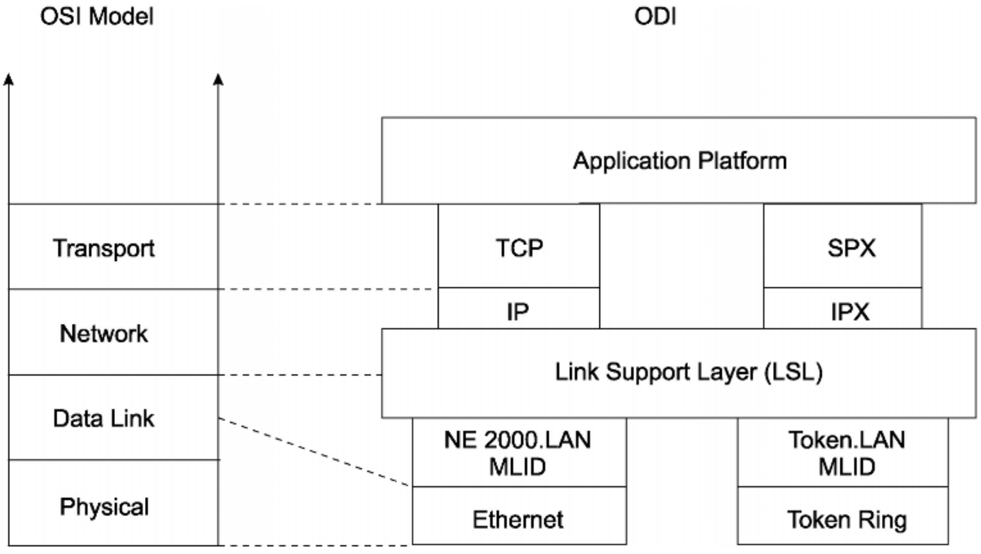


Figure 16.2
ODI architecture

16.5 Packet drivers

The packet driver is the generic interface between the TCP/IP protocol stack and the software responsible for the local area network card hardware. The packet driver hides the hardware specifics from the protocol stacks and likewise hides the protocol issues from the hardware. The packet driver operates at the MAC layer, and its implementation is critically dependent on the specific card hardware. Common driver specifications have evolved to provide standard interfaces to both the protocol stack and the card hardware, thus enabling the protocol to run on top of all network cards, which have that MAC driver specification.

A common example is the packet driver developed by FTP Software, Inc, in 1987. The specification defines how the MAC driver loads and operates under DOS and defines a common software interface for various protocol stacks. The network card is independent of the protocol stacks, and one card can simultaneously handle packets destined for multiple protocols. Each protocol uses a software interrupt in the range 60 h–80 h to communicate with the packet driver.

16.6 Plug and play architecture

Modern PC motherboards support the PC/ISA plug and play (PnP) architecture. With this architecture the PC identifies which slot the particular card is inserted into and the BIOS dynamically assigns the resources the card requires e.g. IRQ, DMA channel etc, resolving any resource conflicts. These resource details are registered in the ESCD (extended system configuration data) and stored in flash memory on the motherboard. The data structure defines the resources used by each device and card on the system. When using PnP it is important to register all legacy cards and devices (non PnP), otherwise resource conflicts are likely.

16.7 PCMCIA interface

16.7.1 Introduction

PCMCIA are the initials of the Personal Computer Memory Card International Association, which was formed in 1989 to promote the standardization and interchangeability of PC cards. As the name indicates initial devices were memory cards implemented as ‘virtual disk drives’ for mobile computer support. PC cards now come in a wide range of memory devices such as RAM, ROM FLASH memory, AT Attachment (ATA) hard drives, and many I/O devices including modems and network interface cards. The interface enables these devices to be powered from the computer and automatically detected by the system as soon as they are installed and then automatically configured. This gives the PC cards the ability to be inserted into a PCMCIA socket after the system has already been powered up.

16.7.2 PCMCIA interface

The PCMCIA interface consists of the following, as shown in Figure 16.3:

Hardware

- The 16-bit PC card, (A 32-bit PC card and socket interface also exists called CardBus)
- PCMCIA socket

- PCMCIA host bus adapter (HBA). These are hardware specific providing the interface between the host expansion bus (EISA, PCI, Micro Channel etc) and the standard interface to the PC card sockets

Software

- Socket services, which provide a standard low level software interface for programmers so that the details of the HBA do not need to be known
- Card services provide a high level software interface for configuration software and a method of allocating system resources to the PC cards
- PC card enablers, or PC card client drivers, read the PC card's **card information structure** (CIS), which is in non-volatile memory on the card, indicating the type of device, the resources it requires, and configuration options. The enabler then configures the HBA and PC card

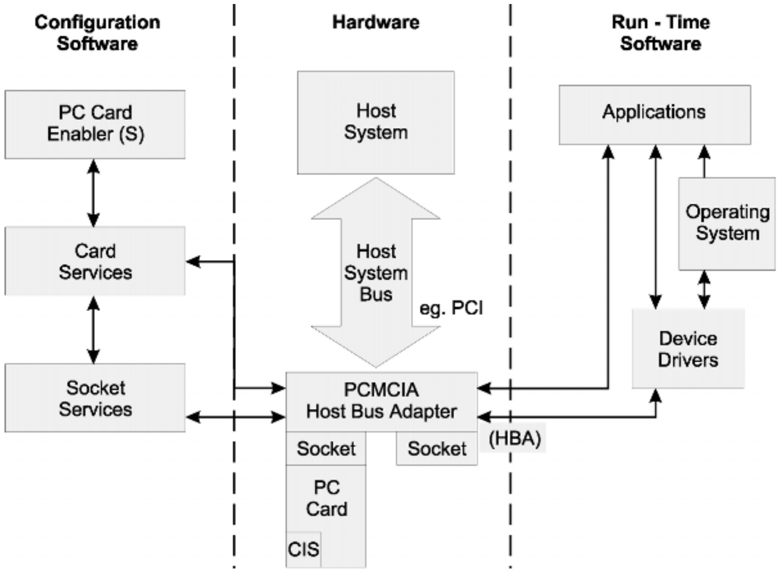


Figure 16.3
PCMCIA software and hardware relationships

After configuration, subsequent accesses to the PC card take place directly, without using the PCMCIA socket or card services, as illustrated above.

16.8 Protocol analyzers

Protocol analyzers enable us to capture data going across the LAN for purposes of analysis. An analyzer inserted into a ring or connected across a bus has the capability of looking at all messages being sent.

These messages can be stored in our computer-based protocol analyzer for subsequent analysis. By looking through the captured packet of data we can examine the message and protocol control information at each layer of the software. Protocol analyzers can operate in 'promiscuous mode' where they capture all messages from all nodes, or filters can be set so that the analyzer only captures those messages to or from specific nodes or in specific protocols. This is very useful for tracking down intermittent faults, so we can leave the analyzer running in the knowledge that it will capture the fault condition, along with all other packets from our faulty node, but will ignore the thousands of other packets from other users.

Protocol analyzers range in cost and complexity from simple analyzers based on a standard LAN card which will capture valid packets, to more sophisticated units which will also analyze the pulses, and capture packet fragments.

17

Troubleshooting TCP/IP

Objectives

When you have completed study of this chapter you should be able to:

- Describe how to do maintenance on TCP/IP networks
- Describe three typical areas requiring troubleshooting
- Describe how to troubleshoot with netstat, ping, tracert, ripquery

17.1 Maintenance and troubleshooting of real TCP/IP networks

Obviously a pro-active approach to maintenance of the TCP/IP network is preferable to that of the troubleshooting which is really a reactive approach.

Network monitoring needs to be objective and the creation of a baseline is a useful start. This comprises a set of monitoring points by which the network can be monitored and indeed be measured. It is important to distinguish between normal and peak network operation. If the network monitoring is done over a peak period such as large database backups being performed, this may result in false statistics being generated.

Typical network statistics that need to be monitored are:

Percentage utilization

This indicates how much of the total available activity on the network is used compared to the total (theoretically) available bandwidth.

Packets/second

This indicates the total number of messages on the bus. This is not the same as the utilization statistic, which indicates the total amount of data.

Kilobytes/second

This provides an indication of the actual throughput on the network.

Errors/second

This gives the total number of errors on the network. This would include such as items as electrical noise on the network. Be careful of some network analyzers which report collisions as errors – these are not errors but an essential part of the operation of Ethernet using the CSMA/CD philosophy.

Overruns

This indicates packets greater than 1518 bytes (1544 bytes in total length) and indicates a failing LAN driver.

Underruns

For similar causes to that of overruns, this indicates packets, which are shorter than 64 bytes.

Jabbers

This arises due to packets longer than 1518 bytes. This arises from a faulty LAN driver or faulty LAN hardware.

CRC/alignments

Packets that are not multiples of 8-bit bytes or have a CRC error may arise from noisy cables or defective components.

Collisions

This results from a collision between two (or more) Ethernet frames, which are greater than 64 bytes in length.

Note that some network analyzers (such as Snooper and the Netboy Suite) may not be able to detect some of these statistics (such as errors/second) when operating in promiscuous mode, as the physical Ethernet card may not provide this information.

Some of the important server-based baseline statistics are:

CPU utilization

This relates to the loading on the server CPU. This gives an indication of the capability of the network interface to maintain the performance levels.

Disk I/O

This indicates the speed of reading and writing the server/hosts file system(s). There should be some indication of the efficiency of the read and write caches.

Memory usage

This gives an indication of the performance of the server/host memory.

This information should be gathered continuously and then used to set up alarms for each measuring point. Typical alarm points should be set to about 10% of the averages recorded for each statistic gathered.

It should be noted that baseline statistics don’t stay static but are constantly changing and as any network change is effected, the new statistics will need to be gathered.

17.2 Network troubleshooting

According to Dr Tim Parker (TCP/IP Unleashed) there are four attributes you require in troubleshooting TCP/IP problems (and indeed most network problems):

- Some basic knowledge of the operation of networking protocols

- A clear understanding of the network's topology and layout
- The ability to utilize the troubleshooting tools (such as a Protocol Sniffer)
- Some luck

The greater the strength of the first three items, the less reliance will be placed on luck. A few typical areas to examine are:

17.2.1 Increasing number of collisions on the Ethernet network

As the utilization of the network increases the number of network errors will increase (although depending on the protocol analyzer and the specific Ethernet card you are using you may not directly observe these statistics). The quickest way to rectify this problem is to reduce the traffic by segmenting the network into smaller sub-networks using bridges or switches. Typical utilization figures would be 2 to 3% on an industrial Ethernet network with a maximum average of 10%. A commercial type Ethernet network (e.g. banking) where the response time may not be as critical can tolerate utilization figures up to 25% without undue problems.

However if the number of errors on an Ethernet network is increasing but the utilization is staying at roughly the same level; this may be due to faulty hardware.

Be careful of some network monitoring packages – they report collisions (a normal part of the CSMA/CD Ethernet system) as errors. In addition, some versions of the TCP/IP Netstat utility also report collisions as errors.

17.2.2 Network utilization low but errors high

This can invariably be traced to a faulty networking component – either hardware or software. If the packets are undersized (less than 64 bytes) and the frame check sequence (FCS) is good, it is likely that the network device driver needs to be replaced. On the other hand if the FCS is bad and the packets are undersized, this may mean that the network device driver is faulty.

On the other hand if the packets are oversized (greater than 1518 bytes) and the FCS is good, this is referred to as excessive jabbering and probably means that the interface board or transceiver needs to be replaced. If the FCS is bad and the packets are greater than 1518 bytes, this probably means that the network device driver needs to be replaced.

17.2.3 High number of packets but low data transfers

As the traffic on a network increases it is expected that the number of packets increases and the total data throughput increases up to the maximum saturation amount of course. Hence the ratio of the number of packets transmitted per second and the total volume of data passed between hosts should remain roughly constant. If the number of packets increases without any corresponding increase in data throughput this could indicate potential routing problems/badly configured network applications or network components that are failing. A data capture will have to be performed to work out what is actually happening here.

17.3 Troubleshooting with TCP/IP Utilities

A complete list of utility programs for the TCP/IP environment is contained in the RFC 1340 and is discussed in an earlier chapter. Typical utilities useful for troubleshooting are:

NETSTAT

The command is usually typed in lower case letters only (netstat). This reports the status of the network interfaces and ports.

NETSTAT -i

The command typed is netstat -i, a space is inserted before the hyphen and the following letter (all letters in lower case only). These statistics relate to running totals from when the system was last started. The total number of packets should be balanced between all network interfaces on the network. The output and input errors for each of the interfaces should be carefully examined. Output errors generally indicate hardware failures of the local system. If the output errors increase to about a quarter of the total packet output, it is time to replace the hardware.

Input errors are more difficult to analyze. These could result from any host on the network – this being caused by faulty network hardware (network interface card/cabling/noise) or the network device driver or an overloaded host.

The collision statistic indicates the number of collisions and should be looked at as a percentage of the total number of packets transmitted and not as an absolute number.

The queue statistic indicates when the network is so busy and thus congested that the Network interface has packets queued and waiting to be sent. It should be zero.

NETSTAT

The most important statistic here is the Send-Q, which should be zero. A non-zero value indicates severe congestion.

NETSTAT -nr

The command is usually typed in lower case, with a space before the hyphen and following letters (netstat -nr). This gives information about the default gateway. This can be useful for troubleshooting why packets don't get to their destination.

Ping

This provides an indication of connectivity between hosts on a network (or a series of interconnected networks) and provides some measure of the time taken to traverse the internetworks (the round-trip delay).

In pinging a host, if it times out, this could mean that either there is no connectivity with the remote host or the subnet mask has been incorrectly set up. If it is possible to ping a remote host but not to Telnet to a remote host, the intermediate routing tables should be examined using ripquery or netstat -r. It may be that the time-to-live field in the IP packet for the Ping protocol is set to considerably longer than for the Telnet packets.

TRACEROUTE

This indicates the route followed by a typical message sent over the interconnected networks. Sending out packets with ever increasing UDP packets, which have successively greater Time-to-live values, does it. The traceroute command will trace the entire route up to a breakage point in the link.

ARP

This provides the mapping between the MAC and IP addresses.

Entries within the ARP table are temporary. The length of time for each entry depends on the arp program implementation. This can cause problems if duplicate IP addresses are on the network. Each of the separate hosts will send out ARP request and response packets detailing different physical addresses for the same IP address. In this case it would be worthwhile manually editing the specific ARP table entries.

Ripquery

This allows the administrator to investigate the contents of a host's routing table.

17.3.1 Example of use of a few of the utilities together

If the error messages such as 'network unreachable', 'host not responding', or 'network host unreachable' are received, the following procedure could be used to troubleshoot the system.

First of all use the 'ping' utility to confirm a connection failure.

Analyze routing table contents by running 'netstat –nr'

If no route is found, then manually enter the desired route to follow using the route add command.

Alternatively if a route exists, use 'traceroute' to find out the precise point of failure.

Interestingly enough, it is a worthwhile exercise to ping both the IP address and the domain name of the host. There is possibly a mismatch between the two addresses. This may only be temporary but it is enough to cause a problem in establishing a connection.

17.3.2 Unreliable connections

Intermittent problems are often more difficult to troubleshoot. Hardware problems such as faulty cables or networking devices can be detected with cable testers. However by the astute use of the TCP/IP utilities it is possible to identify whether the problem is hardware or software related (i.e. it may be the operation of the networking protocols that are the cause of the problem).

Typical steps to follow here would be:

- Use ping to check the connectivity of the remote host
- Use traceroute to check the route followed if ping indicates that there is no connectivity

If ping indicates that there is connectivity, it is possible that the remote host has performance problems. In this case use the 'netstat -i' utility to check the remote host. If there are values recorded within the Queue field, it could mean that packets are being delayed before being transmitted on this interface. The 0errs field should also be zero. If non-zero there could be a physical problem on the interface. Similarly the 1errs field should be zero. If this is not the case, potential problems could be the physical hardware or the interface card device driver or the host is overloaded.

- Use ping -t (or the spray command) to keep up with a sustained data transfer if you suspect the host is overloaded.

Finally check for overload on one of the network segments by examining the collision field within the netstat command. If the number of collisions exceeds 5% of the total number of messages it may be that one of the segments is overloaded.

17.3.3 Network congestion

If the network users report that the connections to a remote host seem slightly on the slow side, the following procedure should be used:

- Use the ping command to check the connectivity

- Hereafter use the ‘traceroute’ utility to check the time for each section of the route and compare this against the baseline statistics that have been recorded
- If a significant increase in response time between two gateways; remedial action is indicated for this section

18

Satellites and TCP/IP

18.1 Introduction

There is a fairly vigorous debate at present over the merits of using TCP/IP over a satellite-based communications channel. One of the greatest challenges with satellite-based networks is the high-latency or delays in transmission and the appropriate solution in dealing with this problem.

This chapter is broken up into the following sections:

- Introduction
- Overview of satellite communications
- Advantages of satellite communications
- Applications of satellite systems
- Review of TCP/IP
- Weaknesses of TCP/IP in satellite usage
- Methods of optimizing TCP/IP over satellites

18.2 Overview of satellite communications

Satellite communications has been around for a considerable time with the VSAT (very small aperture terminal) system being very popular for general use. This system could deliver up to 24 Mbps in a point-to-multi-point link. The alternative of a point-to-point link would deliver up to 1.5 Mbps. Customers have traditionally bought very specific time on a specific satellite. This is where the use of satellite communications distinguishes itself – for predictable communications. Typical applications here have been periodic uplinks by news providers. The more unpredictable Internet usage with surges in demand that often requires a quick response is not very suited to the use of satellites. NASA pioneered a satellite more focussed to personal usage such as for the Internet with the launch of its **advanced communications technology satellite** (ACTS). This is capable of delivering 100 Mbps of bandwidth using a Ka-band (20–30 GHz) spot-beam geo synchronous earth orbit (GEO) satellite system.

When a satellite is used in a telecommunications link there is a delay (referred to as latency) due to the path length from the sending station to the satellite and back to the receiving station at some other location on the surface of the earth. For a satellite in **geostationary earth orbit** (or GEO) the satellite is about 36 000 km above the equator and the propagation time for the radio signal to go to the satellite and return to earth is about 240 milliseconds. When the ground station is at the edge of the satellite footprint this one-way delay can increase to as much as 280 milliseconds. Additional delays may be incurred in the on-board processing in the satellite and any terrestrial- or satellite-to-satellite linking. The comparable delay for a 10 000 km fiber optic cable would be about 60 milliseconds.

Low earth orbit (LEO) satellites are typically located about 500–700 km above the surface of the earth. At this height the satellites are moving rapidly relative to the ground station and to provide continuous coverage a large number of satellites are used, together with inter-satellite linking. The IRIDIUM system uses 66 satellites and GLOBALSTAR uses 48 satellites. The propagation delay from the ground station to the satellite varies due to the satellite position, from about 2 milliseconds when the satellite is directly overhead to about 80 milliseconds when the satellite is near the horizon. Mobile LEO users normally operate with quasi-omnidirectional antennas while large feeder fixed earth stations need steerable antenna with good tracking capability. Large users need seamless hand-off to acquire the next satellite before the previous one disappears over the horizon.

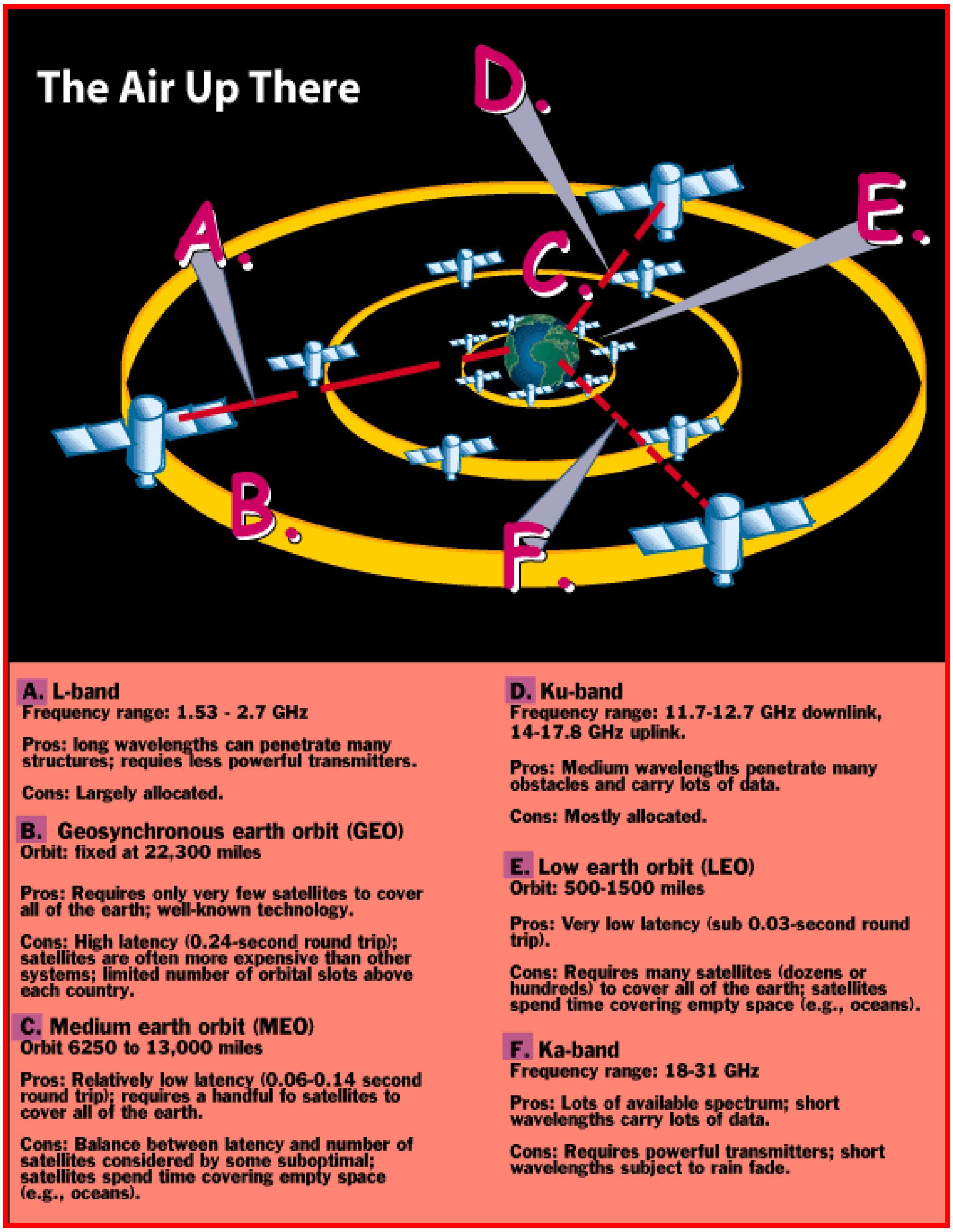
The main application where these satellite systems can be useful for is in providing high-bandwidth access to places where the landline system doesn’t provide this type of infrastructure. But the greatest challenge with satellite systems is the time it takes for your data to get from the one point to the other. A possible solution to reduce this latency is the use of low earth orbit (LEO) satellites. However, the problems with LEOs are the need to provide a large number to get the relevant coverage of the earth’s surface. In partnership with Boeing, Teledisc is targeting the provision of 288 LEO satellites. A further practical problem with LEO satellites is they only last 10 to 12 years before they burn up in falling to earth through the atmosphere. GEOs don’t have this particular problem as they are merely ‘parked’ a lot higher up and left. A further challenge with Leo’s is tracking these swiftly moving satellites, as they are only visible for up to 30 minutes before passing over the horizon. A phased-array antenna comprising a multitude of smaller antennas solves this antenna problem by tracking several different satellites simultaneously with different signals from each satellite. At least two satellites are kept in view at all times and the antenna initiates a link to a new one before it breaks the existing connection to the satellite moving to the bottom of the horizon.

The probable focus on using GEO and LEO satellites will probably be as follows:

- **GEO satellites** – Data downloading and broadcasting (higher latency)
- **LEO satellites** – High-speed networking/teleconferencing (lower latency)

Two other interesting issues for satellites that have not been quite resolved yet are the questions of security and the cost of the service. Theoretically, anyone with a scanner can tune in to the satellite broadcasts; although encryption is being used for critical applications. Also vendors claim that the costs of using satellites will be similar to that of existing landline systems. This is difficult to believe as the investment costs, (e.g. Iridium), in these systems have been extremely high.

A representation of the different satellite systems in use is indicated in the figure on the next page.



A. L-band
Frequency range: 1.53 - 2.7 GHz

Pros: long wavelengths can penetrate many structures; requies less powerful transmitters.

Cons: Largely allocated.

B. Geosynchronous earth orbit (GEO)
Orbit: fixed at 22,300 miles

Pros: Requires only very few satellites to cover all of the earth; well-known technology.

Cons: High latency (0.24-second round trip); satellites are often more expensive than other systems; limited number of orbital slots above each country.

C. Medium earth orbit (MEO)
Orbit 6250 to 13,000 miles

Pros: Relatively low latency (0.06-0.14 second round trip); requires a handful fo satellites to cover all of the earth.

Cons: Balance between latency and number of satellites considered by some suboptimal; satellites spend time covering empty space (e.g., oceans).

D. Ku-band
Frequency range: 11.7-12.7 GHz downlink, 14-17.8 GHz uplink.

Pros: Medium wavelengths penetrate many obstacles and carry lots of data.

Cons: Mostly allocated.

E. Low earth orbit (LEO)
Orbit: 500-1500 miles

Pros: Very low latency (sub 0.03-second round trip).

Cons: Requires many satellites (dozens or hundreds) to cover all of the earth; satellites spend time covering empty space (e.g., oceans).

F. Ka-band
Frequency range: 18-31 GHz

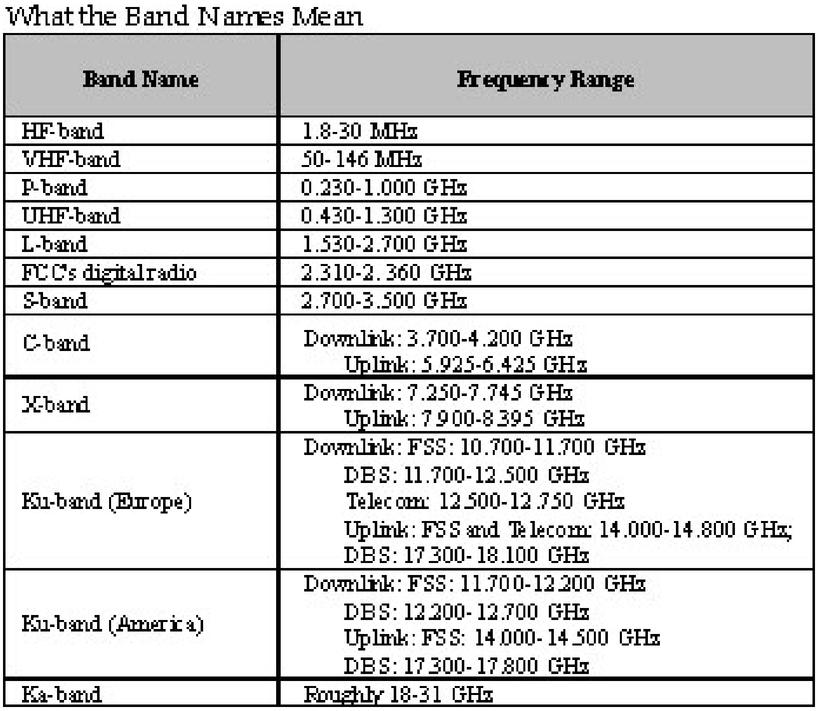
Pros: Lots of available spectrum; short wavelengths carry lots of data.

Cons: Requires powerful transmitters; short wavelengths subject to rain fade.

Figure 18.1
Satellite classifications (courtesy of Byte Publications ref. (1))

The important issues with the satellite system are the distance in which it orbits the earth and the radio frequency it uses. This impacts on the delay in data transfer (the latency) and the power of the signal and the data transfer rate.

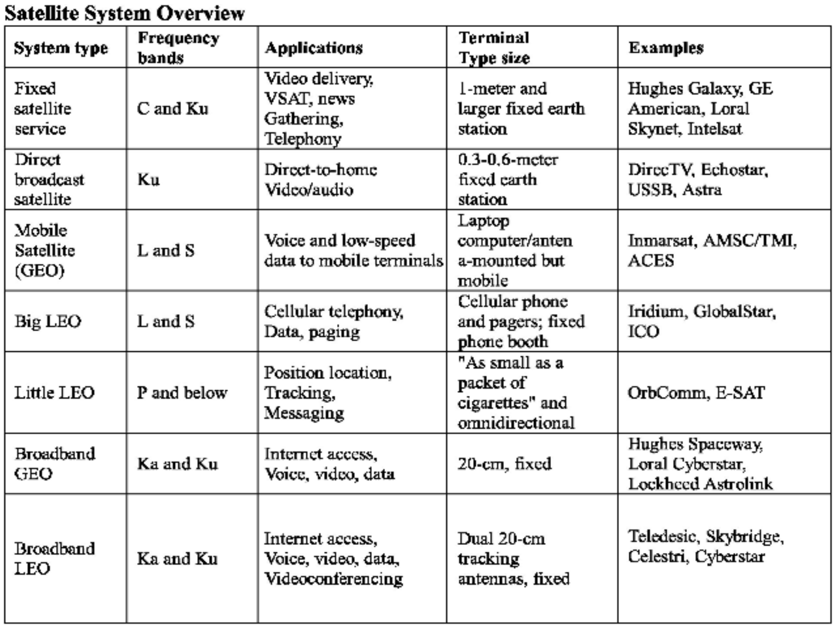
The various satellite bands can be listed as per the tables below. These tables have been sourced from reference (1).



What the Band Names Mean

Band Name	Frequency Range
HF-band	1.8-30 MHz
VHF-band	50-146 MHz
P-band	0.230-1.000 GHz
UHF-band	0.430-1.300 GHz
L-band	1.530-2.700 GHz
FCC's digital radio	2.310-2.360 GHz
S-band	2.700-3.500 GHz
C-band	Downlink: 3.700-4.200 GHz Uplink: 5.925-6.425 GHz
X-band	Downlink: 7.250-7.745 GHz Uplink: 7.900-8.395 GHz
Ku-band (Europe)	Downlink: FSS: 10.700-11.700 GHz DBS: 11.700-12.500 GHz Telecom: 12.500-12.750 GHz Uplink: FSS and Telecom: 14.000-14.800 GHz; DBS: 17.300-18.100 GHz
Ku-band (America)	Downlink: FSS: 11.700-12.200 GHz DBS: 12.200-12.700 GHz Uplink: FSS: 14.000-14.500 GHz DBS: 17.300-17.800 GHz
Ka-band	Roughly 18-31 GHz

Table 18.1
Frequency allocation



Satellite System Overview

System type	Frequency bands	Applications	Terminal Type size	Examples
Fixed satellite service	C and Ku	Video delivery, VSAT, news Gathering, Telephony	1-meter and larger fixed earth station	Hughes Galaxy, GE American, Loral Skynet, Intelsat
Direct broadcast satellite	Ku	Direct-to-home Video/audio	0.3-0.6-meter fixed earth station	DirecTV, Echostar, USSB, Astra
Mobile Satellite (GEO)	L and S	Voice and low-speed data to mobile terminals	Laptop computer/antenna-mounted but mobile	Inmarsat, AMSC/TMI, ACES
Big LEO	L and S	Cellular telephony, Data, paging	Cellular phone and pagers; fixed phone booth	Iridium, GlobalStar, ICO
Little LEO	P and below	Position location, Tracking, Messaging	"As small as a packet of cigarettes" and omnidirectional	OrbComm, E-SAT
Broadband GEO	Ka and Ku	Internet access, Voice, video, data	20-cm, fixed	Hughes Spaceway, Loral Cyberstar, Lockheed Astrolink
Broadband LEO	Ka and Ku	Internet access, Voice, video, data, Videoconferencing	Dual 20-cm tracking antennas, fixed	Teledesic, Skybridge, Celestri, Cyberstar

Table 18.2
Satellite classification

18.3 Advantages of satellite networks

Most TCP/IP traffic occurs over terrestrial networks (such as land lines, cable, telephone, fiber) with bandwidths ranging from 9600 bps to OC-12 with 622 Mbps. There are a number of opportunities for using satellite networks as a useful supplement to these terrestrial services. It is unlikely that the satellite will ever replace landline-based systems. But they will form a useful supplement to landline-based systems.

According to *Satellite Communications in the Global Internet – Issues, Pitfalls and Potential*, (see References at the end of this chapter) using satellites for the Internet (and by default the TCP/IP protocol) has the following advantages:

- The high bandwidth capability means that large amounts of data can be transferred. A Ka-band (20–30 GHz) satellite can deliver many gigabits/second
- Inexpensive means of transmission. There are no land line laying costs and the satellite can cover a huge area. Indeed, in remote areas the high costs may preclude using any land line with significant bandwidth
- Portability in communications in the satellite's range. The ability to move around may have great use for mobile applications
- Simplicity in network topology. The satellite has a very simple star type network structure. This is far easier to handle (especially for network management programs) than the complex interconnected mesh topology of land line-based Internet systems
- Broadcast and multicast. The satellite due to its star connection structure is easy to use in a broadcast capability. The typical mesh interconnection of land line-based systems is far more difficult to implement in a broadcast system

18.4 Applications of satellite systems

According to *Satellite Communications in the Global Internet – Issues, Pitfalls, and Potential* (see References at the end of this chapter), the following typical applications have the following features when running on a satellite system.

18.4.1 Remote control and login

These applications would be very sensitive to any delays in communications. If the delay extended beyond 30 to 40 ms the user would notice and not be very happy. Interestingly enough, one advantage that the satellite system does have over land-based systems is the fact that although the delays can be significant they are predictable and constant. This can be compared to the terrestrial Internet where response times can vary dramatically from one moment to the next.

18.4.2 Video conferencing

Assuming that the video conferencing application can tolerate a certain amount of loss (i.e. caused by transmission errors), the UDP protocol (see later) can be used. This has far less overhead than the TCP protocol as it does not require any handshaking to transfer data and is more compact. Hence satellites would provide an improvement over normal terrestrial communications with a better quality picture due to a greater bandwidth and a simpler topology. Another benefit that satellites would provide for video transmission is the ability to provide isochronous transmission of frames (i.e. a fixed time relationship between frames and thus no jerky pictures).

18.4.3 Electronic mail

This does not require instantaneous responses from the recipient and hence satellite communications would be well suited to this form of communications.

18.4.4 Information retrieval

The transmission of computer files requires a considerable level of integrity (i.e. no errors) and hence a reliable guaranteed protocol such as TCP has to be used on top of the IP protocol. This means that if a particularly fast response is required and a number of small transfers are used to communicate the data; satellite communications will not be a very effective means of operation.

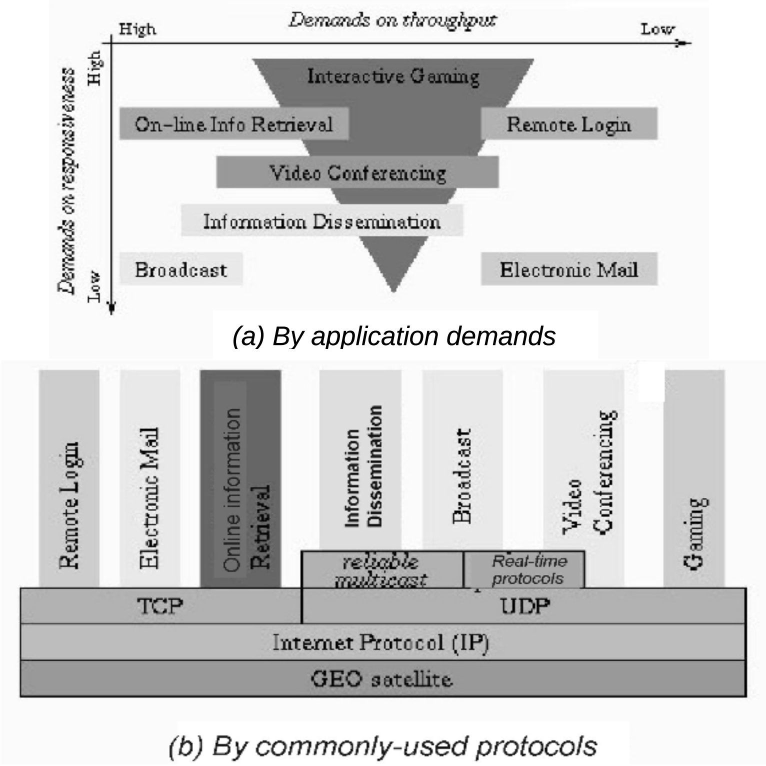
18.4.5 Bulk information broadcasting

Bulk data (such as from stock market databases/medical data/web casting/TV programs) can effectively be distributed by satellite with a vast improvement over the typically mesh-like land line based systems.

18.4.6 Interactive gaming

Computer games played on an interactive basis often require instantaneous reaction times. The inherent latency in a satellite system would mean that this is not particularly effective. Games that require some thought before a response is transmitted (e.g. chess and card games) would however be effective using satellite transmission.

The following diagrams summarize the discussion above.



(a) By application demands

(b) By commonly-used protocols

Figure 18.2
Summary of different satellite applications (*courtesy of* Satellite Communications in the Global Internet – Issues, Pitfalls, and Potential, *see References at the end of this chapter*)

18.5 Review of TCP/IP

As discussed earlier the TCP/IP protocol suite is the one on which the Internet is based. When it was developed very little consideration was taken of its performance over very high-speed fiber optic lines or very high latency satellite links. Some initiatives have been taken (see later) to address these shortcomings.

A quick revision of the main structure of the TCP/IP protocol is found below. The way in which the different protocols fit together is shown in the figure below.

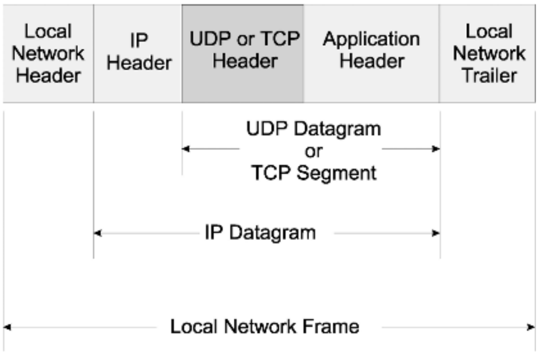


Figure 18.3
Typical structure of Ethernet, TCP/IP and application header

18.5.1 Internet protocol (or IP protocol)

The focus of the IP protocol is on routing data through multiple interconnected networks. The actual IP protocol (containing the data it is routing) can be transported on a number of different mechanisms such as Ethernet, token ring, Arcnet, ATM and frame relay. As each packet arrives at a router (also popularly referred to as a gateway) it is examined for the appropriate destination address and then sent onto the appropriate network. It is important to realize that the actual transport of the IP protocol is done with Ethernet using the Ethernet 48-bit hardware addressing. Ethernet (or any of the other mechanisms listed above) do not see the IP addressing. The IP address provides a hierarchical and unique way of identifying each of the different interconnected networks world-wide. The IP address comprises a host ID (station or PC address) and a net ID (or network address).

A typical protocol stack is shown in the figure below.

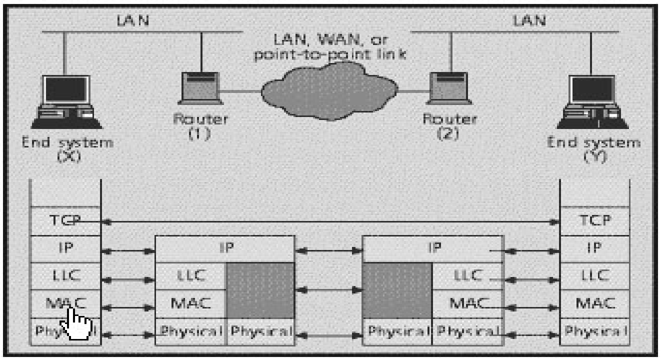


Figure 18.4
Typical protocol stack for TCP/IP (courtesy of Stallings 1997)

Note that datagrams (as the IP packets are known) may have to be broken down (or fragmented) into smaller packets as they pass through a router onto a network with a smaller frame size (e.g. from Ethernet with a maximum size of 1500 bytes to Arcnet with a maximum of approximately 800 bytes).

The IP protocol does not guarantee delivery of any of the packets. It merely handles the routing of the packets to its destination across the different interconnected networks. Packets could be lost due to routers becoming congested (and thus discarding packets) or due to corruption of the packets on a network due to electrical noise, for example. Hence the TCP protocol is used to guarantee delivery of the packets.

18.5.2 Transmission control protocol (or TCP)

Structure of TCP

The TCP protocol is used to guarantee delivery of the packet. Each byte of information is given a unique sequence number. The receiver keeps track of these sequence numbers and sends an acknowledgement to indicate to the originator of the packets that it has received the datagram up to a particular defined byte number.

The TCP protocol initiates the transfer of information using a three-way handshake in which it exchanges parameters with the node it is transferring this data to.

TCP flow control is based on the concept of a window. The window is used to determine how much data can be outstanding (i.e. unacknowledged) from the recipient of the information transfer. The amount of data that can be in transit is referred to as the bandwidth-delay product. The maximum window size is 64 kbytes (but practically it is often limited to 32 kbytes).

Sliding windows

Obviously there is a need to get some sort of acknowledgment back to ensure that there is a guaranteed delivery service. This technique, called positive acknowledgment with retransmission, requires the receiver to send back an acknowledgment message. Inherent in this is the concept of a timeout where a timer is started by the transmitter so that if no response is received from the destination node; another copy of the message will be transmitted. An example of this situation is given in the figure below.

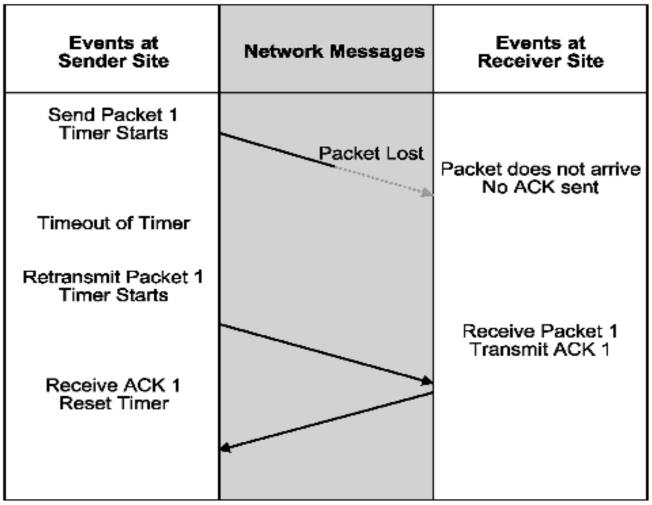

Events at Sender Site
Network Messages
Events at Receiver Site
Send Packet 1 Timer Starts
Packet Lost
Packet does not arrive No ACK sent
Timeout of Timer
Retransmit Packet 1 Timer Starts
Receive Packet 1 Transmit ACK 1
Receive ACK 1 Reset Timer

Figure 18.5
Positive acknowledgment philosophy

The sliding window form of positive acknowledgment is used with most efficient protocols, as it is very time consuming waiting for each individual acknowledgment to be returned for each packet transmitted. Hence the idea is that a number of packets (the window) is transmitted before the source may receive an acknowledgment to the first message (due to time delays, etc). As long as acknowledgments are received, the window slides along and the next packet is transmitted.

TCP uses a variable size-sliding window. Each acknowledgment from the receiver contains a window advertisement indicating how many additional bytes of data the destination will accept. The transmitting node then adjusts the size of its sliding window appropriately (either up or down). This can be considered to be a form of flow control. It is very useful for situations where one node is transmitting more data than the receiver can handle.

Maximum segment size

Both the transmitting and receiving nodes need to agree on the maximum size segments they will transfer. This is specified in the options field. There is an improvement in overall efficiency if the maximum segment size is selected that fills the physical packets that are transmitted across the network. The current specification recommends a maximum segment size of 536 (default size of IP datagram minus IP and TCP headers). If the size is not correctly specified; for example too small, the framing bytes consume most of the packet size resulting in considerable overhead; or too large, the packets have to be fragmented with a higher probability of loss of a packet and the resultant retransmission of the entire packet.

Acknowledgments

TCP/IP segments traveling through the Internet can be lost or arrive out of their sequence order. Each acknowledgment specifies a sequence number which is one greater than the highest byte received. Essentially acknowledgments always specify the sequence number of the next byte that the receiver expects to receive. Note further that the receiver always acknowledges the lowest contiguous prefix of the stream that has been correctly received.

Time out and retransmission

The TCP protocol starts a timer every time it transmits a segment. If no appropriate acknowledgment is received, TCP arranges to retransmit this segment. One of the problems with Internet is the rather variable time in receiving a response to a segment transmitted.

There are various algorithms to calculate the time out time.

A complication arises in calculating the round trip time for retransmitted segments. For example, if the transmitter times out waiting for an acknowledgment and then decides to send another packet and an acknowledgment arises shortly after the second packet is transmitted, the question arises as to which packet the acknowledgment refers to. This can affect the calculation of the round trip time dramatically. Karn’s algorithm, for example, can address this problem.

Congestion

There are two techniques used to reduce congestion on a network:

- **Multiplicative decrease**
  This approach is to reduce the size of the window for bytes to transmit by half on loss of a segment and for these segments still in the window, back off the retransmission time exponentially. This reduces the traffic dramatically and allows the gateways to eliminate the congestion.

- **Slow start recovery**
  When ramping up again in transmission rates, a technique called slow start (additive) recovery is used. This requires the traffic to be increased gradually by using a window of the size of a single segment and then increasing the window by one segment each time an acknowledgment arrives. This is a linear increase as opposed to original exponential increase when the transfer originally started.

The range of increase of the window is reduced once the window reaches one half of its original size. TCP, at this point, increases the window by one only if all segments in the window have been acknowledged.

Establishing/closing/resetting of a TCP connection

A three-way handshake (as indicated in the figure below) is used to establish a connection.

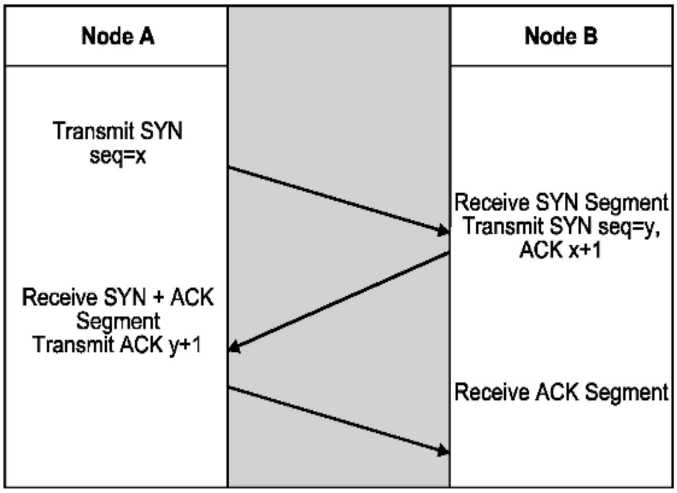


Figure 18.6
Three-way handshake

The SYN bit is set to one in the code field. As this is a full-duplex-based protocol it is possible for a connection to be established from both nodes at the same time.

There are two functions that the three-way handshake accomplishes:

- Both sides are ready to commence transfer of data
- A commencing sequence number is agreed upon

An initial sequence number must be chosen by each node (at random) to identify the bytes in the data stream it is transmitting. It should be realized that the acknowledgments indicate the number of the next byte expected.

When an application program has finished with transmission of its data, it advises the TCP software that it has no more data to transmit. The routine indicated in the figure below is then executed.

When an abnormal condition arises that forces an application program to terminate a connection, the reset bit is used (RST bit in the CODE). The destination responds immediately by aborting the connection.

Another protocol, which can be used to transfer data, is referred to as the **user datagram protocol** (UDP). This does not guarantee transfer of information but has considerably lower overhead than the TCP protocol.

User datagram protocol (UDP)

It should be noted, of course, that the UDP protocol still provides an unreliable connectionless delivery service as for the Internet protocol. Hence the application program must take account of the need for reliability, possibility of message loss, out of order delivery, etc.

The **user datagram protocol** (UDP) is the mechanism by which application programs send datagrams to other application programs. UDP has multiple protocol ports to identify the different programs executing on a particular node. As discussed in an earlier chapter, an abstract destination's source point on a computer is called a protocol port. There are two types of ports – destination ports on the remote computer node, which receives the message and source ports on the local computer node.

The UDP uses the underlying Internet protocol to transport a message from one node to the other. The UDP provides the facility of being able to distinguish among multiple destinations on a given host computer.

18.6 Weaknesses of TCP/IP in satellite usage

There are a number of weaknesses with the TCP/IP protocol (which are exacerbated with the use of high latency satellite links). These are listed below:

18.6.1 Window size too small

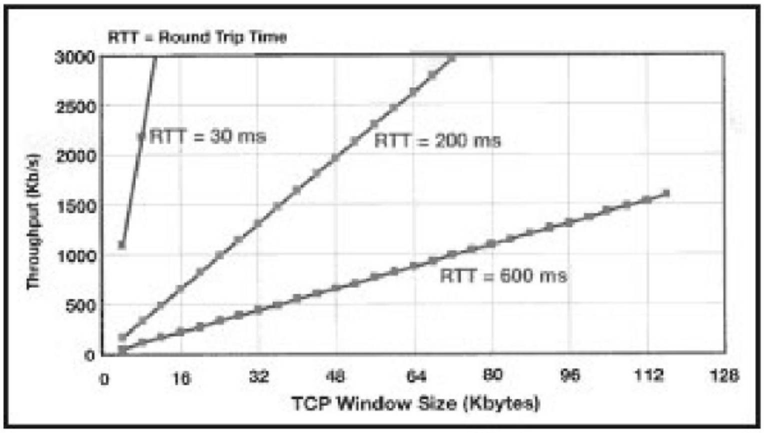


Figure 18.7
Maximum throughput for a single TCP connection as a function of window size and round trip time (RTT) (courtesy of Loyola University, see References at the end of this chapter)

In order to use the bandwidth of a satellite channel more effectively, TCP needs to have a larger window size. If a satellite channel has a round trip delay of say 600 msecs and the bandwidth is 1.54 Mbps, the bandwidth-delay product would be 0.924 Mbits which equates to 113 kbytes – this is considerably larger than the 64 kbyte maximum window size for TCP/IP.

18.6.2 Bandwidth adaptation

Due to the significant latency in the satellite links, TCP adapts rather slowly to bandwidth changes in the channel. TCP adjusts the window size upwards when the channel becomes congested and downward when the bandwidth increases. This means that TCP does not utilize the full bandwidth immediately but has a significant inertia in adapting.

18.6.3 Selective acknowledgment

When a segment is lost, TCP senders will retransmit all the data from the missing segment regardless of whether subsequent segments from the missing one were received correctly or not. This loss of a segment is considered evidence of congestion and the window size is also reduced to half. A more selective mechanism is required. There is a big difference between loss of segments due to real errors on the communications channel and congestion. TCP cannot distinguish between the two forms of missing segments.

18.6.4 Slow start

When a TCP transaction is commenced, an initial window size of one segment (normally about 512 octets) is selected. It then doubles the window size as successful acknowledgements are received from the destination up and until it reaches the network saturation state (where a packet is dropped). Hence again, this is a very slow way of ramping up to full bandwidth utilization. The total time for a TCP slow start period is calculated as:

Slow start time	=	RTT * log (B/MSS)
Where		
RTT	=	Round trip time
B	=	Bandwidth
MSS	=	TCP segment size

18.6.5 TCP for transactions

A TCP/IP transaction involves the use of the client–server interaction. The client sends a request to the server and the server then responds with the appropriate information (i.e. it provides a service to the client). In using the HTTP (**hypertext transfer protocol**), which is what the World Wide Web is based on, every item has to be commenced with the standard three-way handshake as outlined earlier and then the data transferred. This is particularly inefficient for small data transactions, as the process has to be repeated every time.

18.7 Methods of optimizing TCP/IP over satellite channels

There are various ways to optimize the use of TCP/IP over a satellite especially the need to mitigate the effects of latency. Interestingly enough, if these concerns with satellites can be addressed this will assist in the design and operation of future high-speed terrestrial networks because of the similar bandwidth * delay characteristic. The major problems for both satellites and high-speed networks with TCP/IP have been the need for a larger window size, the slow start period and ineffective bandwidth adaptation effects.

The various issues are discussed below:

Large window extension (TCP-LW)

A modification to the existing TCP/IP protocol allows a large window increasing the existing one from 2^{16} to 2^{32} bytes in size. This will allow more effective use of the communications channel with large bandwidth-delay products. Note that both the receiver and sender have to use a version of TCP that implements TCP-LW

Selective acknowledgment (TCP-SACK)

A newly defined standard entitled selective acknowledgment allows for the receiving node to advise the sender immediately of the loss of a packet. The sender will then immediately send a replacement packet thus avoiding the timeout condition and the consequent lengthy recovery in TCP (which would otherwise then have reduced its window size and then very slowly increased bandwidth utilization)

Congestion avoidance

There are two congestion avoidance techniques; but neither has been popular as yet. The first approach, which has to be implemented in a router, is called **random early detection** (RED) where the router sends an explicit notice of congestion (using the ICMP protocol discussed in an earlier chapter) when it believes that congestion will occur shortly if it doesn't take corrective action.

On the other hand an algorithm can be implemented in the sender where it observes the minimum round trip time for the packets it is transmitting to calculate the amount of data queued in the communications channel. If the number of packets being queued is increasing, it can reduce the congestion window. It will then increase the congestion window when it sees the number of queued packets decreasing.

TCP for transactions (T/TCP)

As discussed earlier, the three-way handshake represents a considerable overhead for small data transactions (often associated with HTTP transfers). An extension called T/TCP bypasses the three-way handshake and the slow-start procedure by using the data stored in a cache from previous transactions.

Middleware

It is also possible to effect significant improvements to the operation of TCP/IP without actually modifying the TCP/IP protocol itself using what is called middleware where split-TCP and TCP spoofing could be used.

Split-TCP

The end-to-end TCP connection is broken into two or three segments. This is indicated in the figure below. Each segment is in itself a complete TCP link. This means that the outer two links (which have minimal latency) can be setup as per usual. However the middle TCP satellite link with significant latency would have extensions to TCP such as TCP-LW and T/TCP. This means only minor modifications to the application software at each end of the link.

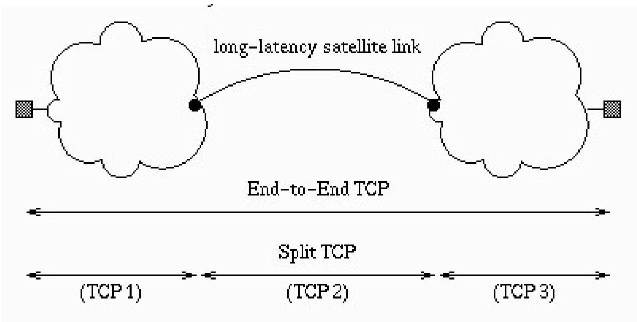


Figure 18.8
Use of Split TCP (courtesy of Loyola University)

TCP spoofing

An intermediate router (such as at the satellite uplink) immediately acknowledges all TCP packets coming through it to the receiver. All the receiver acknowledgment packets are suppressed so that the originator does not get confused. If the receiver does not receive a specific packet and the router has timed out, it will then retransmit this (missing) segment to the receiver. The resultant effect is that the originator believes that it is dealing with a low latency network.

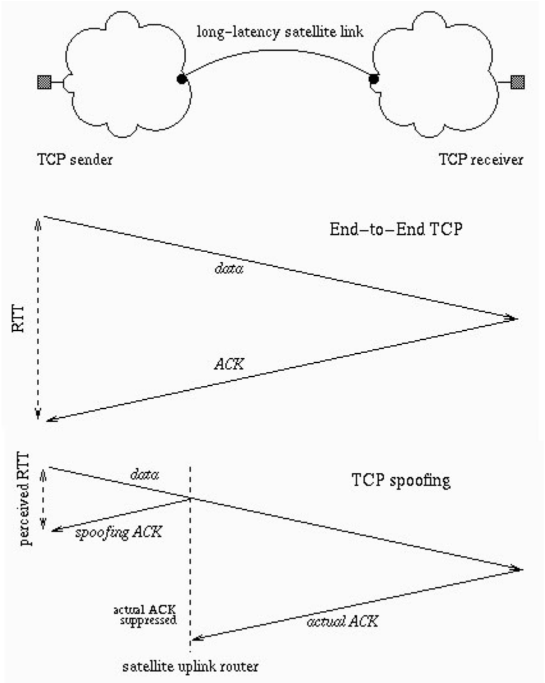


Figure 18.9
TCP spoofing (courtesy of Loyola University)

Application protocol approaches

There are three approaches possible here:

- Persistent TCP connections
- Caching
- Application specific proxies

Persistent TCP connections

In some client–server applications with very small amounts of data transfer, there are considerable inefficiencies. The HTTP 1.1 standard minimizes this problem and takes a persistent connection and combines all these transfers into one fetch. Further to this it pipelines the individual transfers so that there is an overlap of transmission delays thus making for an efficient implementation.

Caching

In this case, the commonly used documents (such as used with HTTP and FTP web protocols) are broadcast to local caches. The web clients then access these local caches rather than having to go through a satellite connection. The web clients thus have a resultant low latency and low network utilization (meaning more bandwidth available for higher speed requirements).

Application specific proxies

In this case, an application specific proxy can use its domain knowledge to pre-fetch web pages so that web clients subsequently requesting these pages considerably reduce the effects of latency.

References

There are a number of excellent references (many web site-based), which have been used in this document. It should be emphasized that due to the rapid changes in satellite communications with respect to TCP/IP, the Web is often the best source of information on this topic.

Montgomery, J. The Orbiting Internet: Fiber in the Sky. John Montgomery. *Byte Magazine*. November 1997.

Yongguang Zhang (ygz@isl.hrl.hac.com)
Dante De Lucia (dante@isl.hrl.hac.com)
Bo Ryu (ryu@isl.hrl.hac.com)
Son K. Dao (son@isl.hrl.hac.com)
Satellite Communications in the Global Internet – Issues, Pitfalls, and Potential.
Hughes Research Laboratories. Malibu, California 90265, U.S.A
Internet: http://www.wins.hrl.com/people/ygz/papers/inet97/index.html

Christoph Mahle (editor), Kul Bhasin, Charles Bostian, William Brandon, John Evans, Alfred Mac Rae. WTEC Panel Report on Global Satellite Communications Technology and Systems.
Internet: http://itri.loyola.edu/satcom2/04_05.htm

Suggested web sites with references:

Alcatel
Paris, France
Phone: +33 1 4058 5858
Internet: http://www.alcatel.com/our_bus/telecom/products/space
whatsnew.htm

Hughes Communications, Inc.
Long Beach, CA
Phone: 310-525-5000
Internet: http://www.spaceway.com

Lockheed
Sunnyvale, CA
Phone: 888-278-7565
Phone: 408-543-3103
Internet: http://www.astrolink.com

Loral
Palo Alto, CA
Phone: 650-852-5736
Internet: http://www.cyberstar.com

Motorola
Chandler, AZ
Phone: 602-732-4018
Internet: http://www.mot.com/

Teledesic
Kirkland, WA
Phone: 425-602-0000
Internet: http://www.teledesic.com

Appendix A

Glossary

10Base2
IEEE 802.3 (or Ethernet) implementation on thin coaxial cable (RG58/Au).

10Base5
IEEE 802.3 (or Ethernet) implementation on thick coaxial cable.

10Base-T
IEEE 802.3 (or Ethernet) implementation on unshielded 22 AWG twisted pair cable.

A

ABM
Asynchronous Balanced Mode

Access control mechanism
The way in which the LAN manages the access to the physical transmission medium.

Address
A normally unique designator for location of data or the identity of a peripheral device, which allows each device on a single communications line to respond to its own message.

Address resolution protocol (ARP)
A TCP/IP process used by a router or a source host to translate the IP address into the physical hardware address, for delivery of the message to a destination on the same physical network.

Algorithm
Normally used as a basis for writing a computer program. This is a set of rules with a finite number of steps for solving a problem.

Alias frequency
A false lower frequency component that appears in data reconstructed from original data acquired at an insufficient sampling rate (which is less than two (2) times the maximum frequency of the original data).

ALU
Arithmetic Logic Unit

Amplitude modulation
A modulation technique (also referred to as AM or ASK) used to allow data to be transmitted across an analog network, such as a switched telephone network. The amplitude of a single (carrier) frequency is varied or modulated between two levels one for binary 0 and one for binary 1.

Analog
A continuous real time phenomenon where the information values are represented in a variable and continuous waveform.

ANSI
American National Standards Institute. The national standards development body in the USA.

API
Application Programming Interface.

Appletalk
A proprietary computer networking standard initiated by the Apple Computer for use in connecting the Macintosh range of computers and peripherals. This standard operates at 230 kilobits/second.

Application layer
The highest layer of the seven-layer ISO/OSI reference model structure, which contains all user or application programs.

Application programming interface (API)
A specification defining how an application program carries out a defined set of services.

Arithmetic logic unit
The element(s) in a processing system that perform(s) the mathematical functions such as addition, subtraction, multiplication, division, inversion, AND, OR, NAND and NOR.

ARP
Address Resolution Protocol.

ARPANET
The packet switching network, funded by the DARPA, which has evolved into the world-wide Internet.

ARP cache
A table of recent mappings of IP addresses to the physical addresses, maintained in each host and router.

AS
Australian Standard

ASCII
American Standard Code for Information Interchange. A universal standard for encoding alphanumeric characters into 7 or 8 binary bits.

ASIC
Application Specific Integrated Circuit

ASN.1
Abstract Syntax Notation One. An abstract syntax used to define the structure of the protocol data units associated with a particular protocol entity.

Asynchronous
Communications where characters can be transmitted at an arbitrary unsynchronized point in time and where the time intervals between transmitted characters may be of varying lengths. Communication is controlled by start and stop bits at the beginning and end of each character.

Attenuation
The decrease in the magnitude of strength (or power) of a signal. In cables, generally expressed in dB per unit length.

Attenuator
A passive network that decreases the amplitude of a signal (without introducing any undesirable characteristics to the signals such as distortion).

AUI cable
Attachment Unit Interface Cable. Sometimes called the drop cable to attach terminals to the transceiver unit.

AWG
American Wire Gauge.

B

Balanced circuit
A circuit so arranged that the impressed voltages on each conductor of the pair are equal in magnitude but opposite in polarity with respect to ground.

Bandwidth
The range of frequencies available expressed as the difference between the highest and lowest frequencies is expressed in hertz (or cycles per second). Also used as an indication of capacity of the communications link.

Base address
A memory address that serves as the reference point. All other points are located by offsetting in relation to the base address.

Baseband
Baseband operation is the direct transmission of data over a transmission medium without the prior modulation on a high frequency carrier band.

Baud
Unit of signaling speed derived from the number of events per second (normally bits per second). However if each event has more than one bit associated with it the baud rate and bits per second are not equal.

BCC
Block Check Character. Error checking scheme with one check character; a good example being block sum check.

BCD
Binary Coded Decimal. A code used for representing decimal digits in a binary code.

BERT/BLERT
Bit Error Rate/Block Error Rate Testing. An error checking technique that compares a received data pattern with a known transmitted data pattern to determine transmission line quality.

BIOS
Basic Input/Output System.

Bipolar
A signal range that includes both positive and negative values.

BIT (binary digit)
Derived from 'BInary DigiT', a one or zero condition in the binary system.

BIT stuffing
Bit stuffing with zero bit insertion. A technique used to allow pure binary data to be transmitted on a synchronous transmission line. Each message block (frame) is encapsulated between two flags, which are special bit sequences. Then if the message data contains a possibly similar sequence, an additional (zero) bit is inserted into the data stream by the sender, and is subsequently removed by the receiving device. The transmission method is then said to be data transparent.

Bits per sec (bps)
Unit of data transmission rate.

Block sum check
This is used for the detection of errors when data is being transmitted. It comprises a set of binary digits (bits) which are the modulo 2 sum of the individual characters or octets in a frame (block) or message.

BNC
Bayonet type coaxial cable connector.

Bridge
A device to connect similar sub-networks without its own network address. Used mostly to reduce the network load.

Broadband
A communications channel that has greater bandwidth than a voice grade line and is potentially capable of greater transmission rates. Opposite of baseband. In wide band operation the data to be transmitted are first modulated on a high frequency carrier signal. They can then be simultaneously transmitted with other data modulated on a different carrier signal on the same transmission medium.

Broadcast
A message on a bus intended for all devices which requires no reply.

BS
British Standard.

BSC
Bisynchronous transmission. A byte or character oriented communication protocol that has become the industry standard (created by IBM). It uses a defined set of control characters for synchronized transmission of binary coded data between stations in a data communications system.

Buffer
An intermediate temporary storage device used to compensate for a difference in data rate and data flow between two devices (also called a spooler for interfacing a computer and a printer).

Burst mode
A high speed data transfer in which the address of the data is sent followed by back-to-back data words while a physical signal is asserted.

Bus
A data path shared by many devices with one or more conductors for transmitting signals, data or power.

Byte
A term referring to eight associated bits of information; sometimes called a ‘character’.

C

Capacitance
Storage of electrically separated charges between two plates having different potentials. The value is proportional to the surface area of the plates and inversely proportional to the distance between them.

Capacitance (mutual)
The capacitance between two conductors with all other conductors, including shield, short-circuited to the ground.

Cascade
Two or more electrical circuits in which the output of one is fed into the input of the next one.

CCITT (see ITU-T)
Consultative Committee on International Telegraphs and Telephone. A committee of the International Telecommunications Union (ITU) that sets world-wide telecommunications standards (e.g. V.21, V.22, V.22bis).

Character
Letter, numeral, punctuation, control code or any other symbol contained in a message.

Characteristic impedance
The impedance that, when connected to the output terminals of a transmission line of any length, makes the line appear infinitely long. The ratio of voltage to current at every point along a transmission line on which there are no standing waves.

Clock
The source(s) of timing signals for sequencing electronic events e.g. synchronous data transfer.

CMRR
Common Mode Rejection Ratio.

CMV
Common Mode Voltage.

CNR
Carrier to Noise Ratio. An indication of the quality of the modulated signal.

Collision
The situation when two or more LAN nodes attempt to transmit at the same time.

Common mode signal
The common voltage to the two parts of a differential signal applied to a balanced circuit.

Common carrier
A private data communications utility company that furnishes communications services to the general public.

Contention
The facility provided by the dial network or a data PABX which allows multiple terminals to compete on a first come, first served basis for a smaller number of computer posts.

CPU
Central Processing Unit.

CRC
Cyclic Redundancy Check. An error-checking mechanism using a polynomial algorithm based on the content of a message frame at the transmitter and included in a field appended to the frame. At the receiver, it is then compared with the result of the calculation that is performed by the receiver.

Cross talk
A situation where a signal from a communications channel interferes with an associated channel's signals.

CSMA/CD
Carrier Sense Multiple Access/Collision Detection. When two situations transmit at the same time on a local area network, they both cease transmission and signal that a collision has occurred. Each then tries again after waiting for a predetermined time period. This forms the basis of the IEEE 802.3 specifications.

D

Data link layer
This corresponds to layer 2 of the ISO reference model for open systems interconnection. It is concerned with the reliable transfer of data (no residual transmission errors) across the data link being used.

Datagram
A type of service offered on a packet-switched data network. A datagram is a self-contained packet of information that is sent through the network with minimum protocol overheads.

Decibel (dB)
A logarithmic measure of the ratio of two signal levels:
Where dB = 20log10V1/V2 or

Where dB = 10log10 P1/P2
And where V refers to Voltage or P refers to Power. Note that it has no units of measure.

Decoder
A device that converts a combination of signals into a single signal representing that combination.

Default
A value or setup condition assigned, which is automatically assumed for the system unless otherwise explicitly specified.

Delay distortion
Distortion of a signal caused by the frequency components making up the signal having different propagation velocities across a transmission medium.

DES
Data Encryption Standard.

Dielectric constant (E)
The ratio of the capacitance using the material in question as the dielectric, to the capacitance resulting when the material is replaced by air.

Digital
A signal, which has definite states (normally two).

DIN
Deutsches Institut für Normierung.

DIP
Acronym for dual in line package referring to integrated circuits and switches.

Direct Memory Access
A technique of transferring data between the computer memory and a device on the computer bus without the intervention of the microprocessor. Also abbreviated to DMA.

DNA
Distributed Network Architecture.

Driver software
A program that acts as the interface between a higher level coding structure and the lower level hardware/firmware component of a computer.

DSP
Digital Signal Processing.

Duplex
The ability to send and receive data simultaneously over the same communications line.

Dynamic range
The difference in decibels between the overload or maximum and minimum discernible signal level in a system.

E

EBCDIC
Extended Binary Coded Decimal Interchange Code. An eight-bit character code used primarily in IBM equipment. The code allows for 256 different bit patterns.

EDAC
Error Detection And Correction.

EEPROM
Electrically Erasable Programmable Read Only Memory. Non-volatile memory in which individual locations can be erased and re-programmed.

EIA
Electronic Industries Association. A standards organization in the USA specializing in the electrical and functional characteristics of interface equipment.

EIA-232-C
Interface between DTE and DCE, employing serial binary data exchange. Typical maximum specifications are 15 m at 19200 baud.

EIA-422
Interface between DTE and DCE employing the electrical characteristics of balanced voltage interface circuits.

EIA-423
Interface between DTE and DCE, employing the electrical characteristics of unbalanced voltage digital interface circuits.

EIA-449
General-purpose 37-pin and 9-pin interface for DCE and DTE employing serial binary interchange.

EIA-485
The recommended standard of the EIA that specifies the electrical characteristics of drivers and receivers for use in balanced digital multi-point systems.

EISA
Enhanced Industry Standard Architecture.

EMI/RFI
Electromagnetic Interference/Radio Frequency Interference. ‘Background noise’ that could modify or destroy data transmission.

EMS
Expanded Memory Specification.

Emulation
The imitation of a computer system performed by a combination of hardware and software that allows programs to run between incompatible systems.

Enabling
The activation of a function of a device by a defined signal.

Encoder
A circuit, which changes a given signal into a coded combination for purposes of optimum transmission of the signal.

EPROM
Erasable Programmable Read Only Memory. Non-volatile semiconductor memory that is erasable in an ultra violet light and reprogrammable.

Equalizer
The device, which compensates for the unequal gain characteristic of the signal received.

Error rate
The ratio of the average number of bits that will be corrupted to the total number of bits that are transmitted for a data link or system.

Ethernet
Name of a widely used LAN, based on the CSMA/CD bus access method (IEEE 802.3).

F

Farad
Unit of capacitance whereby a charge of one coulomb produces a one volt potential difference.

FCC
Federal Communications Commission.

FCS
Frame Check Sequence. A general term given to the additional bits appended to a transmitted frame or message by the source to enable the receiver to detect possible transmission errors.

FIFO
First In, First Out.

Filled cable
A cable construction in which the cable core is filled with a material that will prevent moisture from entering or passing along the cable.

FIP
Factory Instrumentation Protocol.

Firmware
A computer program or software stored permanently in PROM or ROM or semi-permanently in EPROM.

Flame retardancy
The ability of a material not to propagate flame once the flame source is removed.

Floating
An electrical circuit that is above the earth potential.

Flow control
The procedure for regulating the flow of data between two device preventing the loss of data once a device's buffer has reached its capacity.

Frame
The unit of information transferred across a data link. Typically, there are control frames for link management and information frames for the transfer of message data.

Frequency
Refers to the number of cycles per second.

Full-duplex
Simultaneous two way independent transmission in both directions (4 wire). See Duplex.

G

Giga (metric system prefix – 10^9).

Gateway
A device to connect two different networks which translates the different protocols.

Ground
An electrically neutral circuit having the same potential as the earth. A reference point for an electrical system also intended for safety purposes.

H

Half-duplex
Transmissions in either direction, but not simultaneously.

Hamming distance
A measure of the effectiveness of error checking. The higher the Hamming Distance (HD) index, the safer is the data transmission.

Handshaking
Exchange of predetermined signals between two devices establishing a connection.

HDLC
High Level Data Link Control. The international standard communication protocol defined by ISO to control the exchange of data across either a point-to-point data link or a multidrop data link.

Hertz (Hz)
A term replacing cycles per second as a unit of frequency.

Hex
Hexadecimal.

Host
This is normally a computer belonging to a user that contains (hosts) the communication hardware and software necessary to connect the computer to a data communications network.

I

I/O address
A method that allows the CPU to distinguish between different boards in a system. All boards must have different addresses.

IEC
International Electrotechnical Commission.

IEE
Institution of Electrical Engineers.

IEEE
Institute of Electrical and Electronic Engineers. An American-based international professional society that issues its own standards and is a member of ANSI and ISO.

IFC
International FieldBus Consortium.

Impedance (Z)
The total opposition that a circuit offers to the flow of alternating current or any other varying current at a particular frequency. It is a combination of resistance *R* and reactance *X*, measured in ohms.

Inductance
The property of a circuit or circuit element that opposes a change in current flow, thus causing current changes to lag behind voltage changes. It is measured in henrys.

Insulation resistance (IR)
That resistance offered by an insulation to an impressed DC voltage, tending to produce a leakage current though the insulation.

Interface
A shared boundary defined by common physical interconnection characteristics, signal characteristics and measurement of interchanged signals.

Interrupt
An external event indicating that the CPU should suspend its current task to service a designated activity.

Interrupt handler
The section of the program that performs the necessary operation to service an interrupt when it occurs.

IP
Internet protocol

ISA
Industry Standard Architecture (for IBM Personal Computers).

ISB
Intrinsically Safe Barrier.

ISDN

Integrated Services Digital Network. The new generation of world-wide telecommunications network that utilizes digital techniques for both transmission and switching. It supports both voice and data communications.

ISO
International Standards Organization.

ISR
Interrupt Service Routine. See Interrupt Handler.

ITU
International Telecommunications Union.

J

Jabber
Garbage that is transmitted when a LAN node fails and then continuously transmits.

Jumper
A wire connecting one or more pins on the one end of a cable only.

Jumper
A connection between two pins on a circuit board to select an operating function.

K

k
This is 2^{10} or 1024 in computer terminology, e.g. 1 kb = 1024 bytes.

L

LAN
Local Area Network. A data communications system confined to a limited geographic area typically about 3 kms with high data rates (4 Mbps to 155 Mbps).

LCD
Liquid Crystal Display. A low power display system used on many laptops and other digital equipment.

Leased (or private) line
A private telephone line without inter-exchange switching arrangements.

LED
Light emitting diode. A semi-conductor light source that emits visible light or infrared radiation.

Line driver
A signal converter that conditions a signal to ensure reliable transmission over an extended distance.

Linearity
A relationship where the output is directly proportional to the input.

Link layer
Layer two of the ISO/OSI reference model. Also known as the data link layer.

LLC
Logical Link Control (IEEE 802).

Loop resistance
The measured resistance of two conductors forming a circuit.

Loopback
Type of diagnostic test in which the transmitted signal is returned on the sending device after passing through all, or a portion of, a data communication link or network. A loopback test permits the comparison of a returned signal with the transmitted signal.

M

m
Meter. Metric system unit for length.

M
Mega. Metric system prefix for 10^6.

MAC
Media Access Control (IEEE 802).

Manchester encoding
Digital technique (specified for the IEEE 802.3 Ethernet baseband network standard) in which each bit period is divided into two complementary halves; a negative to positive voltage transition in the middle of the bit period designates a binary ‘1’, whilst a positive to negative transition represents a ‘0’. The encoding technique also allows the receiving device to recover the transmitted clock from the incoming data stream (self clocking).

MAP 3.0
Standard profile for manufacturing developed by MAP.

MAP
Manufacturing Automation Protocol. A suite of network protocols originated by General Motors, which follow the seven layers of the OSI model. A reduced implementation is referred to as a mini-MAP.

Mark
This is equivalent to a binary 1.

MAU
Media Access Unit.

MAU
Multistation Access Unit.

Media access unit
This is the Ethernet transceiver unit situated on the coaxial cable that then connects to the terminal with a drop cable.

Microwave
AC signals having frequencies of 1 GHz or more.

MIPS
Million Instructions Per Second.

MMS
Manufacturing Message Services. A protocol entity forming part of the application layer. It is intended for use specifically in the manufacturing or process control industry. It enables a supervisory computer to control the operation of a distributed community of computer-based devices.

MODEM
MODulator–DEModulator. A device used to convert serial digital data from a transmitting terminal to a signal suitable for transmission over a telephone channel or to reconvert the transmitted signal to serial digital data for the receiving terminal.

MOS
Metal Oxide Semiconductor.

MOV
Metal Oxide Varistor.

MTBF
Mean Time Between Failures.

MTTR
Mean Time To Repair.

Multidrop
A single communication line or bus used to connect three or more points.

Multiplexer (MUX)
A device used for division of a communication link into two or more channels either by using frequency division or time division.

Multistation access unit
Passive coupling unit, containing relays and transformers, used to star-wire the lobes of an IBM token ring system.

N

Narrowband
A device that can only operate over a narrow band of frequencies.

Network architecture
A set of design principles including the organization of functions and the description of data formats and procedures used as the basis for the design and implementation of a network (ISO).

Network driver
Program to provide interface between the network card (NIC) and higher layer protocols.

Network layer
Layer 3 in the ISO/OSI reference model, the logical network entity that services the transport layer responsible for ensuring that data passed to it from the transport layer is routed and delivered throughout the network.

Network topology
The physical and logical relationship of nodes in a network; the schematic arrangement of the links and nodes of a network typically in the form of a star, ring, tree or bus topology.

Network
An interconnected group of nodes or stations.

Node
A point of interconnection to a network.

Noise
A term given to the extraneous electrical signals that may be generated or picked up in a transmission line. If the noise signal is large compared with the data carrying signal, the latter may be corrupted resulting in transmission errors.

Non-linearity
A type of error in which the output from a device does not relate to the input in a linear manner.

NRZ
Non Return to Zero. Pulses in alternating directions for successive 1 bits but no change from existing signal voltage for 0 bits.

NRZI
Non Return to Zero Inverted.

O

OHM (Ω)
Unit of resistance such that a constant current of one ampere produces a potential difference of one volt across a conductor.

Optical isolation
Two networks with no electrical continuity in their connection because an optoelectronic transmitter and receiver has been used.

OSI
Open Systems Interconnection.

P

Packet
A group of bits (including data and call control signals) transmitted as a whole on a packet switching network. Usually smaller than a transmission block.

PAD
Packet Assembler/Disassembler. An interface between a terminal or computer and a packet switching network.

Parallel transmission
The transmission model where a number of bits are sent simultaneously over separate parallel lines. Usually unidirectional such as the Centronics interface for a printer.

PCIP
Personal Computer Instrument Products.

PCM
Pulse Code Modulation. The sampling of a signal and encoding the amplitude of each sample into a series of uniform pulses.

PCMCIA
Personal Computer Manufacturers Industries Association. Standard interface for peripherals for laptop computers.

PDU
Protocol Data Unit.

Peripherals
The input/output and data storage devices attached to a computer e.g. disk drives, printers, keyboards, display, communication boards, etc.

Physical layer
Layer one of the ISO/OSI reference model, concerned with the electrical and mechanical specifications of the network termination equipment.

PLC
Programmable Logic Controller.

PLL
Phase Locked Loop.

Point-to-point
A connection between only two items of equipment.

Polyethylene
A family of insulators derived from the polymerization of ethylene gas and characterized by outstanding electrical properties, including high IR, low dielectric constant, and low dielectric loss across the frequency spectrum.

Polyvinyl chloride (PVC)
A general purpose family of insulation whose basic constituent is polyvinyl chloride or its copolymer with vinyl acetate. Plasticizers, stabilizers, pigments and fillers are added to improve mechanical and/or electrical properties of this material.

Port
A place of access to a device or network, used for input/output of digital and analog signals.

Presentation layer
Layer 6 of the ISO/OSI Reference Model, concerned with negotiation of suitable transfer syntax for use during an application. If this is different from the local syntax, the translation to/from this syntax.

Protocol
A formal set of conventions governing the formatting, control procedures and relative timing of message exchange between two communicating systems.

Protocol entity
The code that controls the operation of a protocol layer.

PSDN
Public Switched Data Network. Any switching data communications system, such as Telex and public telephone networks, which provides circuit switching to many customers.

PSTN
Public Switched Telephone Network. This is the term used to describe the (analog) public telephone network.

PTT
Post, Telephone and Telecommunications Authority.

R

R/W
Read/Write.

RAM
Random Access Memory. Semiconductor read/write volatile memory. Data is lost if the power is turned off.

Reactance
The opposition offered to the flow of alternating current by inductance or capacitance of a component or circuit.

Repeater
An amplifier, which regenerates the signal and thus expands the network.

Resistance
The ratio of voltage to electrical current for a given circuit measured in ohms.

Response time
The elapsed time between the generation of the last character of a message at a terminal and the receipt of the first character of the reply. It includes terminal delay and network delay.

RF
Radio Frequency.

RFI
Radio Frequency Interference.

Ring
Network topology commonly used for interconnection of communities of digital devices distributed over a localized area, e.g. a factory or office block. Each device is connected to its nearest neighbors until all the devices are connected in a closed loop or ring. Data are transmitted in one direction only. As each message circulates around the ring, it is read by each device connected in the ring.

Rise time
The time required for a waveform to reach a specified value from some smaller value.

RMS
Root Mean Square.

ROM
Read Only Memory. Computer memory in which data can be routinely read but written to only once using special means when the ROM is manufactured. A ROM is used for storing data or programs on a permanent basis.

Router
A linking device between network segments which may differ in layers 1, 2a and 2b of the ISO/OSI reference model.

S

SAA
Standards Association of Australia.

SAP
Service Access Point.

SDLC
Synchronous Data Link Control. IBM standard protocol superseding the bisynchronous standard.

Serial transmission
The most common transmission mode in which information bits are sent sequentially on a single data channel.

Session layer
Layer 5 of the ISO/OSI reference model, concerned with the establishment of a logical connection between two application entities and with controlling the dialog (message exchange) between them.

Simplex transmissions
Data transmission in one direction only.

Slew rate
This is defined as the rate at which the voltage changes from one value to another.

SNA
Systems Network Architecture.

Standing wave ratio
The ratio of the maximum to minimum voltage (or current) on a transmission line at least a quarter-wavelength long. (VSWR refers to voltage standing wave ratio.)

Star
A type of network topology in which there is a central node that performs all switching (and hence routing) functions.

STP
Shielded Twisted Pair.

Switched line
A communication link for which the physical path may vary with each usage, such as the public telephone network.

Synchronization
The co-ordination of the activities of several circuit elements.

Synchronous transmission
Transmission in which data bits are sent at a fixed rate, with the transmitter and receiver synchronized. Synchronized transmission eliminates the need for start and stop bits.

T

TCP
Transmission Control Protocol.

TDR
Time Domain Reflectometer. This testing device enables the reflections user to determine cable quality with providing information and distance to cable defects.

Telegram
In general a data block which is transmitted on the network. Usually comprises address, information and check characters.

Temperature rating
The maximum, and minimum temperature at which an insulating material may be used in continuous operation without loss of its basic properties.

TIA
Telecommunications Industry Association.

Time sharing
A method of computer operation that allows several interactive terminals to use one computer.

Token ring
Collision free, deterministic bus access method as per IEEE 802.2 ring topology.

TOP
Technical Office Protocol. A user association in USA which is primarily concerned with open communications in offices.

Topology
Physical configuration of network nodes, e.g. bus, ring, star, tree.

Transceiver
A combination of transmitter and receiver.

Transceiver
Transmitter/Receiver. Network access point for IEEE 803.2 networks.

Transient
An abrupt change in voltage of short duration.

Transmission line
One or more conductors used to convey electrical energy from one point to another.

Transport layer
Layer 4 of the ISO/OSI reference model, concerned with providing a network independent reliable message interchange service to the application oriented layers (layers 5 through 7).

Twisted pair
A data transmission medium, consisting of two insulated copper wires twisted together. This improves its immunity to interference from nearby electrical sources that may corrupt the transmitted signal.

U

Unbalanced circuit
A transmission line in which voltages on the two conductors are unequal with respect to ground e.g. a coaxial cable.

UTP
Unshielded Twisted Pair.

V

Velocity of propagation
The speed of an electrical signal down a length of cable compared to speed in free space expressed as a percentage.

VFD
Virtual Field Device. A software image of a field device describing the objects supplied by it e.g. measured data, events, status etc, which can be accessed by another network.

VHF
Very High Frequency.

Volatile memory
AN electronic storage medium that loses all data when power is removed.

Voltage rating
The highest voltage that may be continuously applied to a wire in conformance with standards of specifications.

VSD
Variable Speed Drive.

VT
Virtual Terminal.

W

WAN
Wide Area Network.

Word
The standard number of bits that a processor or memory manipulates at one time. Typically, a word has 16 bits.

X

X.21
CCITT standard governing interface between DTE and DCE devices for synchronous operation on public data networks.

X.25
CCITT standard governing interface between DTE and DCE device for terminals operating in the packet mode on public data networks.

X.25 PAD
A device that permits communication between non X.25 devices and the devices in an X.25 network.

X.3/X.28/X.29
A set of internationally agreed standard protocols defined to allow a character oriented device, such as a visual display terminal, to be connected to a packet switched data network.

Appendix B

Port number allocation

As discussed earlier, there are three levels of addressing:

- The hardware address which resides on the network interface card
- The software or IP address which is broken down into a host and net portion and is set by the network administrator
- A host-to-host address (TCP) level of address known as the port address

Each PC or host is assumed to have numerous applications or processes running. An identifier known as the port number specifies the process the user wishes to access. These port numbers are 16 bits long and are standardized according to their use.

Port numbers 0–255 are assigned by the Internet administrator while the other numbers are available for local use. A complete listing of assigned ports is contained in the RFC 1700 but an abbreviated list is contained below.

Hence a message sent from one host to another requires the three addresses indicated above at the source and destination to complete the communications path. The combination of port address and IP address is often referred to as a socket.

Decimal	Keyword	Description
1	tcpmux	TCP Port service Multiplexer
5	rje	Remote Job Entry
7	echo	Echo
11	systat	Active Users
13	daytime	Daytime
17	qotd	Quote of the day
18	msp	Message Send Protocol
19	chargen	Character generator
20	ftp-data	File Transfer Protocol (Default Data)
21	ftp	File Transfer Protocol (Control)
23	telnet	TELNET
25	smtp	Simple Mail Transfer Protocol
33	dsp	Display Support Protocol
37	time	Time
38	rap	Route Access Protocol
42	nameserver	Host Name Server
43	nicname	Who Is
49	login	Login Host Protocol
53	domain	Domain Name Server
67	bootps	Bootstrap Protocol (Server)
68	bootpc	Bootstrap Protocol (Client)
69	tftp	Trivial File Transfer Protocol
70	gopher	Gopher
79	finger	Finger
80	www-http	World Wide Web HTTP
88	kerberos	Kerberos
92	npp	Network Printing Protocol
93	dcp	Device Control Protocol
101	hostname	NIC Host Name Server
102	iso-tsap	ISO-TSAP
107	rtelnet	Remote TELNET Service
109	pop2	Post Office Protocol - v.2
110	pop3	Post Office Protocol - v.3
111	sunrpc	SUN Remote Procedure Call
115	sftp	Simple File Transfer Protocol
129	pwdgen	Password Generator Protocol
137	netbios-ns	NetBIOS Name Service
138	netbios-dgm	NetBIOS Datagram Service
139	netbios-ssn	NetBIOS Session Service
143	imap2	Interim Mail Access Protocol V.2
144	news	News
146	iso-tp0	ISO-TP0
147	iso-ip	ISO-IP
152	bftp	Background File Transfer Protocol
153	sgmp	Simple Gateway Monitoring Protocol
160	sgmp-traps	SGMP-TRAPS
161	snmp	Simple Network Management Protocol
162	snmptrap	SNMPTRAP
163	cmip-manage	Common Management Information Protocol/TCP Manager
164	cmip-agent	CMIP/TCP Agent
165	xns-courier	Xerox
179	bgp	Border Gateway Protocol
190	gacp	Gateway Access Control Protocol
193	srmp	Spider Remote Monitoring Protocol
194	irc	Internet Relay Chat Protocol
199	smux	SNMP Multiplexing
201	at-rmtp	AppleTalk Routing Maintenance Protocol
202	at-nbp	AppleTalk Name Binding Protocol
203	at-3	AppleTalk Unused
204	at-echo	AppleTalk Echo Protocol
205	at-5	AppleTalk Unused
206	at-zis	AppleTalk Zone Information
207	at-7	AppleTalk Unused
208	at-8	AppleTalk Unused
209	tam	Trivial Authenticated Mail Protocol
220	imap3	Interactive Mail Access Protocol -v.3
246	dsp3270	Display Systems Protocol

Index