

# Networking Hardware

**Edward Beasley**





# **NETWORKING HARDWARE**



# **NETWORKING HARDWARE**

Edward Beasley



Networking Hardware  
by Edward Beasley

Copyright© 2022 BIBLIOTEX

[www.bibliotex.com](http://www.bibliotex.com)

All rights reserved. No part of this book may be reproduced or used in any manner without the prior written permission of the copyright owner, except for the use brief quotations in a book review.

To request permissions, contact the publisher at [info@bibliotex.com](mailto:info@bibliotex.com)

Ebook ISBN: 9781984664099



Published by:

Bibliotex

Canada

Website: [www.bibliotex.com](http://www.bibliotex.com)

# Contents

---

<b>Chapter 1</b>	The Advantages of Networking .....	1
<b>Chapter 2</b>	Local Access Network Design .....	20
<b>Chapter 3</b>	Network Protocols .....	37
<b>Chapter 4</b>	IP Routing .....	87
<b>Chapter 5</b>	Network Cabling.....	119
<b>Chapter 6</b>	Network Capacity .....	132
<b>Chapter 7</b>	Network Management.....	155
<b>Chapter 8</b>	Internet and World Wide Web .....	177





# 1

---

## The Advantages of Networking

---

You have undoubtedly heard the “the whole is greater than the sum of its parts”. This phrase describes networking very well, and explains why it has become so popular. A network isn’t just a bunch of computers with wires running between them.

Properly implemented, a network is a system that provides its users with unique capabilities, above and beyond what the individual machines and their software applications can provide. Most of the benefits of networking can be divided into two generic categories: *connectivity* and *sharing*. Networks allow computers, and hence their users, to be connected together. They also allow for the easy sharing of information and resources, and cooperation between the devices in other ways. Since modern business depends so

## Networking Hardware

much on the intelligent flow and management of information, this tells you a lot about why networking is so valuable.

Here, in no particular order, are some of the specific advantages generally associated with networking:

- *Connectivity and Communication:* Networks connect computers and the users of those computers. Individuals within a building or work group can be connected into *local area networks (LANs)*; LANs in distant locations can be interconnected into larger *wide area networks (WANs)*. Once connected, it is possible for network users to communicate with each other using technologies such as electronic mail. This makes the transmission of business (or non-business) information easier, more efficient and less expensive than it would be without the network.
- *Data Sharing:* One of the most important uses of networking is to allow the sharing of data. Before networking was common, an accounting employee who wanted to prepare a report for her manager would have to produce it on his PC, put it on a floppy disk, and then walk it over to the manager, who would transfer the data to her PC's hard disk. (This sort of "shoe-based network" was sometimes sarcastically called a "sneakernet".)

True networking allows thousands of employees to share data much more easily and quickly than this. More

so, it makes possible applications that rely on the ability of many people to access and share the same data, such as databases, group software development, and much more. Intranets and extranets can be used to distribute corporate information between sites and to business partners.

- *Hardware Sharing:* Networks facilitate the sharing of hardware devices. For example, instead of giving each of 10 employees in a department an expensive colour printer (or resorting to the “sneakernet” again), one printer can be placed on the network for everyone to share.
- *Internet Access:* The Internet is itself an enormous network, so whenever you access the Internet, you are using a network. The significance of the Internet on modern society is hard to exaggerate, especially for those of us in technical fields.
- *Internet Access Sharing:* Small computer networks allow multiple users to share a single Internet connection. Special hardware devices allow the bandwidth of the connection to be easily allocated to various individuals as they need it, and permit an organization to purchase one high-speed connection instead of many slower ones.
- *Data Security and Management:* In a business environment, a network allows the administrators to much better manage the company’s critical data.

### Networking Hardware

Instead of having this data spread over dozens or even hundreds of small computers in a haphazard fashion as their users create it, data can be centralized on shared servers. This makes it easy for everyone to find the data, makes it possible for the administrators to ensure that the data is regularly backed up, and also allows for the implementation of security measures to control who can read or change various pieces of critical information.

- *Performance Enhancement and Balancing:* Under some circumstances, a network can be used to enhance the overall performance of some applications by distributing the computation tasks to various computers on the network.
- *Entertainment:* Networks facilitate many types of games and entertainment. The Internet itself offers many sources of entertainment, of course. In addition, many multi-player games exist that operate over a local area network. Many home networks are set up for this reason, and gaming across wide area networks (including the Internet) has also become quite popular. Of course, if you are running a business and have easily-amused employees, you might insist that this is really a *disadvantage* of networking and not an advantage!

---

## **NETWORKING BASICS**

---

As companies rely on applications like electronic mail and database management for core business operations, computer networking becomes increasingly more important. This tutorial helps to explain Ethernet and Fast Ethernet, which are two of the most popular technologies used in networking.

### **LANs (LOCAL AREA NETWORKS)**

A network is any collection of independent computers that communicate with one another over a shared network medium. LANs are networks usually confined to a geographic area, such as a single building or a college campus. LANs can be small, linking as few as three computers, but often link hundreds of computers used by thousands of people. The development of standard networking protocols and media has resulted in worldwide proliferation of LANs throughout business and educational organizations.

### **WANS (WIDE AREA NETWORKS)**

Often a network is located in multiple physical places. Wide area networking combines multiple LANs that are geographically separate. This is accomplished by connecting the different LANs using services such as dedicated leased phone lines, dial-up phone lines (both synchronous and asynchronous), satellite links, and data packet carrier services. Wide area networking can be as simple as a modem

and remote access server for employees to dial into, or it can be as complex as hundreds of branch offices globally linked using special routing protocols and filters to minimize the expense of sending data sent over vast distances.

## **INTERNET**

The Internet is a system of linked networks that are worldwide in scope and facilitate data communication services such as remote login, file transfer, electronic mail, the World Wide Web and newsgroups.

With the meteoric rise in demand for connectivity, the Internet has become a communications highway for millions of users.

The Internet was initially restricted to military and academic institutions, but now it is a full-fledged conduit for any and all forms of information and commerce. Internet websites now provide personal, educational, political and economic resources to every corner of the planet.

## **INTRANET**

With the advancements made in browser-based software for the Internet, many private organizations are implementing intranets.

An intranet is a private network utilizing Internet-type tools, but available only within that organization. For large organizations, an intranet provides an easy access mode to corporate information for employees.

## **ETHERNET**

Ethernet is the most popular physical layer LAN technology in use today. Other LAN types include Token Ring, Fast Ethernet, Fiber Distributed Data Interface (FDDI), Asynchronous Transfer Mode (ATM) and LocalTalk. Ethernet is popular because it strikes a good balance between speed, cost and ease of installation. These benefits, combined with wide acceptance in the computer marketplace and the ability to support virtually all popular network protocols, make Ethernet an ideal networking technology for most computer users today. The Institute for Electrical and Electronic Engineers (IEEE) defines the Ethernet standard as IEEE Standard 802.3. This standard defines rules for configuring an Ethernet network as well as specifying how elements in an Ethernet network interact with one another. By adhering to the IEEE standard, network equipment and network protocols can communicate efficiently.

### **Fast Ethernet**

For Ethernet networks that need higher transmission speeds, the Fast Ethernet standard (IEEE 802.3u) has been established. This standard raises the Ethernet speed limit from 10 Megabits per second (Mbps) to 100 Mbps with only minimal changes to the existing cable structure. There are three types of Fast Ethernet: 100BASE-TX for use with level 5 UTP cable, 100BASE-FX for use with fibre-optic cable, and 100BASE-T4 which utilizes an extra two wires for use with

level 3 UTP cable. The 100BASE-TX standard has become the most popular due to its close compatibility with the 10BASE-T Ethernet standard. For the network manager, the incorporation of Fast Ethernet into an existing configuration presents a host of decisions. Managers must determine the number of users in each site on the network that need the higher throughput, decide which segments of the backbone need to be reconfigured specifically for 100BASE-T and then choose the necessary hardware to connect the 100BASE-T segments with existing 10BASE-T segments. Gigabit Ethernet is a future technology that promises a migration path beyond Fast Ethernet so the next generation of networks will support even higher data transfer speeds.

### **Token Ring**

Token Ring is another form of network configuration which differs from Ethernet in that all messages are transferred in a unidirectional manner along the ring at all times. Data is transmitted in tokens, which are passed along the ring and viewed by each device.

When a device sees a message addressed to it, that device copies the message and then marks that message as being read. As the message makes its way along the ring, it eventually gets back to the sender who now notes that the message was received by the intended device. The sender can then remove the message and free that token for use by others.



Various PC vendors have been proponents of Token Ring networks at different times and thus these types of networks have been implemented in many organizations.

## **Protocols**

Network protocols are standards that allow computers to communicate. A protocol defines how computers identify one another on a network, the form that the data should take in transit, and how this information is processed once it reaches its final destination.

Protocols also define procedures for handling lost or damaged transmissions or “packets.” TCP/IP (for UNIX, Windows NT, Windows 95 and other platforms), IPX (for Novell NetWare), DECnet (for networking Digital Equipment Corp. computers), AppleTalk (for Macintosh computers), and NetBIOS/NetBEUI (for LAN Manager and Windows NT networks) are the main types of network protocols in use today.

Although each network protocol is different, they all share the same physical cabling. This common method of accessing the physical network allows multiple protocols to peacefully coexist over the network media, and allows the builder of a network to use common hardware for a variety of protocols. This concept is known as “protocol independence,” which means that devices that are compatible at the physical and data link layers allow the user to run many different protocols over the same medium.

## **Media**

An important part of designing and installing an Ethernet is selecting the appropriate Ethernet medium. There are four major types of media in use today: Thickwire for 10BASE5 networks, thin coax for 10BASE2 networks, unshielded twisted pair (UTP) for 10BASE-T networks and fibre optic for 10BASE-FL or Fiber-Optic Inter-Repeater Link (FOIRL) networks. This wide variety of media reflects the evolution of Ethernet and also points to the technology's flexibility. Thickwire was one of the first cabling systems used in Ethernet but was expensive and difficult to use. This evolved to thin coax, which is easier to work with and less expensive.

The most popular wiring schemes are 10BASE-T and 100BASE-TX, which use unshielded twisted pair (UTP) cable. This is similar to telephone cable and comes in a variety of grades, with each higher grade offering better performance. Level 5 cable is the highest, most expensive grade, offering support for transmission rates of up to 100 Mbps. Level 4 and level 3 cable are less expensive, but cannot support the same data throughput speeds; level 4 cable can support speeds of up to 20 Mbps; level 3 up to 16 Mbps. The 100BASE-T4 standard allows for support of 100 Mbps Ethernet over level 3 cable, but at the expense of adding another pair of wires (4 pair instead of the 2 pair used for 10BASE-T); for most users, this is an awkward scheme and therefore 100BASE-T4 has seen little

popularity. Level 2 and level 1 cables are not used in the design of 10BASE-T networks.

For specialized applications, fibre-optic, or 10BASE-FL, Ethernet segments are popular. Fiber-optic cable is more expensive, but it is invaluable for situations where electronic emissions and environmental hazards are a concern. Fiber-optic cable is often used in interbuilding applications to insulate networking equipment from electrical damage caused by lightning. Because it does not conduct electricity, fibre-optic cable can also be useful in areas where large amounts of electromagnetic interference are present, such as on a factory floor. The Ethernet standard allows for fibre-optic cable segments up to 2 kilometres long, making fibre optic Ethernet perfect for connecting nodes and buildings that are otherwise not reachable with copper media.

## **Topologies**

A network topology is the geometric arrangement of nodes and cable links in a LAN, and is used in two general configurations: bus and star. These two topologies define how nodes are connected to one another. A node is an active device connected to the network, such as a computer or a printer. A node can also be a piece of networking equipment such as a hub, switch or a router. A bus topology consists of nodes linked together in a series with each node connected to a long cable or bus. Many nodes can tap into the bus and begin communication with all other nodes on that cable

segment. A break anywhere in the cable will usually cause the entire segment to be inoperable until the break is repaired. Examples of bus topology include 10BASE2 and 10BASE5.

10BASE-T Ethernet and Fast Ethernet use a star topology, in which access is controlled by a central computer. Generally a computer is located at one end of the segment, and the other end is terminated in central location with a hub. Because UTP is often run in conjunction with telephone cabling, this central location can be a telephone closet or other area where it is convenient to connect the UTP segment to a backbone. The primary advantage of this type of network is reliability, for if one of these 'point-to-point' segments has a break, it will only affect the two nodes on that link. Other computer users on the network continue to operate as if that segment were nonexistent.

### **Collisions**

Ethernet is a shared media, so there are rules for sending packets of data to avoid conflicts and protect data integrity. Nodes determine when the network is available for sending packets. It is possible that two nodes at different locations attempt to send data at the same time. When both PCs are transferring a packet to the network at the same time, a collision will result. Minimizing collisions is a crucial element in the design and operation of networks. Increased collisions are often the result of too many users on the network, which results in a lot of contention for network bandwidth. This

can slow the performance of the network from the user's point of view. Segmenting the network, where a network is divided into different pieces joined together logically with a bridge or switch, is one way of reducing an overcrowded network.

### **Ethernet Products**

The standards and technology that have just been discussed help define the specific products that network managers use to build Ethernet networks. The following text discusses the key products needed to build an Ethernet LAN.

#### **Transceivers**

Transceivers are used to connect nodes to the various Ethernet media. Most computers and network interface cards contain a built-in 10BASE-T or 10BASE2 transceiver, allowing them to be connected directly to Ethernet without requiring an external transceiver. Many Ethernet devices provide an AUI connector to allow the user to connect to any media type via an external transceiver. The AUI connector consists of a 15-pin D-shell type connector, female on the computer side, male on the transceiver side. Thickwire (10BASE5) cables also use transceivers to allow connections. For Fast Ethernet networks, a new interface called the MII (Media Independent Interface) was developed to offer a flexible way to support 100 Mbps connections. The MII is a popular way to connect 100BASE-FX links to copper-based Fast Ethernet devices.

## **Network Interface Cards**

Network interface cards, commonly referred to as NICs, are used to connect a PC to a network. The NIC provides a physical connection between the networking cable and the computer's internal bus. Different computers have different bus architectures; PCI bus master slots are most commonly found on 486/Pentium PCs and ISA expansion slots are commonly found on 386 and older PCs. NICs come in three basic varieties: 8-bit, 16-bit, and 32-bit. The larger the number of bits that can be transferred to the NIC, the faster the NIC can transfer data to the network cable. Many NIC adapters comply with Plug-n-Play specifications. On these systems, NICs are automatically configured without user intervention, while on non-Plug-n-Play systems, configuration is done manually through a setup programme and/or DIP switches. Cards are available to support almost all networking standards, including the latest Fast Ethernet environment. Fast Ethernet NICs are often 10/100 capable, and will automatically set to the appropriate speed. Full duplex networking is another option, where a dedicated connection to a switch allows a NIC to operate at twice the speed.

## **Hubs/Repeaters**

Hubs/repeaters are used to connect together two or more Ethernet segments of any media type. In larger designs, signal quality begins to deteriorate as segments exceed their

maximum length. Hubs provide the signal amplification required to allow a segment to be extended a greater distance. A hub takes any incoming signal and repeats it out all ports. Ethernet hubs are necessary in star topologies such as 10BASE-T. A multi-port twisted pair hub allows several point-to-point segments to be joined into one network. One end of the point-to-point link is attached to the hub and the other is attached to the computer. If the hub is attached to a backbone, then all computers at the end of the twisted pair segments can communicate with all the hosts on the backbone.

The number and type of hubs in any one-collision domain is limited by the Ethernet rules. A very important fact to note about hubs is that they only allow users to share Ethernet. A network of hubs/repeaters is termed a “shared Ethernet,” meaning that all members of the network are contending for transmission of data onto a single network (collision domain). This means that individual members of a shared network will only get a percentage of the available network bandwidth. The number and type of hubs in any one collision domain for 10Mbps Ethernet is limited by the following rules:

<b><i>Network Type</i></b>	<b><i>Max Nodes Per Segment</i></b>	<b><i>Max Distance Per Segment</i></b>
10BASE-T	2	100m
10BASE2	30	185m
10BASE5	100	500m
10BASE-FL	2	2000m

---

## **RESEARCH NETWORKING**

---

Research Networking (RN) is about using web-based tools to discover and use research and scholarly information about people and resources. Research Networking tools (RN tools) serve as knowledge management systems for the research enterprise.

RN tools connect institution-level/enterprise systems, national research networks, publicly available research data (*e.g.*, grants and publications), and restricted/proprietary data by harvesting information from disparate sources into compiled expertise profiles for faculty, investigators, scholars, clinicians, community partners, and facilities.

RN tools facilitate the development of new collaborations and team science to address new or existing research challenges through the rapid discovery and recommendation of researchers, expertise, and resources.

RN tools differ from search engines such as Google in that they access information in databases and other data not limited to web pages.

They also differ from social networking systems such as LinkedIn or Facebook in that they represent a compendium of data ingested from authoritative and verifiable sources rather than predominantly individually-asserted information, making RN tools more reliable.

Yet, RN tools have sufficient flexibility to allow for profile editing. RN tools also provide resources to bolster human



connector systems: they can make non-intuitive matches, they do not depend on serendipity, and they do not have a propensity to return only to previously identified collaborations/collaborators. RN tools also generally have associated analytical capabilities that enable evaluation of collaboration and cross-disciplinary research/scholarly activity, especially over time. Importantly, data harvested into robust RN tools is accessible for broad repurposing, especially if available as linked open data (RDF triples). Thus RN tools enhance research support activities by providing data for customized, up-to-date web pages, CV/biosketch generation, and data tables for grant proposals.

---

## **IMPROVE NETWORKING SKILLS**

---

For many of us, networking skills are not something we are born with. You may have to work hard to come out of your shell and make the contacts you need to improve your chances on your career path. With a little practice and the willingness to get out of your “comfort zone,” you can improve your networking skills.

### **START NETWORKING WITH CONFIDENCE**

- Compose a brief introduction for yourself. You may need to compose several versions of this speech, each customized to particular situations. For example, if you are interested in finding a new position or starting a new business, you might write two versions of your

### *Networking Hardware*

speech—one for employment contacts and one for venture capitalists.

- Practice your introduction by standing before a mirror and paying close attention to your posture and stance. You should try to exude confidence by keeping your shoulders back and your eyes up.
- Find a friend or someone you feel comfortable rehearsing your short introduction in front of. This should be someone who is willing to listen and give constructive criticism to help you perfect your introduction or icebreaker.
- Find a networking opportunity that is low key and low pressure at which to unveil your introductory speech. This can be anything from a casual party to an alumni event for your college or university.
- Dress the part. Your appearance makes up a large part of the first impression for a new contact. When networking for your career or business, choose professional attire that is suitable to the occasion and avoid over- or under-dressing.
- Introduce yourself to at least one new person during your first networking event. By overcoming your fear or shyness, you will find the next introduction that much easier.
- Improve the likelihood that the contacts you make will be fruitful by following up. Send a brief e-mail or

### *Networking Hardware*

make a couple of calls, especially if you promised or were promised information or job leads during your initial conversation.

- Keep getting out there. The best way to improve your networking skills is through real-world experience.

# 2

---

## Local Access Network Design

---

### LOCAL AREA NETWORK (LAN)

Is a data communications network connecting terminals, computers and printers within a building or other geographically limited areas. These devices could be connected through wired cables or wireless links.

Ethernet, Token Ring and Wireless LAN using IEEE 802.11 are examples of standard LAN technologies. Ethernet is by far the most commonly used LAN technology. Token Ring technology is still used by some companies. FDDI is sometimes used as a backbone LAN interconnecting Ethernet or Token Ring LANs.

WLAN using IEEE 802.11 technologies is rapidly becoming the new leading LAN technology for its mobility and easy to

use features. Local Area Network could be interconnected using Wide Area Network (WAN) or Metropolitan Area Network (MAN) technologies.

The common WAN technologies include TCP/IP, ATM, Frame Relay etc. The common MAN technologies include SMDS and 10 Gigabit Ethernet. LANs are traditionally used to connect a group of people who are in the same local area. However, the working group are becoming more geographically distributed in today's working environment.

There, virtual LAN (VLAN) technologies are defined for people in different places to share the same networking resource. Local Area Network protocols are mostly at data link layer (layer 2). IEEE is the leading organization defining most of the LAN protocols.

## **LOCAL AREA NETWORK DIAGRAM SOFTWARE**

Edraw Network Diagram is ideal for network engineers and network designers who need to draw wan diagrams. It had defined some common used WAN symbols in drawing WAN diagrams.

## **WIDE AREA NETWORK TECHNOLOGIES**

A Wide Area Network (WAN) is a computer network covering multiple distance areas, which may spread across the entire world. WANs often connect multiple smaller networks, such as local area networks (LANs) or metro area networks (MANs). The world's most popular WAN is the Internet. Some segments

of the Internet are also WANs in themselves. The key difference between WAN and LAN technologies is scalability. C WAN must be able to grow as needed to cover multiple cities, even countries and continents.

A set of switches and routers are interconnected to form a Wide Area Network. The switches can be connected in different topologies such as full mesh and half mesh. A wide area network may be privately owned or rented from a service provider, but the term usually connotes the inclusion of public (shared user) networks.

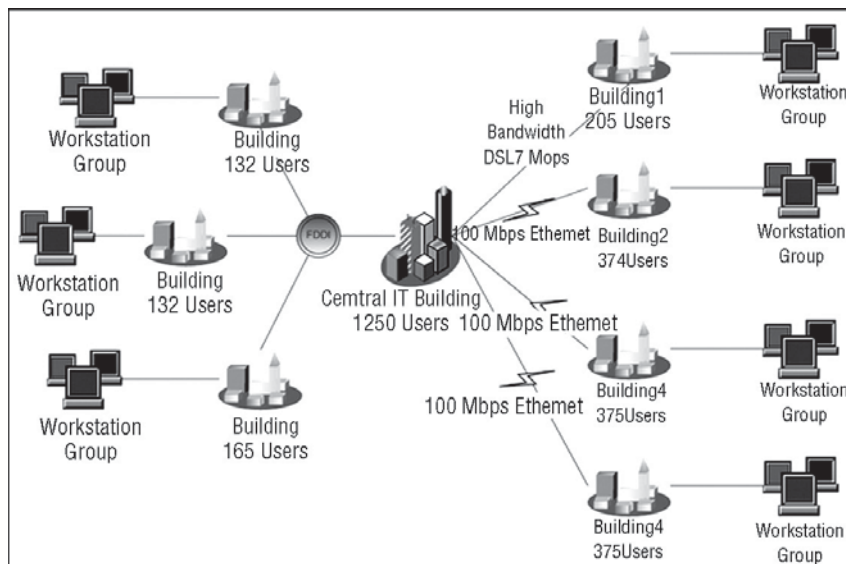
Both packet switching and circuit switching technologies are used in the WAN. Packet switching allows users to share common carrier resources so that the carrier can make more efficient use of its infrastructure. In a packet switching setup, networks have connections into the carrier's network, and many customers share the carrier's network. The carrier can then create virtual circuits between customers' sites by which packets of data are delivered from one to the other through the network.

Circuit Switching allows data connections to be established when needed and then terminated when communication is complete. This works like a normal telephone line works for voice communication. Integrated Services Digital Network (ISDN) is a good example of circuit switching. When a router has data for a remote site, the switched circuit is initiated with the circuit number of the remote network.

## Examples

Virtual private network (VPN) is a technology widely used in a public switched network (PSTN) to provide private and secured WAN for an organization.

VPN uses encryption and other techniques to make it appear that the organisation has a dedicated network, while making use of the shared infrastructure of the WAN.



**Fig.** Wide Area Network

WAN technologies generally function at the lower three layers of the OSI reference model: the physical layer, the data link layer, and the network layer. Key technologies often found in WANs include SONET, Frame Relay, X.25, ATM and PPP.

- *ATM*: A dedicated-connection switching technology that organizes digital data into 53-byte cell units. Individually, a cell is processed asynchronously relative to other related cells and is queued before

being multiplexed over the transmission path. Speeds on ATM networks can reach 10 Gbps.

- *Frame Relay: (FR)*. A high-speed packet-switched data communications service, similar to X.25. Frame relay is widely used for LAN-to-LAN interconnect services, and is well suited to the bursty demands of LAN environments.
- *SONET/SDH*: Synchronous Optical Network is an international standard for high speed communication over fibre-optic networks. The SONET establishes Optical Carrier (OC) levels from 51.8 Mbps to 10 Gbps (OC-192) or even higher. Synchronous Digital Hierarchy (SDH) is a European equivalent of SONET.
- *X.25*: The X.25 protocol allows computers on different public networks to communicate through an intermediary computer at the network layer level.
- *PPP*: A *point-to-point link* provides a single, pre-established WAN communications path from the customer premises through a carrier network, such as a telephone company, to a remote network. Point-to-point lines are usually leased from a carrier and thus are often called leased lines. For a point-to-point line, the carrier allocates pairs of wire and facility hardware to your line only.

IP can also be considered as a WAN technology in the packet switching environment.



## **NETWORK DIAGRAM EXAMPLES**

Seeking a solution for maximizing the efficiencies throughout the network diagram? How to make an network topology?

How indeed does one go about it, without seeing and examples of Network Diagram? Not likely unless one has good Network Diagram examples.

Recommend a new network diagram software similar to Visio, Support flowcharts, organizational charts, business charts, UML diagrams, database and ERD, directional map and network diagrams.

Easy-to-use drawing tools, many pre-drawn flowchart templates, more than 2000 symbols and examples, create network diagrams and business diagrams with minimum time loss.

With Edraw Max, you can create clear and comprehensive network diagrams with no prior experience.

As you can see by studying the examples of network diagram below, these types of diagrams are the ideal way to illustrate the network design idea and network relativity.

## **PHYSICAL LAYER TRANSMISSION**

### **DEFINITION**

The Physical Layer resides immediately below the Data Link Layer. The Physical Layer is responsible for bit-level transmission between network nodes. In copper networks,

the Physical Layer is responsible for defining specifications for electrical signals. In fibre optic networks, the Physical Layer is responsible for defining the characteristics of light signals. The Physical Layer defines items such as: connector types, cable types, voltages, and pin-outs.

The lowest layer of the OSI Reference Model is layer 1, the *physical layer*; it is commonly abbreviated “PHY”. The physical layer is special compared to the other layers of the model, because it is the only one where data is physically moved across the network interface. All of the other layers perform useful functions to create messages to be sent, but they must all be transmitted down the protocol stack to the physical layer, where they are actually sent out over the network. The physical layer is also “special” in that it is the only layer that really does not apply specifically to TCP/IP. Even in studying TCP/IP, however, it is still important to understand its significance and role in relation to the other layers where TCP/IP protocols reside.

## **ROLE**

The name “physical layer” can be a bit problematic. Because of that name, and because of what I just said about the physical layer actually transmitting data, many people who study networking get the impression that the physical layer is only about actual network hardware. Some people may say the physical layer is “the network interface cards and cables”. This is not actually the case, however. The

physical layer defines a number of network functions, not just hardware cables and cards. A related notion is that “all network hardware belongs to the physical layer”. Again, this isn’t strictly accurate. All hardware must have *some* relation to the physical layer in order to send data over the network, but hardware devices generally implement multiple layers of the OSI model, including the physical layer but also others. For example, an Ethernet network interface card performs functions at both the physical layer and the data link layer.

## **FUNCTIONS**

*The following are the main responsibilities of the physical layer in the OSI Reference Model:*

- *Definition of Hardware Specifications:* The details of operation of cables, connectors, wireless radio transceivers, network interface cards and other hardware devices are generally a function of the physical layer (although also partially the data link layer; see below).
- *Encoding and Signaling:* The physical layer is responsible for various encoding and signaling functions that transform the data from bits that reside within a computer or other device into signals that can be sent over the network.
- *Data Transmission and Reception:* After encoding the data appropriately, the physical layer actually transmits the data, and of course, receives it. Note that this

applies equally to wired and wireless networks, even if there is no tangible cable in a wireless network!

- *Topology and Physical Network Design:* The physical layer is also considered the domain of many hardware-related network design issues, such as LAN and WAN topology.

In general, then, physical layer technologies are ones that are at the very lowest level and deal with the actual ones and zeroes that are sent over the network. For example, when considering network interconnection devices, the simplest ones operate at the physical layer: repeaters, conventional hubs and transceivers. These devices have absolutely no knowledge of the contents of a message. They just take input bits and send them as output. Devices like switches and routers operate at higher layers and look at the data they receive as being more than voltage or light pulses that represent one or zero.

### **CAN PHYSICAL LAYER**

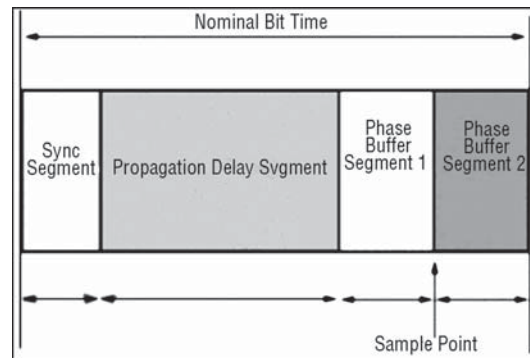
The Controller Area Network (CAN) protocol defines the data link layer and part of the physical layer in the OSI model, which consists of seven layers. The International Standards Organization (ISO) defined a standard, which incorporates the CAN specifications as well as a part of physical layer: the physical signaling, which comprises bit encoding and decoding (Non Return to Zero, NRZ) as well as bit-timing and synchronization.

## **Bit Encoding**

In the chosen Non Return to Zero (NRZ) bit coding the signal level remains constant over the bit time and thus just one time slot is required for the representation of a bit (other methods of bit encoding are e. g. Manchester or Pulse-width-modulation). The signal level can remain constant over a longer period of time; therefore measures must be taken to ensure that the maximum permissible interval between two signal edges is not exceeded. This is important for synchronization purposes. Bit stuffing is applied by inserting a complementary bit after five bits of equal value. Of course the receiver has to un-stuff the stuff-bits so that the original data content is processed.

## **Bit-timing and Synchronization**

On the bit-level (OSI layer 1, physical layer) CAN uses synchronous bit transmission. This enhances the transmitting capacity but also means that a sophisticated method of bit synchronization is required. While bit synchronization in a character-oriented transmission (asynchronous) is performed upon the reception of the start bit available with each character, a synchronous transmission protocol there is just one start bit available at the beginning of a frame. To enable the receiver to correctly read the messages, continuous resynchronization is required. Phase buffer segments are therefore inserted before and after the nominal sample point within a bit interval.



**Fig.** Nominal Bit-time

The CAN protocol regulates bus access by bit-wise arbitration. The signal propagation from sender to receiver and back to the sender must be completed within one bit-time. For synchronization purposes a further time segment, the propagation delay segment, is needed in addition to the time reserved for synchronization, the phase buffer segments. The propagation delay segment takes into account the signal propagation on the bus as well as signal delays caused by transmitting and receiving nodes.

Two types of synchronization are distinguished: hard synchronization at the start of a frame and resynchronization within a frame.

- After a hard synchronization the bit time is restarted at the end of the sync segment. Therefore the edge, which caused the hard synchronization, lies within the sync segment of the restarted bit time.
- Resynchronization shortens or lengthens the bit time so that the sample point is shifted according to the detected edge

The device designer may program the bit-timing parameters in the CAN controller by means of the appropriate registers.

### **Interdependency of Data Rate and Bus Length**

Depending on the size of the propagation delay segment the maximum possible bus length at a specific data rate (or the maximum possible data rate at a specific bus length) can be determined.

The signal propagation is determined by the two nodes within the system that are farthest apart from each other. It is the time that it takes a signal to travel from one node to the one farthest apart (taking into account the delay caused by the transmitting and receiving node), synchronization and the signal from the second node to travel back to the first one. Only then can the first node decide whether its own signal level (recessive in this case) is the actual level on the bus or whether it has been replaced by the dominant level by another node. This fact is important for bus arbitration.

Some modern transceivers support no low data rates. Therefore on acquisition of transceivers the maximal required network length must be considered.

### **PHYSICAL MEDIA**

*This clause is most interesting for system designers:* The basis for transmitting CAN messages and for competing for bus access is the ability to represent a dominant and a recessive bit value. This is possible for electrical and optical

media so far. Also powerline and wireless transmission is possible. The physical media most commonly used to implement CAN networks is a differentially driven pair of wires with common return. For vehicle body electronics single wire bus lines are also used. Some efforts have been made to develop a solution for the transmission of CAN signals on the same line as the power supply.

The parameters of the electrical medium become important when the bus length is increased. Signal propagation, the line resistance and wire cross sections are factors when dimensioning a network. In order to achieve the highest possible bit rate at a given length, a high signal speed is required. For long bus lines the voltage drops over the length of the bus line. The wire cross section necessary is calculated by the permissible voltage drop of the signal level between the two nodes farthest apart in the system and the overall input resistance of all connected receivers. The permissible voltage drop must be such that the signal level can be reliably interpreted at any receiving node. The consideration of electromagnetic compatibility and choice of cables and connectors belongs also to the tasks of a system integrator.

**Assumed Line Length 100 m**

<b>Specific signal propagation time (ns/m)</b>	<b>Maximum bit rate (kbit/s)</b>
5.0	80
5.5	73
6.0	67
6.5	62
7.0	58



## **Network Topology**

This clause is most interesting for system designers. Electrical signals on the bus are reflected at the ends of the electrical line unless measures against that have been taken. For the node to read the bus level correctly it is important that signal reflections are avoided. This is done by terminating the bus line with a termination resistor at both ends of the bus and by avoiding unnecessarily long stubs lines of the bus. The highest possible product of transmission rate and bus length line is achieved by keeping as close as possible to a single line structure and by terminating both ends of the line. Specific recommendations for this can be found in the according standards.

It is possible to overcome the limitations of the basic line topology by using repeaters, bridges or gateways. A repeater transfers an electrical signal from one physical bus segment to another segment. The signal is only refreshed and the repeater can be regarded as a passive component comparable to a cable. The repeater divides a bus into two physically independent segments. This causes an additional signal propagation time. However, it is logically just one bus system.

A bridge connects two logically separated networks on the data link layer (OSI layer 2). This is so that the CAN identifiers are unique in each of the two bus systems. Bridges implement a storage function and can forward messages or parts thereof in an independent time-delayed transmission. Bridges differ

from repeaters since they forward messages, which are not local, while repeaters forward all electrical signals including the CAN identifier.

A gateway provides the connection of networks with different higher-layer protocols. It therefore performs the translation of protocol data between two communication systems. This translation takes place on the application layer (OSI layer 7).

### **Bus Access**

The connection between a CAN controller chip and a two-wire differential bus a variety of CAN transceiver chips according to different physical layer standards are available.

This interface basically consists of a transmitting amplifier and a receiving amplifier (transceiver = transmit and receive). Aside from the adaptation of the signal representation between chip and bus medium the transceiver has to meet a series of additional requirements. As a transmitter it provides sufficient driver output capacity and protects the on-controller-chip driver against overloading. It also reduces electromagnetic radiation.

As a receiver the CAN transceiver provides a defined recessive signal level and protects the on-controller-chip input comparator against over-voltages on the bus lines. It also extends the common mode range of the input comparator in the CAN controller and provides sufficient input sensitivity. Furthermore it detects bus errors such as line breakage,

short circuits, shorts to ground, etc. A further function of the transceiver can also be the galvanic isolation of a CAN node and the bus line.

---

## **PHYSICAL LAYER STANDARDS**

---

### **ISO 11898-2 HIGH SPEED**

ISO 11898-2 is the most used physical layer standard for CAN networks. It describes the bus access unit (implemented as CAN high-speed transceiver) functions as well as some medium-dependent interface features.

In this standard the data rate is defined up to 1 Mbit/s with a theoretically possible bus length of 40 m at 1 Mbit/s. The high-speed standard specifies a two-wire differential bus whereby the number of nodes is limited by the electrical busload. The characteristic line impedance is 120 Ohm, the common mode voltage ranges from -2 V on CAN\_L to +7 V on CAN\_H. The nominal specific propagation delay of the two-wire bus line is specified at 5 ns/m. In order to achieve physical compatibility all nodes in the network must use the same or a similar bit-timing. For automotive applications the SAE published the SAE J2284 specification. For industrial and other non-automotive applications the system designer may use the CiA 102 recommendation. This specification defines the bit-timing for rates of 10 kbit/s to 1 Mbit/s. It also provides recommendations for bus lines and for connectors and pin assignment.

### **ISO 11898-3 Fault-tolerant**

An alternative form of bus interfacing and arrangement of bus lines is specified in ISO 11898-3 (fault-tolerant CAN). This standard is mainly used for body electronics in the automotive industry. Since for this specification a short network was assumed, the problem of signal reflection is not as important as for long bus lines. This makes the use of an open bus line possible.

This means low bus drivers can be used for networks with very low power consumption and the bus topology is no longer limited to a linear structure. It is possible to transmit data asymmetrically over just one bus line in case of an electrical failure of one of the bus lines.

ISO 11898-3 defines data rates up to 125 kbit/s with the maximum bus length depending on the data rate used and the busload. Up to 32 nodes per network are specified. The common mode voltage ranges between  $-2\text{ V}$  and  $+7\text{ V}$ . The power supply is defined at  $5\text{ V}$ .

Transceiver chips, which support this standard, are available from several companies. The fault-tolerant transceivers support the complete error management including the detection of bus errors and automatic switching to asymmetrical signal transmission.

# 3

---

## Network Protocols

---

A protocol is a set of rules that governs the communications between computers on a network. In order for two computers to talk to each other, they must be speaking the same language. Many different types of network protocols and standards are required to ensure that your computer (no matter which operating system, network card, or application you are using) can communicate with another computer located on the next desk or half-way around the world. The OSI (Open Systems Interconnection) Reference Model defines seven layers of networking protocols.

The complexity of these layers is beyond the scope of this tutorial; however, they can be simplified into four layers to help identify some of the protocols with which you should be familiar.

**Table. OSI model related to common network protocols**

<b>OSI Layer</b>	<b>Name</b>	<b>Common Protocols</b>
7 Telnet	Application	HTTP   FTP   SMTP   DNS
6	Presentation	
5	Session	
4	Transport	TCP   SPX
3	Network	IP   IPX
2	Data Link	Ethernet
1	Physical	

Some of the major protocols would correlate to the OSI model in order to communicate via the Internet. In this model, there are four layers, including:

- Ethernet (Physical/Data Link Layers)
- IP/IPX (Network Layer)
- TCP/SPX (Transport Layer)
- HTTP, FTP, Telnet, SMTP, and DNS (combined Session/Presentation/Application Layers).

Assuming you want to send an e-mail message to someone in Italy, we will examine the layers “from the bottom up” — beginning with Ethernet (physical/data link layers).

## **ARP CACHE**

There are two types of ARP entries—static and dynamic. Most of the time, you will use dynamic ARP entries. What this means is that the ARP entry (the Ethernet MAC to IP address link) is kept on a device for some period of time, as long as it is being used. The opposite of a dynamic ARP entry is static ARP entry. With a static ARP entry, you are manually entering the link between the Ethernet MAC address and

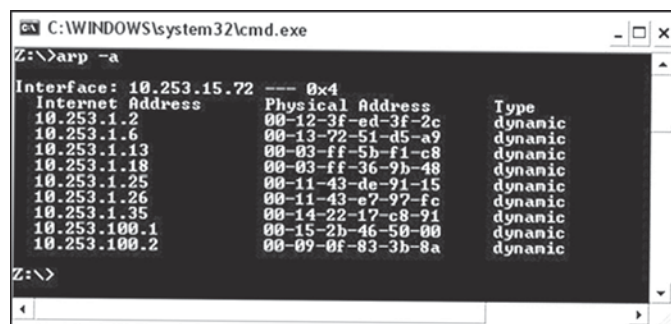
## Networking Hardware

the IP address. Because of management headaches and the lack of significant negatives to using dynamic ARP entries, dynamic ARP entries are used most of the time.

So how is the dynamic ARP entry created? The answer is that the ARP protocol is used. Let's say that a PC wants to communicate with host Myserver.Bluecrabfood.com. Before it can do that, it has to first resolve the hostname with the DNS server. Let's say that it is successfully resolved to 10.10.10.10. Before the PC can communicate with that IP address, it must first resolve the IP address to the MAC address.

To do this, it does an ARP request. This is a broadcast to the local LAN that says who has IP address 10.10.10.10 and what is your Ethernet MAC address? Say that server responds and says I have IP address 10.10.10.10 and my MAC address is 1234.4567.890A.

The PC will put that entry into its local ARP cache and it will stay there until the entry has not been used and the ARP cache timeout has expired. Here is an ARP cache looks like on a Windows PC:



```
C:\WINDOWS\system32\cmd.exe
Z:\>arp -a

Interface: 10.253.15.72 --- 0x4
Internet Address      Physical Address      Type
10.253.1.2            00-12-3f-ed-3f-2c     dynamic
10.253.1.6            00-13-72-51-d5-a9     dynamic
10.253.1.13           00-03-ff-5b-f1-c8     dynamic
10.253.1.18           00-03-ff-36-9b-48     dynamic
10.253.1.25           00-11-43-de-91-15     dynamic
10.253.1.26           00-11-43-e7-97-fc     dynamic
10.253.1.35           00-14-22-17-c8-91     dynamic
10.253.100.1          00-15-2b-46-50-00     dynamic
10.253.100.2          00-09-0f-83-3b-8a     dynamic

Z:\>
```

## Networking Hardware

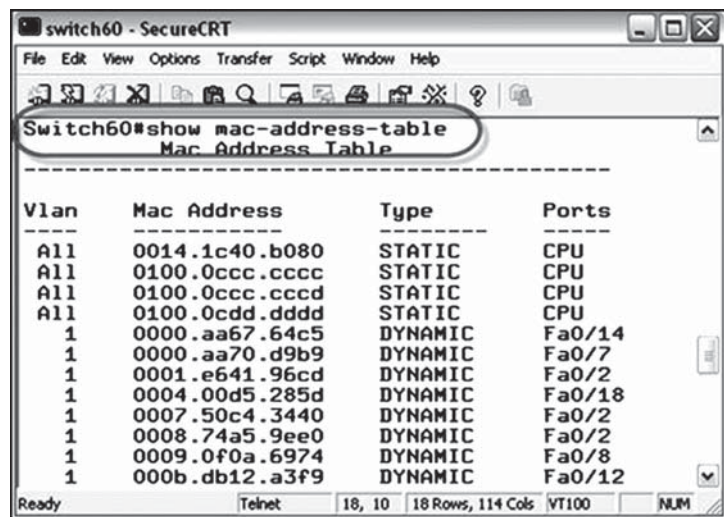
If a router is communicating with a device, it will have its own ARP cache. Here is an example of the show arp command on a Cisco IOS router:



```
com port 1 - SecureCRT
Router#sh arp
Protocol Address      Age (min) Hardware Addr  Type   Interface
Internet 1.1.1.1          -         0003.e39b.9220 ARPA    Ethernet0/0
Internet 1.1.1.2          0         Incomplete     ARPA
Router#
```

In this example, you can see how IP address 1.1.1.1 is mapped to Ethernet MAC address 0003.e39b.9220. Notice the Incomplete entry; this is a sign of trouble.

Switches will have their own ARP cache but they will also keep track of which MAC address is connected to which port on the switch. This can be seen with the show mac-address-table commands on a Cisco IOS switch:



```
switch60 - SecureCRT
Switch60#show mac-address-table
Mac Address Table
-----
Vlan    Mac Address      Type        Ports
-----
All     0014.1c40.b080   STATIC      CPU
All     0100.0ccc.cccc   STATIC      CPU
All     0100.0ccc.cccc   STATIC      CPU
All     0100.0cdd.dddd   STATIC      CPU
1       0000.aa67.64c5   DYNAMIC     Fa0/14
1       0000.aa70.d9b9   DYNAMIC     Fa0/7
1       0001.e641.96cd   DYNAMIC     Fa0/2
1       0004.00d5.285d   DYNAMIC     Fa0/18
1       0007.50c4.3440   DYNAMIC     Fa0/2
1       0008.74a5.9ee0   DYNAMIC     Fa0/2
1       0009.0f0a.6974   DYNAMIC     Fa0/8
1       000b.db12.a3f9   DYNAMIC     Fa0/12
```

## PROXY ARP

Proxy ARP is the name given when a node responds to an arp request on behalf of another node.



This is commonly used to redirect traffic sent to one IP address to another system. Proxy ARP can also be used to subvert traffic away from the intended recipient. By responding instead of the intended recipient, a node can pretend to be a different node in a network, and therefore force traffic directed to the node to be redirected to itself. The node can then view the traffic (*e.g.* before forwarding this to the originally intended node) or could modify the traffic. Improper use of Proxy ARP is therefore a significant security vulnerability and some networks therefore implement systems to detect this. Gratuitous ARP can also help defend the correct IP to MAC bindings.

Modern IP hosts, such as workstations and PCs, transmit directly to either a destination host or router. If the destination is on the same IP network and subnetwork as the sender's, the sender transmits an ARP request to determine the destination MAC address and then transmits directly to it over the LAN.

If the destination's net/subnet is not the same as the sender's, the sender transmits the packet to a router. Hosts are usually configured manually with a default router, which is the IP address of a router on their LAN. Older hosts may always attempt to ARP for a destination address, even if it is not on the local LAN. The older host expects the router to respond to the ARP request with the router's MAC address. This is called

## **Hosts With No Subnet Support**

If the host attempts to send a packet to a network subnet, it sends an ARP request to find the MAC address of the destination host. If the subnet is not on the local wire, a router configured for ARP subnet routing may respond to the ARP request with its own MAC address if the following conditions exist:

- The router has the location of the subnet in its routing table.
- The router sends packets to that subnet via a different interface than the interface that received the ARP request.

Because of the second condition, configure all routers on a local wire for ARP subnet routing when you use hosts without network subnet support.

## **Proxy ARP Request Example**

The following list describes the sequence when a station requiring Proxy ARP wants to send an IP packet to a host on a remote network:

- The host issues an ARP request that contains the destination IP address.
- Any router enabled to respond looks at the IP address for a match in its routing table.
- If there is a match and the route does not pass back through the same LAN port where the ARP host resides, the router responds with an ARP response

supplying its MAC address. Finding a match without passing back through the ARP host port implies another router is present, has a shorter path to the destination, and replies to the ARP itself.

- The host then sends the packet to the router using the newly learned MAC address.
- The host stores this information (that is, the mapping of the IP address to the MAC address) in a local cache so that if it sends another packet to the same destination, it can do so without sending an ARP Request.
- The information is not used. The information is aged out of the cache and may be relearned by resending an ARP Request.

### **Caution When Using Proxy ARP**

The use of proxy ARP is discouraged in modern IP operation. Few hosts require it.

### **PROXY SUBNET ARP**

Proxy Subnet ARP is the same as Proxy ARP except that the router responds to ARP requests for hosts it knows are on other subnets remote from the local subnetwork. Sometimes hosts forward to a router for destinations with different class A, B, or C addresses, but ARP for any destination with the same class A, B, or C address as their own. They do not know about subnets of the class A, B, or C

addresses. They expect the router to respond to the ARP for all subnets of the local class A, B, and C net and to forward to the proper subnet.

### **Proxy Subnet ARP Example**

The following example shows that a host functioning with ARP does not use subnetting (*i.e.*, subnetting is not configured or software does not include subnetting). Unless the router is enabled to respond using Proxy ARP subnet, it does not respond to this ARP and denies connectivity to other subnets of the same IP network.

### **Example Addressing Description**

A single IP class B network number 128.12.0.0 is used to define two subnetworks connected by a router: 128.12.1.0 and 128.12.2.0 (mask 255.255.255.0). The host is on 128.12.1.0 and is attempting to send to 128.12.2.1.

If the host used subnetting, then it sends a packet to its default router and relies on the router to get the packet delivered to the destination 128.12.2.0.

If the host does not use subnetting then it sees the IP network address as 128.12.0.0 (it only knows IP network addresses and therefore uses a class B mask of 255.255.0.0 to obtain 128.12.0.0) and calculates that the destination is on the local LAN (because it has the same network number as itself). It therefore ARPs for the 128.12.2.1 address.

The router must enable Proxy Subnet ARP in order to respond with the router's MAC address. It sends a packet to its default router and relies on the router to get the packet delivered to the destination 128.12.2.0. The host does not use subnetting. It sees the IP network address as 128.12.0.0 (it only knows IP network addresses and therefore uses a class B mask of 255.255.0.0 to obtain 128.12.0.0) and calculates that the destination is on the local LAN (because it has the same network number as itself). It therefore ARPs for the 128.12.2.1 address. The router must enable Proxy Subnet ARP in order to respond with the router's MAC address.

### **Inverse ARP Description**

Inverse ARP is a protocol which allows a device to automatically determine the IP Address of a remote device in a Frame Relay network.

### **Duplicate IP Address Detection**

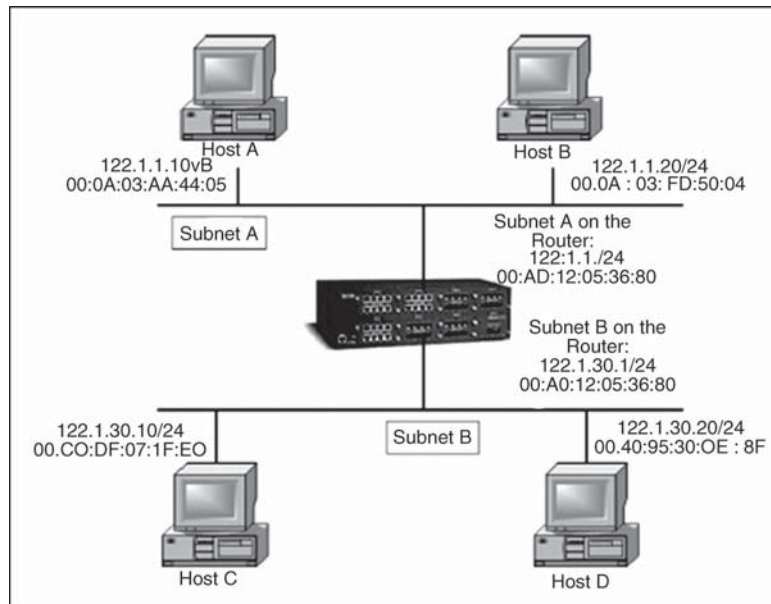
#### **Duplicate IP Address Detection Defined**

Duplicate IP Address Detection is used to detect if the same IP address has been configured on multiple IP devices on the same LAN. If a user configures interface with the same IP address as another device on the same LAN, the network will not work properly. Both devices could receive and respond to packets with that common IP address.

**Note:** Duplicate IP Address Detection cannot detect all the address duplication problems. There is not a central database

to hold all the IP address configurations of a full network. Only unicast addresses are checked.

## RARP



RARP (Reverse Address Resolution Protocol) is a protocol by which a physical machine in a local area network can request to learn its IP address from a gateway server's Address Resolution Protocol (ARP) table or cache. A network administrator creates a table in a local area network's gateway router that maps the physical machine (or Media Access Control-MAC address) addresses to corresponding Internet Protocol addresses. When a new machine is set up, its RARP client program requests from the RARP server on the router to be sent its IP address. Assuming that an entry has been set up in the router table, the RARP server will return the IP address to the machine which can store it for future use.

A reverse address resolution protocol (RARP) is used for diskless computers to determine their IP address using the network. The RARP message format is very similar to the ARP format. When the booting computer sends the broadcast ARP request, it places its own hardware address in both the sending and receiving fields in the encapsulated ARP data packet. The RARP server will fill in the correct sending and receiving IP addresses in its response to the message. This way, the booting computer will know its IP address when it gets the message from the RARP server.

RARP request packet is usually generated during the booting sequence of a host. A host must determine its IP address during the booting sequence. The IP address is needed to communicate with other hosts in the network. When a RARP server receives a RARP request packet, it performs the following steps:

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

When a host receives a RARP reply packet, it gets its IP address from the packet and completes the booting process. This IP address is used for communicating with other hosts,

till it is rebooted. The length of a RARP request or a RARP reply packet is 28 bytes.

The 'operation' field in the RARP packet is used to differentiate between a RARP request and a RARP reply packet. In an RARP request packet, the source and destination IP address values are undefined. In a RARP reply packet, the source IP address is the IP address of the RARP server responding to the RARP request and the destination IP address is the IP address of the host that sent the said RARP request.

Since a RARP request packet is a broadcast packet, it is received by all the hosts in the network. But only a RARP server processes a RARP request packet, all the other hosts discard the packet. The RARP reply packet is not broadcast, it is sent directly to the host, which sent the RARP request. If more than one RARP server responds to a RARP request, then only the first RARP reply received is used. All other replies are discarded. If a RARP reply is not received within a reasonable amount of time, the host, which sent the RARP request, will not be able to complete its booting sequence. Usually, the host will again retry sending the RARP request after a timeout period.

The BOOTP and DHCP protocols can be used instead of RARP to get the IP address from the MAC address.

### **DIFFERENCE BETWEEN ARP AND RARP**

The address resolution protocol (ARP) is used to associate the 32 bit IP address with the 48 bit physical address, used



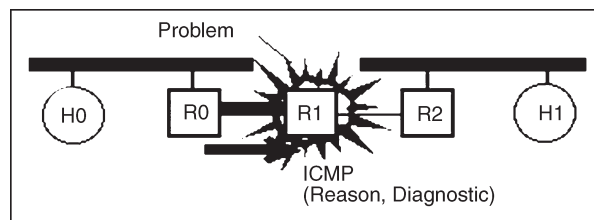
by a host or a router to find the physical address of another host on its network by sending a ARP query packet that includes the IP address of the receiver. The reverse address resolution protocol (RARP) allows a host to discover its Internet address when it knows only its physical address.

---

## **INTERNET MESSAGE CONTROL PROTOCOL (ICMP)**

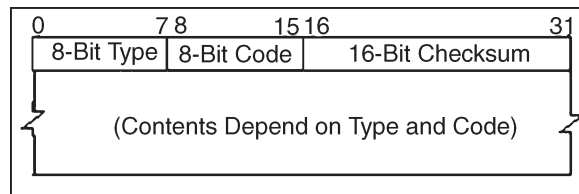
---

The Internet Control Message Protocol (ICMP) [RFC792] protocol is classic example of a client server application. The ICMP server executes on all IP end system computers and all IP intermediate systems (*i.e.* routers). The protocol is used to report problems with delivery of IP datagrams within an IP network. It can be used to show when a particular End System (ES) is not responding, when an IP network is not reachable, when a node is overloaded, when an error occurs in the IP header information, etc. The protocol is also frequently used by Internet managers to verify correct operations of End Systems (ES) and to check that routers are correctly routing packets to the specified destination address.



ICMP messages generated by router R1, in response to message sent by H0 to H1 and forwarded by R0. This message

could, for instance be generated if the MTU of the link between R0 and R1 was smaller than size of the IP packet, and the packet had the Don't Fragment (DF) bit set in the IP packet header. The ICMP message is returned to H0, since this is the source address specified in the IP packet that suffered the problem. A modern version of Path MTU Discovery provides a mechanism to verify the Path MTU [RFC4821].



**Fig.** An ICMP Message Consisting of 4 Bytes of PCI and an Optional Message Payload.

The format of an ICMP message. The 8-bit type code identifies the types of message. This is followed by at least the first 28 bytes of the packet that resulted in generation of the error message (*i.e.* the network-layer header and first 8 bytes of transport header). This payload is, for instance used by a sender that receives the ICMP message to perform Path MTU Discovery so that it may determine IP destination address of the packet that resulted in the error. Longer payloads are also encouraged (which can help better identify the reason why the ICMP message was generated and which program generated the original packet).

The encapsulation of ICMP over an Ethernet LAN using an IP network layer header, and a MAC link layer header and trailer containing the 32-bit checksum:



**Fig.** Encapsulation for a Complete ICMP Packet.

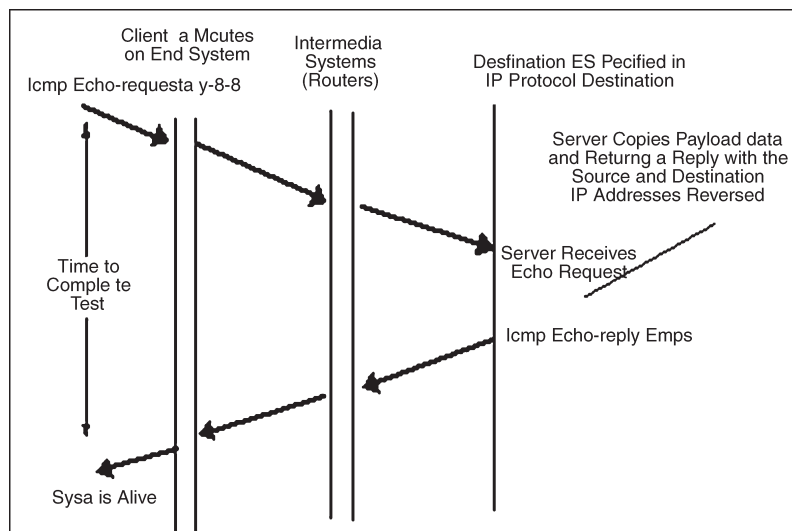
It is the responsibility of the network layer (IP) protocol to ensure that the ICMP message is sent to the correct destination. This is achieved by setting the destination address of the IP packet carrying the ICMP message. The source address is set to the address of the computer that generated the IP packet (carried in the IP source address field) and the IP protocol type is set to “ICMP” to indicate that the packet is to be handled by the remote end system’s ICMP client interface.

RFC792 specifies the Internet Control Message Protocol (ICMP) that is used with the Internet Protocol version 4 (IPv4). It defines, among other things, a number of error messages that can be used by an end-system and intermediate systems to report errors back to the sending system. The host requirements [RFC1122] classifies ICMP these error messages into those that indicate “soft errors” (advising of problems), and those that indicate “hard errors” (which need to be responded to).

A version of ICMP has also been defined for IPv6, called ICMPv6 [RFC4443]. This subsumes all the equivalent functions of ICMP for IPv4 and adds other network-layer functions. ICMP error messages are up to 1280 bytes in size, and therefore always carry a substantial number of bytes from the packet that generated the error being reported.

## THE PING APPLICATION

The “ping” program contains a client interface to ICMP. It may be used by a user to verify an end-to-end Internet Path is operational. The ping program also collects performance statistics (*i.e.* the measured round trip time and the number of times the remote server fails to reply. Each time an ICMP echo reply message is received, the ping program displays a single line of text. The text printed by ping shows the received sequence number, and the measured round trip time (in milliseconds). Each ICMP Echo message contains a sequence number (starting at 0) that is incremented after each transmission, and a timestamp value indicating the transmission time.



**Fig.** Use of the Ping Program to Test Whether a Particular Computer (“Sysa”) is Operational.

The operation of ICMP is illustrated in the frame transition. In this case there is only one Intermediate System (IS) (*i.e.* IP

router). In this case two types of message are involved the ECHO request (sent by the client) and the ECHO reply (the response by the server). Each message may contain some optional data. When data are sent by a server, the server returns the data in the reply which is generated. ICMP packets are encapsulated in IP for transmission across an internet.

### **THE TRACEROUTE APPLICATION**

The “traceroute” program contains a client interface to ICMP. Like the “ping” program, it may be used by a user to verify an end-to-end Internet Path is operational, but also provides information on each of the Intermediate Systems (*i.e.* IP routers) to be found along the IP Path from the sender to the receiver. Traceroute uses ICMP echo messages. These are addressed to the target IP address. The sender manipulates the TTL (hop count) value at the IP layer to force each hop in turn to return an error message.

- The program starts by sending an ICMP Echo request message with an IP destination address of the system to be tested and with a Time To Live (TTL) value set to 1. The first router that receives this packet decrements the TTL and discards the message, since this now has a value of zero. Before it deletes the message, the system constructs an ICMP error message (with an ICMP message type of “TTL exceeded”) and returns this back to the sender.

Receipt of this message allows the sender to identify which system is one link away along the path to the specified destination.

- The sender repeats this two more times, each time reporting the system that received the packet. If all packets travel along the same path, each ICMP error message will be received from the same system. Where two or more alternate paths are being used, the results may vary.
- If the system that responded was not the intended destination, the sender repeats the process by sending a set of three identical messages, but using a TTL value that is one larger than the previous attempt. The first system forwards the packet (decrementing the TTL value in the IP header), but a subsequent system that reduces the TTL value to zero, generates an ICMP error message with its own source address. In this way, the sender learns the identity of another system along the IP path to the destination.
- This process repeats until the sender receives a response from the intended destination (or the maximum TTL value is reached).

Note: Some Routers are configured to discard ICMP messages, while others process them but do not return ICMP Error Messages. Such routers hide the “topology” of the network, but also can impact correct operation of protocols.

Some routers will process the ICMP Messages, providing that they do not impose a significant load on the routers, such routers do not always respond to ICMP messages. When “traceroute” encounters a router that does not respond, it prints a “\*” character.

An example:

```
>traceroute bbc.co.uk traceroute to bbc.co.uk
(212.58.224.131), 64 hops max, 40 byte packets
 1 10.10.10.1 (10.10.10.1) 51.940 ms 18.491 ms 1.260 ms
 2 lo0-plusnet.ptn-ag2.plus.net (195.166.128.53) 49.263
ms 55.061 ms 53.525 ms
 3 ge1-0-0-204.ptn-gw2.plus.net (84.92.3.106) 139.647 ms
52.525 ms 127.196 ms
 4 gil-1-22.ptn-gw5.plus.net (212.159.4.6) 76.505 ms
57.524 ms 52.404 ms
 5 rt0.thdo.bbc.co.uk (212.58.239.25) 89.200 ms 49.666
ms 144.629 ms
 6 212.58.238.133 (212.58.238.133) 48.786 ms 68.650 ms
51.599 ms
```

### **ICMP Type Numbers**

The Internet Protocol [IP] is not designed to be absolutely reliable. The purpose of these control messages [ICMP] is to provide feedback about problems in the communication environment, not to make IP reliable. There are still no guarantees that a datagram will be delivered or a control message will be returned. Some datagrams may still be

undelivered without any report of their loss. The higher level protocols that use IP (the transport layer, if TCP, or the application, if UDP) must implement their own reliability procedures if reliable communication is required.

The ICMP messages typically report errors in the processing of datagrams. To avoid the infinite regress of messages about messages etc., no ICMP messages are sent about ICMP messages. Also ICMP messages are only sent about errors in handling fragment zero of fragmented datagrams. (Fragment zero has the fragment offset equal zero).

The Internet Control Message Protocol (ICMP) has many messages that are identified by a “type” field, these are defined by RFCs.

Many of the types of ICMP message are now obsolete and are no longer seen in the Internet. Some important ones which are widely used include:

Echo Reply (0), Echo Request (8), Redirect (5), Destination Unreachable (3), Traceroute (30), Time Exceeded (11).

A list is shown below (the full list may be retrieved from the link at the base of this page):

<b>Type</b>	<b>Name</b>	<b>Reference</b>
0	Echo Reply	[RFC792]
1	Unassigned	[JBP]
2	Unassigned	[JBP]
3	Destination Unreachable	[RFC792]
4	Source Quench	[RFC792]
5	Redirect	[RFC792]
6	Alternate Host Address	[JBP]
7	Unassigned	[JBP]
8	Echo	[RFC792]



## Networking Hardware

9	Router Advertisement	[RFC1256]
10	Router Selection	[RFC1256]
11	Time Exceeded	[RFC792]
12	Parameter Problem	[RFC792]
13	Timestamp	[RFC792]
14	Timestamp Reply	[RFC792]
15	Information Request	[RFC792]
16	Information Reply	[RFC792]
17	Address Mask Request	[RFC950]
18	Address Mask Reply	[RFC950]
19	Reserved (for Security)	[Solo]
20-29	Reserved (for Robustness Experiment)	[ZSu]
30	Traceroute	[RFC1393]
31	Datagram Conversion Error	[RFC1475]
32	Mobile Host Redirect	[David Johnson]
33	IPv6 Where-Are-You	[Bill Simpson]
34	IPv6 I-Am-Here	[Bill Simpson]
35	Mobile Registration Request	[Bill Simpson]
36	Mobile Registration Reply	[Bill Simpson]
37	Domain Name Request	[RFC1788]
38	Domain Name Reply	[RFC1788]
39	SKIP	[Markson]
40	Photuris	[RFC2521]

---

## **PATH MTU DISCOVERY (PMTUD)**

The IP MTU is the largest size of IP datagram which may be transferred using a specific data link connection. The MTU value is a design parameter of a LAN and is a mutually agreed value (*i.e.* both ends of a link agree to use the same specific value) for most wide area network links. The size of MTU may vary greatly between different links. Note, people who design lower-layer networks (below IP), often define the MTU in a different way to the IP-oriented people. This can catch you out, if you are not careful.

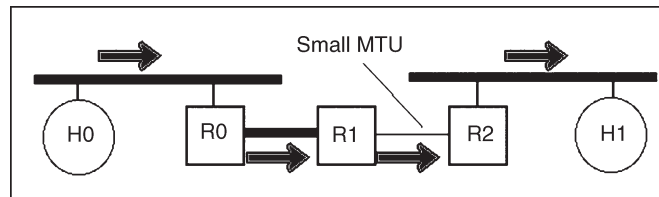
IP fragmentation could be used to break larger packets into a sequence of smaller packets for transmission across

the network. This however, lowers the efficiency and reliability of Internet communication. The loss of a single fragment results in the loss of an entire fragmented packet, because even if all other fragments are received correctly, the original packet cannot be reassembled and delivered and has to be resent. This is one example where router IP fragmentation can be considered harmful. In addition, some Network Address Translators (NATs) and firewalls drop IP fragments. Often network address translation performed by a NATs only operates on complete IP packets. Some firewall policies also require inspection of a complete IP packet. Even with these being the case, some NATs and firewalls simply do not implement the necessary reassembly functionality, and instead choose to drop all fragments. These fundamental issues with fragmentation exist for both IPv4 and IPv6.

Instead of making routers fragment packets, an end system could try to find out the largest IP packet that may be sent to a specific destination. The Path MTU Discovery algorithm operates at the sender at the boundary of the Transport Layer and Network Layers, generating probe messages and responding to ICMP error reports that indicate a low MTU. In Intermediate Systems (IS), *i.e.* routers, this operates in the Network Layer, returning ICMP error messages based on the link-layer configuration.

The way in which an end system finds out this large packet size associated with a specific path (*i.e.* on the series of links

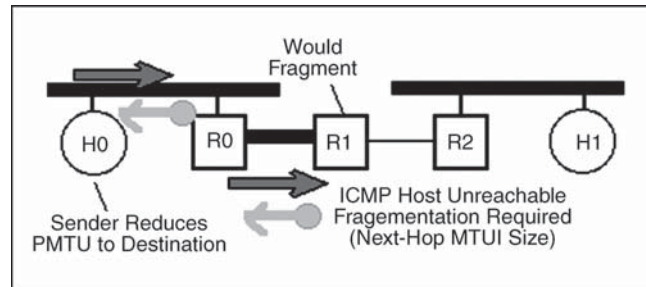
used to reach a destination), is to send a large IP packet (up to the MTU of the link to which it is connected). The packet is sent with the Don't Fragment (DF) flag set in the IP protocol header.



### ICMP-BASED PMTUD

The simplest form of PMTUD relies on the ICMP server that executes on all IP end system computers and all IP intermediate systems (*i.e.* routers). If a router receives a packet that is less than or equal to the MTU of the link that it wishes to forward the packet to, then it sends the packet. If however, a router finds that the MTU of the next link exceeds the packet size and the DF flag is set, this tells the router not to segment the packet. Instead, the router will discard the packet. An Internet Control Message Protocol (ICMP) message is returned by the router back to the sender (H0), with a code saying the packet has been discarded, but importantly, this message also says the reason (*i.e.* the fragmentation would have been required) and indicates the maximum MTU allowed (in this case the MTU of the link between R1 and R2).

Occasionally the end system will generate a large packet, just to see if a new Internet path has been found (*i.e.* a different route). The new path may allow a larger P-MTU.



If an IPv4 end system receives an ICMP message saying a packet is too large (type = 3, code = 4), it sets a variable called the Path-MTU (P-MTU) to the appropriate maximum size and then itself fragments the packet to make sure it will not be discarded next time. The end system keeps (caches) a set of P-MTU values for each IP address in use.

When there are a series of links along the path, each with increasingly smaller MTUs, the above process may take place a number of times, before the sender finally determines the minimum value of the P-MTU. Once the P-MTU has been found, all packets are sent segmented to this new value.

In this model, Routers do not therefore have to do any additional processing for these packets. This is much more efficient than router fragmentation.

However, practical problems are being experienced within the current Internet, caused by systems (*e.g.* some firewalls) that do not return the required ICMP messages back to the sender. The widely deployed version of PMTUD relies on messages received from the ICMP protocol, specified by RFC1191.

## **PMTUD WITHOUT RELYING ON ICMP**

There are issues with deploying PMTUD in practical networks, you may need to think harder, and the best method is to avoid having to receive ICMP messages (RFC2923). One standard method is to send probes and validate which get through at the transport layer- this is known as Packet-Layer PMTUD (PLPMTUD) and is standardised in RFC4821. PLPMTUD can of course also utilise ICMP messages (if it wishes) as a part of the process of learning what size of MTU will work across an entire Internet path.

## **PMTUD with IPv6**

RFC4443, which defines ICMPv6 specifies a “Packet Too Big” (type 2, code 0) error message, that is analogous to the ICMP “fragmentation needed and DF bit set” (type 3, code 4) error message of ICMP for IPv4. RFC1981 defines the Path MTU Discovery mechanism for IP Version 6, that makes use of these messages to determine the MTU of an arbitrary internet path. PLPMTUD also works with IPv6.

---

## **THE BORDER GATEWAY PROTOCOL (BGP)**

---

The Border Gateway Protocol (BGP) is the routing protocol used to exchange routing information across the Internet. It makes it possible for ISPs to connect to each other and for end-users to connect to more than one ISP. BGP is the only protocol that is designed to deal with a network of the Internet’s size, and the only protocol that can deal well with

having multiple connections to unrelated routing domains. BGP has proven to be scalable, stable and provides the mechanisms needed to support complex routing policies. When people talk about “BGP” today, they implicitly mean BGP4. There is no need to specify the-4 version number because no one uses earlier versions, and very few vendors even still support them.

The Border Gateway Protocol is an inter-Autonomous System routing protocol. The primary function of a BGP speaking system is to exchange network reachability information with other BGP systems.

This network reachability information includes information on the list of Autonomous Systems (AS) that reachability information traverses. This information is sufficient to construct a graph of AS connectivity from which routing loops may be pruned and some policy decisions at the AS level may be enforced. BGP4 provides a set of mechanisms for supporting Classless Inter-Domain Routing (CIDR) defined in RFC 4632.

These mechanisms include support for advertising a set of destinations as an IP prefix and eliminating the concept of network “class” within BGP. BGP version 4 also introduces mechanisms which allow aggregation of routes, including aggregation of AS paths.

Routing information exchanged via BGP supports only the destination-based forwarding paradigm, which assumes that

a router forwards a packet based solely on the destination address carried in the IP header of the packet. This, in turn, reflects the set of policy decisions that can (and can not) be enforced using BGP. BGP can support only the policies conforming to the destination-based forwarding paradigm.

A unique AS number (ASN) is allocated to each AS for use in BGP routing. The numbers are assigned by IANA and the Regional Internet Registries (RIR), the same authorities that allocate IP addresses. There are public numbers, which may be used on the Internet and range from 1 to 64511, and private numbers from 64512 to 65535, which can be used within an organization.

Most AS numbers are currently 16-bit integers, but deployment of 32-bit AS Numbers (4-Byte ASN) is starting. Why? Because of expected ASN shortage in the 16-bit range around the year 2010. In RFC 4893 (May 2007), some new extensions to BGP are described to carry the Autonomous System number as a four-octet entity. In RFC 5668 (Oct 2009), a new type of a BGP extended community which carries the 4-octet Autonomous System (AS) number, is defined.

---

## **HISTORY AND EVOLUTION**

---

TCP and IP were developed by a Department of Defence (DOD) research project to connect a number different networks designed by different vendors into a network of networks (the “Internet”). It was initially successful because

it delivered a few basic services that everyone needs (file transfer, electronic mail, remote logon) across a very large number of client and server systems. Several computers in a small department can use TCP/IP (along with other protocols) on a single LAN. The IP component provides routing from the department to the enterprise network, then to regional networks, and finally to the global Internet. On the battlefield a communications network will sustain damage, so the DOD designed TCP/IP to be robust and automatically recover from any node or phone line failure. This design allows the construction of very large networks with less central management. However, because of the automatic recovery, network problems can go undiagnosed and uncorrected for long periods of time.

*As with all other communications protocol, TCP/IP is composed of layers:*

- IP-is responsible for moving packet of data from node to node. IP forwards each packet based on a four byte destination address (the IP number). The Internet authorities assign ranges of numbers to different organizations. The organizations assign groups of their numbers to departments. IP operates on gateway machines that move data from department to organization to region and then around the world.
- TCP-is responsible for verifying the correct delivery of data from client to server. Data can be lost in the



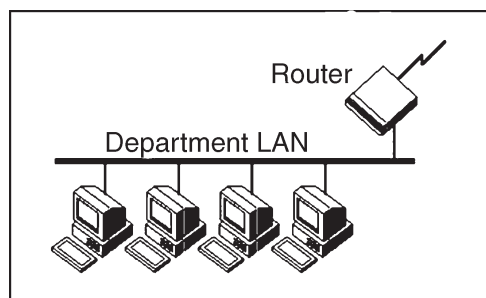
intermediate network. TCP adds support to detect errors or lost data and to trigger retransmission until the data is correctly and completely received.

- Sockets-is a name given to the package of subroutines that provide access to TCP/IP on most systems.

## **NETWORK OF LOWEST BIDDERS**

The Army puts out a bid on a computer and DEC wins the bid. The Air Force puts out a bid and IBM wins. The Navy bid is won by Unisys. Then the President decides to invade Grenada and the armed forces discover that their computers cannot talk to each other. The DOD must build a “network” out of systems each of which, by law, was delivered by the lowest bidder on a single contract.

The Internet Protocol was developed to create a Network of Networks (the “Internet”). Individual machines are first connected to a LAN (Ethernet or Token Ring). TCP/IP shares the LAN with other uses (a Novell file server, Windows for Workgroups peer systems). One device provides the TCP/IP connection between the LAN and the rest of the world.



To insure that all types of systems from all vendors can communicate, TCP/IP is absolutely standardized on the LAN. However, larger networks based on long distances and phone lines are more volatile. In the US, many large corporations would wish to reuse large internal networks based on IBM's SNA. In Europe, the national phone companies traditionally standardize on X.25. However, the sudden explosion of high speed microprocessors, fibre optics, and digital phone systems has created a burst of new options: ISDN, frame relay, FDDI, Asynchronous Transfer Mode (ATM). New technologies arise and become obsolete within a few years. With cable TV and phone companies competing to build the National Information Superhighway, no single standard can govern citywide, nationwide, or worldwide communications.

The original design of TCP/IP as a Network of Networks fits nicely within the current technological uncertainty. TCP/IP data can be sent across a LAN, or it can be carried within an internal corporate SNA network, or it can piggyback on the cable TV service. Furthermore, machines connected to any of these networks can communicate to any other network through gateways supplied by the network vendor.

## **ADDRESSES**

Each technology has its own convention for transmitting messages between two machines within the same network. On a LAN, messages are sent between machines by supplying the six byte unique identifier (the "MAC" address). In an SNA

### *Networking Hardware*

network, every machine has Logical Units with their own network address. DECNET, Appletalk, and Novell IPX all have a scheme for assigning numbers to each local network and to each workstation attached to the network.

On top of these local or vendor specific network addresses, TCP/IP assigns a unique number to every workstation in the world. This “IP number” is a four byte value that, by convention, is expressed by converting each byte into a decimal number (0 to 255) and separating the bytes with a period. For example, the PC Lube and Tune server is 130.132.59.234. An organization begins by sending electronic mail to Hostmaster@ INTERNIC.NET requesting assignment of a network number. It is still possible for almost anyone to get assignment of a number for a small “Class C” network in which the first three bytes identify the network and the last byte identifies the individual computer. The author followed this procedure and was assigned the numbers 192.35.91.\* for a network of computers at his house. Larger organizations can get a “Class B” network where the first two bytes identify the network and the last two bytes identify each of up to 64 thousand individual workstations. Yale’s Class B network is 130.132, so all computers with IP address 130.132.\*.\* are connected through Yale.

The organization then connects to the Internet through one of a dozen regional or specialized network suppliers. The network vendor is given the subscriber network number and

adds it to the routing configuration in its own machines and those of the other major network suppliers.

There is no mathematical formula that translates the numbers 192.35.91 or 130.132 into “Yale University” or “New Haven, CT.” The machines that manage large regional networks or the central Internet routers managed by the National Science Foundation can only locate these networks by looking each network number up in a table. There are potentially thousands of Class B networks, and millions of Class C networks, but computer memory costs are low, so the tables are reasonable. Customers that connect to the Internet, even customers as large as IBM, do not need to maintain any information on other networks. They send all external data to the regional carrier to which they subscribe, and the regional carrier maintains the tables and does the appropriate routing.

New Haven is in a border state, split 50-50 between the Yankees and the Red Sox. In this spirit, Yale recently switched its connection from the Middle Atlantic regional network to the New England carrier.

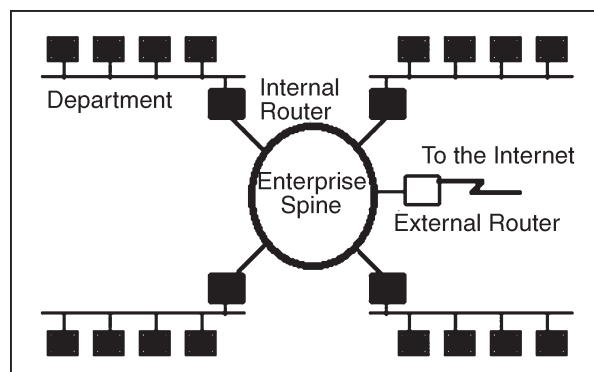
When the switch occurred, tables in the other regional areas and in the national spine had to be updated, so that traffic for 130.132 was routed through Boston instead of New Jersey.

The large network carriers handle the paperwork and can perform such a switch given sufficient notice. During a

conversion period, the university was connected to both networks so that messages could arrive through either path.

## SUBNETS

Although the individual subscribers do not need to tabulate network numbers or provide explicit routing, it is convenient for most Class B networks to be internally managed as a much smaller and simpler version of the larger network organizations. It is common to subdivide the two bytes available for internal assignment into a one byte department number and a one byte workstation ID.



The enterprise network is built using commercially available TCP/IP router boxes. Each router has small tables with 255 entries to translate the one byte department number into selection of a destination Ethernet connected to one of the routers. Messages to the PC Lube and Tune server (130.132.59.234) are sent through the national and New England regional networks based on the 130.132 part of the number. Arriving at Yale, the 59 department ID selects an Ethernet connector in the C& IS building. The 234 selects a

particular workstation on that LAN. The Yale network must be updated as new Ethernets and departments are added, but it is not effected by changes outside the university or the movement of machines within the department.

### **A UNCERTAIN PATH**

Every time a message arrives at an IP router, it makes an individual decision about where to send it next. There is concept of a session with a preselected path for all traffic. Consider a company with facilities in New York, Los Angeles, Chicago and Atlanta. It could build a network from four phone lines forming a loop (NY to Chicago to LA to Atlanta to NY). A message arriving at the NY router could go to LA via either Chicago or Atlanta. The reply could come back the other way.

How does the router make a decision between routes? There is no correct answer. Traffic could be routed by the “clockwise” algorithm (go NY to Atlanta, LA to Chicago). The routers could alternate, sending one message to Atlanta and the next to Chicago. More sophisticated routing measures traffic patterns and sends data through the least busy link.

If one phone line in this network breaks down, traffic can still reach its destination through a roundabout path. After losing the NY to Chicago line, data can be sent NY to Atlanta to LA to Chicago. This provides continued service though with degraded performance. This kind of recovery is the primary design feature of IP. The loss of the line is immediately detected

by the routers in NY and Chicago, but somehow this information must be sent to the other nodes. Otherwise, LA could continue to send NY messages through Chicago, where they arrive at a “dead end.” Each network adopts some Router Protocol which periodically updates the routing tables throughout the network with information about changes in route status.

If the size of the network grows, then the complexity of the routing updates will increase as will the cost of transmitting them. Building a single network that covers the entire US would be unreasonably complicated. Fortunately, the Internet is designed as a Network of Networks. This means that loops and redundancy are built into each regional carrier. The regional network handles its own problems and reroutes messages internally. Its Router Protocol updates the tables in its own routers, but no routing updates need to propagate from a regional carrier to the NSF spine or to the other regions (unless, of course, a subscriber switches permanently from one region to another).

## **UNDIAGNOSED PROBLEMS**

IBM designs its SNA networks to be centrally managed. If any error occurs, it is reported to the network authorities. By design, any error is a problem that should be corrected or repaired. IP networks, however, were designed to be robust. In battlefield conditions, the loss of a node or line is a normal circumstance. Casualties can be sorted out later on, but the network must stay up. So IP networks are robust. They

automatically (and silently) reconfigure themselves when something goes wrong. If there is enough redundancy built into the system, then communication is maintained. In 1975 when SNA was designed, such redundancy would be prohibitively expensive, or it might have been argued that only the Defence Department could afford it. Today, however, simple routers cost no more than a PC. However, the TCP/IP design that, “Errors are normal and can be largely ignored,” produces problems of its own.

Data traffic is frequently organized around “hubs,” much like airline traffic. One could imagine an IP router in Atlanta routing messages for smaller cities throughout the Southeast. The problem is that data arrives without a reservation. Airline companies experience the problem around major events, like the Super Bowl. Just before the game, everyone wants to fly into the city. After the game, everyone wants to fly out. Imbalance occurs on the network when something new gets advertised. Adam Curry announced the server at “mtv.com” and his regional carrier was swamped with traffic the next day. The problem is that messages come in from the entire world over high speed lines, but they go out to mtv.com over what was then a slow speed phone line.

Occasionally a snow storm cancels flights and airports fill up with stranded passengers. Many go off to hotels in town. When data arrives at a congested router, there is no place to send the overflow. Excess packets are simply discarded. It



becomes the responsibility of the sender to retry the data a few seconds later and to persist until it finally gets through. This recovery is provided by the TCP component of the Internet protocol.

TCP was designed to recover from node or line failures where the network propagates routing table changes to all router nodes. Since the update takes some time, TCP is slow to initiate recovery. The TCP algorithms are not tuned to optimally handle packet loss due to traffic congestion. Instead, the traditional Internet response to traffic problems has been to increase the speed of lines and equipment in order to stay ahead of growth in demand.

TCP treats the data as a stream of bytes. It logically assigns a sequence number to each byte. The TCP packet has a header that says, in effect, "This packet starts with byte 379642 and contains 200 bytes of data." The receiver can detect missing or incorrectly sequenced packets. TCP acknowledges data that has been received and retransmits data that has been lost. The TCP design means that error recovery is done end-to-end between the Client and Server machine. There is no formal standard for tracking problems in the middle of the network, though each network has adopted some ad hoc tools.

### **Need to Know**

There are three levels of TCP/IP knowledge. Those who administer a regional or national network must design a

system of long distance phone lines, dedicated routing devices, and very large configuration files. They must know the IP numbers and physical locations of thousands of subscriber networks. They must also have a formal network monitor strategy to detect problems and respond quickly.

Each large company or university that subscribes to the Internet must have an intermediate level of network organization and expertise. A half dozen routers might be configured to connect several dozen departmental LANs in several buildings. All traffic outside the organization would typically be routed to a single connection to a regional network provider.

However, the end user can install TCP/IP on a personal computer without any knowledge of either the corporate or regional network.

Three pieces of information are required:

- The IP address assigned to this personal computer
- The part of the IP address (the subnet mask) that distinguishes other machines on the same LAN (messages can be sent to them directly) from machines in other departments or elsewhere in the world (which are sent to a router machine)
- The IP address of the router machine that connects this LAN to the rest of the world.

In the case of the PCLT server, the IP address is 130.132.59.234. Since the first three bytes designate this

department, a “subnet mask” is defined as 255.255.255.0 (255 is the largest byte value and represents the number with all bits turned on). It is a Yale convention (which we recommend to everyone) that the router for each department have station number 1 within the department network. Thus the PCLT router is 130.132.59.1. Thus the PCLT server is configured with the values:

- My IP address: 130.132.59.234
- Subnet mask: 255.255.255.0
- Default router: 130.132.59.1

The subnet mask tells the server that any other machine with an IP address beginning 130.132.59.\* is on the same department LAN, so messages are sent to it directly. Any IP address beginning with a different value is accessed indirectly by sending the message through the router at 130.132.59.1 (which is on the departmental LAN).

---

## **ETHERNET (PHYSICAL/DATA LINK LAYERS)**

---

The physical layer of the network focuses on hardware elements, such as cables, repeaters, and network interface cards. By far the most common protocol used at the physical layer is Ethernet. For example, an Ethernet network (such as 10BaseT or 100BaseTX) specifies the type of cables that can be used, the optimal topology (star vs. bus, etc.), the maximum length of cables, etc.

The data link layer of the network addresses the way that data packets are sent from one node to another. Ethernet uses an access method called CSMA/CD (Carrier Sense Multiple Access/Collision Detection). This is a system where each computer listens to the cable before sending anything through the network. If the network is clear, the computer will transmit.

If some other node is already transmitting on the cable, the computer will wait and try again when the line is clear. Sometimes, two computers attempt to transmit at the same instant. When this happens a collision occurs. Each computer then backs off and waits a random amount of time before attempting to retransmit. With this access method, it is normal to have collisions. However, the delay caused by collisions and retransmitting is very small and does not normally effect the speed of transmission on the network.

## **ETHERNET**

The original Ethernet standard was developed in 1983 and had a maximum speed of 10 Mbps (phenomenal at the time) over coaxial cable. The Ethernet protocol allows for bus, star, or tree topologies, depending on the type of cables used and other factors. This heavy coaxial cabling was expensive to purchase, install, and maintain, and very difficult to retrofit into existing facilities.

The current standards are now built around the use of twisted pair wire. Common twisted pair standards are

10BaseT, 100BaseT, and 1000BaseT. The number (10, 100, 1000) stands for the speed of transmission (10/100/1000 megabits per second); the “Base” stands for “baseband” meaning it has full control of the wire on a single frequency; and the “T” stands for “twisted pair” cable. Fibre cable can also be used at this level in 10BaseFL.

### **FAST ETHERNET**

The Fast Ethernet protocol supports transmission up to 100 Mbps. Fast Ethernet requires the use of different, more expensive network concentrators/hubs and network interface cards. In addition, category 5 twisted pair or fibre optic cable is necessary. Fast Ethernet standards include:

- 100BaseT- 100 Mbps over 2-pair category 5 or better UTP cable.
- 100BaseFX- 100 Mbps over fibre cable.
- 100BaseSX-100 Mbps over multimode fibre cable.
- 100BaseBX- 100 Mbps over single mode fibre cable.

### **GIGABIT ETHERNET**

Gigabit Ethernet standard is a protocol that has a transmission speed of 1 Gbps (1000 Mbps). It can be used with both fibre optic cabling and copper.

1000BaseT- 1000 Mbps over 2-pair category 5 or better UTP cable.

- 1000BaseTX- 1000 Mbps over 2-pair category 6 or better UTP cable.

- 1000BaseFX- 1000 Mbps over fibre cable.
- 1000BaseSX-1000 Mbps over multimode fibre cable.
- 1000BaseBX- 1000 Mbps over single mode fibre cable.

The Ethernet standards continue to evolve. with 10 Gigabit Ethernet (10,000 Mbps) and 100 Gigabit Ethernet (100,000 Mbps),

**Table. Ethernet Protocol Summary**

<b>Protocol</b>	<b>Cable</b>	<b>Speed</b>
Ethernet	Twisted Pair, Coaxial, Fibre	10 Mbps
Fast Ethernet	Twisted Pair, Fibre	100 Mbps
Gigabit Ethernet	Twisted Pair, Fibre	1000 Mbps

## **OLDER NETWORK PROTOCOLS**

Several very popular network protocols, commonly used in the 90's and early 21st century have now largely fallen into disuse. While you may hear terms from time to time, such as "LocalTalk" (Apple) or "Token Ring" (IBM), you will rarely find these systems still in operation. Although they played an important role in the evolution of networking, their performance and capacity limitations have relegated them to the past, in the wake of the standardization of Ethernet driven by the success of the Internet.

## **IP AND IPX (NETWORK LAYER)**

The network layer is in charge of routing network messages (data) from one computer to another. The common protocols at this layer are IP (which is paired with TCP at the transport layer for Internet network) and IPX (which is paired with

SPX at the transport layer for some older Macintosh, Linus, UNIX, Novell and Windows networks). Because of the growth in Internet-based networks, IP/TCP are becoming the leading protocols for most networks.

Every network device (such as network interface cards and printers) have a physical address called a MAC (Media Access Control) address. When you purchase a network card, the MAC address is fixed and cannot be changed. Networks using the IP and IPX protocols assign logical addresses (which are made up of the MAC address and the network address) to the devices on the network, This can all become quite complex — suffice it to say that the network layer takes care of assigning the correct addresses (via IP or IPX) and then uses routers to send the data packets to other networks.

### **TCP AND SPX (TRANSPORT LAYER)**

The transport layer is concerned with efficient and reliable transportation of the data packets from one network to another. In most cases, a document, e-mail message or other piece of information is not sent as one unit. Instead, it is broken into small data packets, each with header information that identifies its correct sequence and document. When the data packets are sent over a network, they may or may not take the same route — it doesn't matter. At the receiving end, the data packets are re-assembled into the proper order. After all packets are received, a message goes back to the originating network. If a packet does not arrive, a message

to “re-send” is sent back to the originating network. TCP, paired with IP, is by far the most popular protocol at the transport level. If the IPX protocol is used at the network layer (on networks such as Novell or Microsoft), then it is paired with SPX at the transport layer.

### **HTTP, FTP, SMTP AND DNS (SESSION/ PRESENTATION/APPLICATION LAYERS)**

Several protocols overlap the session, presentation, and application layers of networks. There protocols listed below are a few of the more well-known:

- DNS- Domain Name System- translates network address (such as IP addresses) into terms understood by humans (such as Domain Names) and *vice-versa*
- DHCP- Dynamic Host Configuration Protocol-can automatically assign Internet addresses to computers and users
- FTP- File Transfer Protocol- a protocol that is used to transfer and manipulate files on the Internet
- HTTP- HyperText Transfer Protocol-An Internet-based protocol for sending and receiving webpages
- IMAP- Internet Message Access Protocol- A protocol for e-mail messages on the Internet
- IRC- Internet Relay Chat- a protocol used for Internet chat and other communications
- POP3- Post Office protocol Version 3- a protocol used by e-mail clients to retrieve messages from remote servers



- SMTP- Simple Mail Transfer Protocol- A protocol for e-mail messages on the Internet.

---

## **ADDRESS RESOLUTION PROTOCOL**

---

The address resolution protocol (arp) is a protocol used by the Internet Protocol (IP) [RFC826], specifically IPv4, to map IP network addresses to the hardware addresses used by a data link protocol. The protocol operates below the network layer as a part of the interface between the OSI network and OSI link layer. It is used when IPv4 is used over Ethernet. The term address resolution refers to the process of finding an address of a computer in a network. The address is “resolved” using a protocol in which a piece of information is sent by a client process executing on the local computer to a server process executing on a remote computer. The information received by the server allows the server to uniquely identify the network system for which the address was required and therefore to provide the required address. The address resolution procedure is completed when the client receives a response from the server containing the required address.

An Ethernet network uses two hardware addresses which identify the source and destination of each frame sent by the Ethernet. The destination address (all 1’s) may also identify a broadcast packet (to be sent to all connected computers). The hardware address is also known as the

Medium Access Control (MAC) address, in reference to the standards which define Ethernet. Each computer network interface card is allocated a globally unique 6 byte link address when the factory manufactures the card (stored in a PROM). This is the normal link source address used by an interface. A computer sends all packets which it creates with its own hardware source link address, and receives all packets which match the same hardware address in the destination field or one (or more) pre-selected broadcast/multicast addresses.

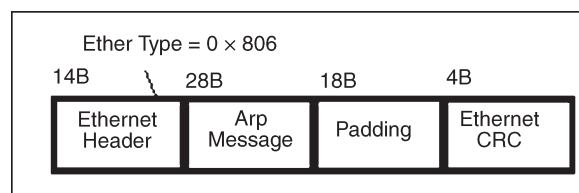
The Ethernet address is a link layer address and is dependent on the interface card which is used. IP operates at the network layer and is not concerned with the link addresses of individual nodes which are to be used. The address resolution protocol (arp) is therefore used to translate between the two types of address. The arp client and server processes operate on all computers using IP over Ethernet. The processes are normally implemented as part of the software driver that drives the network interface card. There are four types of arp messages that may be sent by the arp protocol. These are identified by four values in the “operation” field of an arp message. The types of message are:

- ARP request
- ARP reply
- RARP request
- RARP reply.

To reduce the number of address resolution requests, a client normally caches resolved addresses for a (short) period of time. The arp cache is of a finite size, and would become full of incomplete and obsolete entries for computers that are not in use if it was allowed to grow without check. The arp cache is therefore periodically flushed of all entries. This deletes unused entries and frees space in the cache. It also removes any unsuccessful attempts to contact computers which are not currently running. If a host changes the MAC address it is using, this can be detected by other hosts when the cache entry is deleted and a fresh arp message is sent to establish the new association. The use of gratuitous arp (*e.g.* triggered when the new NIC interface is enabled with an IP address) provides a more rapid update of this information.

### **EXAMPLE OF USE OF THE ADDRESS RESOLUTION PROTOCOL (ARP)**

The use of arp when a computer tries to contact a remote computer on the same LAN (known as “sysa”) using the “ping” program. It is assumed that no previous IP datagrams have been received from this computer, and therefore arp must first be used to identify the MAC address of the remote computer.



The arp request message (“who is X.X.X.X tell Y.Y.Y.Y”, where X.X.X.X and Y.Y.Y.Y are IP addresses) is sent using the Ethernet broadcast address, and an Ethernet protocol type of value  $0 \times 806$ . Since it is broadcast, it is received by all systems in the same collision domain (LAN). This ensures that if the target of the query is connected to the network, it will receive a copy of the query. Only this system responds. The other systems discard the packet silently.

The target system forms an arp response (“X.X.X.X is hh:hh:hh:hh:hh:hh”, where hh:hh:hh:hh:hh:hh is the Ethernet source address of the computer with the IP address of X.X.X.X). This packet is unicast to the address of the computer sending the query (in this case Y.Y.Y.Y). Since the original request also included the hardware address (Ethernet source address) of the requesting computer, this is already known, and doesn’t require another arp message to find this out.

### **GRATUITOUS ARP**

Gratuitous ARP is used when a node (end system) has selected an IP address and then wishes to defend its chosen address on the local area network (*i.e.* to check no other node is using the same IP address). It can also be used to force a common view of the node’s IP address (*e.g.* after the IP address has changed). Use of this is common when an interface is first configured, as the node attempts to clear out any stale caches that might be present on other hosts.

The node simply sends an arp request for itself. The Address Resolution Protocol (ARP) is a low-level protocol that dynamically learns and maps network layer IP addresses to physical Medium Access Control (MAC) addresses, for example, Ethernet. Given only the network layer IP address of the destination system, ARP lets a router find the MAC address of the destination host on the same network segment. For example, a router receives an IP packet destined for a host connected to one of its LANs. The packet contains only a 32-bit IP destination address. To be able to forward the packet on the LAN, the router must construct the Data Link layer header using the physical MAC address of the destination host. The router must acquire this physical MAC address of the destination host and map that address to the 32-bit IP address.

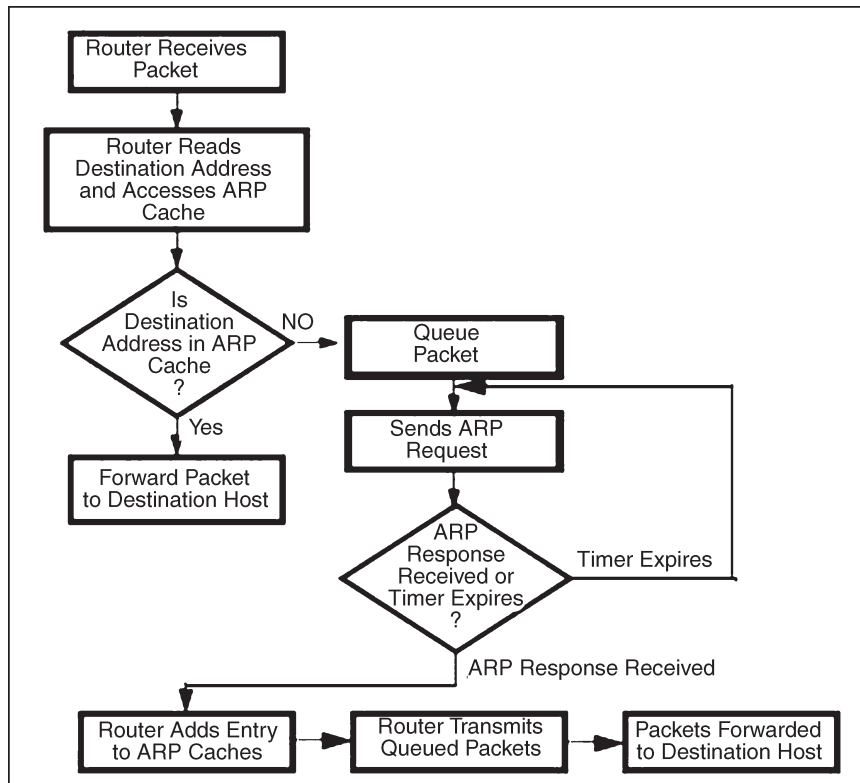
To obtain the physical address of the host, the router broadcasts an ARP request to all host of the network. Only the host with that IP address responds with its physical MAC address. The router saves the IP/MAC address mapping in a table called ARP cache and it can use this mapping in the future when forwarding packets to the destination host.

**RFC:** RFC 826 documents the ARP protocol.

### **ARP Physical Address Broadcast**

**Note:** If the ARP cache does not contain an entry for a destination, the packet is queued pending an ARP Response. This means that the first packet sent between IP Hosts is

queued until the expiration of the Time to Retry timer. If an ARP Response is not received within this time an ARP Request is retransmitted. All IP-based protocols perform this function.



**Note:** If a second IP packet, intended for the same Destination Address, arrives while the device is awaiting an ARP Response, the packet is queued but a second ARP Request is not sent. When another IP packet, intended for a different Destination Address, arrives while the device is awaiting an ARP Response for the first packet, an ARP Request for the second Destination Address is immediately broadcast to the network.

# 4

---

## IP Routing

---

IP Routing is an umbrella term for the set of protocols that determine the path that data follows in order to travel across multiple networks from its source to its destination. Data is routed from its source to its destination through a series of routers, and across multiple networks. The IP Routing protocols enable routers to build up a forwarding table that correlates final destinations with next hop addresses.

*These protocols include:*

- BGP (Border Gateway Protocol)
- IS-IS (Intermediate System-Intermediate System)
- OSPF (Open Shortest Path First)
- RIP (Routing Information Protocol).

When an IP packet is to be forwarded, a router uses its forwarding table to determine the next hop for the packet's

destination (based on the destination IP address in the IP packet header), and forwards the packet appropriately. The next router then repeats this process using its own forwarding table, and so on until the packet reaches its destination. At each stage, the IP address in the packet header is sufficient information to determine the next hop; no additional protocol headers are required.

The Internet, for the purpose of routing, is divided into Autonomous Systems (ASs). An AS is a group of routers that are under the control of a single administration and exchange routing information using a common routing protocol. For example, a corporate intranet or an ISP network can usually be regarded as an individual AS. The Internet can be visualized as a partial mesh of ASs. An AS can be classified as one of the following three types.

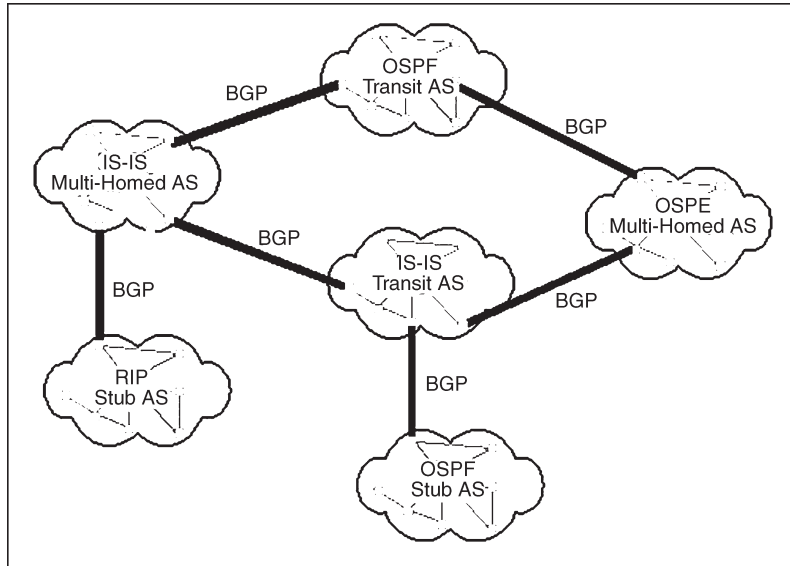
- A Stub AS has a single connection to one other AS. Any data sent to, or received from, a destination outside the AS must travel over that connection. A small campus network is an example of a stub AS.
- A Transit AS has multiple connections to one or more ASs, which permits data that is not destined for a node within that AS to travel through it. An ISP network is an example of a transit AS.
- A Multihomed AS also has multiple connections to one or more ASs, but it does not permit data received over one of these connections to be forwarded out of the



AS again. In other words, it does not provide a transit service to other ASs. A Multihomed AS is similar to a Stub AS, except that the ingress and egress points for data travelling to or from the AS can be chosen from one of a number of connections, depending on which connection offers the shortest route to the eventual destination. A large enterprise network would normally be a multihomed AS.

An Interior Gateway Protocol (IGP) calculates routes within a single AS. The IGP enables nodes on different networks within an AS to send data to one another. The IGP also enables data to be forwarded across an AS from ingress to egress, when the AS is providing transit services. Routes are distributed between ASs by an Exterior Gateway Protocol (EGP).

The EGP enables routers within an AS to choose the best point of egress from the AS for the data they are trying to route. The EGP and the IGPs running within each AS cooperate to route data across the Internet. The EGP determines the ASs that data must cross in order to reach its destination, and the IGP determines the path within each AS that data must follow to get from the point of ingress (or the point of origin) to the point of egress (or the final destination). The diagram below illustrates the different types of AS in a network. OSPF, IS-IS and RIP are IGPs used within the individual ASs; BGP is the EGP used between ASs.



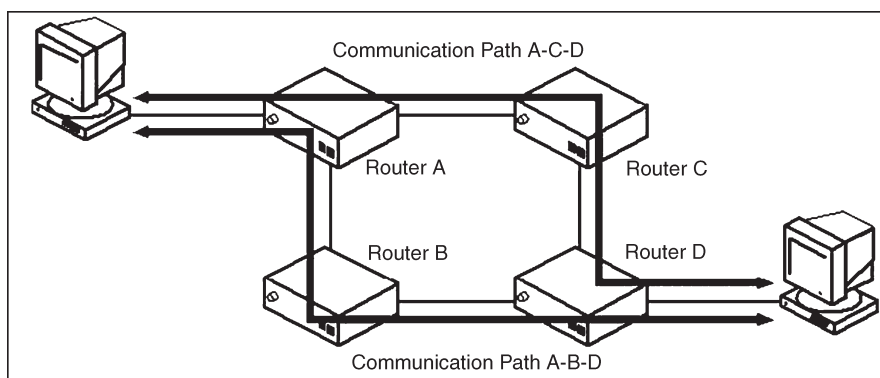
## ROUTING PRINCIPLES

Routing is one of the most important and most complex operations performed by TCP/IP. The protocols were designed with scalability in mind, but no one in the 1970s could have predicted the massive growth of the Internet that would occur two decades later. Whereas packets might pass through a handful of routers on a private internetwork, Internet packets routinely pass through a dozen or more routers on the way to their destinations. Some of the routers on the Internet have to maintain information about several networks, and the process of compiling and maintaining this information makes the Internet routing process very complex.

A *router* is a system connected to two or more networks that forwards packets from one network to another. Routers operate at the network layer of the OSI reference model, so

they can connect networks running different data-link layer protocols and different network media.

On a small internetwork, a router's job can be quite simple. When you have two local area networks (LANs) connected by one router, for example, the router simply receives packets from one network and forwards only those destined for the other network. On a large internetwork, however, routers connect several different networks together, and in many cases, networks have more than one router connected to them.



**Figure:** Internetworks with Redundant Routers Provide Multiple Paths between Two End Systems.

This enables packets to take different paths to a given destination. If one router on the network should fail, packets can bypass it and still reach their destinations.

In a complex internetwork, an important part of a router's job is to select the most efficient route to a packet's destination. Usually, this is the path that enables a packet to reach the destination with the fewest number of hops (that is, by passing through the smallest number of routers).

Routers share information about the networks to which they are attached with other routers in the immediate vicinity. As a result, a composite picture of the internetwork eventually develops, but on a large internetwork such as the Internet, no single router possesses the entire image. Instead, the routers work together by passing each packet from router to router, one hop at a time.

## **ROUTER PRODUCTS**

A router can be a stand-alone hardware device or a regular computer. Operating systems like Microsoft Windows 2000, Microsoft Windows NT, and Novell NetWare have the ability to route IP traffic, so creating a router out of a computer running one of these operating systems is simply a matter of installing two network interface adapters, connecting the computer to two different networks, and configuring it to route traffic between those networks. In TCP/IP parlance, a computer with two or more network interfaces is called a *multihomed* system. Microsoft Windows 95, Microsoft Windows 98, and Microsoft Windows Me on their own can't route IP traffic between two network interface adapters, but you can use systems running these operating systems as dial-in servers that enable you to access a network from a remote location using the NetBIOS Enhanced User Interface (NetBEUI) or Internetwork Packet Exchange (IPX) protocols.

Windows 98 Second Edition and Windows Me also include an Internet Connection Sharing (ICS) feature, which enables

other computers on the LAN to access the Internet through one computer's dial-up connection to an Internet service provider (ISP).

There are also third-party software products that provide Internet connection sharing. In essence, these products are software routers that enable your computer to forward packets between the local network and the network run by your ISP. Using these products, all of the computers on a LAN, such as one installed in a home or a small business, can share a single computer's connection to the Internet, whether it uses a dial-up modem, cable modem, or other type of connection.

When you use a computer as an IP router, each of the network interface adapters must have its own IP address appropriate for the network to which it is attached. When one of the two networks is an ISP connection, the ISP's server typically supplies the address for that interface. The other IP address is the one that you assign to your network interface adapter when you install it.

A stand-alone router is a hardware device that is essentially a special-purpose computer. The unit has multiple built-in network interface adapters, a processor, and memory in which it stores its routing information and temporary packet buffers. Routers are now available for a wide range of prices and with a variety of capabilities. You can purchase an inexpensive stand-alone router that enables you to share an

Internet connection with a small network for a few hundred dollars, or you can move up to enormously expensive rack-mounted models that connect the LANs of a large internetwork or provide wide area connectivity to remote offices or ISPs.

## **ROUTING TABLES**

The routing table is the heart of any router; without it, all that's left is mechanics. The routing table holds the information that the router uses to forward packets to the proper destinations. However, it is not only routers that have routing tables; every TCP/IP system has one, which it uses to determine where to send its packets.

Routing is essentially the process of determining what data-link layer protocol address the system should use to reach a particular IP address. If a system wants to transmit a packet to a computer on the local network, for example, the routing table instructs it to address the packet directly to that system. This is called a *direct route*. In this case, the Destination IP Address field in the IP header and the Destination Address field in the data-link layer protocol header refer to the same computer.

If a packet's destination is on another network, the routing table contains the address of the router that the system should use to reach that destination. In this case, the Destination IP Address and Destination Address fields specify different systems because the data-link layer address has to

refer to a system on the local network, and for the packet to reach a computer on a different network, that local system must be a router. Because the two addresses refer to different systems, this is called an *indirect route*.

### **Routing Table Format**

A routing table is essentially a list of networks (and possibly hosts) and addresses of routers that the system can use to reach them. The arrangement of the information in the routing table can differ depending on the operating system, but it generally appears in something like the following format, which is the routing table from a Windows 2000 system.

*The functions of the various columns in the table are as follows:*

- *Network Address.* This column specifies the address of the network or host for which routing information is provided in the other columns.
- *Netmask.* This column specifies the subnet mask for the value in the Network Address column. As with any subnet mask, the system uses the Netmask value to determine which parts of the Network Address value are the network identifier, the subnet identifier (if any), and the host identifier.
- *Gateway Address.* This column specifies the address of the router that the system should use to send datagrams to the network or host identified in the

## Networking Hardware

Network Address column. On a LAN, the hardware address for the system identified by the Gateway Address value will become the Destination Address value in the packet's data-link layer protocol header.

- *Interface*. This column specifies the address of the network interface adapter that the computer should use to transmit packets to the system identified in the Gateway Address column.
- *Metric*. This column contains a value that enables the system to compare the relative efficiency of routes to the same destination.

<b>Network Address</b>	<b>Netmask</b>	<b>Gateway Address</b>	
<b>Interface</b>	<b>Metric</b>		
0.0.0.0	0.0.0.0	192.168.2.99	192.168.2.2 1
127.0.0.0	255.0.0.0	127.0.0.1	127.0.0.1 1
192.168.2.0	255.255.255.0	192.168.2.2	192.168.2.2 1
192.168.2.2	255.255.255.25	5	127.0.0.1
127.0.0.1	1		
192.168.2.255	255.255.255.25	5	192.168.2.2
192.168.2.2	1		
224.0.0.0	224.0.0.0	192.168.2.2	192.168.2.2 1
255.255.255.25	5	255.255.255.25	5 192.168.2.2
192.168.2.2	1		

### Routing Table Entries

The sample routing table shown previously contains typical entries for a workstation that is not functioning as a router. The value 0.0.0.0 in the Network Address column, found in the first entry in the table, identifies the default gateway entry. The *default gateway* is the router on the LAN that the system uses when there are no routing table entries that match the Destination IP Address of an outgoing packet.



Even if there are multiple routers available on the local network, a routing table can have only one functional default gateway entry. On a typical workstation that is not a router, the majority of packets go to the default gateway; only packets destined for systems on the local network do not use this router. The Gateway Address column in the default gateway entry contains the IP address of a router on the local network, and the Interface column contains the IP address of the network interface adapter that connects the system to the network.

In TCP/IP terminology, the term *gateway* is synonymous with the term *router*. However, this is not the case in other networking disciplines, in which a gateway can refer to a different device that connects networks at the application layer instead of the network layer.

The second entry in the sample routing table contains a special IP address that is designated as the TCP/IP loopback address. IP automatically routes all packets destined for any address on the 127.0.0.0 network right back to the incoming packet queue on the same computer. The packets never reach the data-link layer or leave the computer. The entry ensures this by specifying that the system should use its own loopback address (127.0.0.1) as the “router” to the destination.

The IP address of the network interface adapter in the computer to which this routing table belongs is 192.168.2.2. Therefore, the third entry in the sample routing table contains

the address of the local network on which the computer is located. The Network Address and Netmask values indicate that it is a Class C network with the address 192.168.2.0. This is the entry that the system uses for direct routes when it transmits packets to other systems on the local network. The Gateway Address and Interface columns both contain the IP address of the network interface adapter for the computer, indicating that the computer should use itself as the gateway. In other words, the computer should transmit the data-link layer frames to the same computer identified by the Destination IP Address value in the datagrams.

The fourth entry in the sample routing table contains the host address of the computer itself. It instructs the system to transmit data addressed to itself to the loopback address. IP always searches the routing table for host address entries before network address entries, so when processing any packets addressed to the computer's own address (192.168.2.2), IP would select this entry before the entry above it, which specifies the system's network address.

The fifth and seventh entries in the sample routing table contain broadcast addresses, both the generic IP broadcast address (255.255.255.255) and the local network's broadcast address (192.168.2.255). In both of these cases, packets are transmitted to the computers on the local network, so the system again uses itself as a gateway. The sixth entry in the sample routing table contains the network address for the

multicast addresses designated by the Internet Assigned Numbers Authority (IANA) for specific purposes.

The routing table on a router is considerably more complex because it contains entries for all of the networks to which it's attached, as well as entries provided manually by administrators or dynamically by routing protocols. A router also makes more use of the Interface and Metric columns. On a system with one network interface adapter, there is only one interface to use, so the Interface column is actually superfluous.

Routers and multihomed systems have at least two network interfaces, so the value in the Interface column is a crucial part of transmitting a packet correctly. In the same way, the Metric values in a singlehomed system's routing table are superfluous as well, because the computer has no information about routes more distant than those on the local network. As a result, the Metric value for all of the entries is 1.

### **Selecting a Table Entry**

When a TCP/IP system has data to transmit, the IP protocol selects a route for each packet using the procedure.

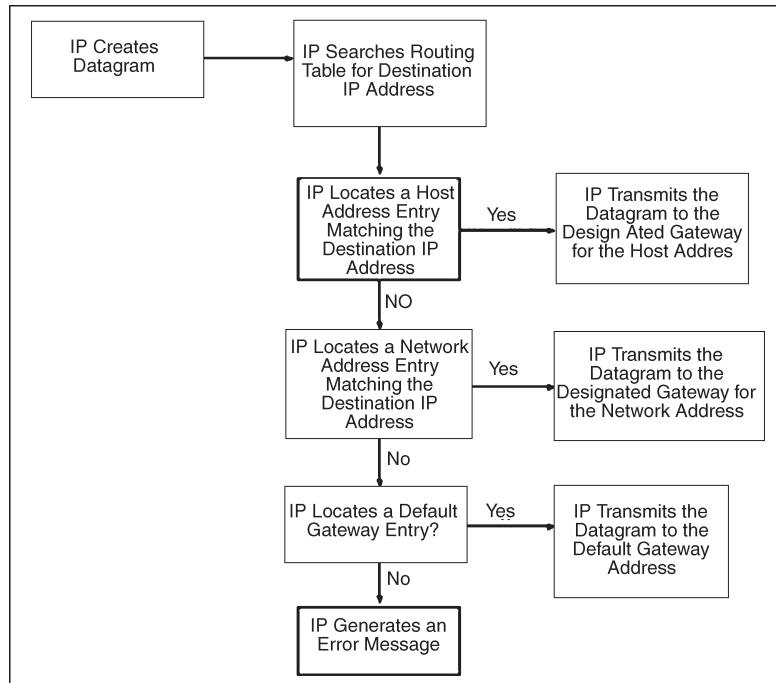
*The IP protocol selects a route using the following procedure:*

1. After packaging the transport layer information into a datagram, IP compares the Destination IP Address for the packet with the routing table, looking for a host address with the same value. A host address

entry in the table has a full IP address in the Network Address column and the value 255.255.255.255 in the Netmask column.

2. If there is no host address entry that exactly matches the Destination IP Address value, the system then scans the routing table's Network Address and Netmask columns for an entry that matches the address's network and subnet identifiers. If there is more than one entry in the routing table that contains the desired network and subnet identifiers, IP uses the entry with the lower value in the Metric column.
3. If there are no table entries that match the network and subnet identifiers of the Destination IP Address value, the system searches for a default gateway entry that has a value of 0.0.0.0 in the Network Address and Netmask columns.
4. If there is no default gateway entry, the system generates an error message. If the system transmitting the datagram is a router, it transmits an Internet Control Message Protocol (ICMP) Destination Unreachable message back to the end system that originated the datagram. If the system transmitting the datagram is itself an end system, the error message gets passed back up to the application that generated the data.

## Networking Hardware



**Figure:** TCP/IP Systems Search the Routing Table for an Address that Matches the Destination IP Address Value Found in the Header of Each Datagram.

5. When the system locates a viable routing table entry, IP prepares to transmit the datagram to the router identified in the Gateway Address column. The system consults the Address Resolution Protocol (ARP) cache or performs an ARP procedure to obtain the hardware address of the router.
6. Once it has the router's hardware address, IP passes it and the datagram down to the data-link layer protocol associated with the address specified in the Interface column. The data-link layer protocol constructs a frame using the router's hardware address in its Destination Address field and transmits it out over the designated interface.

---

## **DYNAMIC ROUTING**

---

Dynamic routing protocols not only perform these path determination and route table update functions but also determine the next-best path if the best path to a destination becomes unusable. The capability to compensate for topology changes is the most important advantage dynamic routing offers over static routing.

Obviously, for communications to occur the communicators must speak the same language. There are eight major IP routing protocols from which to choose; if one router speaks RIP and another speaks OSPF, they cannot share routing information because they are not speaking the same language.

Subsequent chapters examine all the IP routing protocols in current use, and even consider how to make a router “bilingual,” but first it is necessary to explore some characteristics and issues common to all routing protocols—IP or otherwise.

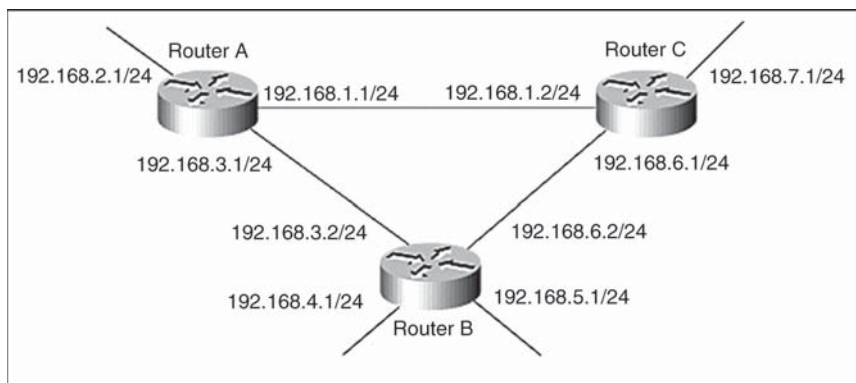
All dynamic routing protocols are built around an algorithm. Generally, an *algorithm* is a step-by-step procedure for solving a problem. A routing algorithm must, at a minimum, specify the following:

- A procedure for passing reachability information about networks to other routers
- A procedure for receiving reachability information from other routers

## Networking Hardware

- A procedure for determining optimal routes based on the reachability information it has and for recording this information in a route table
- A procedure for reacting to, compensating for, and advertising topology changes in an internetwork

A few issues common to any routing protocol are path determination, metrics, convergence, and load balancing.



## PATH DETERMINATION

All networks within an internetwork must be connected to a router, and wherever a router has an interface on a network that interface must have an address on the network. This address is the originating point for reachability information. Each router knows about its directly connected networks from its assigned addresses and masks.

## Metrics

When there are multiple routes to the same destination, a router must have a mechanism for calculating the best path. A *metric* is a variable assigned to routes as a means of ranking them from best to worst or from most preferred to least

preferred. Consider the following example of why metrics are needed.

Different routing protocols use different, and sometimes multiple, metrics. For example, RIP defines the “best” route as the one with

### ***Hop Count***

A hop count metric simply counts router hops. For instance, from router A it is 1 hop to network 192.168.5.0 if packets are sent out interface 192.168.3.1 (through router B) and 2 hops if packets are sent out 192.168.1.1 (through routers C and B). Assuming hop count is the only metric being applied, the best route is the one with the fewest hops, in this case, A-B.

But is the A-B link really the best path? If the A-B link is a DS-0 link and the A-C and C-B links are T-1 links, the 2-hop route may actually be best because bandwidth plays a role in how efficiently traffic travels through the network.

### ***Bandwidth***

A bandwidth metric would choose a higher-bandwidth path over a lower-bandwidth link. However, bandwidth by itself still may not be a good metric. What if one or both of the T1 links are heavily loaded with other traffic and the 56K link is lightly loaded? Or what if the higher-bandwidth link also has a higher delay?



### **Load**

This metric reflects the amount of traffic utilizing the links along the path. The best path is the one with the lowest load.

Unlike hop count and bandwidth, the load on a route changes, and therefore the metric will change. Care must be taken here. If the metric changes too frequently, *route flapping*—the frequent

### **Delay**

*Delay* is a measure of the time a packet takes to traverse a route. A routing protocol using delay as a metric would choose the path with the least delay as the best path. There may be many ways to measure delay. Delay may take into account not only the delay of the links along the route but also such factors as router latency and queuing delay. On the other hand, the delay of a route may be not measured at all; it may be a sum of static quantities defined for each interface along the path. Each individual delay quantity would be an estimate based on the type of link to which the interface is connected.

### **Convergence**

A dynamic routing protocol must include a set of procedures for a router to inform other routers about its directly connected networks, to receive and process the same information from other routers, and to pass along the



administrator. The path with highest reliability would be selected as best.

### **Cost**

This metric is configured by a network administrator to reflect more- or less-preferred routes. Cost may be defined by any policy or link characteristic or may reflect the arbitrary judgment of the network administrator.

The term *cost* is often used as a generic term when speaking of route choices. For example, “RIP chooses the lowest-cost path based on hop count.” Another generic term is *shortest*, as in “RIP chooses the shortest path based on hop count.” When used in this context, either *lowest-cost* (or *highest-cost*) and *shortest* (or *longest*) merely refer to a routing protocol’s view of paths based on its specific metrics.

## **STATIC VS. DYNAMIC ROUTING**

### **Static Routing**

Static routing is not really a routing protocol. Static routing is simply the process of manually entering routes into a device’s routing table via a configuration file that is loaded when the routing device starts up. As an alternative, these routes can be entered by a network administrator who configures the routes manually. Since these manually configured routes don’t change after they are configured (unless a human changes them) they are called ‘static’ routes.

Static routing is the simplest form of routing, but it is a manual process.

Use static routing when you have very few devices to configure (<5) and when you know the routes will probably never change.

Static routing also does not handle failures in external networks well because any route that is configured manually must be updated or reconfigured manually to fix or repair any lost connectivity.

### **Dynamic Routing**

Dynamic routing protocols are supported by software applications running on the routing device (the router) which dynamically learn network destinations and how to get to them and also advertise those destinations to other routers. This advertisement function allows all the routers to learn about all the destination networks that exist and how to those networks.

A router using dynamic routing will 'learn' the routes to all networks that are directly connected to the device. Next, the router will learn routes from other routers that run the same routing protocol (RIP, RIP2, EIGRP, OSPF, IS-IS, BGP etc). Each router will then sort through it's list of routes and select one or more 'best' routes for each network destination the router knows or has learned. Dynamic routing protocols will then distribute this 'best route' information to other routers running the same routing protocol, thereby extending

the information on what networks exist and can be reached. This gives dynamic routing protocols the ability to adapt to logical network topology changes, equipment failures or network outages 'on the fly'.

## **RIPV2**

When it comes to internal routing protocols, Routing Information Protocol version 2 (RIPv2) is one of the most common routing protocols in use today. In addition, RIPv2 is one of the easiest protocols to configure.

Meant for smaller networks, RIPv2 does have a number of limitations.

However, version 2 boasts some critical improvements over the previous version. RIPv2 is a valuable tool that provides quick and simple IP routing, and it should be in every network administrator's toolbox.

### **WHAT TYPE OF ROUTING PROTOCOL IS RIP?**

Based on RFC 1388, 1723, and 2453, RIP is a distance-vector routing protocol. Its primary limitation is that it can't support a network that has more than 15 hops. RIP assumes that anything more than 15 hops is infinity, so it considers the route invalid. Despite this limitation, RIP works great for basic route communications between devices.

Its other benefit is that it's so widespread. Many routers come with RIP by default, even many small office routers. In addition, many firewalls support RIP, but not OSPF or EIGRP.

*Here are some important RIP facts:*

- RIP's administrative distance is 120 for both RIPv1 and RIPv2.
- RIPv2 sends routing updates via multicast address 224.0.0.9.
- Cisco routers don't enable RIPv2 by default. To use RIPv2, you must use the *ver 2* command in RIP Router Configuration Mode.
- RIP automatically summarizes routing updates. You can disable this by using the *no auto-summary* command.
- RIP uses hop count as its metric.

### **How does RIP Work?**

With RIP, a router sends its full routing table to all other connected routers every 30 seconds. Triggered updates can also occur if a router goes down before the 30-second timer has expired.

RIP performs "routing by rumour" and is more prone to loops than other routing protocols. That's because a RIP router sends its entire routing table to every other router. All other routers do the same, and because there's no real neighbour relationship or calculation of routes, the routers have little firsthand knowledge of available networks. Because of this, it can turn out like the old telephone game you used to play as a kid. At the end of the game, the resulting message is usually quite different from the original. This type of

problem can affect RIP because, unlike OSPF (a link-state protocol), RIP routers don't calculate their own routes; they must trust their neighbour's routes.

### **What's New in RIP Version 2?**

*RIPv2 boasts the following enhancements:*

- Support for variable length subnet masks (VLSM) (Because of this, RIP doesn't assume that all networks are classful.)
- Multicast routing updates
- Authentication with an encrypted password for routing updates.

### **How do I Configure RIP?**

Unlike some other routing protocols, RIP doesn't use any kind of autonomous system numbers to identify areas of the network that are under a single administrative domain. Because of this, entering RIP Routing Configuration Mode is very simple. Here's an example: Router(config)# router rip Router(config-router)# Once in RIP Configuration Mode, you need to specify that you're using RIPv2. To do so, enter the *ver 2* command at the config-router prompt. Here's an example:

```
Router(config-router)# ver 2
```

The most common network administrator's task is to specify which networks RIP will advertise and listen on. You can define this using the *network* command. Here's an example:

```
Router(config-router)# network 10.0.0.0
```

The 10.0.0.0 parameter is the network IP address covering any interfaces that you want RIP to perform the following three functions on:

- Send out routing updates.
- Listen for incoming routing updates.
- Include the network that the interface is on in any routing updates sent from the router.

If you would prefer not to send out advertisements on an interface, use the *passive interface* command.

---

## **OPEN SHORTEST PATH FIRST (OSPF)**

---

Open Shortest Path First (OSPF) is a link state routing protocol which was first defined as version 2 in RFC 2328. It is used to allow routers to dynamically learn routes from other routers and to advertise routes to other routers. Advertisements containing routes are referred to as Link State Advertisements (LSAs) in OSPF. OSPF router keeps track of the *state* of all the various network connections (*links*) between itself and a network it is trying to send data to. This makes it a *link-state* routing protocol. OSPF supports the use of classless IP address ranges and is very efficient. OSPF uses areas to organize a network into a hierarchal structure; it summarizes route information to reduce the number of advertised routes and thereby reduce network load and uses a designated router (elected via a process that is part of OSPF)



to reduce the quantity and frequency of Link State Advertisements.

OSPF does require the router have a more powerful processor and more memory than other routing protocols.

OSPF selects the best routes by finding the lowest cost paths to a destination. All router interfaces (links) are given a cost. The cost of a route is equal to the sum of all the costs configured on all the outbound links between the router and the destination network, plus the cost configured on the interface that OSPF received the Link State Advertisement on.

This tutorial will focus on explaining the basic components of OSPF, the operation of OSPF, basic configuration of OSPF and finally close with troubleshooting techniques used to verify correct OSPF configuration and operation.

## **OSPF ROUTER TYPES**

- Internal Router
- Backbone Router
- Area Border Router (ABR)
- Autonomous System Boundary Router (ASBR)
- Designated Router (DR)
- Backup Designated Router (BDR).

In this part of our OSPF tutorial, when speaking of an *OSPF router*, we are speaking of the OSPF routing process running on a given routing device. OSPF routers serve in various roles depending upon where they are located and which areas they participate in.

## **Internal Routers**

An internal router connects only to one OSPF area. All of its interfaces connect to the area in which it is located and does not connect to any other area.

If a router connects to more than one area, it will be one of the following types of routers.

## **Backbone Routers**

Backbone routers have one or more interfaces in Area 0 (the backbone area).

## **Area Border Router (ABR)**

A router that connects more than one area is called an area border router or ABR. Usually an ABR is used to connect non-backbone areas to the backbone. If OSPF virtual links are used an ABR will also be used to connect the area using the virtual link to another non-backbone area.

## **Autonomous System Boundary Router (ASBR)**

If the router connects the OSPF Autonomous System to another Autonomous System, it is called an Autonomous System Boundary Router (ASBR).

OSPF elects two or more routers to manage the Link State Advertisements:

## **Designated Router (DR)**

Every OSPF area will have a designated router and a backup designated router. The Designated Router (DR) is the router to which all other routers within an area send

their Link State Advertisements. The Designated Router will keep track of all link state updates and make sure the LSAs are flooded to the rest of the network using Reliable Multicast transport.

### **Backup Designated Router (BDR)**

The election process which determines the Designated Router will also elect a Backup Designated Router (BDR). The BDR takes over from the DR when the DR fails.

### **OSPF AREAS**

OSPF areas are used to impose a hierarchical structure to the flow of data over the network. A network using OSPF will always have at least one area and if there is more than one area, one of the two areas must be the backbone area. OSPF has only 2 levels to its hierarchy, the backbone, and all other areas attached to it.

Areas are used to group routers into manageable groups that exchange routing information locally, but summarize that routing information when advertising the routes externally. A standard OSPF network looks something like a big bubble (the backbone area) with a lot of smaller bubbles (stub areas) attached directly to it. Area Border Routers (ABR) are used to connect the areas. Each area will elect a designated router (DR) and a backup designated router (BDR) to assist in flooding Link State Advertisements (LSAs) throughout the area.

## **Backbone (Area 0)**

The backbone is the first area you should always build in any network using OSPF and the backbone is always Area 0 (zero). All areas are connected directly to the OSPF backbone area. When designing an OSPF backbone area, you should make sure there is little or no possibility of the backbone area being split into two or more parts by a router or link failure. If the OSPF backbone is split due to hardware failures or access lists, sizeable areas of the network will become unreachable.

## **Totally Stub Area**

A totally stubby area is only connected to the backbone area. A totally stubby/totally stub area does not advertise the routes it knows. It does not send any Link State Advertisements. The only route a totally stub area receives is the default route from an external area, which must be the backbone area. This default route allows the totally stub area to communicate with the rest of the network.

## **Stub Area**

Stub areas are connected only to the backbone area. Stub areas do not receive routes from outside the autonomous system, but do receive the routes from within the autonomous system, even if the route comes from another area.

## **Not-So-Stubby (NSSA)**

Frequently, it is advisable to use a separate network to connect the internal enterprise network to the Internet. OSPF

makes provisions for placing an Autonomous System Boundary Router (ASBR) within a non-backbone area. In this case, the stub area must learn routes from outside the OSPF autonomous system.

Thus, a new type of LSA was required—the Type 7 LSA. Type 7 LSA's are created by the Autonomous System Boundary Router and forwarded via the stub area's border router (ABR) to the backbone. This allows the other areas to learn routes that are external to the OSPF routing domain.

## **VIRTUAL LINKS**

Virtual links are used when you have a network that must be connected to an existing OSPF system, but cannot be physically connected directly to the routers in the OSPF backbone area. You can configure an OSPF virtual link from the area to a backbone router, creating a virtual direct connection to the backbone area. This virtual link acts as a tunnel which forwards LSAs to the backbone via a second intermediate area.

## **OSPF Operation**

- OSPF Router ID
- Designated Router Election
- OSPF Link States
- OSPF Timers
- Neighbour Discovery
- Forming Neighbour Adjacencies

- Link State Advertisements
- Route Summarization
- Shortest Path Algorithm.

### **OSPF Router ID**

The OSPF Router ID identifies a specific router in the OSPF topology. The Router ID is either a) the IP address assigned to the loopback interface, or b) the IP address of the interface with the highest IP address number. Using the loopback interface makes a more stable OSPF environment as the loopback interface is always up, unlike physical interfaces, which can fail. If a physical interface fails, the OSPF router ID may change, triggering router election and link state advertisement flooding.

### **Designated Router Election**

Once the designated router has been chosen, it remains the designated router until it fails.

# 5

---

## Network Cabling

---

Cable is the medium through which information usually moves from one network device to another. There are several types of cable which are commonly used with LANs. In some cases, a network will utilize only one type of cable, other networks will use a variety of cable types.

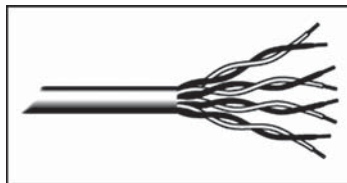
The type of cable chosen for a network is related to the network's topology, protocol, and size. Understanding the characteristics of different types of cable and how they relate to other aspects of a network is necessary for the development of a successful network.

The following sections discuss the types of cables used in networks and other related topics.

- Unshielded Twisted Pair (UTP) Cable
- Shielded Twisted Pair (STP) Cable

- Coaxial Cable
- Fibre Optic Cable
- Cable Installation Guides
- Wireless LANs
- Unshielded Twisted Pair (UTP) Cable.

Twisted pair cabling comes in two varieties: shielded and unshielded. Unshielded twisted pair (UTP) is the most popular and is generally the best option for school networks.



**Figure:** Unshielded Twisted Pair.

**Categories of Unshielded Twisted Pair**

<b>Category</b>	<b>Speed</b>	<b>Use</b>
1	1 Mbps	Voice Only (Telephone Wire)
2	4 Mbps	LocalTalk & Telephone (Rarely used)
3	16 Mbps	10BaseT Ethernet
4	20 Mbps	Token Ring (Rarely used)
5	100 Mbps (2 pair)	100BaseT Ethernet
	1000 Mbps (4 pair)	Gigabit Ethernet
5e	1,000 Mbps	Gigabit Ethernet
6	10,000 Mbps	Gigabit Ethernet

The quality of UTP may vary from telephone-grade wire to extremely high-speed cable. The cable has four pairs of wires inside the jacket. Each pair is twisted with a different number of twists per inch to help eliminate interference from adjacent pairs and other electrical devices. The tighter the twisting, the higher the supported transmission rate and the greater the cost per foot. The EIA/TIA (Electronic Industry



Association/Telecommunication Industry Association) has established standards of UTP and rated six categories of wire (additional categories are emerging).

### **UNSHIELDED TWISTED PAIR CONNECTOR**

The standard connector for unshielded twisted pair cabling is an RJ-45 connector. This is a plastic connector that looks like a large telephone-style connector. A slot allows the RJ-45 to be inserted only one way. RJ stands for Registered Jack, implying that the connector follows a standard borrowed from the telephone industry. This standard designates which wire goes with each pin inside the connector.



**Figure:** RJ-45 Connector.

### **SHIELDED TWISTED PAIR (STP) CABLE**

Although UTP cable is the least expensive cable, it may be susceptible to radio and electrical frequency interference (it should not be too close to electric motors, fluorescent lights, etc.). If you must place cable in environments with lots of potential interference, or if you must place cable in extremely sensitive environments that may be susceptible to the electrical current in the UTP, shielded twisted pair may be

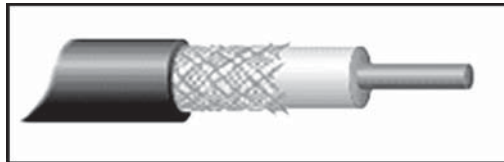
the solution. Shielded cables can also help to extend the maximum distance of the cables. Shielded twisted pair cable is available in three different configurations:

- Each pair of wires is individually shielded with foil.
- There is a foil or braid shield inside the jacket covering all wires (as a group).
- There is a shield around each individual pair, as well as around the entire group of wires (referred to as double shield twisted pair).

## **COAXIAL CABLE**

Coaxial cabling has a single copper conductor at its centre. A plastic layer provides insulation between the centre conductor and a braided metal shield.

The metal shield helps to block any outside interference from fluorescent lights, motors, and other computers.



**Figure:** Coaxial Cable.

Although coaxial cabling is difficult to install, it is highly resistant to signal interference. In addition, it can support greater cable lengths between network devices than twisted pair cable. The two types of coaxial cabling are thick coaxial and thin coaxial.

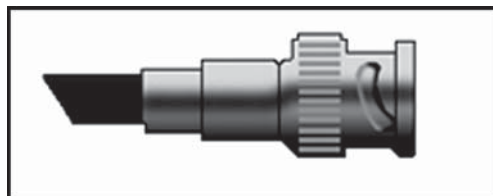
Thin coaxial cable is also referred to as thinnet. 10Base2 refers to the specifications for thin coaxial cable carrying

Ethernet signals. The 2 refers to the approximate maximum segment length being 200 meters. In actual fact the maximum segment length is 185 meters. Thin coaxial cable has been popular in school networks, especially linear bus networks.

Thick coaxial cable is also referred to as thicknet. 10Base5 refers to the specifications for thick coaxial cable carrying Ethernet signals. The 5 refers to the maximum segment length being 500 meters. Thick coaxial cable has an extra protective plastic cover that helps keep moisture away from the centre conductor. This makes thick coaxial a great choice when running longer lengths in a linear bus network. One disadvantage of thick coaxial is that it does not bend easily and is difficult to install.

## **COAXIAL CABLE CONNECTORS**

The most common type of connector used with coaxial cables is the Bayone-Neill-Concelman (BNC) connector. Different types of adapters are available for BNC connectors, including a T-connector, barrel connector, and terminator. Connectors on the cable are the weakest points in any network. To help avoid problems with your network, always use the BNC connectors that crimp, rather screw, onto the cable.



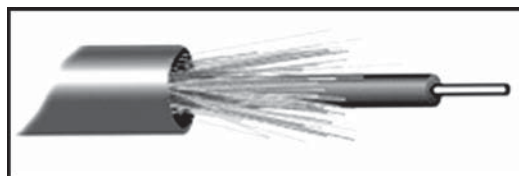
**Figure:** BNC Connector.

## **FIBRE OPTIC CABLE**

Fibre optic cabling consists of a centre glass core surrounded by several layers of protective materials. It transmits light rather than electronic signals eliminating the problem of electrical interference. This makes it ideal for certain environments that contain a large amount of electrical interference. It has also made it the standard for connecting networks between buildings, due to its immunity to the effects of moisture and lightning.

Fibre optic cable has the ability to transmit signals over much longer distances than coaxial and twisted pair. It also has the capability to carry information at vastly greater speeds. This capacity broadens communication possibilities to include services such as video conferencing and interactive services. The cost of fibre optic cabling is comparable to copper cabling; however, it is more difficult to install and modify. 10BaseF refers to the specifications for fibre optic cable carrying Ethernet signals.

The centre core of fibre cables is made from glass or plastic fibres. A plastic coating then cushions the fibre centre, and kevlar fibres help to strengthen the cables and prevent breakage. The outer insulating jacket made of teflon or PVC.



**Figure:** Fibre Optic Cable.

There are two common types of fibre cables — single mode and multimode. Multimode cable has a larger diameter; however, both cables provide high bandwidth at high speeds. Single mode can provide more distance, but it is more expensive.

<b>Specification</b>	<b>Cable Type</b>
10BaseT	Unshielded Twisted Pair
10Base2	Thin Coaxial
10Base5	Thick Coaxial
100BaseT	Unshielded Twisted Pair
100BaseFX	Fibre Optic
100BaseBX	Single mode Fibre
100BaseSX	Multimode Fibre
1000BaseT	Unshielded Twisted Pair
1000BaseFX	Fibre Optic
1000BaseBX	Single mode Fibre
1000BaseSX	Multimode Fibre

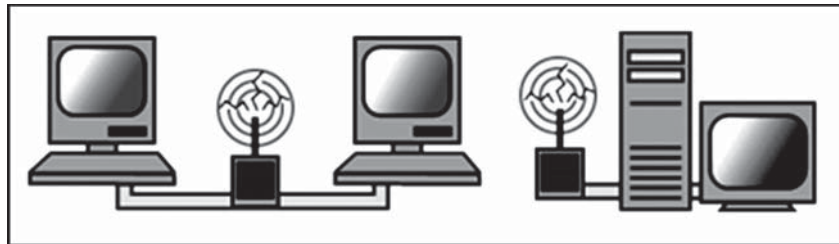
## **INSTALLING CABLE-SOME GUIDELINES**

When running cable, it is best to follow a few simple rules:

- Always use more cable than you need. Leave plenty of slack.
- Test every part of a network as you install it. Even if it is brand new, it may have problems that will be difficult to isolate later.
- Stay at least 3 feet away from fluorescent light boxes and other sources of electrical interference.
- If it is necessary to run cable across the floor, cover the cable with cable protectors.
- Label both ends of each cable.
- Use cable ties (not tape) to keep cables in the same location together.

## **WIRELESS LANS**

More and more networks are operating without cables, in the wireless mode. Wireless LANs use high frequency radio signals, infrared light beams, or lasers to communicate between the workstations, servers, or hubs. Each workstation and file server on a wireless network has some sort of transceiver/antenna to send and receive the data. Information is relayed between transceivers as if they were physically connected. For longer distance, wireless communications can also take place through cellular telephone technology, microwave transmission, or by satellite.



Wireless networks are great for allowing laptop computers, portable devices, or remote computers to connect to the LAN. Wireless networks are also beneficial in older buildings where it may be difficult or impossible to install cables.

The two most common types of infrared communications used in schools are line-of-sight and scattered broadcast. Line-of-sight communication means that there must be an unblocked direct line between the workstation and the transceiver. If a person walks within the line-of-sight while there is a transmission, the information would need to be sent again. This kind of obstruction can slow down the

wireless network. Scattered infrared communication is a broadcast of infrared transmissions sent out in multiple directions that bounces off walls and ceilings until it eventually hits the receiver. Networking communications with laser are virtually the same as line-of-sight infrared networks.

## **WIRELESS STANDARDS AND SPEEDS**

The Wi-Fi Alliance is a global, non-profit organization that helps to ensure standards and interoperability for wireless networks, and wireless networks are often referred to as Wi-Fi (Wireless Fidelity). The original Wi-Fi standard (IEEE 802.11) was adopted in 1997. Since then many variations have emerged (and will continue to emerge). Wi-Fi networks use the Ethernet protocol.

<b>Standard</b>	<b>Max Speed</b>	<b>Typical Range</b>
802.11a	54 Mbps	150 feet
802.11b	11 Mbps	300 feet
802.11g	54 Mbps	300 feet
802.11n	100 Mbps	300+ feet

## **WIRELESS SECURITY**

Wireless networks are much more susceptible to unauthorized use than cabled networks. Wireless network devices use radio waves to communicate with each other. The greatest vulnerability to the network is that rogue machines can “eves-drop” on the radio wave communications. Unencrypted information transmitted can be monitored by a third-party, which, with the right tools (free to download), could quickly gain access to your entire network, steal valuable

passwords to local servers and online services, alter or destroy data, and/or access personal and confidential information stored in your network servers. To minimize the possibility of this, all modern access points and devices have configuration options to encrypt transmissions. These encryption methodologies are still evolving, as are the tools used by malicious hackers, so always use the strongest encryption available in your access point and connecting devices.

A NOTE ON ENCRYPTION: As of this writing WEP (Wired Equivalent Privacy) encryption can be easily hacked with readily-available free tools which circulate the internet. WPA and WPA2 (WiFi Protected Access versions 1 and 2) are much better at protecting information, but using weak passwords or passphrases when enabling these encryptions may allow them to be easily hacked. If your network is running WEP, you must be very careful about your use of sensitive passwords or other data. Three basic techniques are used to protect networks from unauthorized wireless use. Use any and all of these techniques when setting up your wireless access points:

### **Encryption**

Enable the strongest encryption supported by the devices you will be connecting to the network. Use strong passwords (strong passwords are generally defined as passwords containing symbols, numbers, and mixed case letters, at least 14 characters long).



## **Isolation**

Use a wireless router that places all wireless connections on a subnet independent of the primary private network. This protects your private network data from pass-through internet traffic.

## **Hidden SSID**

Every access point has a Service Set Identifier (SSID) that by default is broadcast to client devices so that the access point can be found. By disabling this feature, standard client connection software won't be able to "see" the access point. However, the eaves-dropping programs discussed previously can easily find these access points, so this alone does little more than keep the access point name out of sight for casual wireless users.

### *Advantages of wireless networks:*

- **Mobility-** With a laptop computer or mobile device, access can be available throughout a school, at the mall, on an airplane, etc. More and more businesses are also offering free WiFi access ("Hot spots").
- **Fast setup-** If your computer has a wireless adapter, locating a wireless network can be as simple as clicking "Connect to a Network" — in some cases, you will connect automatically to networks within range.
- **Cost-** Setting up a wireless network can be much more cost effective than buying and installing cables.

### *Networking Hardware*

- **Expandability-** Adding new computers to a wireless network is as easy as turning the computer on (as long as you do not exceed the maximum number of devices).

#### *Disadvantages of Wireless Networks:*

- **Security-** Be careful. Be vigilant. Protect your sensitive data with backups, isolated private networks, strong encryption and passwords, and monitor network access traffic to and from your wireless network.
- **Interference-** Because wireless networks use radio signals and similar techniques for transmission, they are susceptible to interference from lights and electronic devices.
- **Inconsistent connections-** How many times have you hears “Wait a minute, I just lost my connection?” Because of the interference caused by electrical devices and/or items blocking the path of transmission, wireless connections are not nearly as stable as those through a dedicated cable.
- **Speed-** The transmission speed of wireless networks is improving; however, faster options (such as gigabit Ethernet) are available via cables. If you are only using wireless for internet access, the actual internet connection for your home or school is generally slower than the wireless network devices, so that connection

### *Networking Hardware*

is the bottleneck. If you are also moving large amounts of data around a private network, a cabled connection will enable that work to proceed much faster.

# 6

---

## Network Capacity

---

The fundamental problem is that all network resources are limited, including router processing time and link throughput.

*For example:*

- Today's (2006) Wireless LAN effective bandwidth throughput (15-100Mbit/s) is easily filled by a single personal computer.
- Even on fast computer networks (*e.g.* 1 Gbit), the backbone can easily be congested by a few servers and client PCs.
- Because P2P scales very well, file transmissions by P2P have no problem filling and will fill an uplink or some other network bottleneck, particularly when nearby peers are preferred over distant peers.

- Denial of service attacks by botnets are capable of filling even the largest Internet backbone network links (40 Gbit/s as of 2007), generating large-scale network congestion.

## **CONGESTIVE COLLAPSE**

Congestive collapse (or congestion collapse) is a condition which a packet switched computer network can reach, when little or no useful communication is happening due to congestion. Congestion collapse generally occurs at choke points in the network, where the total incoming traffic to a node exceeds the outgoing bandwidth. Connection points between a local area network and a wide area network are the most likely choke points.

When a network is in such a condition, it has settled (under overload) into a stable state where traffic demand is high but little useful throughput is available, and there are high levels of packet delay and loss (caused by routers discarding packets because their output queues are too full) and general quality of service is extremely poor.

## **History**

Congestion collapse was identified as a possible problem as far back as 1984 (RFC 896, dated 6 January). It was first observed on the early Internet in October 1986, when the NSFnet phase-I backbone dropped three orders of magnitude from its capacity of 32 kbit/s to 40 bit/s, and this continued

to occur until end nodes started implementing Van Jacobson's congestion control between 1987 and 1988.

### **Cause**

When more packets were sent than could be handled by intermediate routers, the intermediate routers discarded many packets, expecting the end points of the network to retransmit the information. However, early TCP implementations had very bad retransmission behaviour. When this packet loss occurred, the end points sent *extra* packets that repeated the information lost; doubling the data rate sent, exactly the opposite of what should be done during congestion. This pushed the entire network into a 'congestion collapse' where most packets were lost and the resultant throughput was negligible.

### **Congestion Control**

Congestion control concerns controlling traffic entry into a telecommunications network, so as to avoid congestive collapse by attempting to avoid oversubscription of any of the processing or link capabilities of the intermediate nodes and networks and taking resource reducing steps, such as reducing the rate of sending packets. It should not be confused with flow control, which prevents the sender from overwhelming the receiver.

### **Theory of Congestion Control**

The modern theory of congestion control was pioneered by Frank Kelly, who applied microeconomic theory and

convex optimization theory to describe how individuals controlling their own rates can interact to achieve an “optimal” network-wide rate allocation.

Examples of “optimal” rate allocation are max-min fair allocation and Kelly’s suggestion of proportional fair allocation, although many others are possible. The mathematical expression for optimal rate allocation is as follows. Let  $x_i$  be the rate of flow  $i$ ,  $C_l$  be the capacity of link  $l$ , and  $r_{li}$  be 1 if flow  $i$  uses link  $l$  and 0 otherwise. Let  $x$ , and  $R$  be the corresponding vectors and matrix. Let  $U(x)$  be an increasing, strictly convex function, called the utility, which measures how much benefit a user obtains by transmitting at rate  $x$ . The optimal rate allocation then satisfies

$$\max_x \sum_i U(x_i)$$

such that  $Rx \leq c$ .

The Lagrange dual of this problem decouples, so that each flow sets its own rate, based only on a “price” signalled by the network. Each link capacity imposes a constraint, which gives rise to a Lagrange multiplier,  $p_l$ . The sum of these Lagrange multipliers,  $y_i = \sum_l p_l r_{li}$ , is the price to which the flow responds.

Congestion control then becomes a distributed optimisation algorithm for solving the above problem. Many current congestion control algorithms can be modelled in this framework, with  $p_l$  being either the loss probability or the queueing delay at link  $l$ .

A major weakness of this model is that it assumes all flows observe the same price, while sliding window flow control causes “burstiness” which causes different flows to observe different loss or delay at a given link.

### **Classification of Congestion Control Algorithms**

*There are many ways to classify congestion control algorithms:*

- By the type and amount of feedback received from the network: Loss; delay; single-bit or multi-bit explicit signals
- By incremental deployability on the current Internet: Only sender needs modification; sender and receiver need modification; only router needs modification; sender, receiver and routers need modification.
- By the aspect of performance it aims to improve: high bandwidth-delay product networks; lossy links; fairness; advantage to short flows; variable-rate links
- By the fairness criterion it uses: max-min, proportional, “minimum potential delay”.

### **AVOIDANCE**

*The prevention of network congestion and collapse requires two major components:*

- A mechanism in routers to reorder or drop packets under overload,



- End-to-end flow control mechanisms designed into the end points which respond to congestion and behave appropriately.

The correct end point behaviour is usually still to repeat dropped information, but progressively slow the rate that information is repeated. Provided all end points do this, the congestion lifts and good use of the network occurs, and the end points all get a fair share of the available bandwidth. Other strategies such as slow-start ensure that new connections don't overwhelm the router before the congestion detection can kick in.

The most common router mechanisms used to prevent congestive collapses are fair queueing and other scheduling algorithms, and random early detection, or RED, where packets are randomly dropped proactively triggering the end points to slow transmission before congestion collapse actually occurs. Fair queueing is most useful in routers at choke points with a small number of connections passing through them. Larger routers must rely on RED.

Some end-to-end protocols are better behaved under congested conditions than others. TCP is perhaps the best behaved. The first TCP implementations to handle congestion well were developed in 1984, but it was not until Van Jacobson's inclusion of an open source solution in the Berkeley Standard Distribution UNIX ("BSD") in 1988 that good TCP implementations became widespread.

UDP does not, in itself, have any congestion control mechanism. Protocols built atop UDP must handle congestion in their own way. Protocols atop UDP which transmit at a fixed rate, independent of congestion, can be troublesome. Real-time streaming protocols, including many Voice over IP protocols, have this property. Thus, special measures, such as quality-of-service routing, must be taken to keep packets from being dropped from streams.

In general, congestion in pure datagram networks must be kept out at the periphery of the network, where the mechanisms described above can handle it. Congestion in the Internet backbone is very difficult to deal with. Fortunately, cheap fibre-optic lines have reduced costs in the Internet backbone. The backbone can thus be provisioned with enough bandwidth to keep congestion at the periphery.

### **Practical Network Congestion Avoidance**

Implementations of connection-oriented protocols, such as the widely-used TCP protocol, generally watch for packet errors, losses, or delays in order to adjust the transmit speed. There are many different network congestion avoidance processes, since there are a number of different trade-offs available.

### **TCP/IP Congestion Avoidance**

Transmission Control Protocol (TCP) uses a network congestion avoidance algorithm that includes various aspects

of an additive increase/multiplicative decrease (AIMD) scheme, with other schemes such as slow-start in order to achieve congestion avoidance. The TCP congestion avoidance algorithm is the primary basis for congestion control in the Internet.

### **TCP TAHOE AND RENO**

To avoid congestion collapse, TCP uses a multifaceted congestion control strategy. For each connection, TCP maintains a *congestion window*, limiting the total number of unacknowledged packets that may be in transit end-to-end. This is somewhat analogous to TCP's sliding window used for flow control. TCP uses a mechanism called *slow start* to increase the congestion window after a connection is initialized and after a timeout. It starts with a window of two times the maximum segment size (MSS). Although the initial rate is low, the rate of increase is very rapid: for every packet acknowledged, the congestion window increases by 1 MSS so that the congestion window effectively doubles for every round trip time (RTT). When the congestion window exceeds a threshold *ssthresh* the algorithm enters a new state, called congestion avoidance. In some implementations (*e.g.*, Linux), the initial *ssthresh* is large, and so the first slow start usually ends after a loss. However, *ssthresh* is updated at the end of each slow start, and will often affect subsequent slow starts triggered by timeouts.

Congestion avoidance: As long as non-duplicate ACKs are received, the congestion window is additively increased by one MSS every round trip time.

When a packet is lost, the likelihood of duplicate ACKs being received is very high (it's possible though unlikely that the stream just underwent extreme packet reordering, which would also prompt duplicate ACKs). The behaviour of Tahoe and Reno differ in how they detect and react to packet loss:

- *Tahoe*: Triple duplicate ACKS are treated the same as a timeout. Tahoe will perform “fast retransmit”, reduce congestion window to 1 MSS, and reset to slow-start state.
- *Reno*: If three duplicate ACKs are received (*i.e.*, four ACKs acknowledging the same packet, which are not piggybacked on data, and do not change the receiver's advertised window), Reno will halve the congestion window, perform a fast retransmit, and enter a phase called Fast Recovery. If an ACK times out, slow start is used as it is with Tahoe.

**Fast Recovery:** (Reno Only) In this state, TCP retransmits the missing packet that was signaled by three duplicate ACKs, and waits for an acknowledgment of the entire transmit window before returning to congestion avoidance. If there is no acknowledgment, TCP Reno experiences a timeout and enters the slow-start state.

Both algorithms reduce congestion window to 1 MSS on a timeout event.

### **TCP VEGAS**

Until the mid 1990s, all of TCP's set timeouts and measured round-trip delays were based upon only the last transmitted packet in the transmit buffer. University of Arizona researchers Larry Peterson and Lawrence Brakmo introduced TCP Vegas, in which timeouts were set and round-trip delays were measured for every packet in the transmit buffer. In addition, TCP Vegas uses additive increases in the congestion window. This variant was not widely deployed outside Peterson's laboratory. In a comparison study of various TCP congestion control algorithms, TCP Vegas appeared to be the smoothest followed by TCP CUBIC. However, TCP Vegas was deployed as default congestion control method for DD-WRT firmwares v24 SP2.

### **TCP NEW RENO**

TCP New Reno, defined by RFC 3782, improves retransmission during the fast recovery phase of TCP Reno. During fast recovery, for every duplicate ACK that is returned to TCP New Reno, a new unsent packet from the end of the congestion window is sent, to keep the transmit window full. For every ACK that makes partial progress in the sequence space, the sender assumes that the ACK points to a new hole, and the next packet beyond the ACKed sequence

number is sent. Because the timeout timer is reset whenever there is progress in the transmit buffer, this allows New Reno to fill large holes, or multiple holes, in the sequence space—much like TCP SACK. Because New Reno can send new packets at the end of the congestion window during fast recovery, high throughput is maintained during the hole-filling process, even when there are multiple holes, of multiple packets each. When TCP enters fast recovery it records the highest outstanding unacknowledged packet sequence number. When this sequence number is acknowledged, TCP returns to the congestion avoidance state. A problem occurs with New Reno when there are no packet losses but instead, packets are reordered by more than 3 packet sequence numbers. When this happens, New Reno mistakenly enters fast recovery, but when the reordered packet is delivered, ACK sequence-number progress occurs and from there until the end of fast recovery, every bit of sequence-number progress produces a duplicate and needless retransmission that is immediately ACKed.

New Reno performs as well as SACK at low packet error rates, and substantially outperforms Reno at high error rates.

### **TCP HYBLA**

TCP Hybla aims to eliminate penalization of TCP connections that incorporate a high-latency terrestrial or satellite radio link, due to their longer round trip times. It stems from an analytical evaluation of the congestion window

dynamics, which suggests the necessary modifications to remove the performance dependence on RTT.

### **TCP BIC**

Binary Increase Congestion control is an implementation of TCP with an optimized congestion control algorithm for high speed networks with high latency (called LFN, long fat networks, in RFC 1072). BIC is used by default in Linux kernels 2.6.8 through 2.6.18.

### **TCP CUBIC**

CUBIC is a less aggressive and more systematic derivative of BIC, in which the window is a cubic function of time since the last congestion event, with the inflection point set to the window prior to the event. CUBIC is used by default in Linux kernels since version 2.6.19.

### **Compound TCP**

Compound TCP is a Microsoft implementation of TCP which maintains two different congestion windows simultaneously, with the goal of achieving good performance on LFNs while not impairing fairness. It has been widely deployed with Microsoft Windows Vista and Windows Server 2008 and has been ported to older Microsoft Windows versions as well as Linux.

Compound TCP is a TCP algorithm designed by Song et. al. for use on windows operating systems. The algorithm is designed to better use available bandwidth on high-speed

links while still maintaining intra-protocol and inter-protocol fairness.

The algorithm essentially builds on previous work by combining aggressive algorithms for increasing link usage on idle links with delay information-based algorithms that pro-actively control the TCP window size to reduce congestion before loss occurs.

### **Previous Work**

Existing TCP algorithms such as TCP Reno and Tahoe are very conservative about bandwidth increase in order to maintain TCP fairness. The authors show that, during the congestion avoidance phase for links with sufficiently high latency and available bandwidth, standard TCP is too conservative and typical loss rates too high for the link to ever be fully utilized. This is because a long delay high-speed network requires a window size on the order of thousands of segments to maximize link usage.

One way to improve link use efficiency is to increase the aggressiveness of existing TCP algorithms. In one example algorithm, STCP, multiplicative increase replaces additive increase. In this algorithm CWND increases by 1% per ACK and only reduces to 87.5% on a packet loss. A less aggressive example algorithm, HSTCP, uses AIMD with varying increase-decrease values based on the size of CWND.

Both of these algorithms have been shown to exhibit poor intra-protocol fairness when RTT of competing links varies.



Furthermore, the authors conduct experiments to show that these algorithms steal bandwidth from standard TCP flows and therefore exhibit poor inter-protocol fairness.

Another branch of algorithms to improve link efficiency descend from the work of TCP Vegas. FAST TCP, in particular, is a scaled version of Vegas that attempts to maintain a fixed buffer size in the bottleneck queue by varying its sending rate based on measured versus expected RTT values. If the bottleneck RTT is far less than expected then multiplicative increase is used. Otherwise the algorithm switches to additive increase or decrease modes. While these algorithms are overall much more fair and efficient, they fail to perform well against the loss-based congestion control methods of standard TCP because they are not aggressive enough in highly-congested scenarios.

Lastly, TCP Africa is mentioned as the algorithm most similar to that proposed in the paper in that it also combines delay information with a loss-based approach. However, since TCP Africa never decreases its window size based on delay information, it is still fundamentally a loss-based approach with the same inherent limitations.

## **COMPOUND TCP ALGORITHM**

Compound TCP works by maintaining two windows: the standard CWND and a new DWND or delay window. The overall window size is the sum of CWND and DWND. When congestion is lower than expected, CTCP is tuned to operate

like HSTCP because HSTCP has been shown to be aggressive enough and was already made into an experimental standard. Furthermore, in cases where the network is poorly buffered, CTCP's worst case behaviour will be similar to HSTCP's.

When congestion passes a particular threshold, the delay-based component begins to shrink based on a configurable parameter, and similarly, when a packet loss occurs, DWND multiplicatively shrinks based on another configurable parameter. Lastly, if a retransmission timeout occurs DWND is reset to 0 and the sender is placed in slow start state. The delay-based component is then only re-enabled when the algorithm leaves slow start state.

### **Analysis and Conclusions**

The authors provide a mathematical analysis to show that CTCP is efficient, maintains good intra-protocol fairness across varying RTTs, and maintains good inter-protocol fairness with other TCP algorithms. They then detail a performance evaluation on a network by using DummyNet to simulate different network conditions and by modifying the TCP implementation on all workstations to be able to dynamically configure their congestion control algorithm via a socket option. Each experiment conducted on the network lasts 300 seconds and results are presented as averages over 5 runs of a test.

Using DummyNet to generate random packet losses, the authors first show that in a long-delay high-bandwidth

environment that CTCP can still effectively use available bandwidth even in situations involving high throughput from a competing UDP connection.

In a second set of experiments, the authors run 8 TCP flows simultaneously and measure bandwidth stolen from the regular TCP flows under varying loss conditions. In these experiments CTCP steals far less bandwidth than HSTCP. Finally, the authors run 4 competing flows with RTTs significantly different from each other. In these experiments CTCP shows significantly better RTT fairness than regular TCP and HSTCP.

---

## **TCP CONGESTION HANDLING AND CONGESTION AVOIDANCE ALGORITHMS**

---

By changing the window size that a device advertises to a peer on a TCP connection, the device can increase or decrease the rate at which its peer sends it data. This is how the TCP sliding window system implements flow control between the two connected devices. We've seen how this works in the last few topics, including the changes required to the "basic" mechanism to ensure performance remains high by reducing the number of small segments sent.

### **WHY TCP MUST MONITOR AND DEAL WITH INTERNETWORK CONGESTION**

Flow control is a very important part of regulating the transmission of data between devices, but it is limited in the

following respect: it only considers what is going on within each of the devices on the connection, and *not* what is happening in devices between them. In fact, this “self-centeredness” is symptomatic of architectural layering. Since we are dealing with how TCP works between a typical server and client at layer four, we don’t worry about how data gets between them; that’s the job of the Internet Protocol at layer three.

In practice, what is going on at layer three can be quite important. Considered from an abstract point of view, our server and client may be connected “directly” using TCP, but all the segments we transmit are carried across an internetwork of networks and routers between them. These networks and routers are also carrying data from many other connections and higher-layer protocols. If the internetwork becomes very busy, the speed at which segments are carried between the endpoints of our connection will be reduced, and they could even be dropped. This is called *congestion*. Again, at the TCP level, there is no way to directly comprehend what is causing congestion or why. It is perceived simply as inefficiencies in moving data from one device to another, through the need for some segments to be retransmitted. However, even though TCP is mostly oblivious of what is happening on the internetwork, it *must* be smart enough to understand how to deal with congestion and not exacerbate it.

Recall that each segment that is transmitted is placed on the retransmission queue with a retransmission timer. Now, suppose congestion dramatically increased on the internetwork, and there were no mechanisms in place to handle congestion.

Segments would be delayed or dropped, which would cause them to time out and be retransmitted. This would increase the amount of traffic on the internetwork between our client and server. Furthermore, there might be thousands of other TCP connections behaving similarly. Each would keep retransmitting more and more segments, increasing congestion further, leading to a vicious circle. Performance of the entire internetwork would decrease dramatically, resulting in a condition called *congestion collapse*.

The message is clear: TCP cannot just ignore what is happening on the internetwork between its connection endpoints. To this end, TCP includes several specific algorithms that are designed to respond to congestion, or avoid it in the first place. Many of these techniques can be considered, in a way, to be methods by which a TCP connection is made less “selfish”—that is, it tries to take into account the existence of other users of the internetwork over which it operates. While no single connection by itself can solve congestion of an entire internetwork, having all devices implement these measures collectively reduces congestion due to TCP. The first issue is that we need to

know when congestion is taking place. By definition, congestion means intermediate devices—routers—are overloaded. Routers respond to overloading by dropping datagrams. When these datagrams contain TCP segments, the segments don't reach their destination, and they are therefore left unacknowledged and will eventually expire and be retransmitted.

This means that when a device sends TCP segments and does not receive acknowledgments for them, it can be assumed that in most cases, they have been dropped by intermediate devices due to congestion. By detecting the rate at which segments are sent and not acknowledged, a TCP device can infer the level of congestion on the network between itself and its TCP connection peer.

## **ACTIVE QUEUE MANAGEMENT (AQM)**

### **Random Early Detection**

One solution is to use random early detection (RED) on the network equipment's port queue buffer. On network equipment ports with more than one queue buffer, weighted random early detection (WRED) could be used if available. RED indirectly signals to sender and receiver by deleting some packets, *e.g.* when the average queue buffer lengths are more than *e.g.* 50% (lower threshold) filled and deletes linearly more or cubical more packets, up to *e.g.* 100% (higher threshold). The average queue buffer lengths are computed over 1 second at a time.

## **Robust Random Early Detection (RRED)**

Robust Random Early Detection (RRED) algorithm was proposed to improve the TCP throughput against Denial-of-Service (DoS) attacks, particularly Low-rate Denial-of-Service (LDoS) attacks. Experiments have confirmed that the existing RED-like algorithms are notably vulnerable under Low-rate Denial-of-Service (LDoS) attacks due to the oscillating TCP queue size caused by the attacks. RRED algorithm can significantly improve the performance of TCP under Low-rate Denial-of-Service attacks.

## **Flowbased-RED/WRED**

Some network equipment are equipped with ports that can follow and measure each flow (flowbased-RED/WRED) and are hereby able to signal to a too big bandwidth flow according to some QoS policy. A policy could divide the bandwidth among all flows by some criteria.

## **IP ECN**

Another approach is to use IP ECN. ECN is only used when the two hosts signal that they want to use it. With this method, an ECN bit is used to signal that there is explicit congestion.

This is better than the indirect packet delete congestion notification performed by the RED/WRED algorithms, but it requires explicit support by both hosts to be effective. Some outdated or buggy network equipment drops packets with the ECN bit set, rather than ignoring the bit. More information

on the status of ECN including the version required for Cisco IOS, by Sally Floyd, one of the authors of ECN.

When a router receives a packet marked as ECN capable and anticipates (using RED) congestion, it will set an ECN-flag notifying the sender of congestion. The sender then ought to decrease its transmission bandwidth; *e.g.* by decreasing the tcp window size (sending rate) or by other means.

### **Cisco AQM: Dynamic Buffer Limiting (DBL)**

Cisco has taken a step further in their Catalyst 4000 series with engine IV and V. Engine IV and V has the possibility to classify all flows in “aggressive” (bad) and “adaptive” (good). It ensures that no flows fill the port queues for a long time. DBL can utilize IP ECN instead of packet-delete-signalling.

### **TCP Window Shaping**

Congestion avoidance can also efficiently be achieved by reducing the amount of traffic flowing into a network. When an application requests a large file, graphic or web page, it usually advertises a “window” of between 32K and 64K. This results in the server sending a full window of data (assuming the file is larger than the window). When there are many applications simultaneously requesting downloads, this data creates a congestion point at an upstream provider by flooding the queue much faster than it can be emptied. By using a device to reduce the window advertisement, the remote servers will send less data, thus reducing the congestion



and allowing traffic to flow more freely. This technique can reduce congestion in a network by a factor of 40.

## **SIDE EFFECTS OF CONGESTIVE COLLAPSE AVOIDANCE**

### **Radio Links**

The protocols that avoid congestive collapse are often based on the idea that data loss on the Internet is caused by congestion. This is true in nearly all cases; errors during transmission are rare on today's fibre based Internet. However, this causes WiFi, 3G or other networks with a radio layer to have poor throughput in some cases since wireless networks are susceptible to data loss due to interference. The TCP connections running over a radio based physical layer see the data loss and tend to believe that congestion is occurring when it isn't and erroneously reduce the data rate sent.

### **Short-lived Connections**

The slow-start protocol performs badly for short-lived connections. Older web browsers would create many consecutive short-lived connections to the web server, and would open and close the connection for each file requested. This kept most connections in the slow start mode, which resulted in poor response time.

To avoid this problem, modern browsers either open multiple connections simultaneously or reuse one connection

### *Networking Hardware*

for all files requested from a particular web server. However, the initial performance can be poor, and many connections never get out of the slow-start regime, significantly increasing latency.

# 7

---

## Network Management

---

Network management refers to the broad subject of managing computer networks. There exists a wide variety of software and hardware products that help network system administrators manage a network. Network management covers a wide area, including:

- *Security*: Ensuring that the network is protected from unauthorized users.
- *Performance*: Eliminating bottlenecks in the network.
- *Reliability*: Making sure the network is available to users and responding to hardware and software malfunctions.

Since the 21st century, with the popularization of computer and the development of computer technologies and communications technologies, network has been integrated

into daily life more and more extensively, and has become the pillar of today's information age. Network originates from the combination of computer and communications. The dependence of information society on network has made it very important for the network to be reliable itself, and has raised the bar on its management. To improve network management, and to spread information in a safe and swift manner. The computer network management system has become more and more important.

## **BASIC KNOWLEDGE OF COMPUTER NETWORK MANAGEMENT SYSTEM**

### **Definition of Computer Network Management System**

The computer network management system is the software system used to manage the network. The computer network management is to gather the static and dynamic operating information of the various parts of the network, to analyse and to process the information, so as to ensure the safe, reliable, efficient running of the network, to distribute the network resources reasonably, to allocate network load dynamically, to optimize the network performance, and to reduce the cost of network maintenance.

*The fundamental components of the network management system:*

- *Network Manager:* The network manager is the substantial part to implement the network

management. Being the core of the network system, he/she stays at the management workstation, performing the various complex tasks related to network management.

- *Management Agent*: The network management agent is the software module set in the network equipment, including UNIX workstation, network printers, and some other network equipment. It can acquire the information on the operating status of local equipment, its features and system allocation.
- *Management Information Base (MIB)*: The management information base is stored at the store of the managed target. As a dynamically refreshed information base, MIB includes the information on network equipment allocation, data communications, security and equipment features. The information delivered dynamically to the MIB, forms the data source of the network management system.
- *Agent Server and Management Protocol*: The agent server serves as a bridge between the standard network management software and the system that does not support this standard protocol. With the server, the protocol can be upgraded to the newer version without upgrading the entire network. As for the network management system, it is the network management protocol used by the manager and the agent.

## **Functions of Network Management System**

The network management system has five big functions: allocation management, performance management, security management, failure management and fee calculation management.

These five basic functions are both independent of and related to each other. Among them, failure management is the core of entire network management; allocation management is the basis of all functions because other functions all need to use the information of allocation management; performance management, security management and fee calculation management are relatively more independent, especially fee calculation management because the fee calculation policies are very different for different application units, and the development environments to which fee calculation applies are very different; therefore, fee calculation management is usually developed specific to actual individual conditions.

## **Network Management Protocol**

Due to the features commonly found in network, such as multi-manufacturer, isomerization and inherent distribution, standard is introduced to network management to standardize manufacture of network equipment and development of network management system. Network management protocol is such a standard. Currently, the Simple Network Management Protocol (SNMP) and the

Common Management Information Service (CMIS)/Common Management Information Protocol (CMIP) are the most influential ones. They represent the two big solutions for network management now. Due to its complexity, CMIP is slow to be standardized, and has not been widely accepted, SNMP, on the other hand, has been supported by various manufacturers, and has been widely applied for its practical simplicity.

### **THE DEVELOPING TREND OF COMPUTER NETWORK MANAGEMENT SYSTEM**

Computer network management system is now starting to enter the application layer. Traditionally, computer network management system mainly concerned about the various network equipment, which was in the network layer. Centered around equipment or equipment assemblage, it used SNMP to control and manage equipment. Web users have higher demand on network as well as network bandwidth. Some demands concern the transmission of time sensitive data, such as real-time audio and video, but some data are not time sensitive. Therefore, considering the limit current network bandwidth, it is imperative to change the previous practice of not differentiating service content, but to provide high quality service to all individual users according to the service contents, so as to better utilize the resources of bandwidth. This is called QOS (Quality of Services). With this idea, network management starts to move the controlling

force from network layer to application layer. R1MON2 has tried this way, which was an important change to network management system. In spite of all the versatile technologies used in network management system, as a result of standardization activities and the need for system interconnection.

### **Distributed Network Management**

The key of distributed object is to solve the problem of cross-platform connection and interaction, and to realise distributed application system. The CORBA presented by OMG is quite an ideal platform. Distributed network management is to set up multi domain management processes. Domain management process takes charges of the objects in the domain, and at the same time, different domains coordinates and interacts with each other, so as to perform the management of global area network. Thus, not only is the load of central network management reduced, but time lag for transmission of information on network management is decreased as well, which makes management more effective. Distributed technologies mainly have two aspects: one utilizes CORBA, and the other mobile agent technology. In the near future, centralized distributed network management model can be used to realise the functions of centralization and data acquisition distribution.



## **Integrated Network Management**

Integrated network management requires network management system provide the multitier management support. It can keep all the sub networks in perspective through one operating platform, understanding its operating businesses, identifying and eliminating failures. Thus, the multi interlinked networks management is fulfilled. With network management having become more and more important, many different network management systems have emerged, including those that manage SDH networks and IP networks. The networks managed by these systems interlinks with and interdependent on with each other. There are multi network management systems at the same time. They are independent, in charge of different parts of the network. There can even be a few network management systems with same contents existing concurrently. They come from different manufacturers, and manage their equipment respectively, which has greatly made network management more complex.

## **Businesses Monitoring**

Traditionally, network management aims at network equipment, and could not directly reflect the impact of equipment failure on businesses. Up until now, some network products have realised the monitoring of processes. However, for some services, even though the services end, but the processes still exist. The monitoring of services cannot be

clearly shown. For customers, they are concerned more about the services they get, such as quantity and quality of programs. Therefore, the monitoring of services and businesses is the further goal of management.

### **Intelligent Management**

It supports strategic management and network management system self diagnosis and self adjustment. Network management is the method of managing the tools that belong to a network and maintaining, administering all the systems that are connected in the network. For one to be able to efficiently manage a network that person should be a qualified network administrator and should have in depth knowledge of the functionalities of the network and different topologies of network. There are two aspects in any network, one is the logical aspect and the other is the physical level. The network administrator should be good at both the logical and the physical aspects to be able to troubleshoot efficiently.

### **Network Administration Functions**

The main task of network administration is to keep the network running smoothly 24 hours a day. The main function is to monitor the network constantly and look for possible trouble, detect and rectify the trouble before the network gets affected.

The network administration part of the job involves the resources available on the network and their functioning.

The administrator is required to keep a track of all the resources available in the network and ensure they are functioning properly.

Network maintenance part of the job involves installing updates frequently, updating the service packs for the network software's and also applying patches for the routers when needed. The software and the hardware for the network must be constantly monitored for updates and maintained in order for the network to function properly.

The provisioning part of the job is accommodating extra resources on the network or upgrading the devices on the network. Suddenly if an entirely new device is introduced to the network, the network may stop functioning, so in order to avoid this the administrator needs to make extra provisions for possible devices at the beginning itself with precision.

### **Network Architecture**

The network administrator needs to have a thorough understanding of the network architecture in order to be able to troubleshoot when there are problems. Most of the networks follow a similar architecture and function in the same way. The network architecture is designed in such a way that if the normal functioning is disrupted in any way then the network will send out an e-mail to the administrator, of any unwanted event, or shutdowns. Since the administrator is notified of the problem it gets taken care of immediately. The disaster recovery is also apart of network

planning and management. The network architecture itself plays a pro-active role in the network by aiding in the functioning of the network. The other crucial part of the network is the network protocol. Most networks use the Simple Network Management Protocol and some others use the Common Management Information Protocol.

### **Network Management History**

By the time computers gained popularity, the demand for networks has also been increasing. Companies and people started needing systems which would work in an environment and still required the controls to be in one place. As and when more devices were introduced it became difficult to add one individual device for each computer. Companies needed an easier way of managing the resources. A Network was the perfect answer; however networks were not easy and needed advanced working technology. When the network initially began it was difficult, but today networks have advanced through software and topologies and there are many efficient methods to handle them.

There are many kinds of networks today like cable networks, wireless networks, digital networks, Satellite connections and all these networks work on similar network topologies. Companies use a combination of these networks for their functionalities. Based on these networks internet and intra company networks have promoted business to a large extent.

---

## **FAULT MANAGEMENT-STATE OF THE ART**

---

The terms *fault*, *error* and *failure* have often been confused. Incorrect interpretation of these terms may lead to their misuses. Definitions distinguishing them can be found in Wang's paper. A *fault* is a software or hardware defect in a system that disrupts communication or degrades performance. An *error* is the incorrect output of a system component. If a component presents an error, we say the component fails. This is a *component failure*.

*Failure* or component failure corresponds to the production of an error by this component. It is essential that we distinguish the terms. The *fault* is the direct or indirect cause of the errors. The *errors* are manifestations of the *fault*. The *failure* is the overall result of the *fault*. However, if a component produces errors, we cannot conclude that there is a fault within the component.

Network faults can be characterised by several aspects such as their symptoms, propagation (transmission of an error from a component to another), duration in the network and severity. Though network faults can be distinguished through their characteristics, it is worth noting that it is difficult to measure these properties accurately since they are subjected to the manner in which they are controlled and managed. The occurrence of a fault may be detected by users through symptoms that may be produced by some network components as a result of error. Fault symptoms

can be associated to four types of error-timing errors, timely errors, commission errors and omission errors.

*These symptoms may take one of the following forms:*

- *An output with an expected value comes either too early or too late:* This situation is due to a timing error. It is usually seen by users as a slow response or a time-out, when their applications are indirectly influenced by the effects of the faults.
- *An output with an unexpected value within the specified time interval:* This situation is due to a timely error which usually indicates a minor fault in the applications or underlying software and hardware.
- *An output with an unexpected value outside the specified time interval:* This is due to a commission error. If no response is produced, it is associated with an omission error. An omission error can be regarded as a special case of commission error. A commission error usually implies a severe fault has occurred in the network.

If these symptoms are observed, there is a possibility that a fault has occurred somewhere in the network resulting in the components to produce erroneous outputs (errors). A fault in one component may have consequences on other associated components. Besides failing its own system, it may produce errors which could be transmitted to other components and degrade other systems as well. In this way,

a fault in a single component may have global effects on the network. This phenomenon is referred to as fault propagation. A fault which occurs in an isolated systems may not affect other systems because there is no interaction between them. However, when the isolated systems are connected together by communications, errors produced by a fault may travel in a packet to other systems. A component can fail as a result of faults within it and the erroneous input produced by faults in other components.

This is one of the major characteristics of network faults. Media for fault propagation include parameter, data and traffic. The duration of a network fault, though recognised as one of its important criteria is somewhat difficult to measure.

This is due to three reasons. Firstly, a fault will not be perceived until it produces errors. Secondly, it may take a long time for a particular fault to be isolated. Finally, the effects of network faults may not be eliminated automatically when the faults are removed. They will remain for some time until the operation is completely restored. Therefore, network faults can only be generally divided according to their duration into three groups: permanent, intermittent and transient. A permanent fault will exist in the network until a repair action has been taken. This results in permanent maximum degradation of the service. An intermittent fault occurs in a discontinuous and a periodic manner. Its outcome

will be failures in current processes. This implies maximum degradation of the service level for a short period of time. A transient fault will momentarily cause a minor degradation of the service. Faults of the first type will cause an event report to be sent out and changes made in the network configuration to prohibit further utilisation of this resource.

For a fault of the second type, the severity of the fault may transfer from being intermittent to being permanent if an excessive occurrence of this kind of fault becomes significant. Finally, a transient fault will usually be masked by the error recovery procedures of network protocols and therefore may not be observed by the users. It is fundamental for a designer of a fault management system to have knowledge of fault characteristics. This is because not all faults will have the same priority. The fault management system designer will have to decide which faults must be managed.

## **FAULT MANAGEMENT PROCESS**

Recent literature suggests that a comprehensive fault management system is composed of monitoring, reporting, logging, trouble ticketing, filtering, correlating, diagnosis and recovery activities. The domain in which the fault management system will operate. For the purpose, we have chosen to divide the activities associated with fault management into four major categories, namely detection, isolation, correction and administration. Error detection provides the capability to recognise faults. It consists of



monitoring and reporting activities. Information provided by monitoring devices must be current, timely, accurate, relevant and complete. Reporting activity include investigation of critical criteria which require notification and a mechanism for report generation. It also involves determining appropriate destination for sending notifications.

Its purpose is to isolate the actual fault, given a number of possible hypotheses of faults. Testing may be the most appropriate way of isolating the fault at this stage. Isolation comprises four activities: filtration, interpretation, correlation and diagnosis. Filtration involves analysing management information in order to identify new faults or if the fault has occurred before, to update its count.

Filtration discards management information notifications that are of no significance and routes applicable notifications to their appropriate destinations within the system. Important information contained in the event reports must be extracted. Here the nature, structure and significance of the event report are examined.

Important information such as the name of the event report which normally represents the predefined condition that was met and triggers the generation of the event report itself. Other useful information is the time when the event report was generated. This information is important when performing correlation activity so that we may distinguish which events are related, and which are not. Correlation

proves to be helpful, when two or more notifications received are actually due to a single fault. Through correlation, some faults may be indirectly detected.

Hypotheses can be drawn from alarm correlation giving possible causes of fault. The objective of the diagnosis process is to isolate the cause of a fault down to a network resource. Given a set of probable causes, the diagnosis process is carried out. It involves identification and analysis of problems by gathering, examining and testing the symptoms, information and facts.

Once isolated, the effect of the fault must be minimized through bypass and recovery, and permanent repair instituted. Where applicable, steps must be taken to ensure that problems do not recur. This procedure consists of three activities: reconfiguration, recovery and restoration.

Reconfiguration or bypassing involves activating redundant resources specifically assigned to backup critical entities, suspending services, or re-allocating resources to more important uses. The objective is to reduce the immediate impact of the failure. This function may be in a mixture of manual, semi-auto and automatic procedures. It may be possible for a fault to recover before any reconfiguration attempt is made. This depends on the nature of the failure, the criticality of the service and the expected time required to recover/reconfigure. Once a fault has been rectified, the repair needs to be tested and the entity returned to service.

This needs to be scheduled at an appropriate time and depends on the expected service disruption in doing so.

Fault administration service ensures that faults are not lost or neglected, but they are solved in a timeous fashion. This involves monitoring fault records, maintaining an archive of fault information, analysing trends, tracking costs, educating personnel and enforcing company policy with regards to problem resolution. It consists of three activities: logging, tracking and trend analysis. Logging maintains a log of event reports on faults that have occurred in the network. This will be used for trend analysis, reporting and future diagnosis of the same type of or similar failures. Tracking keeps track of existing problems and persons responsible for and/or working on each one, facilitates communication between problem solving entities and prevents duplicate problem solving efforts. This includes prioritisation of open faults due to their severity, and escalation of fault isolation or correction processes based on duration and severity of the faults. The current open fault records need to be ordered according to a priority scheme such that the most costly, or potentially costly failures are timeously resolved. The whole activities may be accomplished using a trouble ticket system. Important information includes the frequency of occurrence of a particular failure and how much down time the various users are experiencing. Trends may indicate a need to redesign areas of the communication

environment, replace inferior equipment, enhance problem solving expertise, acquire new problem solving tools, improve problem solving procedures, improve education, renegotiate service level agreements. In this project, all activities in fault detection and isolation procedures and some aspects in recovery and administration procedures are implemented.

### **Typical Problems**

One of the most critical problem associated with fault monitoring as given by Dupuy and Stallings is *unobservable faults*. In this situation, certain faults are inherently unobservable through local observation. For example, the existence of a deadlock between co-operating distributed processes may not be observable locally. Other faults may not be observable because the vendor equipment is not instrumented to record the occurrence of a fault.

Other problems are defined by Fried and Tjong as follows. *Too many related observation*: A single failure can affect many active communication paths. The failure of a WAN back-bone will affect all active communication between the token-ring stations and stations on the Ethernet LANs, as well as voice communication between the PBXs. Furthermore, a failure in one layer of the communications architecture can cause degradation or failures in all the dependent higher layers. This kind of failure is an example of propagation of failures. Because a single failure may generate many secondary failures, they may occur around the same time and may often

obscure the single underlying problem. *Absence of automated testing tools:* Testing to isolate faults is difficult, expensive and time consuming. It requires significant expertise in device behaviours and tools to pursue testing. Even such a simple task as tracing the progressions of packets along a virtual circuit is typically impossible to accomplish. This leads to empirical rules of operation as “the only way to test if a virtual circuit is up, is to take it down”.

The process of recovery typically involves a combination of automatic local resetting combined with manual activation of recovery procedures. Recovery presents a number of interesting technical challenges. Which include *automatic recovery as a source of fault:* Since most network devices or processes are designed to recover automatically from local failures, this can also be a source of faults. The problem in fault administration is the maintenance of fault reports log which has been made difficult due to the lack of functionalities for creating or deleting records. The logging facility is usually performed in a static manner, where logging characteristics are stagnant. Therefore, some important fault occurrences are unable to be recorded. If log attribute values can be changed, the logging behaviour may also be altered.

### **FMS: A FAULT MANAGEMENT SYSTEM BASED ON THE OSI STANDARDS**

In order to overcome the problems, we have designed and implemented FMS. It offers three types of fault management

applications as depicted namely the Fault Maintenance Application (FMA), the Log Maintenance Application (LMA), and the Diagnostic Test

Application (DTA). These fault management applications work together with the OSI Agent to perform fault management tasks on the network resources. The OSI Agent serves as a fault-monitoring agent. It has the capability to independently report errors to FMA. It can also issue a report when a monitored variable crosses a threshold. This allows the FMS to anticipate faults. In addition, it maintains a log of events. These logs can be accessed and manipulated by LMA. DTA provides a set of diagnostic tests which may be invoked by the user. This facility is beneficial to the network administrator whenever there is any suspicion that some of the resources are not functioning as desired.

The paragraphs that follow explain how these facilities help in increasing fault management efficiency. The FMA solves the problems of unobservable fault due to ill-equipped vendor equipment to record the occurrence of a fault. This is done by monitoring the real resource critical properties and when significant events involving these properties occur, these events are reported to FMA so that further investigation is initiated. FMA acts upon the receipt of these events by performing other fault management processes on them. These processes include event interpretation and filtration, event correlation, invocation of predefined diagnostic tests

and initiating recovery process. Event or alarm correlation is used by the FMA to solve the problems of too many related observation. In addition, the reporting criteria is designed to be reasonably tight to reduce the volume of alarms received by the FMA. In accomplishing this objective, only events that require attention are reported. Thus, in the FMS implementation, event reports are equal to alarms. Nevertheless, the objective to anticipate failure is neither neglected nor sacrificed. Hence, a number of managed objects are designed to issue event reports when the monitored attributes cross thresholds. Automated testing is provided in FMA by scheduling the execution of predefined diagnostic tests for every event report that is received, so that the source of failure becomes apparent.

Subsequently, recovery action pertaining to the diagnosis is carried out automatically. On the other hand, the DTA provides a function that allows diagnostic tests to be invoked by the user. This facility proves to be beneficial to the experienced user who wants to skip trivial tests and choose only specific ones. This results in lower consumption of the network resources for the purpose of management. The lack of adequate tools for systematic auditing is overcome by the supports provided by the LMA. The LMA has the capability to initiate error condition (events) logging. Furthermore, the LMA can access these logs and control logging behaviour by setting their log attribute values. Other supports include

*Networking Hardware*

facilities for deleting logs and log records, and reviewing events (by reviewing log records) for diagnostic purpose or trend analysis.



# 8

---

## Internet and World Wide Web

---

The Internet and the World Wide Web have a whole-to-part relationship. The Internet is the large container, and the Web is a part within the container. It is common in daily conversation to abbreviate them as the “Net” and the “Web”, and then swap the words interchangeably. But to be technically precise, the Net is the restaurant, and the Web is the most popular dish on the menu.

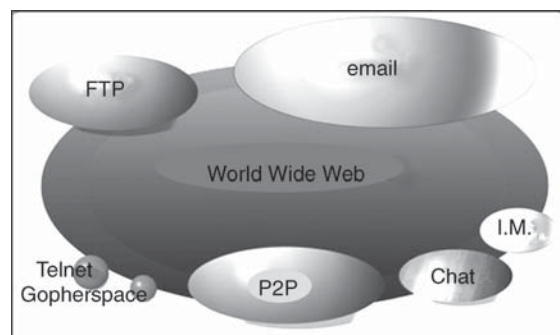
Here is the detailed explanation:

***The Internet is a Big Collection of Computers and Cables.*** The Internet is named for “interconnection of computer networks”. It is a massive hardware combination of millions of personal, business, and governmental computers, all connected like roads and highways. The Internet started in the 1960’s under the original name

“ARPAnet”. ARPAnet was originally an experiment in how the US military could maintain communications in case of a possible nuclear strike. With time, ARPAnet became a civilian experiment, connecting university mainframe computers for academic purposes.

As personal computers became more mainstream in the 1980’s and 1990’s, the Internet grew exponentially as more users plugged their computers into the massive network. Today, the Internet has grown into a public spiderweb of millions of personal, government, and commercial computers, all connected by cables and by wireless signals.

No single person owns the Internet. No single government has authority over its operations. Some technical rules and hardware/software standards enforce how people plug into the Internet, but for the most part, the Internet is a free and open broadcast medium of hardware networking.



## **THE WEB IS A BIG COLLECTION OF HTML PAGES ON THE INTERNET**

The World Wide Web, or “Web” for short, is that large software subset of the Internet dedicated to broadcasting

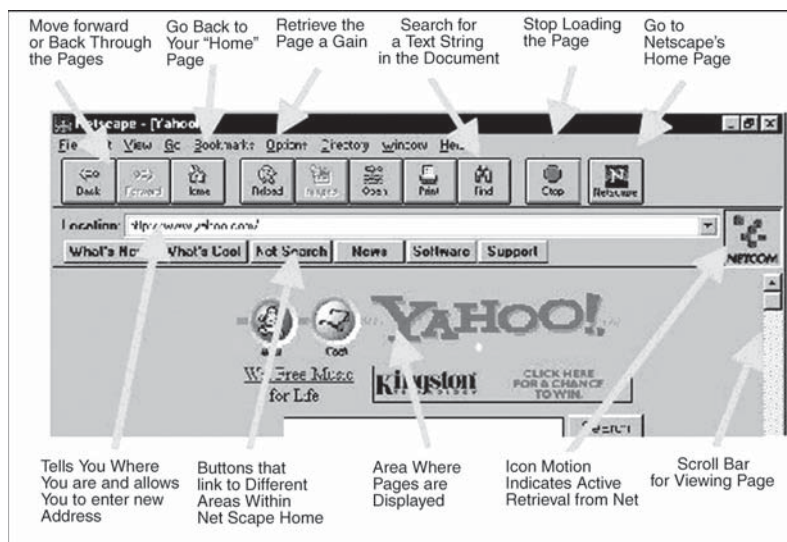
HTML pages. The Web is viewed by using free software called web browsers. Born in 1989, the Web is based on hypertext transfer protocol, the language which allows you and me to “jump” (hyperlink) to any other public web page. There are over 40 billion public web pages on the Web today.

The Internet is a worldwide network of computers that use common communication standards and interfaces to provide the physical backbone for a number of interesting applications.

One of the most utilized of these Internet applications is the World Wide Web. What sets the Web apart is an easy-to-use interface to a complex network of computers and data.

## WWW BASICS

### BROWSERS



A browser is an application which provides a window to the Web. All browsers are designed to display the pages of information located at Web sites around the world. The most popular browsers on the market today include Microsoft's Internet Explorer and Netscape Navigator. Here is a brief overview of the most commonly used features of a browser:

### **WEB SITES**

Information on the Web is displayed in pages. These pages are written in a standard language called HTML (HyperText Markup Language) which describes how the information should be displayed regardless of the browser used or the type of computer. Pages also include hypertext links which allow users to jump to other related information. Hypertext is usually underlined and in a different colour and can include individual words, sentences, or even graphics. A Web site is a collection of related Web pages with a common Web address.

### **WEB ADDRESSES**

Web sites and the pages they contain each have a unique worldwide address. This address (or Uniform Resource Locator, URL, in Internet jargon). The address for Microsoft is `www.microsoft.com`. For most sites, this is all you need to specify and it defaults to the main page (or home page) for the site. In some cases, you may also need or want to specify the path and file name such as `www.microsoft.com/office97`.

Note the extension.com after microsoft. There are six of extensions that help to divide the computers on the Internet into understandable groups or domains. These six domains include:.com = commercial,.gov = government,.edu = education,.org = organizations,.net = networks,.mil = military. There are also extensions for sites outside of the U.S. including:.jp = Japan,.uk = United Kingdom,.fr = France, and so on.

## How to “Surf” the Web

With the tidal wave of information on the Web, learning how to surf is an important skill.

All you need to do is...

Enter a Web site address in the “Location” box and hit the return key. You will jump to the home page of the site. If you are not looking for a particular site, a good place to start is Netscape’s “What’s Cool” page which can be found by pressing the “What’s Cool” button located under the address location box on Netscape browsers.



Mouse click on any words on the page that are underlined and highlighted, like this. These words are hypertext links which jump you to other related information located on the

page, on the site, or other sites. As you jump from page to page and site to site, remember that you can always hit the “Back” arrow button to return to any page. The browser automatically saves all the Web pages to your hard drive (the disk cache) so you can immediately go back without having to reload the pages. In most cases, you will start out surfing a particular site or topic and through numerous hypertext links find yourself somewhere completely unrelated but interesting. Now you’re surfing!

### **How to Search the Web**

There are basically three major search services available for handling different tasks: Directories, Search Engines, and Meta Search Engines.

Directories are sites that, like a gigantic phone book, provide a listing of the sites on the web. Sites are typically categorized and you can search by descriptive keywords. Directories do not include all of the sites on the Web, but generally include all of the major sites and companies. Yahoo is a great directory.

Search Engines read the entire text of all sites on the Web and creates an index based on the occurrence of key words for each site. AltaVista and Infoseek are powerful search engines.

Meta Search Engines submit your query to both directory and search engines. Metacrawler is a popular meta search engine.

## **Downloading Software**

In addition to serving Web pages to your browser, the Web also provides opportunities to easily download programs and files.

## **Browser Extensions**

Both Netscape Navigator and Microsoft Internet Explorer browsers provide the ability to extend the functionality of your browser by downloading additional programs that work within the browser. Navigator calls such programs “Plug-Ins” and you can find a collection of these at [www.netscape.com/plugins](http://www.netscape.com/plugins). Microsoft calls them “ActiveX Controls” and can be found at [www.activex.com](http://www.activex.com).

## **File Compression**

Most files are compressed to make them smaller in size and faster to download. You will need to know how it was compressed and have the corresponding decompression program to view the file (most decompression programs are available as shareware). There are different compression programs for different computers, but the most common for the PC include WinZip and PKZIP (files end in .zip), and for the Mac, BinHex (.hqx) and Stuffit (.sit).

---

## **WEB MULTIMEDIA**

---

The Web is rapidly evolving from primarily text-based documents to multimedia experience of sight, sound and

motion which rival CD-ROM titles. There are a number of new multimedia technologies and browser add-ins that can enhance your Web surfing.

### **AUDIO**

Hear live broadcasts, sample songs from your favourite bands, or even use the Web to have two-way “web phone” conversations. Some good sites to start include:

Real Audio at [www.realaudio.com](http://www.realaudio.com)

Web Phone at [www.webphone.com](http://www.webphone.com)

### **VIDEO**

Participate in a live video-conference or see the latest movie clips.

RealVideo at [www.realnworks.com](http://www.realnworks.com)

CU-SeeMe at [www.whitepine.com](http://www.whitepine.com)

### **3D**

Manipulate three dimensional objects and experience virtual reality on the Web.

VRML at [www.vrml.sgi.com](http://www.vrml.sgi.com)

### **ANIMATION**

Interact with some of the most engaging and entertaining sites on the web.

Macromedia Shockwave at [www.macromedia.com](http://www.macromedia.com)

Narrative Enliven at [www.narrative.com](http://www.narrative.com)



---

## **OTHER INTERNET APPLICATIONS**

---

### **ELECTRONIC MAIL**

One of the most widely used applications in business, electronic mail (or e-mail) provides very fast delivery of messages to any enabled site on the Internet. Users must have an e-mail account established with their Internet service provider and a unique e-mail address (such as `santa@northpole.com`). Most browsers include integrated e-mail software.

### **USENET AND NEWSGROUPS**

One of the most popular applications for non-business use on the Internet is the UseNet. UseNet is a very large public bulletin board where individuals can engage in a wide range of activities including: publish ideas, ask questions, sell items, etc. E-mail is the primary method of posting to a newsgroup. Most browsers include an integrated “News Reader” to read and post to Newsgroups. UseNet topics are organized into Newsgroups which start with prefixes such as `rec.` and `alt.` There is a whole culture of jargon and net etiquette (or netiquette) associated with the UseNet.

### **FTP**

FTP, or File Transfer Protocol, is used primarily as a tool to efficiently uploading and downloading files on the Internet. It is often used transparently on Web sites where there are a large number of downloads.

---

## **INTERNET AND WEB PROTOCOLS**

---

The Internet works as a greater system of special networks, each with its own function and set of data transmission procedures. The World Wide Web represents one of the services offered through the international communication system known as the Internet. Additional services include e-mail, instant messaging and file sharing.

### **NETWORK PROTOCOLS**

Internet network protocols refer to data transmission processes between computers, or routers, within a network. Data are broken down and portioned into message packets, a procedure that optimizes transmission speed. An additional part of a network protocol includes how the network router sends and receives message packets.

### **WEB (INTERNET) PROTOCOLS**

The standard network protocol of the Internet, TCP/IP, stands for Transmission Control Protocol/Internet Protocol. The Internet Protocol part of the standard refers to the addressing of data message packets. Additional protocols that operate within the TCP/IP framework include UDP, HTTP and FTP. Each has different functions and purposes that ultimately work together to provide assorted capabilities through what's currently known as the World Wide Web.

## **Web Protocols**

The Internet relies on a number of *protocols* in order to function properly. A protocol is simply a standard for enabling the connection, communication, and data transfer between two places on a network. Here are some of the key protocols that are used for transferring data across the Internet.

### **HTTP**

HTTP stands for Hypertext Transfer Protocol. It is the standard protocol for transferring web pages (and their content) across the Internet.

You may have noticed that when you browse a web page, the URL is preceded by “HTTP://”. This is telling the web browser to use HTTP to transfer the data. Most browsers will default to HTTP if you don’t specify it. You can test this by typing in say... `www.quackit.com` (instead of `http://www.quackit.com`”).

### **HTTPS**

HTTPS stands for Hypertext Transfer Protocol over Secure Socket Layer. Think of it as a secure version of HTTP. HTTPS is used primarily on web pages that ask you to provide personal or sensitive information (such as a password or your credit card details). When you browse a web page using HTTPS, you are using SSL (Secure Sockets Layer). For a website to use HTTPS it needs to have an *SSL certificate* installed on the server. These are usually issued by a trusted 3rd party, referred to as a Certificate Authority (CA).

When you browse a web page using HTTPS, you can check the details of the SSL certificate. For example, you could check the validity of it. You could also check that the website does actually belong to the organization you think it does. You can usually do this by double clicking on the browser's padlock icon. The padlock icon only appears when you view a secure site.

## **FTP**

FTP stands for File Transfer Protocol. It is used to transfer files across the Internet. FTP is commonly used by web developers to publish updates to a website (*i.e.* to upload a new version of the website).

Where HTTP is used for displaying the file in your browser, FTP is used simply to transfer the file from one computer to a specified location on another computer. You can use FTP to transfer the files from your computer to a remote computer (such as a web server), or to transfer from the remote computer to your local computer.

---

## **THE NEXT GENERATION INTERNETWORK PROTOCOL (IPV6)**

---

The IP (Internet Protocol) is a protocol that uses datagrams to communicate over a packet-switched network. When the IETF defined IPv4, it specified a 32-bit address space [RFC3330], from which people and companies now receive allocations of Internet addresses. A set of addresses have

been assigned for special purposes (multicast, private addressing, loopback, etc). The remainder (219,914 blocks, each with 16,777,216 addresses) is being currently allocated by the Internet Assigned Numbers Agency (IANA). IANA regularly delegate address blocks to each of a number of Regional Internet Registries (RIR's). These, in-turn, allocate addresses to operators (*e.g.* ISPs or large organisations), responsible for assigning addresses to individual users and companies. It was commonly held in the 1990's that the Internet would run out of IPv4 addresses, and to avoid this, the world would need to transition to use a new version of the IP protocol that permitted a larger address space. The IETF embarked on a 2.5-year IETF activity called "IP next generation", [IPNG], which performed an exhaustive search to identify user requirements, and to assess the good and bad parts of the existing IPv4 protocol. The IPng working group realised that an upgrade also provided a "once in a lifetime" opportunity to fix deficiencies/inefficiencies in IPv4 [RFC1751]. In July 1994, work started on creation of the new protocol, which would finally be published as IPv6 [RFC2460] in 1998. The key features introduced by IPv6 are:

### **SIMPLIFIED HEADER FORMAT**

The network header of the currently deployed IPv4 protocol is 20 bytes (plus options). IPv6 omits the group of fields in the second 32-byte word of the IPv4 header. A decision was made that IPv6 routers would not support fragmentation

within the network, following poor performance for router-based IPv4 fragmentation. IPv6 still supports host-fragmentation and transparent link/tunnel fragmentation (where the packet is reassembled at the next-hop).

The IPv6 header also omits the network checksum. This was removed on the basis that routers were reliable, and that checksum processing incurred an unnecessary overhead in high-speed routers. Instead, IPv6 relies upon the presence of the pseudo-header in the transport checksum to validate that a packet has been delivered to the intended recipient.

The resulting IPv6 base header is 40 bytes. This increase in header size was not accompanied by an increase in complexity; the IPv6 base header is much simpler, comprising just 8 fields. The main reason for the larger size is to accommodate a pair of larger network addresses, increasing the size from 32 to 128 bits.

### **IPv6 Addressing**

While the larger address space is a major feature, IPv6 also benefits from a new hierarchical addressing architecture [RFC2491]. Each IPv6 address is formed of a scope, prefix and interface ID. Various types of prefixes are specified: global, link-local, Unique Local (ULA), Multicast, Anycast, or Reserved. A site-local and IPv4-compatible scope were previously defined, but have since been deprecated. Based on this structured approach to addressing, scalable IP routing protocols are being developed.

Each IPv6 unicast address refers to a single interface, rather than the address of a host or router (as in IPv4). All interfaces have at least one Link-Local unicast address, although a single interface may have multiple IPv6 addresses of any type (unicast, anycast, and multicast) and a single or set of unicast addresses may be assigned to multiple physical interfaces if all are treated as one network interface. Any unicast interface address may be used as a node identifier.

### **Representing IPv6 Addresses**

An IPv6 address is written as eight sets of four hex characters separated by colons, *e.g.* 2001:0db8:0001:0035:0bad:beef:0000:cafe. To simplify reading, the leading zeros within each group of four hex digits may be omitted, allowing this to be written as 2001:db8:d:46:bad:beef:0:cafe. Further, one can use '::' once in an address to abbreviate a sequence of consecutive zeros, *e.g.* 2001:db8:d0:0:0:0:0:1 can be more concisely written as 2001:db8:d0::1.

### **Neighbour Discovery**

Neighbour Discovery (ND) [RFC 2461] replaces the function of the Address Resolution Protocol in IPv4. It uses IP multicast to determine which hosts and routers are available on-link, or to determine the link address of a specific neighbouring node. Key ND functions are: Router Discovery, Parameter discovery, Redirect, Duplicate Address

Detection (DAD) and Neighbour Unreachability Detection (NUD). The Secure Neighbour Discovery (SEND) protocol [RFC3971] extends ND to address major security concerns associated with ARP.

### **Address Configuration**

IPv6 defines both stateful and stateless autoconfiguration. The choice depends upon the expected deployment scenario. Stateless Address Autoconfiguration (SLAAC) supports ‘plug and play’.

Nodes dynamically discover the network to which they are connected and can provide an appropriate network-layer configuration [RFC4861]. Stateful configuration for IPv6 allows a router to assign Interface IDs and prefixes using a set of extensions to the Dynamic Host Configuration Protocol (DHCP) [RFC 3315].

### **Extension Headers**

IPv6 differs from IPv4 in the way the protocol can be extended. IPv4 may be extended by either assigning the (few) reserved bits in the header for new functions, or by including an option field between the network and transport headers. In practice, most IPv4 routers do not efficiently process packets with options, and IPv4 options are therefore normally only used for packets that need to be inspected by all routers along a path. IPv6 uses a modular ‘Next Header’ mechanism, consisting of zero or more extension headers each containing



a field identifying the next header. The last header (or base header, if no extensions) identifies the type of the payload. This eases processing and frees IPv6 from a limit to the maximum number of option bytes.

### **Flow Label and QoS**

IPv6 refined the QoS model for the Internet, and defined a new header element, the Flow Label, to assist router look-up and to identify sub-flows encrypted using IPsec. Since publishing the base specification of IPv6, IPv4 and IPv6 QoS models have converged. This is not now a major differentiator between the protocols, and the merits of the flow label field remain a matter of controversy among the networking community.

### **IPv6 Security**

From the outset, security was seen as an important part of the IPv6 stack. The security solution advocated use of IPsec by hosts (required in full IPv6 implementations), however this model did not see widespread use, with IPsec now mostly used in tunnel mode, and additional security commonly implemented above the network layer. IPsec continued to evolve for both IPv4 and IPv6 and is no longer a major differentiator in favour of IPv6. IPv6 does provide other features that can improve security (*e.g.* SEND) and its addressing architecture can provide greater resilience to some forms of denial of service attacks.

---

## **SSL: FOUNDATION FOR WEB SECURITY**

---

Virtually all businesses, most government agencies, and many individuals now have Web sites. The number of individuals and companies with Internet access is expanding rapidly, and all of them have graphical Web browsers. As a result, businesses are enthusiastic about setting up facilities on the Web for electronic commerce. But the reality is that the Internet and the Web are extremely vulnerable to compromises of various sorts. As businesses utilize the Internet for more than information dissemination, they will need to use trusted security mechanisms.

An increasingly popular general-purpose solution is to implement security as a protocol that sits between the underlying transport protocol (TCP) and the application. The foremost example of this approach is the *Secure Sockets Layer* (SSL) and the follow-on Internet standard of SSL known as *Transport Layer Security* (TLS). At this level, there are two implementation choices. For full generality, SSL (or TLS) could be provided as part of the underlying protocol suite and therefore be transparent to applications. Alternatively, SSL can be embedded in specific packages. For example, Netscape and Microsoft Explorer browsers come equipped with SSL, and most Web servers have implemented the protocol. Although it is possible to use SSL for applications other than Web transactions, its use at present is typically as part of Web browsers and servers and hence limited to Web traffic.

If you have viewed an HTML source document, you have seen that the links are referenced with href=http://www.cisco.com//www.cisco.com//www.cisco.com/within an anchor (A) tag. In most cases, the reference is to another document through the use of the *Hyper Text Transfer Protocol*, or HTTP. For this, the browser initiates one or more sessions to the destination port of TCP/80 (the well-known port for HTTP) on the server. In some cases, a plug-in can be called, and data specific to that plug-in can be transferred to or from the browser. For that, the browser would initiate a session to the well-known TCP port of the plug-in.

SSL is called when the reference starts like the following: href="https://.. By calling "https" within the browser, it is mandating that the data be transferred through the use of SSL. By clicking on this hot link, the browser initiates a session to the server on port TCP/443. SSL attempts to negotiate a secure link and transfers the data across it. If the negotiation fails, no data is transferred. The browser usually indicates that a secure connection has been requested. Netscape Navigator version 3 indicates this with a blue border around the page and a highlighted key in the lower left corner. Netscape Communicator version 4 displays this with a closed padlock in a lower status window. Microsoft's Internet Explorer indicates it with a padlock in a lower information window. Display of these

signs indicates that the information within the browser window has been delivered through the security of SSL.

SSL was originated by Netscape. Version 3 of the protocol was designed with public review and input from industry and was published as an Internet Draft document. Subsequently, when a consensus was reached to submit the protocol for Internet standardization, the TLS working group was formed within the *Internet Engineering Task Force* (IETF) to develop a common standard. The current work on TLS is aimed at producing an initial version as an Internet Standard. This first version of TLS can be viewed as essentially an SSLv3.1, and is very close to SSLv3. TLS includes a mechanism by which a TLS entity can back down to the SSLv3.0 protocol; in that sense, TLS is backward compatible with SSL.

## **SSL ARCHITECTURE**

SSL is designed to make use of TCP to provide a reliable end-to-end secure service. SSL is not a single protocol but rather two layers of protocols.

The SSL Record Protocol provides basic security services to various higher-layer protocols. In particular, the HTTP, which provides the transfer service for Web client/server interaction, can operate on top of SSL. Three higher-layer protocols are defined as part of SSL: the *Handshake Protocol*, the *Change CipherSpec Protocol*, and the *Alert Protocol*. These SSL-specific protocols are used in the management of SSL exchanges.

Two important SSL concepts are the SSL session and the SSL connection, which are defined in the specification as follows:

**Connection:** A logical client/server link that provides a suitable type of service. For SSL, such connections are peer-to-peer relationships. The connections are transient. Every connection is associated with one session.

**Session:** An association between a client and a server. Sessions are created by the Handshake Protocol. Sessions define a set of cryptographic security parameters, which can be shared among multiple connections. Sessions are used to avoid the expensive negotiation of new security parameters for each connection.

Between any pair of parties (applications such as HTTP on client and server), there may be multiple secure connections. In theory, there may also be multiple simultaneous sessions between parties, but this feature is not used in practice.

Several states are associated with each session. When a session is established, there is a current operating state for both read and write (that is, receive and send). In addition, during the Handshake Protocol, pending read and write states are created. Upon successful conclusion of the Handshake Protocol, the pending states become the current states. A session state is defined by the following parameters (definitions taken from the SSL specification):

**Session Identifier:** An arbitrary byte sequence chosen by the server to identify an active or resumable session state.

**Peer Certificate:** An X509.v3 certificate of the peer. This element of the state may be null.

**Compression Method:** The algorithm used to compress data prior to encryption.

**CipherSpec:** Specifies the bulk data encryption algorithm (such as DES) and a hash algorithm (such as MD5 or SHA-1). It also defines cryptographic attributes such as the hash size.

**Master Secret:** 48-byte secret shared between the client and server.

**Is Resumable:** A flag indicating whether the session can be used to initiate new connections.

*A connection state is defined by the following parameters:*

- **Server and client random:** Byte sequences that are chosen by the server and client for each connection.
- **Server write MAC secret:** The secret key used in MAC operations on data sent by the server.
- **Client write MAC secret:** The secret key used in MAC operations on data sent by the client.
- **Server write key:** The conventional encryption key for data encrypted by the server and decrypted by the client.
- **Client write key:** The conventional encryption key for data encrypted by the client and decrypted by the server.

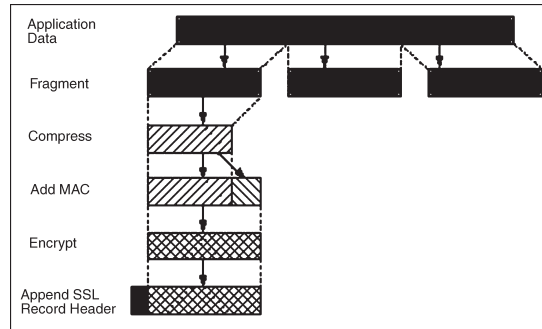
- Initialization vectors: When a block cipher in CBC mode is used, an initialization vector (IV) is maintained for each key. This field is first initialized by the SSL Handshake Protocol. Thereafter the final ciphertext block from each record is preserved for use as the IV for the next record.
- Sequence numbers: Each party maintains separate sequence numbers for transmitted and received messages for each connection. When a party sends or receives a change CipherSpec message, the appropriate sequence number is set to zero.

## **SSL RECORD PROTOCOL**

The SSL Record Protocol provides two services for SSL connections: confidentiality, by encrypting application data; and message integrity, by using a *message authentication code* (MAC). The Record Protocol is a base protocol that can be utilized by some of the upper-layer protocols of SSL. One of these is the handshake protocol which, as described later, is used to exchange the encryption and authentication keys. It is vital that this key exchange be invisible to anyone who may be watching this session.

The overall operation of the SSL Record Protocol. The Record Protocol takes an application message to be transmitted, fragments the data into manageable blocks, optionally compresses the data, applies a MAC, encrypts, adds a header, and transmits the resulting unit in a TCP

segment. Received data is decrypted, verified, decompressed, and reassembled and then delivered to the calling application, such as the browser.



**Figure:** SSL Record Protocol Operation.

The first step is fragmentation. Each upper-layer message is fragmented into blocks of 2<sup>14</sup> bytes (16,384 bytes) or less. Next, compression is optionally applied. In SLLv3 (as well as the current version of TLS), no compression algorithm is specified, so the default compression algorithm is null. However, specific implementations may include a compression algorithm.

The next step in processing is to compute a message authentication code over the compressed data. For this purpose, a shared secret key is used. In essence, the hash code (for example, MD5) is calculated over a combination of the message, a secret key, and some padding. The receiver performs the same calculation and compares the incoming MAC value with the value it computes. If the two values match, the receiver is assured that the message has not been altered in transit. An attacker would not be able to alter both



the message and the MAC, because the attacker does not know the secret key needed to generate the MAC.

Next, the compressed message plus the MAC are encrypted using symmetric encryption. A variety of encryption algorithms may be used, including the Data Encryption Standard (DES) and triple DES. The final step of SSL Record Protocol processing is to prepend a header, consisting of the following fields:

- Content Type (8 bits): The higher-layer protocol used to process the enclosed fragment.
- Major Version (8 bits): Indicates major version of SSL in use. For SSLv3, the value is 3.
- Minor Version (8 bits): Indicates minor version in use. For SSLv3, the value is 0.
- Compressed Length (16 bits): The length in bytes of the plain-text fragment (or compressed fragment if compression is used).

The content types that have been defined are `change_cipher_spec`, `alert`, `handshake`, and `application_data`. The first three are the SSL-specific protocols, mentioned previously. The application-data type refers to the payload from any application that would normally use TCP but is now using SSL, which in turn uses TCP. In particular, the HTTP protocol that is used for Web transactions falls into the application-data category. A message from HTTP is passed down to SSL, which then wraps this message into an SSL record.

## **Change CipherSpec Protocol**

The Change CipherSpec Protocol is one of the three SSL-specific protocols that use the SSL Record Protocol, and it is the simplest. This protocol consists of a single message, which consists of a single byte with the value 1. The sole purpose of this message is to cause the pending state to be copied into the current state, which updates the CipherSuite to be used on this connection. This signal is used as a coordination signal. The client must send it to the server and the server must send it to the client. After each side has received it, all of the following messages are sent using the agreed-upon ciphers and keys.

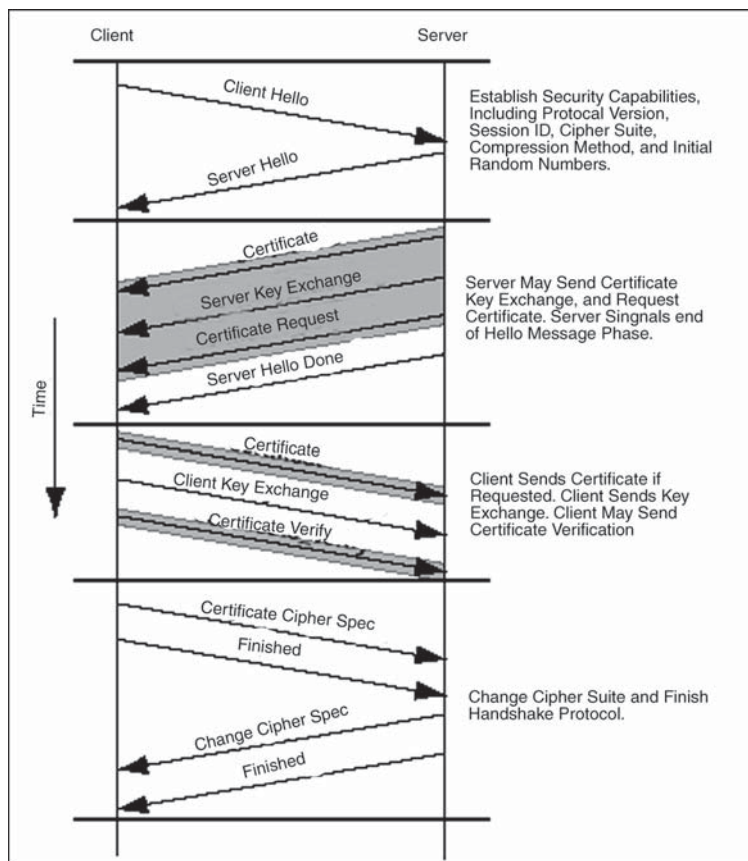
## **Alert Protocol**

The Alert Protocol is used to convey SSL-related alerts to the peer entity. As with other applications that use SSL, alert messages are compressed and encrypted, as specified by the current state.

Each message in this protocol consists of two bytes. The first byte takes the value “warning” (1) or “fatal”(2) to convey the severity of the message. If the level is fatal, SSL immediately terminates the connection. Other connections on the same session may continue, but no new connections on this session may be established. The second byte contains a code that indicates the specific alert. An example of a fatal message is `illegal_parameter` (a field in a handshake message was out of range or inconsistent with other fields). An example

of a warning message is `close_notify` (notifies the recipient that the sender will not send any more messages on this connection; each party is required to send a `close_notify` alert before closing the write side of a connection).

## Handshake Protocol



**Figure:** Handshake Protocol Action.

The most complex part of SSL is the Handshake Protocol. This protocol allows the server and client to authenticate each other and to negotiate an encryption and MAC algorithm and cryptographic keys to be used to protect data sent in an SSL record. The Handshake Protocol is used before any

application data is transmitted. The Handshake Protocol consists of a series of messages exchanged by the client and the server. The initial exchange needed to establish a logical connection between the client and the server. The exchange can be viewed as having four phases.

Phase 1 is used to initiate a logical connection and to establish the security capabilities that will be associated with it. The exchange is initiated by the client, which sends a `client_hello` message with the following parameters:

- **Version:** The highest SSL version understood by the client.
- **Random:** A client-generated random structure, consisting of a 32-bit timestamp and 28 bytes generated by a secure random number generator. These values serve as nonces and are used during key exchange to prevent replay attacks.
- **Session ID:** A variable-length session identifier. A nonzero value indicates that the client wishes to update the parameters of an existing connection or create a new connection on this session. A zero value indicates that the client wishes to establish a new connection on a new session.
- **CipherSuite:** A list that contains the combinations of cryptographic algorithms supported by the client, in decreasing order of preference. Each element of the list (each CipherSuite) defines both a key exchange

algorithm and a CipherSpec; these are discussed subsequently.

- Compression Method: A list of the compression methods the client supports.

After sending the client\_hello message, the client waits for the server\_hello message, which contains the same parameters as the client\_hello message. For the server\_hello message, the following conventions apply. The Version field contains the lower of the version suggested by the client and the highest version supported by the server. The Random field is generated by the server and is independent of the client's Random field. If the SessionID field of the client was nonzero, the same value is used by the server; otherwise the server's SessionID field contains the value for a new session. The CipherSuite field contains the single CipherSuite selected by the server from those proposed by the client. The Compression field contains the compression method selected by the server from those proposed by the client.

The first element of the CipherSuite parameter is the key exchange method (that is, the means by which the cryptographic keys for conventional encryption and MAC are exchanged). The following key exchange methods are supported:

- RSA: The secret key is encrypted with the receiver's RSA public key. A public-key certificate for the receiver's key must be made available.

- **Fixed Diffie-Hellman:** This is a Diffie-Hellman key exchange in which the server's certificate contains the Diffie-Hellman public parameters signed by the *certificate authority* (CA). That is, the public-key certificate contains the Diffie-Hellman public-key parameters. The client provides its Diffie-Hellman public key parameters either in a certificate, if client authentication is required, or in a key exchange message. This method results in a fixed secret key between two peers, based on the Diffie-Hellman calculation using the fixed public keys.
- **Ephemeral Diffie-Hellman:** This technique is used to create ephemeral (temporary, one-time) secret keys. In this case, the Diffie-Hellman public keys are exchanged, and signed using the sender's private RSA or DSS key. The receiver can use the corresponding public key to verify the signature. Certificates are used to authenticate the public keys. This option appears to be the most secure of the three Diffie-Hellman options because it results in a temporary, authenticated key.
- **Anonymous Diffie-Hellman:** The base Diffie-Hellman algorithm is used, with no authentication. That is, each side sends its public Diffie-Hellman parameters to the other, with no authentication. This approach is vulnerable to man-in-the-middle attacks, in which

the attacker conducts anonymous Diffie-Hellman exchanges with both parties.

Following the definition of a key exchange method is the CipherSpec, which indicates the encryption and hash algorithms and other related parameters.

The server begins Phase 2 by sending its certificate, if it needs to be authenticated; the message contains one or a chain of X.509 certificates. The certificate message is required for any agreed-on key exchange method except anonymous Diffie-Hellman. Note that if fixed Diffie-Hellman is used, this certificate message functions as the server's key exchange message because it contains the server's public Diffie-Hellman parameters.

Next, a `server_key_exchange` message may be sent, if it is required. It is not required in two instances: (1) The server has sent a certificate with fixed Diffie-Hellman parameters; or (2) RSA key exchange is to be used.

Next, a nonanonymous server (server not using anonymous Diffie-Hellman) can request a certificate from the client. The `certificate_request` message includes two parameters: `certificate_type` and `certificate_authorities`. The `certificate_type` indicates the type of public-key algorithm. The second parameter in the `certificate_request` message is a list of the distinguished names of acceptable certificate authorities.

The final message in Phase 2, and one that is always required, is the `server_done` message, which is sent by the

server to indicate the end of the server hello and associated messages. After sending this message, the server waits for a client response. This message has no parameters.

Upon receipt of the `server_done` message, the client should verify that the server provided a valid certificate, if required, and check that the server hello parameters are acceptable. If all is satisfactory, the client sends one or more messages back to the server in Phase 3. If the server has requested a certificate, the client begins this phase by sending a certificate message. If no suitable certificate is available, the client sends a `no_certificate` alert instead. Next is the `client_key_exchange` message, which must be sent in this phase. The content of the message depends on the type of key exchange.

Finally, in this phase, the client may send a `certificate_verify` message to provide explicit verification of a client certificate. This message is only sent following any client certificate that has signing capability (that is, all certificates except those containing fixed Diffie-Hellman parameters).

Phase 4 completes the setting up of a secure connection. The client sends a `change_cipher_spec` message and copies the pending `CipherSpec` into the current `CipherSpec`. Note that this message is not considered part of the Handshake Protocol but is sent using the Change CipherSpec Protocol. The client then immediately sends the finished message under the new algorithms, keys, and secrets. The finished message verifies that the key exchange and authentication processes



were successful. In response to these two messages, the server sends its own `change_cipher_spec` message, transfers the pending to the current `CipherSpec`, and sends its finished message. At this point the handshake is complete and the client and server may begin to exchange application layer data.

After the records have been transferred, the TCP session is closed. However, since there is no direct link between TCP and SSL, the state of SSL may be maintained.

For further communications between the client and the server, many of the negotiated parameters are retained. This may occur if, in the case of Web traffic, the user clicks on another link that also specifies HTTPs on the same server. If the clients or servers wish to resume the transfer of records, they don't have to again negotiate encryption algorithms or totally new keys.

The SSL specifications suggest that the state information be cached for no longer than 24 hours. If no sessions are resumed within that time, all information is deleted and any new sessions have to go through the handshake again. The specifications also recommend that neither the client nor the server have to retain this information, and shouldn't if either of them suspects that the encryption keys have been compromised. If either the client or the server does not agree to resume the session, for any reason, then both will have to go through the full handshake.