



Virtual University of Pakistan

FEDERAL GOVERNMENT UNIVERSITY

COURSE HANDOUTS

Introduction to Network Design and Analysis

CS206

Order Your VU Books Now

Get all Virtual University books in high-quality hard copy, delivered right to your doorstep. Visit vubookshoppk.com



Order Now

INTRODUCTION TO NETWORK DESIGN AND ANALYSIS CS206

Table of Contents

Topic 1: Course Introduction.....	9
Topic 2: What is a Network?	10
Topic 3: Traditional Computer Networks	12
Topic 4: Using the Network by Accident	14
Topic 5: Using the Network on Purpose.....	15
Topic 6: Well-known Network Applications	16
Topic 7: Building a Network: Starts with a Plan.....	18
Topic 8: Examples of Good Rules for Networking	20
Topic 9: Proprietary and Public Models	21
Topic 10: How TCP/IP Standard Grows?.....	23
Topic 11: Two-Well Known Networking Models	25
Topic 12: What is a Local Area Network?	26
Topic 13: LAN Cables and Connectors	28
Topic 14: Ethernet Hubs.....	32
Topic 15: Ethernet Frames and Collisions	33
Topic 16: How to avoid Collisions over Ethernet?	34
Topic 17: Importance of Ethernet Addresses	36
Topic 18: Frame Check Sequence.....	37
Topic 19: Two Ethernet Standards	38
Topic 20: Working of a LAN Switch	39
Topic 21: Switch: Collision Avoidance	41
Topic 22: Full Duplex and Full Switching	43
Topic 23: Learning Ethernet addresses	45
Topic 24: Ethernet Network Speeds.....	47
Topic 25: Multiple Physical LANs.....	49

Topic 26: Introduction to Virtual LANs	51
Topic 27: Packing VLAN's frames in a Trunk	52
Topic 28: Email: A Network Application	54
Topic 29: TCP/IP Email Standards: SMTP, POP3	56
Topic 30: File Transfer Protocol (FTP).....	58
Topic 31: World Wide Web and HTTP Protocol.....	61
Topic 32: Main Features of TCP.....	63
Topic 33: Working of TCP	65
Topic 34: TCP Flow and Congestion Control.....	66
Topic 35: User Datagram Protocol (UDP)	68
Topic 36: TCP Ports Numbers.....	69
Topic 37: TCP Segmentation	71
Topic 38: Basics of Internet Protocol (IP)	72
Topic 39: Working of IP and IP addresses.....	74
Topic 40: How to run an IP Network?	75
Topic 41: Classes of IP Networks.....	77
Topic 42: IP Subnetting	78
Topic 43: Network Address Translation (NAT)	80
Topic 44: Dynamic Host Configuration Protocol (DHCP).....	81
Topic 45: Internet Control Message Protocol (ICMP)	83
Topic 46: End-to-End: Processing IP Packet.....	85
Topic 47: A Link State Routing Algorithm	86
Topic 48: Distance Vector Routing Algorithm.....	88
Topic 49: Working of Default Gateway Router.....	90
Topic 50: Address Resolution Protocol.....	91
Topic 51: Router's Routing Logic and Table.....	94
Topic 52: Routing with Subnets.....	96
Topic 53: Router Hardware Architecture	97

Topic 54: Routing to Nearby Places.....	100
Topic 55: Dynamically Learning Routing Tables	101
Topic 56: How to Pick the Best Route	103
Topic 57: Interior & Exterior Routing Protocols.....	104
Topic 58: Introduction to Domain Name System	105
Topic 59: Domain Name System (DNS) Servers.....	106
Topic 60: Working of Domain Name System	108
Topic 61: DNS Resource Records.....	109
Topic 62: Introduction to Wide Area Network	111
Topic 63: Different Aspects of WAN Link	112
Topic 64: A Cross-over Cable versus Leased Circuit.....	113
Topic 65: Routers and WANS	114
Topic 66: Frame Relay.....	116
Topic 67: Frame Relay Switching.....	117
Topic 68: Virtual Circuits	118
Topic 69: Point-to-Point WANs & Frame Relay	120
Topic 70: Routing over PVCs	121
Topic 71: The Internet: A Large IP Network	122
Topic 72: Using a Phone Line for Data.....	124
Topic 73: Digital Subscriber Line (DSL)	125
Topic 74: Sending Data without a Phone Line	126
Topic 75: Introduction to AAA Security Model.....	128
Topic 76: Password Authentication Protocol	130
Topic 77: Virtual Private Network (VPN)	132
Topic 78: Enterprise Network and the Internet.....	133
Topic 79: Firewalls, Demilitarized Zone, IDS.....	134
Topic 80: Introduction to Wireshark	138
Topic 81: Five Common Network Problems	142

Topic 82: Next 5 Common Network Problems	143
Topic 83: Next 6 Common Network Problems	144
Topic 84: Next 4 Common Network Problems	145
Topic 85: Next 5 Common Network Problems	145
Topic 86: Next 8 Common Network Problems	146
Topic 87: A Four-Part Analysis Methodology	148
Topic 88: Using a Troubleshooting Checklist	148
Topic 89: Wireshark Lab 1.....	150
Topic 90: Wireshark Lab 2.....	151
Topic 91: Wireshark Lab 3.....	153
Topic 92: Wireshark Lab 4.....	154
Topic 93: Wireshark Lab 5.....	155
Topic 94: Wireshark Lab 6.....	157
Topic 95: Wireshark Lab 7	157
Topic 96: Wireshark Lab 8.....	159
Topic 97: Wireshark Lab 9.....	160
Topic 98: Wireshark Lab 10.....	161
Topic 99: Wireshark Lab 11.....	163
Topic 100: Wireshark Lab 12.....	165
Topic 101: Wireshark Lab 13.....	166
Topic 102: Wireshark Lab 14.....	168
Topic 103: Capture Options for a Switched Network	170
Topic 104: Wireshark Lab 15.....	171
Topic 105: Wireshark Lab 16.....	172
Topic 106: Wireshark Lab 17	173
Topic 107: Verify the Target Host Traffic.....	173
Topic 108: Wireshark Lab 18.....	174
Topic 109: Wireshark Lab 19.....	174

Topic 110: Wireshark Lab 20	175
Topic 111: Wireshark Lab 21	176
Topic 112: Do not Focus on Acceptable Delays	177
Topic 113: Watch for the Delays that DO Matter	178
Topic 114: Wireshark Lab 22	179
Topic 115: Wireshark Lab 23	180
Topic 116: Wireshark Lab 24	181
Topic 117: Wireshark Lab 25	183
Topic 118: Wireshark Lab 26	183
Topic 119: Wireshark Lab 27	185
Topic 120: Wireshark Lab 28	186
Topic 121: Wireshark Lab 29	188
Topic 122: Wireshark Lab 30	189
Topic 123: Wireshark Lab 31	191
Topic 124: RTT: Packets 2 and 3 of TCP Handshake	192
Topic 125: Wireshark Lab 32	193
Topic 126: Wireshark Lab 33	194
Topic 127: Wireshark Lab 34	196
Topic 128: Wireshark Lab 35	198
Topic 129: Wireshark Lab 36	199
Topic 130: Wireshark Lab 37	200
Topic 131: Wireshark Lab 38	201
Topic 132: Wireshark Lab 39	202
Topic 133: Wireshark Lab 40	203
Topic 134: Wireshark Lab 41	205
Topic 135: Wireshark Lab 42	207
Topic 136: Wireshark Lab 43	208
Topic 137: Wireshark Lab 44	209

Topic 138: Wireshark's Expert Infos System.....	210
Topic 139: Wireshark's Packet Loss Detection	211
Topic 140: Packet Loss Recovery Methods.....	212
Topic 141: Wireshark Lab 45	213
Topic 142: Wireshark Lab 46	214
Topic 143: Wireshark Lab 47	215
Topic 144: Wireshark Lab 48	217
Topic 145: Wireshark Lab 49	218
Topic 146: Duplicate ACKs and their Causes	219
Topic 147: Wireshark Lab 50	220
Topic 148: Wireshark Lab 51	221
Topic 149: Wireshark Lab 52	223
Topic 150: Out-of-Order Packets and their Causes	224
Topic 151: Wireshark Lab 53	225
Topic 152: Wireshark Lab 54	226
Topic 153: Causes of Fast Retransmissions	228
Topic 154: Wireshark Lab 55	228
Topic 155: Wireshark Lab 56	229
Topic 156: Causes of Retransmissions.....	230
Topic 157: Wireshark Lab 57	231
Topic 158: Wireshark Lab 58	232
Topic 159: Causes of ACKed Unseen Segments.....	234
Topic 160: Wireshark Lab 59	235
Topic 161: Wireshark Lab 60	236
Topic 162: Causes of Keep Alives	237
Topic 163: Wireshark Lab 61	237
Topic 164: Wireshark Lab 62	238
Topic 165: Wireshark Lab 63	239

Topic 166: Causes of Reused Ports.....	241
Topic 167: Wireshark Lab 74.....	242
Topic 168: Wireshark Lab 75.....	243
Topic 169: Causes of Checksum Errors.....	244
Topic 170: Wireshark Lab 76.....	245
Topic 171: Wireshark Lab 77.....	246
Topic 172: Wireshark Lab 78.....	248
Topic 173: Introduction to Packet Tracer	249
Topic 174: Packet Tracer's Interface Overview	250
Topic 175: Creating a Simple topology.....	254
Topic 176: Introduction to Cisco and PT devices	256
Topic 177: Customizing Devices with Modules	258
Topic 178: Accessing CLI of a Device	259
Topic 179: Configuring Devices with Config Tab.....	261
Topic 180: Generic IP End Devices in PT.....	264
Topic 181: Configuring End Devices in PT.....	266
Topic 182: Packet Tracer's Simulation Mode	268
Topic 183: Connecting Devices and Link Status.....	271
Topic 184: Testing Connectivity with PDUs	272
Topic 185: Clustering a Topology	273
Topic 186: Creating Cities, Offices & Wiring Closets.....	275
Topic 187: Managing Cables and Distances.....	277
Topic 188: Static Routing with GUI.....	278
Topic 189: Static Routing with CLI	280
Topic 190: Configuring RIP with GUI	282
Topic 191: Configuring RIP with CLI.....	284
Topic 192: Load Sharing.....	286

Topic 1: Course Introduction

This topic provides motivation to study this course.

Motivation1: Information I can receive: Now-a-days, we can get a lot of information while sitting in front of the computer. For example, we can access our bank accounts; we can receive world news, look at the weather forecast etc. This has become possible because of computer networking. This is shown next.



Figure taken from "CCNA Exploration Network Fundamentals" Cisco Press

Motivation 2: Daily life Applications: Email, web, instant messaging, P2P file sharing, network games, streaming videos, real-time video conferring are those applications that we use in our daily lives. We may encounter issues such as response delays, slow application at the server.

Motivation 3: Internet: The largest engineered system ever created by mankind is the Internet which consists of hundreds of millions of connected computers, communication links, and switches. There billions of users who connect via laptops, tablets, and smartphones.

In this course, we will look at how the computer networks work, how we can troubleshoot their problems using Wireshark, and how we can simulate new networks using Packet Tracer.

Course Composition: We have divided the course into three Main Parts.

- Part 1 presents the networking fundamentals from practical point-of-view and does not delve into the theoretical nitty-gritties.
- Part 2 describes how analyzing packets with the Wireshark tool can help a student to troubleshoot his network.

- Part 3 discusses how to configure Cisco devices with practical examples and simulate networking with Packet Tracer.

Figures and Material used for Part 1(Topics 2 to 79 have been adapted from W. Odom’s book with the title “*Computer Networking First-Step*”, Cisco Press, 2004 unless stated otherwise.

Topic 2: What is a Network?

This topic provides the basic know how of a simple computer network.

A **computer network** is a combination of hardware, software, and cabling. This combination allows multiple computing devices to communicate with each other and thus gives computers the ability to communicate with each other. Next we show a simple network:

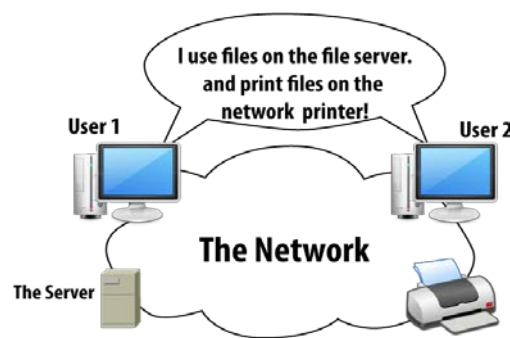


Figure 2.1: Simple Single Site Network

This is a typical small network used by a company at a single site. In a network diagram, we use a cloud to hide details such as hardware, software, and cabling. This is done, when details are not important to the current discussion. A server provides

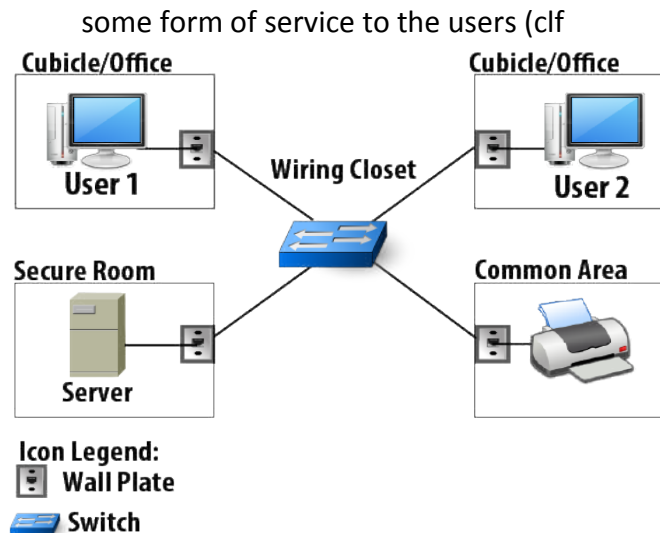


Figure 2.2: A closer Look at the simple Network

We can see that User1's PC has networking software installed, and a cable that connects his PC to a socket on the wall. That socket has a cable on the (hidden) other side of the wall plate. Cables normally run under the floor, in the ceiling, or some other hidden place, with the other end being in a wiring closet. All the cables connect to a switch inside the wiring closet. A **switch** consists of specialized hardware and software that forwards the network traffic back and forth between the various network devices on the network. There are a lots of places (called switch ports) into which you can plug in one of the networking cables.

Now, we look at the different perceptions three different people have according to their work on a different aspect of a network. First of all, we study the **Server Guy**, who is responsible for the server and needs some PC hardware skills, but more importantly, he needs strong skills with software. His perception is shown next.

The server guy views the rest of the network as a network utility i.e. he treats the rest of the network just like you think of the telephone, electrical power, and water system. There is no need to think about it unless it's not working. Next is the **Cabling Guy**. He is also known as an electrician. He runs the cables from each cubicle back to the wiring closet. His job requires physical dexterity, knowledge of how to conform to the electrical building standards. He focuses on installing, testing, and troubleshooting the cabling from each wall plate to the wiring closet. He makes sure that there's a working cable running from the wiring closet to each place in the building where a computer needs to connect to the network.

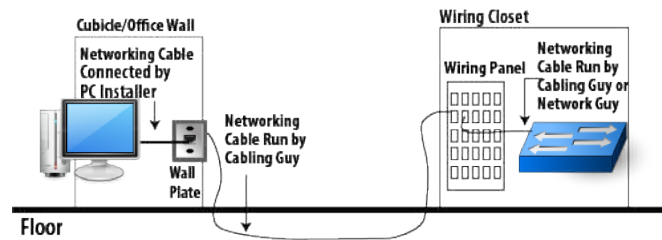


Figure 2.3: The cabling Guy's perception of the network "Elephant"

The third guy is the **Network Guy**. He is responsible for the switch, as well as any other hardware and software used to create a network utility for the computers. He installs, supports, and troubleshoots the hardware and software on the switch. He knows which computer's cable plugs into the various numbered ports on the switch. The network guy's perception is shown next.

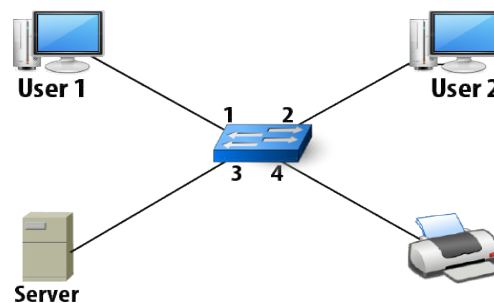


Figure 2.4: The network Guy's perception of the Network "Elephant"

Topic 3: Traditional Computer Networks

In this topic, we study different traditional computer networks.

There are two well-known examples of computer networks. They are named an **Enterprise WAN** and the **Internet**.

An Enterprise WAN is a network that should have the ability to connect all sites of a big company, located at multiple sites. You call it an enterprise network because the network is owned by and created by one company, and a company can be considered to be an enterprise.

An enterprise network is shown next.

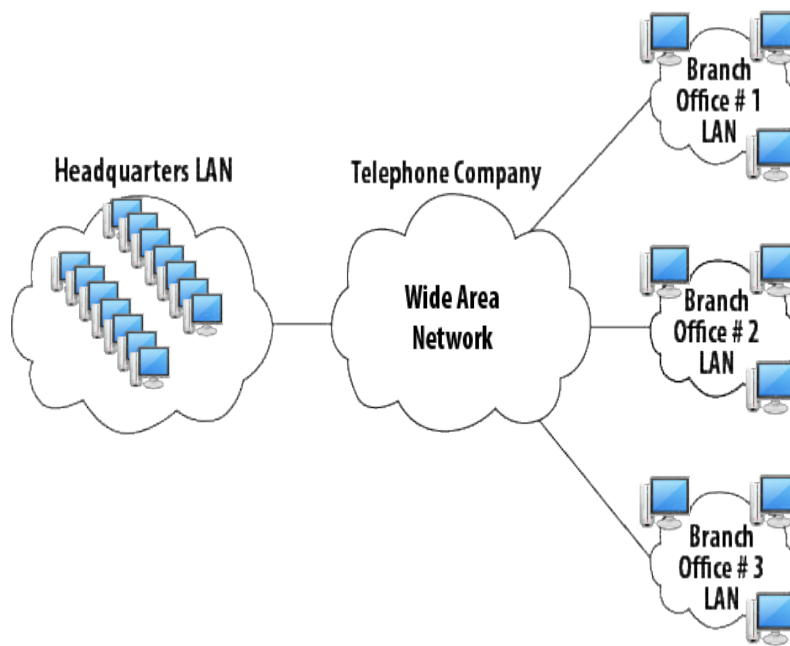


Figure 3.1: A Larger Network: AN Enterprise WAN

This typical enterprise has three remote branch offices, each with few PCs and a printer. The headquarters site has more users (hence, more PCs), as well as several servers. Each remote site consists of a simple network.

The **Internet** is a unique computer network as it connects almost all enterprise networks and individual users. We next show a part of the Internet.

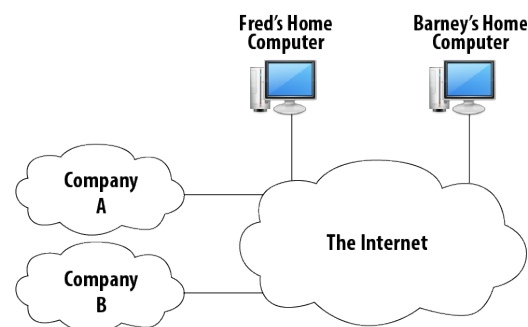


Figure 3.2: The Internet with Enterprise and individuals connected to it

The Internet works with the help of companies called Internet service providers (ISPs), which provide service to companies and individuals to connect them to the Internet. This makes almost all computers on the planet can communicate with each other.

Topic 4: Using the Network by Accident

This topic discusses the scenario when we use the network by accident.

Some of the most common network services hide the network from the end user. We call them “accidental” network services. To understand this concept, let’s assume an employee named Fred working at a remote office. He is the master of his own domain; he has a PC, a printer, and he can do all his work without much interruption from the home office. Fred’s daily tasks include opening a document with a word processor, changing some of the text contained in it, and printing it. He then mails the letter to the customer and saves the changes to the letter. Fred’s PC is shown next.

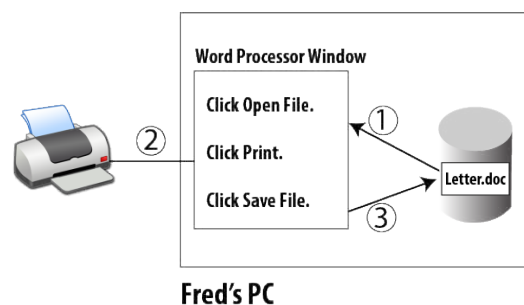


Figure 4.1: Daily task computer

As Fred’s company grows, Home office sends Wilma to Fred's office to work with him and help him get all the work done. She brings her fast high-end PC with her to the new office and installs a network. She takes over Fred's old printer and connects it to her own PC. Also copies all the customer letters from Fred's PC over to her PC. The network looks like this:

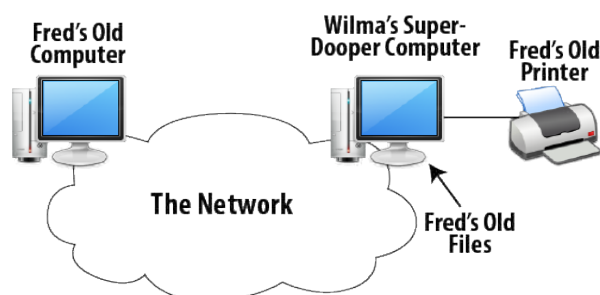


Figure 4.2: Network with Fred and Wilma’s computers

As Wilma has set everything up, including the network, so she knows how it works. But Fred has no idea about the network.

To do his job on the very next day, Fred starts up his word processor, grabs a diskette, and walks over to Wilma's computer and copies a customer letter onto the diskette. He walks back to his desk, updates the letter using his old computer, and then walks back to Wilma's computer with the diskette. Then Fred copies the file

back onto Wilma's computer, replacing the old file, and brings up the word processor on Wilma's computer so that he can print a copy for mailing to the customer. This is depicted next.



Figure 4.2: Fred's sneakernet

Fred could have done the same job in a simpler and more efficient way if he had used the network. He doesn't even need to know it's there. If Fred had just looked on his C drive in the folder called Customers, he would have seen all the same files he is used to working with. These files are on Wilma's PC - a file server but that's hidden from Fred. Also, when he clicks on the Print icon from his word processor, he'll see a printer called *same old printer*. If he prints to that printer, it will print on his same old printer, even though it's connected to Wilma's PC, which is set up as a print server.

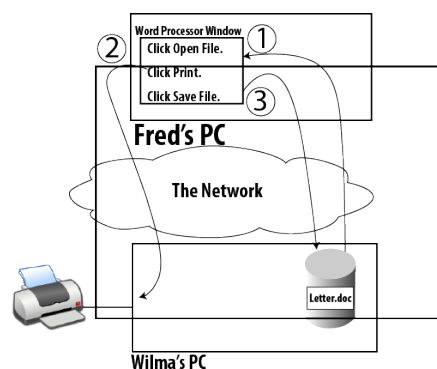


Figure 4.3: Fred's PC using the new Network

Topic 5: Using the Network on Purpose

In topic, we study the case when we use the network on purpose.

When we use the Internet, chances are high that we will be aware that we are using a network. The Internet is the global network to which almost every company and organization in the world is connected. We can make a phone call to almost anyone

on the planet because all the telephone companies in the world connect to each other. Similarly, most computers can communicate with each other over the Internet because most computer networks connect to each other through Internet Service Providers (ISPs). Networking connections among enterprises, individual home users, and other ISPs are provided by ISPs. A chunk of the Internet is shown next.

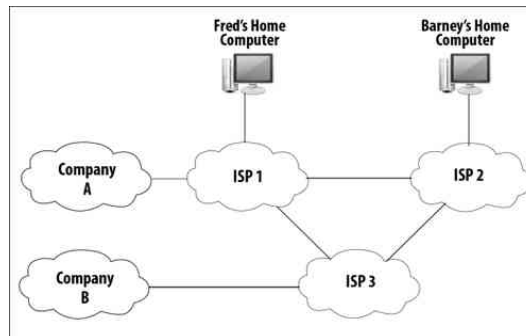


Figure 5.1: Conceptual View of the Internet

ISP1's network allows individual computers, such as Fred's, to connect to it. Fred could communicate with computers inside Company A, assuming that the security policies at Company A allowed it. Company A created its enterprise network using hardware, software, and cabling paid for by Company A. ISP1 creates its network as well using its own funds. ISP1 agrees to allow Company A's traffic to pass through ISP1 and on to other ISPs so that Company A can communicate with the rest of the world. In return, Company A pays ISP1 an ongoing fee.

Topic 6: Well-known Network Applications

This topic discusses working of the well-known network applications.

Web Browsing

A **web browser** allows you to sit at one computer and display information that resides in a **web server** somewhere on the Internet. A **web server** consists of software that resides on the computer that is accessible to the end user via the Internet. The information can be in many forms, including simple text, graphics images, animation, video streams, and audio clips. Well-known examples of web browsers are Internet Explorer, Google Chrome, Firefox Mozilla. After a browser requests a web page from a web server, the server replies by sending the contents of the web page back to the browser. Companies, organizations, and even individuals can create their own websites on the Internet. Anyone can then ask for the content in those web pages and see the results. Next, we show the working of web browsing application.

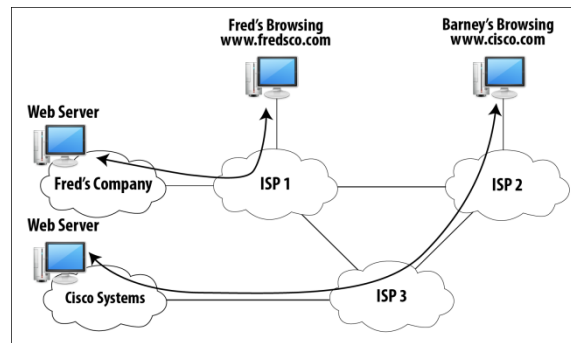


Figure 6.1: Web Browsers and Web Pages on the Internet

A client such as Fred brings up the web browser software on his computer and points to the Universal Resource Locator (URL) of the website. For example, in this case he wants to access www.fredsco.com. The URL or a web address is a string of characters that uniquely identifies a particular web page.

E-Mail

It allows a user to create, send, and receive messages electronically. E-mail is sent between one user and another user, or in some cases, to multiple other users. The person sending the mail needs to know the e-mail address of the person who needs to receive the e-mail. An e-mail address is a text string that represents the address of a person for the purposes of sending and receiving e-mail, much like a mailing address used for postal mail. Example: abc@yahoo.com. An example is shown below.

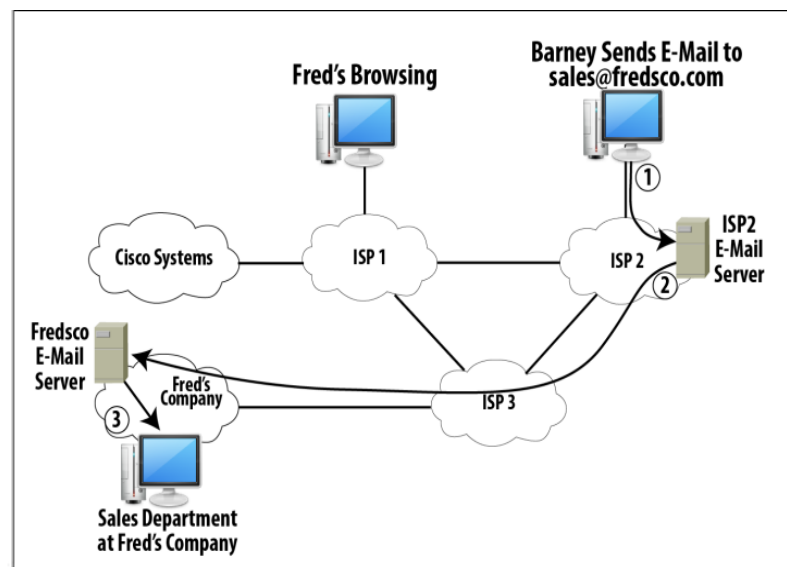


Figure 6.2: E-Mail in the Internet Using Mail Servers

In this diagram, Barney wants to e-mail the sales department at Fred's Company by sending an e-mail to the address of sales@fredsco.com. Barney does not actually

send the e-mail directly to a computer at Fred's Company. Barney sends the e-mail to his e-mail server, which for an individual Internet user like Barney, typically sits inside the ISP network to which the user is connected. An **e-mail server** is a server that receives, forwards, or holds e-mail, much like the service performed by the postal service. That e-mail server forwards the e-mail to the e-mail server at Fred's Company.

Downloading and Transferring Files

The central theme of this application is to move files into and out of your computer using a network. In most cases, the end user does not have the file, and he wants to get it. Next, we show an example where **myproject.doc** is being copied between Barney and Fred.

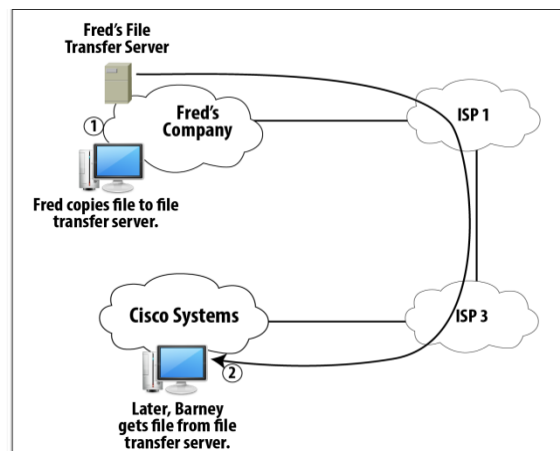


Figure 6.3: Copying myproject.doc Between Barney and Fred

File transfer is a two-step procedure:

- 1. Fred places a copy of the file onto a file transfer server, using file transfer client software.
- 2. Barney gets a copy of the file from the file transfer server, also using file transfer client software.

Topic 7: Building a Network: Starts with a Plan

In topic, we study why rules and standards are important for networking.

Standards are rules that make life a lot easier. To understand this idea, let's have a look at a power socket on a nearby wall. You may find an electrical socket with three holes two that accept flat metal prongs, and one that accepts a round metal prong. The flat metal prongs are parallel to each other. If you go to a store to buy an electrical device such as a lamp, you will expect that its power cord would fit into the

wall socket of your house. In case, does not fit into the power socket, you will get very confused.

Let's now imagine that you bought a new lamp. You plug it in, and the light bulb instantly is broken. You put in a new bulb, and it doesn't light up. You decide that there is some problem with the lamp. So you bring it back to the store and replace it with an identical lamp. When you get it home, the same thing happens to this lamp too. What to do next? Let's call the customer service to inquire from about the problem. This is our special 'We light up your life' model of lamp. It uses the same kind of power cord you are used to using but requires less electrical voltage, saving you money. If you read the instructions for the lamp, you will see that it directs you to rewire and change the voltage coming out of the sockets you want to use for the lamp. 'If you plug this lamp into a normal wall socket, the extra voltage will fry the lamp, and it will no longer light up your life.' So contact an electrician and get your socket changed.

A standard therefore tends to define a particular thing, such as the shape of the wall socket and connector used by an electrical socket and electrical power cable. Another standard might dictate how much voltage flows through the wall socket, whether it is AC or DC, how much current, and the like. Both standards are important and must be followed to prevent exploding lamps. Similarly, networking has standards. First, the PCs in the network need some form of physical connectivity. Physical connectivity refers to the combination of cabling, networking devices, and network interface cards (NICs) in the computers, which together provide the physical capability to transmit and receive data across a network. Next, we show this idea.

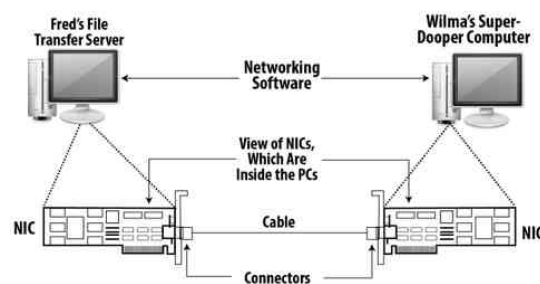


Figure 7.1: Components of a Simple Network

Both PCs have an NIC installed, and each card has a receptacle into which a cable can be connected. To emphasize their existence, the NICs are shown outside the PCs. The two computers must implement the same standards for how the networking software on each computer tells the other what it wants to do.

Topic 8: Examples of Good Rules for Networking

This topic presents examples for implementing networking.

Example 1

When two humans communicate one person says something and the other person listens. The two people need to understand the same language. Similarly, when computers communicate with each other over a network, the sending application needs to send some information to an application on the receiving side. Let's assume Fred's computer is using Wilma's computer, which is configured as a file server.

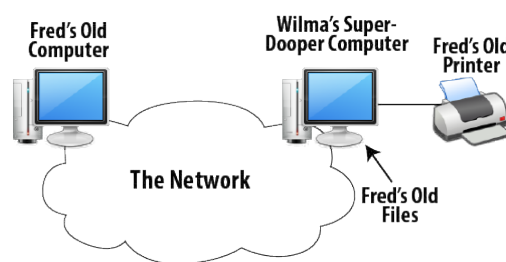


Figure 7.2: Data Transmission Using Packets

Over the network, the file will be transmitted in the form of binary digits (bits). The file server reads the file from the disk. A networking standard will define the encoding rules so that data can be transmitted. For example, it can ask the sender to change the voltage to one level to mean a binary 0, and another to mean a binary 1. In our example shown above, the NIC inside Wilma's computer sends some electrical signal over the cable. The device on the other end of the cable Fred's PC NIC interprets the incoming electrical signal. This will work if both NICs agree to some standard means of transmission. For instance, imagine that Wilma's NIC sends a +5 volts to transmit a binary 0 and +10 volts to send a binary 1. If Fred's NIC expects to receive a +2 volt signal for binary 0 and +4 volts for binary 1, the network will not work because Fred will not understand what Wilma is sending him.

Example 2

Devices in a network send bits in groups, generally called packets. Assume Wilma wants to send a file to Fred. She sends the data bits in the form of three packets as shown next.

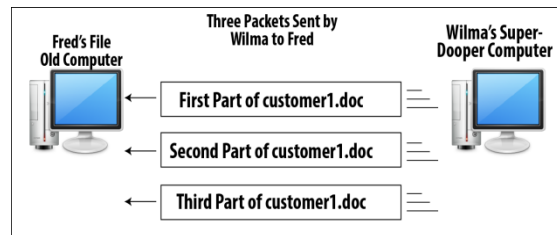


Figure 7.3: A Simple protocol for Error Recovery

We assume that Fred received all the three packets. The second packet has some bits that he is unable to understand. This may have occurred due to electrical interference (one electrical signal changes because of some other nearby electrical signal). If both NICs have agreed upon +5 volts to be a binary 0, and +10 volts to be a 1, then the receiving NIC will get confused upon receiving let's say +7.5 volts. A standard needs to be defined to address this issue. Both Wilma and Fred's computer software must agree to use the same networking protocol that provides a method to recognize and recover from errors. In the diagram below, we have shown such as error recovery mechanism:

- 1- The sender of the packet numbers the packets.
- 2- When the receiver (Fred) notices the error, he can send a message back to the sender (Wilma) asking her to send the specific packet again.
- 3- Wilma replies to Fred's request by resending packet 2.

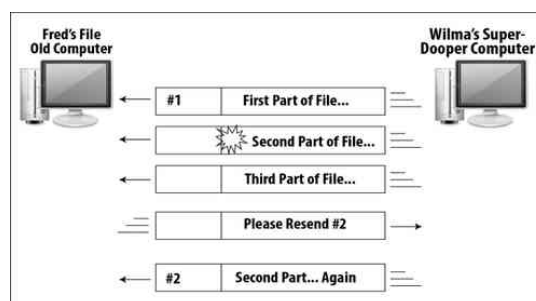


Figure 7.4: Error Recovery

Topic 9: Proprietary and Public Models

In this topic, we describe Proprietary and Public Models.

Networks began to be developed as part of each computer vendor's offerings by the late 1960s, and they became popular by the late 1970s. Each computer vendor created its own networking model, which helped computers from that one vendor communicate easily. At the advent of networking, the two largest computer vendors in the world were: International Business Machines (IBM) and Digital Equipment Corporation (DEC). IBM created its own networking model called Systems Network

Architecture (SNA), and DEC created its own DECnet. Each of these vendor-proprietary networking models allowed networks to be created and implemented and they worked but they were issues. Proprietary networks cannot communicate with other networks. IBM computers could not communicate with DEC computers.

Let's assume a company that owns some IBM computers and some DEC computers. How the company can form a network? The company should have two separate networks?

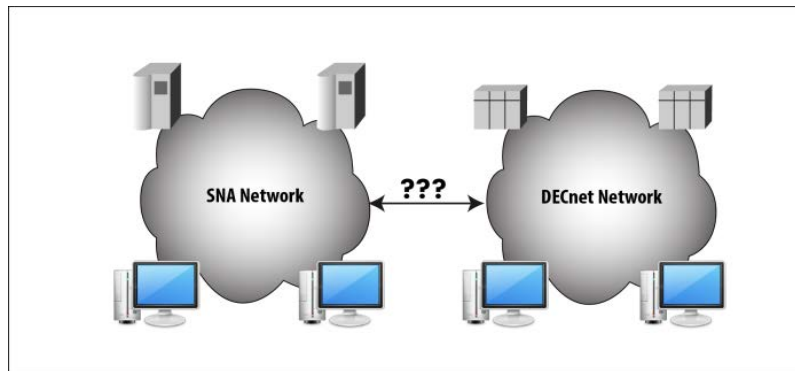


Figure 9.1: Non-Networking of IBM and DEC Networks in a Single company

Two solutions emerged: one short term, and one long term.

Short term solution

DEC made its computers conform to the IBM SNA model as IBM was roughly 10 times larger at the time in terms of gross revenues. So, DEC created software that converted between DECnet standards and SNA standards. DEC created a DEC-to-SNA gateway, which allowed the DEC and IBM SNA devices to talk.

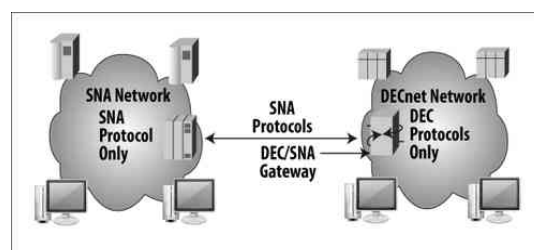


Figure 9.2: DECnet Emulating SNA Using a Gateway

Long term solution

The second, better, and long-term solution was to get IBM and DEC to stop using their proprietary networking models and use a public, open networking model. Due to this, all computers can communicate easily. Public network models provide pervasively popular networks. Today, we live in a world in which practically every computer uses the same public network model, called Transmission Control Protocol/Internet Protocol, or TCP/IP. No one vendor dictates the standards and protocols for it. Individuals from many companies and organizations have participated in the standards definition process. The Internet Engineering Task Force (IETF) manages the process of creating TCP/IP standards. IETF is open to any interested individual and its URL is <http://www.ietf.org>.

Topic 10: How TCP/IP Standard Grows?

This topic describes how TCP/IP standard grows.

The Internet Engineering Task Force (IETF) manages the creation and approval of TCP/IP standards and protocols. Each standard or protocol of the TCP/IP model is defined in a document. It is posted on the Internet so that anyone can look at it and comment before it becomes a Requests for Comments (RFC). TCP/IP is composed of a lot of individual protocols. The name TCP/IP is a combination of the two most popular protocols inside the TCP/IP model.

Transmission Control protocol (TCP)

TCP is described in RFC 793. TCP appends a TCP header in front of the user data. A header is a bunch of overhead bits added to the user data so that a protocol can do its job. Acknowledgment number is a field in the TCP header. TCP uses it for error recovery. On the receiver, TCP uses it to inform the sender which packet he expects to receive next and can thus be used to inform about erroneous packets.

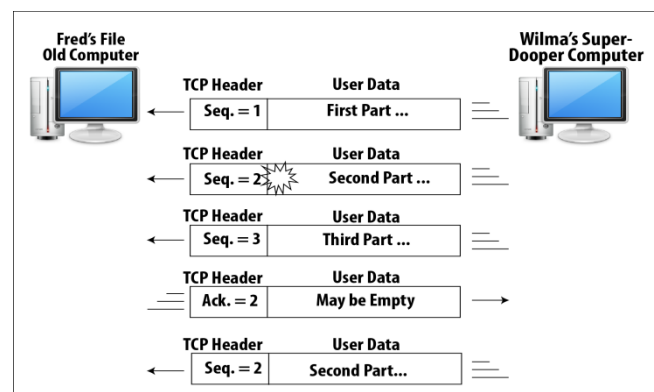
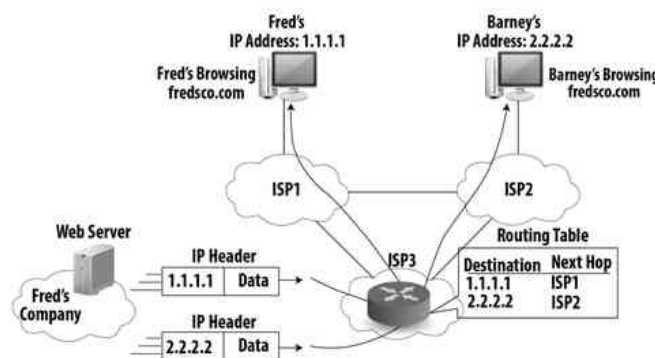


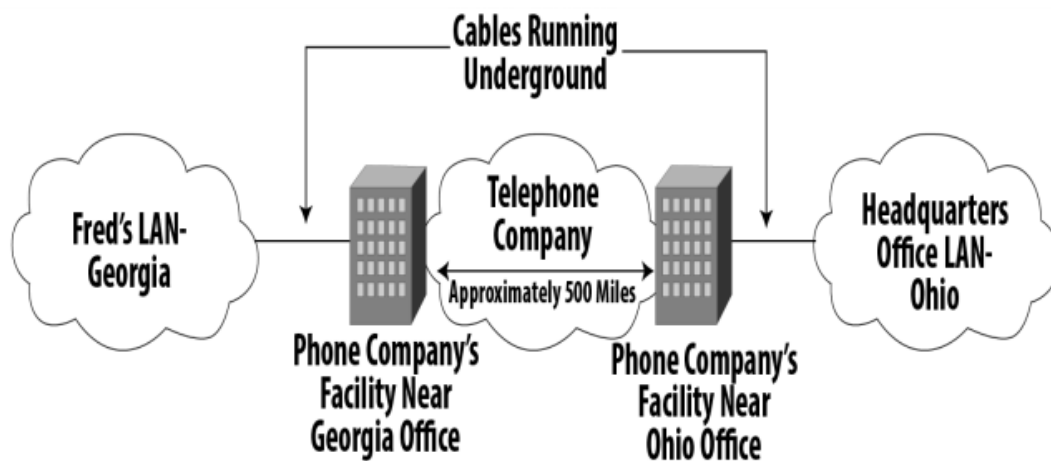
Figure 10.1: TCP Error Recovery

The Internet Protocol (IP)

IP is another TCP/IP standard protocol. It defines logical addressing and routing for the TCP/IP model and is explained in RFC 791. The best analogy for understanding IP's working relates to the postal service. Before you put a letter in the mailbox, you put an address on the front. Similarly, a computer sends a packet over the physical network by putting an address in front of the packet. In addition to TCP header, IP adds its own header. IP address is used for identifying a computer. Besides names, NTN/CNIC numbers are identifiers for different purposes. www.fredsco.com is a string for a computer which humans can remember easily. A number such as 3.3.3.3 is computer-friendly. Computers are identified by IP addresses. When packets are received by a router, the router looks at the IP address and decides where to forward the packet. A router uses a table, called a routing table. This is shown next.



There are other standards bodies who have already defined standards that TCP/IP can easily use. IETF happily references other well-known standards created by other standards organizations. For example, The Institute of Electrical and Electronic Engineers (IEEE) defines standards for LANs, which defines a type of network, or a part of a larger network, in which the devices are relatively close together. By “close”, we mean in the same building or in the same small campus of buildings. Like any other network, a LAN includes computers, hardware, software, and cabling that allow communications to happen. TCP/IP standards simply say "Use IEEE LAN standards if you want to use a LAN." Another example is the International Telecommunications Union (ITU), which defines standards for WANs. A wide-area network (WAN) defines a type of network, or part of a network, in which the devices are relatively far apart. A WAN is a network, or part of a network, for which the cabling must pass outside the property of one company. The distance might only be a few miles, or it might be thousands of miles!



Topic 11: Two-Well Known Networking Models

In this topic, we study TCP/IP and OSI networking models.

Systems can be implemented more efficiently and more effectively if the activities are broken down into layers. Layering helps in product development, and it helps in keeping each protocol simple. The reduced complexity makes for better products and more stable networks.

The TCP/IP Style

TCP/IP really is a set of protocols and standards, some not even defined directly by TCP/IP, that allow you to create networks. Some people segment the TCP/IP network model into the four layers. Others consider it to be a five-layer model. From a practical perspective, it does not matter.

TCP/IP Network Model		TCP/IP Network Model (Alternate)	
Application	HTTP (Web)	Application	
Transport	TCP	Transport	
Internetwork	IP	Internetwork	
Network Interface	LAN, WAN	Data Link	
		Physical	

Figure 11.1: TCP/IP Model

Each layer represents a general function that must be accomplished.

The OSI Style

While TCP/IP was being evolved into a legitimate networking model, a competing public networking model, called Open Systems Interconnect (OSI), was being

developed. The OSI model was developed by an organization called the International Organization for Standardization (ISO). The aim was to build the end-all networking model i.e. once implemented by all computers on the planet; it would allow pervasive communications among all computers from all vendors in all countries everywhere! OSI uses a seven-layer model, instead of the four (or five) layers in the TCP/IP model.

	OSI Model	TCP/IP Model	TCP/IP Protocols
7	Application	Application	HTTP, SMTP POP3
6	Presentation		
5	Session		
4	Transport	Transport	TCP, UDP
3	Network	Internetwork	IP
2	Data Link	Network Interface	Ethernet, Frame Relay, PPP
1	Physical		

Figure 11.2: OSI & TCP/IP Models

TCP/IP versus OSI:

OSI was developed much more slowly than TCP/IP. TCP/IP took over the marketplace before OSI could be finished. ISO plays an active role in standards development even today, working with the IETF, ITU, and other standards bodies.

Topic 12: What is a Local Area Network?

This topic describes the basics of local area network.

A LAN is a network in which the devices are relatively close together i.e. in the same building or in the same small campus of buildings. The simplest LAN consists of at least 2 computers, networking software, a cable and network interface cards (NICs). It is shown next.

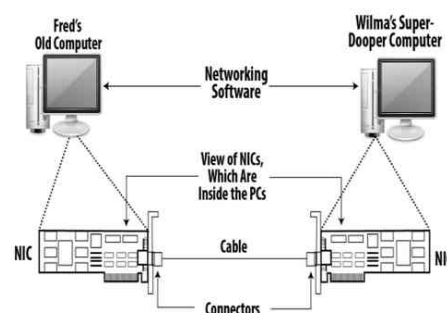


Figure 12.1: Components of a simple Network

Assume Fred opens a file on his computer that resides on Wilma's disk drive. Then, prints the file on the printer connected to Wilma's computer, and finally, saves the file back on Wilma's disk drive. This sequence is shown

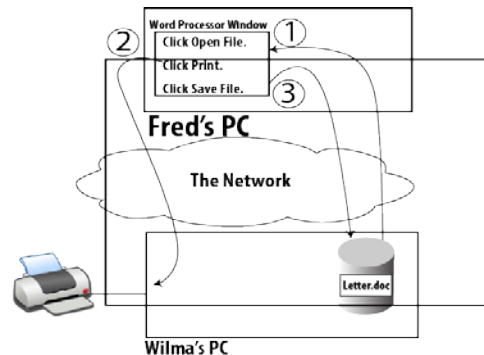


Figure 12.2: Basic Flow with Fred Using a file/ print server

A file is just a bunch of bits. In the previous example, the network needs to be able to transfer a bunch of bits from one computer to another. To send a binary code from one device to another, the sending device puts some electricity on the wire. A NIC can vary the voltage level of an electrical signal to different values. One value represents a binary 1 while the other means binary 0. Assume that a company employs an encoding standard that makes the NICs to define a binary 0 with a voltage of ± 5 volts, and a binary 1 with ± 10 volts. This is shown next.

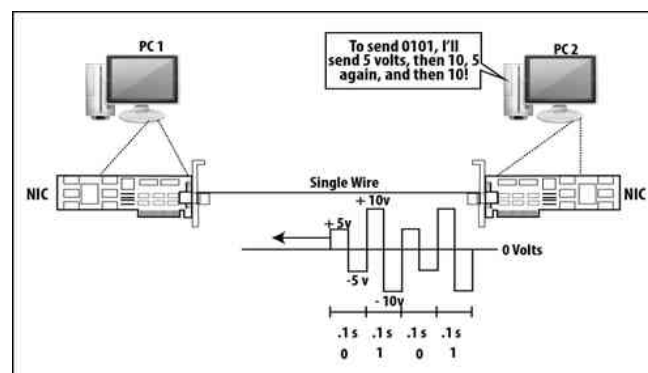


Figure 12.3: Basics Data of Transmission across a wire

PC2 wants to send a binary value 0101. It generates some electricity on the wire according to the encoding scheme. PC1, on the other end of the wire, senses the incoming electrical signal. It interprets the electricity, using the same set of encoding rules to mean 0101, exactly as PC2 intended. **Digital Transmission** is the use of discrete, constant values, which are then instantly changed to other possible discrete values. When both sender and receiver agree to encoding scheme, and the

rate at which the bits are transmitted over the wire. The receiver (PC1) must think about the electrical signal at different points in time, on a regular interval. The sender (PC2) must use the same regular time interval to decide when it should change signal.

An Example

Assume PC2 varies the voltage to mean either 0 or 1 every .1 seconds, and PC1 samples the incoming electrical signal every .1 seconds. The speed of this connection is 10 bits per second (bps). Digital Transmission does not work correctly if the two PCs do not agree on the transmission speed. For example, if we assume that PC2 thought the speed was 10 bits per second, meaning it should encode a new bit every 1/10 of a second. If PC1 thought that it should be receiving a bit 20 times per second, it would sample the incoming electrical signal every 1/20 of a second. PC1 would think it was sending 10 bits each second, and PC2 would think it received 20 bits. In real life, LANs typically run at much higher speeds, with a slow LAN transmitting at 10 million bits per second (Mbps, also called megabits per second). Can PC1 and PC2 talk simultaneously in both directions over the same wire? The electrical signals would overlap, and neither PC1 nor PC2 could understand what was sent. This is shown next.

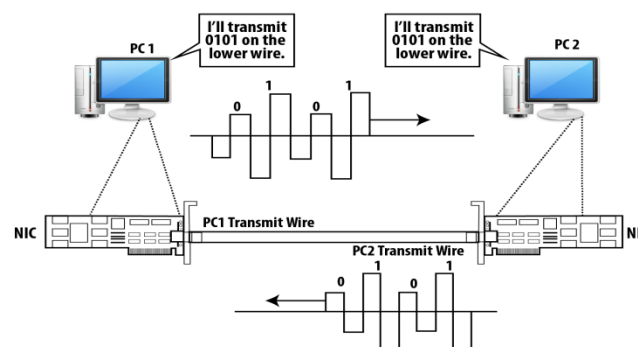


Figure 12.4: Concurrent Data Transmission across Two Different Wires

Topic 13: LAN Cables and Connectors

In this topic, we study LAN cables and connectors.

Cabling and connectors are used to manage the wires.

Cables

The copper wires that networking cards use are encased inside a cable which is made from plastic. A thin plastic coating painted onto each wire helps prevent it from breaking. Different colors of plastic coating for each wire helps in looking at each end of the cable and figuring out which wire is which.

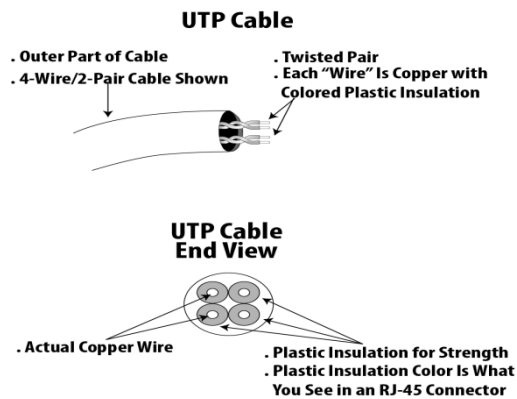


Figure 13.1: Typical LAN Cable

The electrical currents on the wire can change when electrical signals that exist in the air caused by other wires or other nearby electrically powered devices – Electromagnetic interference (EMI). A computer might misinterpret a 0 as a 1 or a 1 as a 0, or might not understand what the sender really sent. To reduce electrical interference, the wires are twisted together in pairs. Such a pair is called a twisted pair. Shielding can be added to the cable to reduce EMI. The cable becomes less bendable but more expensive. Such a cable is called shielded twisted pair (STP). This is shown next.

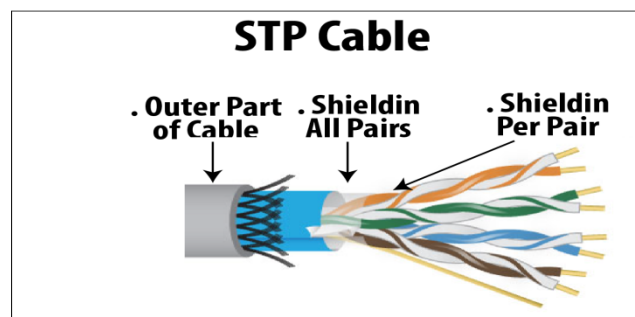


Figure 13.2: Shielded Twisted Pair (STP) cabling

Unshielded cables are called unshielded twisted pair (UTP). Mostly, the LAN technology employs less expensive UTP cabling. STP cabling is preferred in environments where significant EMI exists.

Connectors

They line up the wires on the end of a cable. Each colored wire has a specific reserved place inside the connector. A pin is a physical position in the end of the connector in which the copper part of the wire sits.



Figure 13.3: Typical Networking connector (RJ 45)

The Telecommunications Industry Association (TIA) and Electrical Industries Alliance (EIA) define standards for which wires fit into which pins when you make a cable for use with LAN. Eight wires can fit into an RJ-45 connector. EIA/TIA standards suggest numbering schemes for the eight pin locations and the pairs of wires. Two of them are shown next.

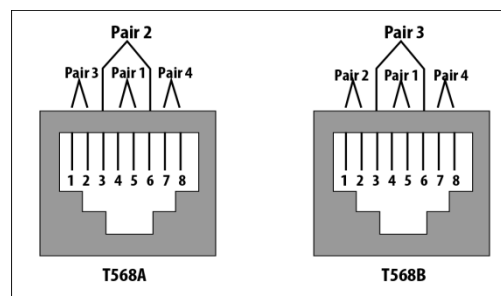


Figure 13.4: Pinout options for (RJ 45) Connectors

The standard also specifies which color of wire goes into pin position 1, 2, and so on. The **RJ-45 Color Code** scheme is shown next.

RJ-45 Color Code

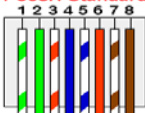
T-568B Standard



Pin #1



T-568A Standard



Pin #	Ethernet 10BASE-T 100BASE-TX	EIA/TIA 568A	EIA/TIA 568B or AT&T 258A
1	Transmit +	White with green stripe	White with orange stripe
2	Transmit -	Green with white stripe or solid green	Orange with white stripe or solid orange
3	Receive +	White with orange stripe	White with green stripe
4	N/A	Blue with white stripe or solid blue	Blue with white stripe or solid blue
5	N/A	White with blue stripe	White with blue stripe
6	Receive -	Orange with white stripe or solid orange	Green with white stripe or solid green
7	N/A	White with brown stripe or solid brown	White with brown stripe or solid brown
8	N/A	Brown with white stripe or solid brown	Brown with white stripe or solid brown

Figure 13.5: Color code scheme of RJ45 Connector

NICs send data over the twisted pair that uses pins 1 and 2 of an RJ-45 connector and receive data on the twisted pair that uses pins 3 and 6. Let's cable one end of a wire in pin 1 of one connector, and the other end into pin1 of the other connector. Pin 2 on one end of the cable connects to pin 2 on the other side and so on. This arrangement is called a straight-through cable. This is shown next.

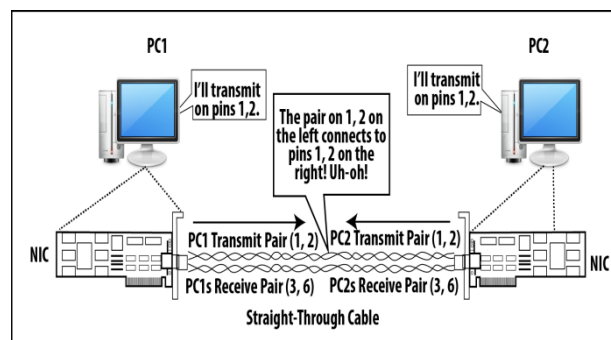


Figure 13.6: Both PCs Using the Same Pair (Lane) to send data

NICs of both PCs send data at pins 1 and 2. That electricity goes over the wires and enters the other NIC on pins 1 and 2. But, the NICs aren't receiving data on pins 1 and 2! Hence, both PCs send, but neither receives data. If we connect the wire at pin 1 on one end of the cable with pin 3 on the other end; the wire at pin 2 with pin 6 on the other end; the wire at pin 3 with pin 1 at the other side; and the wire at pin 6 with pin 2 at the other side. This arrangement is called cross-over cables. The NICs of the two PCs can receive the data sent by the other device! This is shown next.

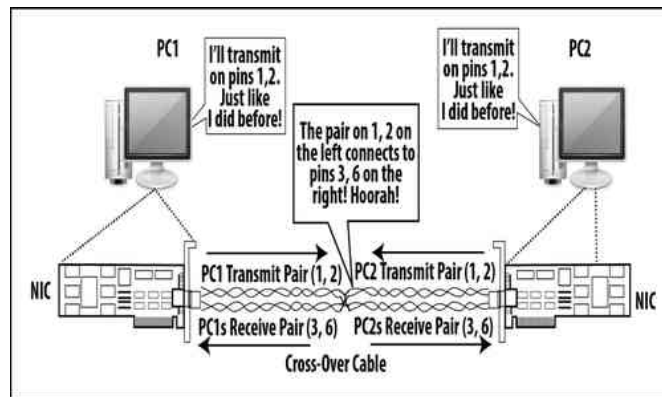


Figure 13.7: Both PCs Using Different Pair (Lane) to send data

Topic 14: Ethernet Hubs

This topic explains the basics of Ethernet hubs.

Ethernet is by far the most popular type of LAN today. Ethernet LANs are highly used at work and school. To connect to other PCs, you would need more LAN NICs, and your PC probably does not have enough room for all the cards. Also, you would need to run cables between your PC and all the other PCs. If you tried to do this for 100 PCs on the same floor of the building, and every PC wanted to connect to every other, you would have 99 cables connected to 99 NICs inside each PC!

The alternative to running a cable to every other PC is to run a cable from each PC to a wiring closet and connect the cables to a networking device, called an Ethernet hub. An Ethernet hub allows the electrician to cable each device to the hub using only a single NIC and single cable. It eliminates the cabling problem. This is shown next.

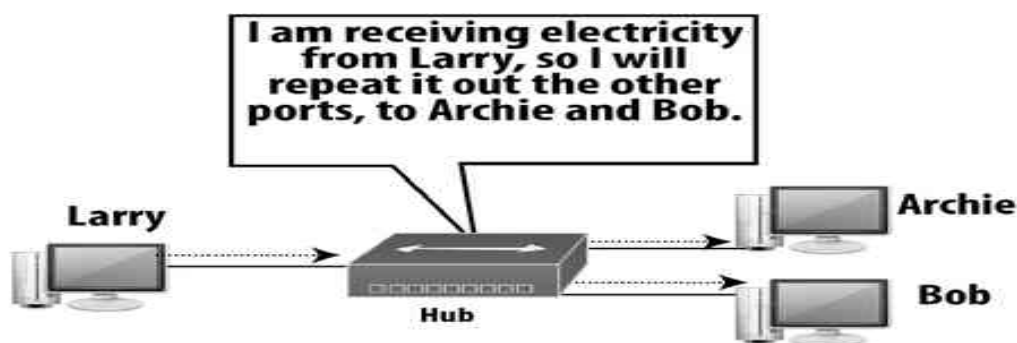
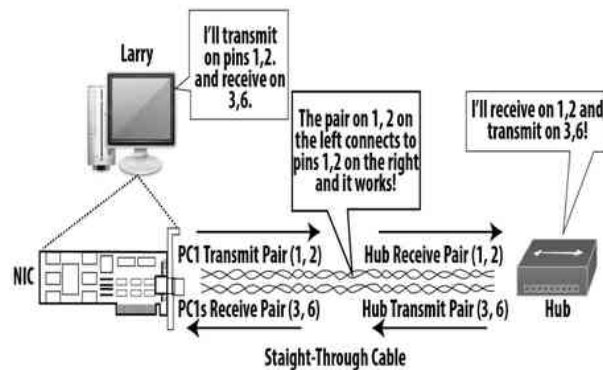


Figure 13.6: Ethernet Hub repeats everything it hears

The hub simply listens for incoming electrical signals, and when received, the hub repeats the same electrical signal to every other device that's connected to the hub. The hub expects straight-through Ethernet cabling between itself and the PCs. This is

opposite of Ethernet's pin configuration. A straight-through cable works between a PC NIC and a hub. This is shown next.



Hubs Logic:

- 1- Receive traffic on pins 1 and 2 on each physical interface.
- 2-When received, repeat the same electrical signal out all other ports, except the one in which the data was received.
- 3-When repeating out other ports, repeat the traffic out pins 3 and 6 so that the PCs will be listening.

Topic 15: Ethernet Frames and Collisions

In this topic, we study Ethernet frames and learn why collisions occur.

Ethernet is world's most popular LAN standard. It consists of a set of standards and protocols for LAN communication, as defined by the IEEE. For instance, Ethernet standards define how a network interface card (NIC) should encode binary 0s and 1s on a wire by varying the voltage. Before a PC could ask a NIC to send data, the PC must encapsulate the user data inside an Ethernet frame. i.e. take the data and add a header and trailer to it. Ethernet standard defines the formats of headers and trailers. The resulting bunch of bits created including the Ethernet header and trailer is called an Ethernet frame. This is shown next.

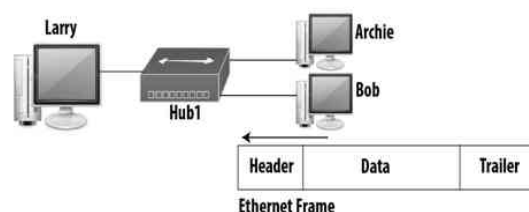


Figure 15.1: Ethernet Frame sent from Bobs to Larry

The information in header and trailer are used by the computers, NICs, and networking devices to make the Ethernet work smoothly. The first 8 bytes in an Ethernet header are called the preamble. The preamble contains alternating 1s and 0s so that the NICs receiving the data know that a new frame is being sent across the LAN. A collision occurs when two or more frames are sent over a single twisted pair at the same point in time. The result is that none of the frames is intelligible. Even when all devices follow Ethernet rules, collisions can happen. Let's focus on two key facts. **Fact1:** When two or more electrical signals travel over the same pair, both electrical signals are distorted and become a single signal. The receiving device cannot interpret the signal as 0s and 1s. **Fact2:** A hub repeats received electrical signals out on all other physical ports on the hub, except the one over which the signal was received, even if other electrical signals are already being repeated. This is shown next.

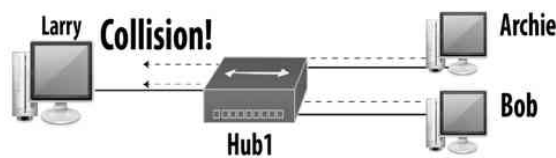


Figure 15.2: Collision between Bob's and Archie's Frames

What happens when both Bob and Archie send an Ethernet frame to Larry at the same time? The hub blindly repeats each of the frames sent by Bob and Archie out all other ports, including the one connected to Larry. Larry cannot understand either frame because the hub is trying to send both electrical signals over the cable to Larry at the same time.

Topic 16: How to avoid Collisions over Ethernet?

This topic discusses how to avoid collisions over Ethernet.

Imagine that you are connected to a hub, and you are currently receiving a frame. What would happen if you sent a frame at that time? This will cause a collision. Ethernet standards define a basic algorithm that helps reduce collisions, as well as defining what to do when collisions occur. The algorithm is called the carrier sense multiple access collision detect (CSMA/CD) algorithm

The CSMA/CD Algorithm

It is based on a simple concept: Listen before sending, and wait until you are not receiving a frame before you try to send your frame. Well, that's exactly what the algorithm calls for. Why not just wait a moment? This is shown next.

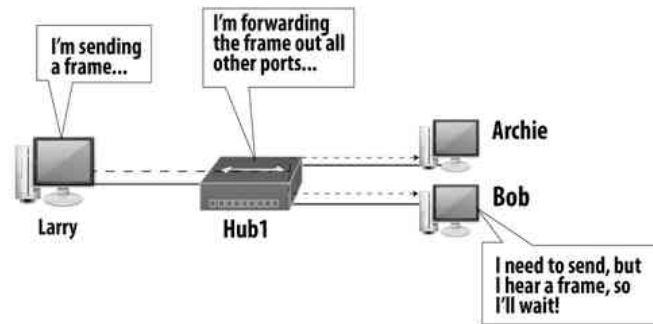
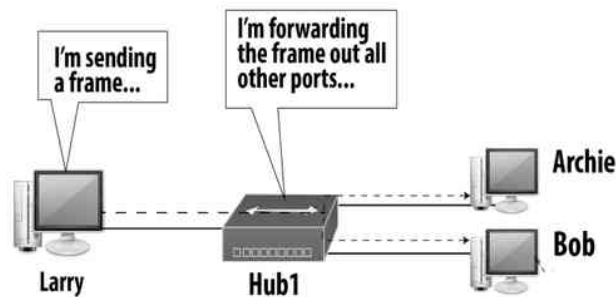


Figure 16.1: Collision Avoidance by Listening before sending

Even when you are using CSMA/CD, collisions can still occur. Why? Let's imagine Larry, Bob, and Archie are connected via an Ethernet. Currently, Larry is sending a frame. Bob and Archie want to send frames over the LAN. Possible after Larry finishes. This is shown next.



When no one is sending anything, there is no electricity flowing over the wires and we say that LAN is silent. As soon as both Bob and Archie stop receiving an electrical signal, both try to send their frames at roughly the same time. This is shown next.

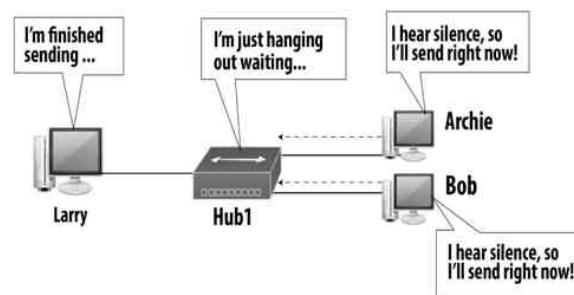


Figure 16.2: An imminent and Unavoidable Collision

The hub will repeat both frames out to Larry, so Larry will know about the collision. The hub won't forward Bob's frame back to Bob, Bob receives one frame. The same happens with Archie. Bob and Archie do not know there's a collision! When a NIC transmits a frame, it receives its own frame as well. This is called a loopback circuit. By using loopback, while Bob is sending a frame and if Archie also sends a frame at

the same time, the hub forwards Archie's frame to Bob and now Bob knows there is a collision.

CSMA/CD: what to Do When a Collision Happens

1. The senders of the collided frames send a jamming signal to make sure everyone knows a collision has occurred.
2. The senders of the collided frames independently pick a random timer value.
3. Each sender waits until his own random timer has expired and then tries to send his frames again.

Topic 17: Importance of Ethernet Addresses

In this topic, we learn the importance of Ethernet Addresses.

In an Ethernet LAN with a hub, a computer receives a lot of Ethernet frames due to the hub logic. Only some of them contain data that is meant for that computer. Consider Larry, Archie and Bob are connected over an Ethernet LAN with a hub. A frame sent by Larry to Bob will be received by Bob and Archie.

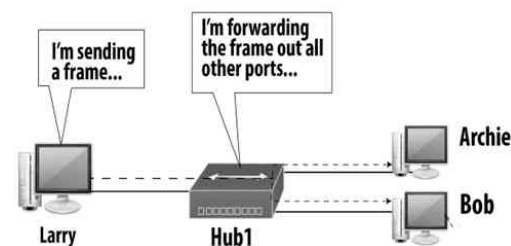


Figure 17.1: Collision Avoidance by Listening Before Sending

Archie should simply ignore the frame, and Bob should examine the data and process it. Practically, this becomes possible with the help of Ethernet Addresses. Before Larry sends the frame, he puts Bob's Ethernet address into a field in the Ethernet header called the destination address field. Next, we show working of this concept:

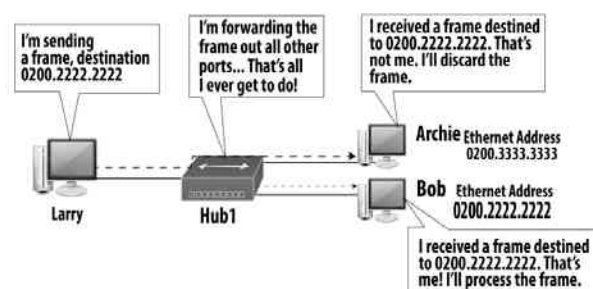


Figure 17.2: How Bobs Decides the frame was sent to Him

Each NIC has a unique Ethernet address, which is put onto the NIC by the NIC manufacturer. Ethernet standards specify that addresses be 48 bits long (6 bytes). The Ethernet header includes both a source address field and a destination address field. The source address identifies the Ethernet address of the NIC that sent the frame. When an NIC receives a frame, it uses the destination address to decide whether to process it or ignore it.

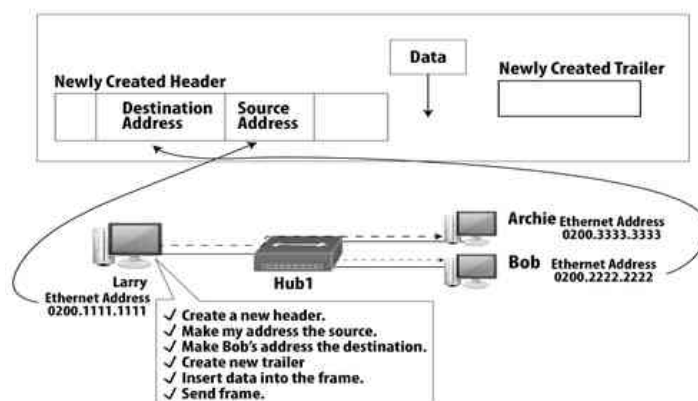


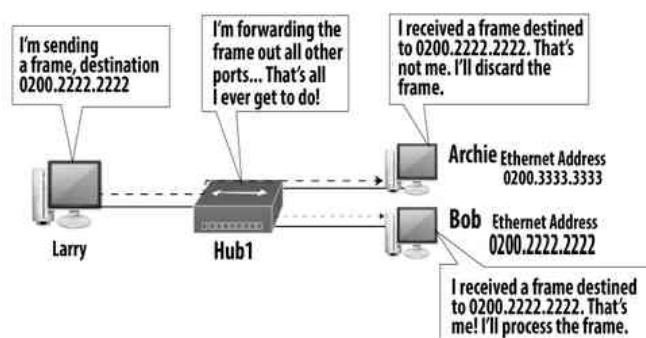
Figure 17.3: Ethernet Source and Destination Addresses

After Larry assembles the frame, he can use CSMA/CD logic to determine when he can send the frame.

Topic 18: Frame Check Sequence

This topic describes the usage of Frame Check Sequence.

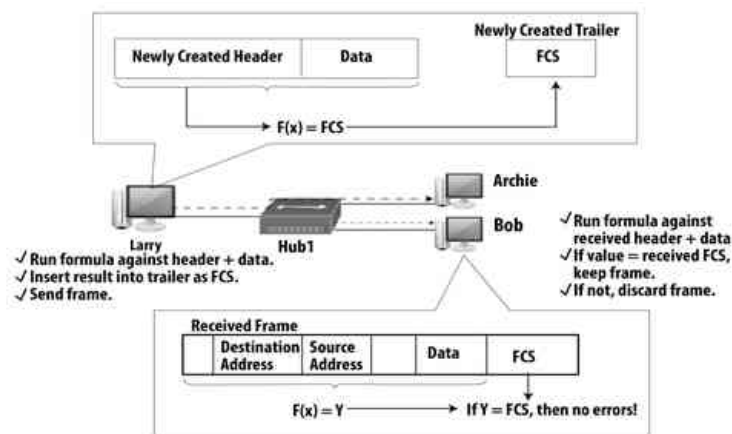
Consider Larry, Archie and Bob are connected over an Ethernet LAN with a hub. Larry sends a frame to Bob. This is shown next.



When Bob's NIC interprets the incoming electrical signal as meaning some string of binary 0s and 1s, it might misinterpret the meaning. Electrical signal has changed as it passed over the wire, affected by resistance in the wire, EMI, collisions, and other factors. Sender might send a 1 and the receiver might think it was a 0, or vice versa. Ethernet uses a field in the Ethernet trailer called the frame check sequence (FCS)

field. It holds a 4-byte number that is set by a sender and allows a receiver to determine whether a frame contains bit errors.

A sender runs a mathematical formula, with the input being the contents of the Ethernet frame to be sent, up to the Ethernet trailer. The sender places the results of the formula into the trailer FCS field. On the receiver's side, the receiver applies the same formula to the same part of the Ethernet frame (everything up to the trailer). If the result of the formula as calculated by the receiving NIC is the same as the value in the transmitted FCS, then no errors occurred. Otherwise an error has occurred.



Ethernet standards specify that the receiver should simply discard frames that have errors. The receiver does not request that the frame be re-sent, and the sender does not know that the frame was in error. This simple process is called error detection. Error recovery refers to the process whereby the receiver requests that the sender retransmit the frames that did not pass the FCS check. Transmission Control Protocol (TCP) performs error recovery, but Ethernet does not.

Topic 19: Two Ethernet Standards

This topic describes two different standards of Ethernet.

Ethernet was originally created in the 1970s by Robert Metcalfe and some others working for Xerox corporation. Later Intel got involved, and convinced them to put Ethernet logic on a computer chip, making the mass production of Ethernet cards less expensive. Also Digital Equipment Corporation (DEC), the second-largest computer maker at the time, got involved to support Ethernet. The original standard, as defined by those three companies, came to be called DIX Ethernet. The final version created by these three companies is known as Ethernet Version 2. In the 1980s, the IEEE while standardizing several LAN standards, created a committee to define the Ethernet standards and protocols. The committee was named the 802.3 committee. Another committee was created by the IEEE to define the common LAN features and was known as the 802.2 committee. The 802.3 standard is also called

Media Access Control (MAC), and the 802.2 standard is also called Logical Link Control (LLC). Next we show the header defined by 802.3 and 802.2, as well as the trailer, as defined by 802.3.

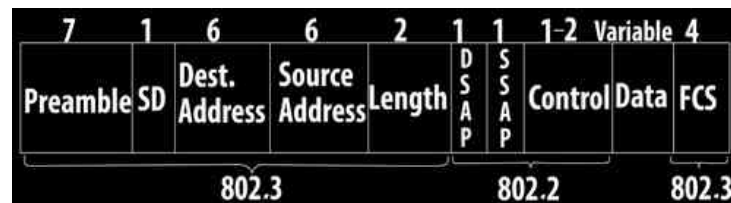


Figure 19.1: IEEE Ethernet Headers and Trailers

The preamble with starting delimiter (SD) serves to “wake up” the receiving adapters and to synchronize their clocks to that of the sender’s clock. The original DIX Ethernet specifications used an 8-byte preamble field. IEEE renamed the eighth byte of the preamble to starting delimiter, with no change in the value of the field. Ethernet addresses are often called MAC addresses and are 6 bytes in length.

IEEE Ethernet Technologies

The early 10BASE-2 and 10BASE-5 standards specify 10 Mbps Ethernet over two types of coaxial cable, each limited in length to 500 meters. In mid-1990s, Ethernet was standardized at 100 Mbps, which is 10 times faster than 10 Mbps Ethernet. 100BASE-T was defined for twisted pair copper wires and 100BASE-FX was defined over fiber. 100 Mbps Ethernet is limited to a 100 meter distance over twisted pair and to several kilometers over fiber. Gigabit Ethernet offers a raw data rate of 1,000 Mbps. 10 Gbps Ethernet was initially operating over optical fiber, but is now able to run over category UTP cabling.

Topic 20: Working of a LAN Switch

In this topic, we learn how a LAN switch works.

Only one device can transmit data at a time over a LAN connected via a hub. A switch allows multiple devices to transmit at the same time, increasing the amount of data that can be sent over the LAN. Let’s assume PCs of Fred, Barney, Betty and Wilma are connected over a LAN via a hub. Fred wants to send a frame to Barney while Betty wants to send a frame to Wilma. **Hub’s logic:** When I receive an electrical signal, I repeat it out on all ports, except the port over which the signal was received. **CSMA/CD logic:** A PC needs to listen before sending i.e. if you're currently receiving a frame, wait until it is finished before you try to send your frame. Otherwise a collision can occur.

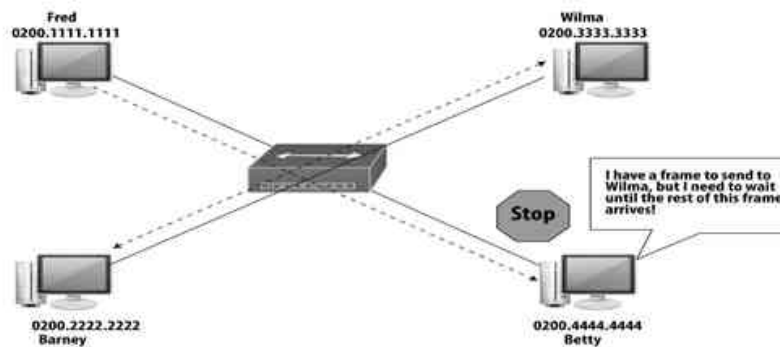


Figure 20.1: Betty Waiting on Fred's Frame that was sent to Barney

It's like paving a beautiful wide road and only having one lane, so everyone must wait until one car finishes driving past before the next car can use the road.

Switch

A LAN switch provides the same cabling advantage that a hub does, while providing significant performance improvements. **Switch's logic:** When receiving a frame, examine the destination Ethernet address. Forward the frame out on the one port and only that port through which that address can be reached. In comparison to a hub, a switch does not simply repeat the electrical signal out on all other ports. A switch forwards frames selectively only forwarding the frame where it really needs to go.

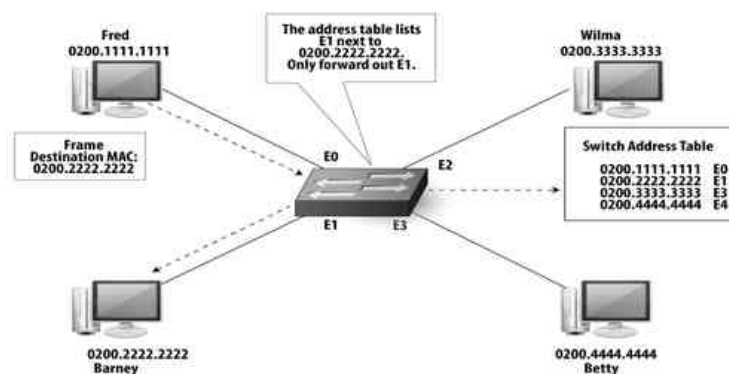


Figure 20.2: Switch logic

In real life, we can simply replace a hub with a switch, and use the same cables. The switch still uses a twisted pair for transmission and another for receiving traffic in each cable. It uses the same pin outs in the RJ-45 connector. The switch receives on pins 1 and 2 and transmits on pins 3 and 6 just like a hub i.e. you need a straight-through cable between the switch and each computer. The physical details can remain the same. To make a forwarding decision, a switch uses a table that lists the MAC addresses in the network. We can refer to it as either the switching table or the MAC address table.

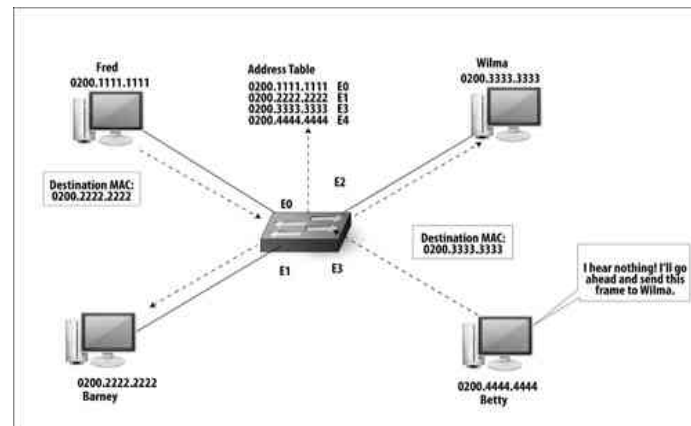


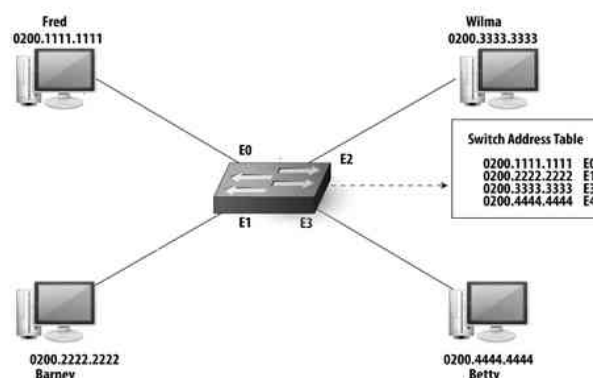
Figure 20.3: Fred Sending to Barney, While Betty send to Wilma

LAN switching logic improves LAN performance because the frame is not repeated to every computer that is attached to the switch. Betty still uses CSMA/CD logic. With two devices sending at the same time, the capacity of the LAN to forward frames doubles! If we are using a 10 Mbps Ethernet, separate transmissions occur, each at 10 Mbps. This LAN has 20 Mbps of capacity. Now imagine a switch with 24 ports, with the device on port 1 sending to the device on port 2; the device on port 3 sending to the device on port 4; and so on. This switch supports 12 x 10 Mbps, or 120 Mbps, of capacity.

Topic 21: Switch: Collision Avoidance

In this topic, we study how a switch avoids collisions.

Let's assume PCs of Fred, Barney, Betty and Wilma are connected over a LAN via a switch. When a switch receives a frame, it examines the destination Ethernet address. It forwards the frame out on the one port and only that port through which that address can be reached.



Let's discuss the scenario: what happens if everyone wanted to talk (send) to Fred at the same time? Keeping the logic of a switch in mind, the switch would try to

forward all three frames, which would cause a collision. A switch receives on pins 1 and 2 and transmits on pins 3 and 6.

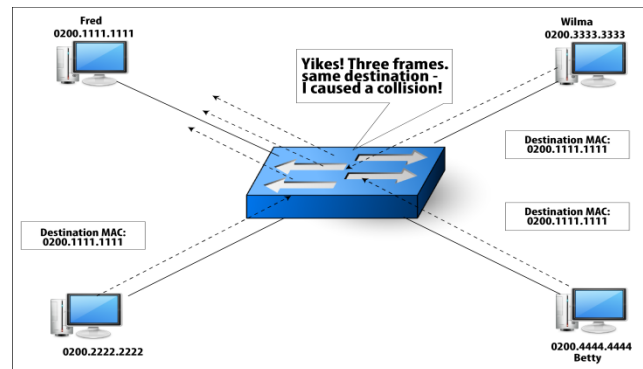


Figure 21.1: Potential collision when forwarding multiple frames

Buffers

To avoid sending all three frames at the same time, the switch uses buffers. Buffers consist of memory inside the switch that is used to store frames temporarily. The switch sends one frame, and it keeps the other two frames in buffers. After the switch finishes sending the first frame, it gets one of the frames from the buffers and sends it. Finally, the switch grabs the third frame from the buffer and sends it. By doing so, the switch usually avoids causing a collision.

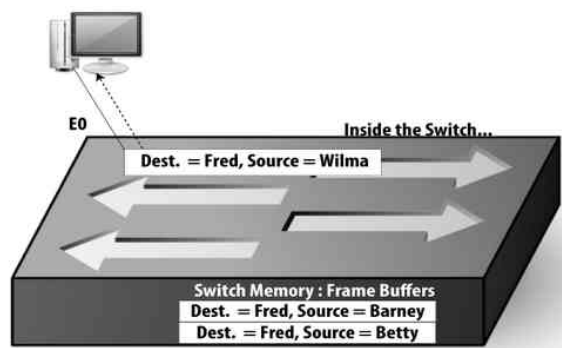


Figure 21.3: Switch Avoids Collisions by Buffering the Frames

Switch Logic Updated

When receiving a frame, examine the destination Ethernet address. Forward the frame out the one port and only that port through which that address can be reached. If multiple frames need to be sent out to the same port, send one frame and buffer the rest. As soon as the port becomes available, send the other frames. By buffering frames, delay will be introduced called buffering delay. How much? For instance, if those three frames were each 1250 bytes long, it would only take 3 milliseconds (.003 seconds) for all three to be sent over the cable with a 10 Mbps rate from the switch to Fred. If collisions had occurred, each frame would have taken

longer to reach Fred, and, both the collision and the time taken to resend the frames could have prevented other user traffic from crossing the LAN. Buffering of frames definitely improves LAN performance.

Topic 22: Full Duplex and Full Switching

This topic presents concepts of full duplex and full switching.

Let's assume PCs of Fred, Barney, Betty and Wilma are connected over a LAN via a switch. With carrier sense multiple access/collision detect (CSMA/CD), if a NIC is not receiving a frame, it can send. The switch avoids collisions by buffering the frame if the output port is busy.

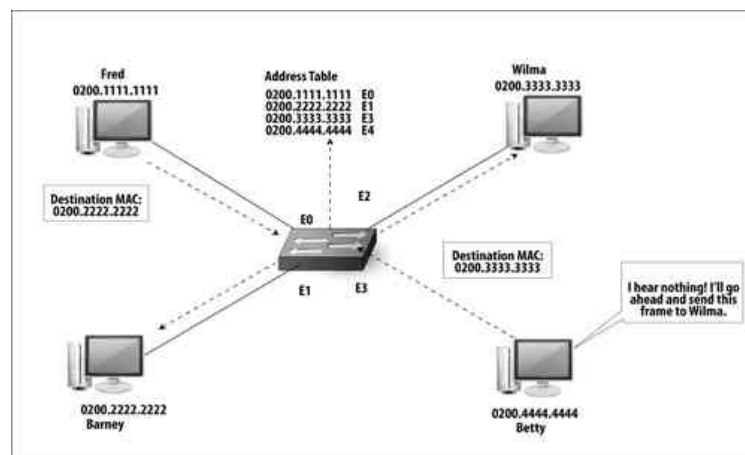


Figure 22.1: Potential Collision When Forwarding Multiple Frames

A computer NIC, using CSMA/CD, believes it should not send a frame: when the NIC is actually receiving a frame. Imagine that Fred is sending a frame to Barney. Barney wants to send a frame. (In this case, Barney wants to send a frame to Fred, but the destination does not really matter in this scenario.)

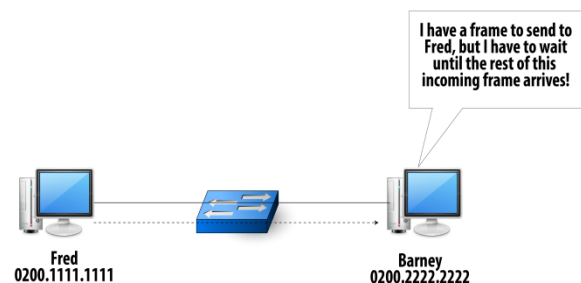


Figure 22.2: Barney Waiting to Send, When Fred Sends to Barney

With Barney's CSMA/CD logic enabled, he must wait before sending a frame. But can a collision really occur? Physically, there is a single cable between Barney and the switch. Barney sends on the pair using pins 1 and 2, and the switch sends to Barney on the pair using pins 3 and 6. The switch will buffer any frames if pins 3 and 6 are busy. There is truly no danger of a collision. Barney would not cause a collision, if he sends the frame. Barney can send and receive at the same instant in time as receiving the frame from Fred. This ability is called full duplex. The CSMA/CD imposes restriction of only sending or only receiving at one point in time. This is called half duplex. This is shown next.

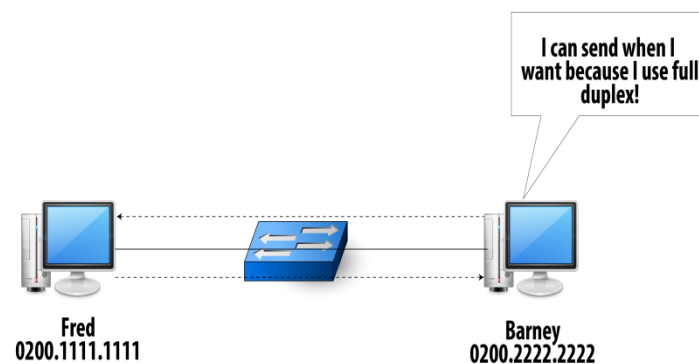


Figure 22.3: Barney and Fred Sending and Receiving at the Same Time Full Duplex

With switches and full duplex enabled, the LAN works like you have a two-lane road between the switch and each device, plus another two-lane road between each port on the switch. This is shown next.

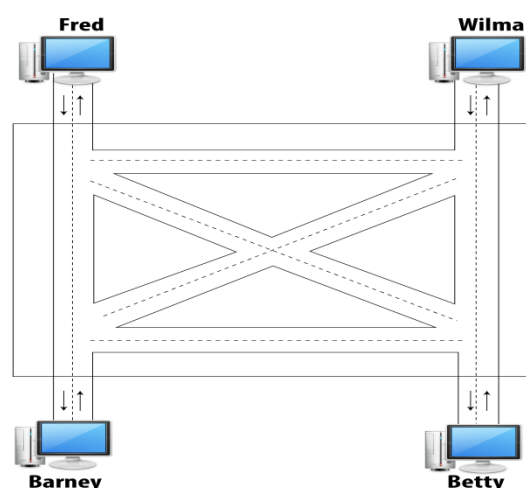


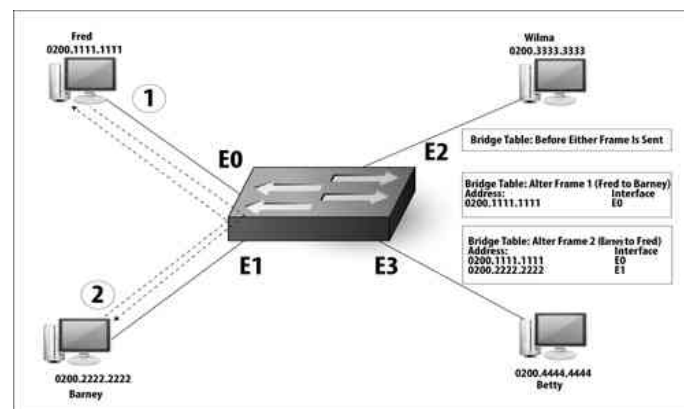
Figure 22.4: Full Duplex and Full Switching

Topic 23: Learning Ethernet addresses

In this topic, we study how Ethernet addresses are learnt.

A switch maintains a switching table or Media Access Control (MAC) address table. It makes its decisions about where to forward the frames based on that table. How does the switch figure out what should be in the MAC address table in the first place? The switch has no entries in the MAC address table when the switch first powers up. A switch dynamically learns MAC addresses and the corresponding ports. This process of learning MAC addresses is called learning.

Whenever a NIC sends an Ethernet frame, the NIC places its own MAC address into the frame as the source MAC address. The switch learns the MAC address of the sender of each frame by examining the source MAC address.



When a switch receives a frame, it sends the frame out on one port, based on a comparison of the destination MAC address and the MAC address table – forwarding decision. Conversely, by not sending the frame out on other ports, the switch has simply filtered the frame from exiting those ports, which is a filtering decision. What should a switch do with a frame that is sent before the switch has learned the MAC address table? The switch performs flooding.

Flooding

When a switch receives a frame whose destination address is not in the MAC address table, the switch forwards the frame out on all ports except the one over which the frame was received.

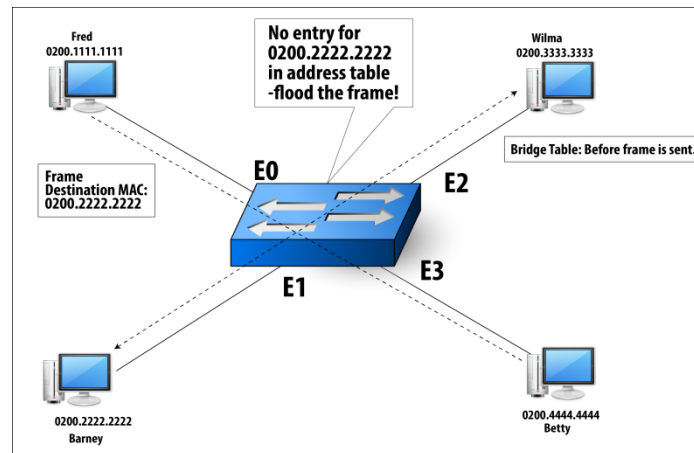


Figure 23.1: Switch Logic for Unknown Destinations

By flooding the frame, Barney, Wilma, and Betty all get a copy. Barney replies, and the switch enters Barney's MAC address in the table. Any future frames sent to Barney will be forwarded correctly.

An Ethernet or MAC address represents an individual NIC attached to a LAN –unicast MAC addresses. When a computer wants to forward a frame to all devices on the LAN, Ethernet defines a special MAC address, called the broadcast address. The value is FFFF.FFFF.FFFF.

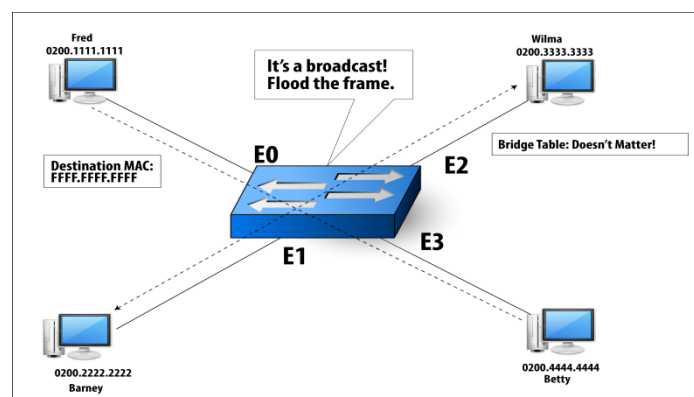


Figure 23.2: Switch Flood Broadcast Frames

Topic 24: Ethernet Network Speeds

This topic lists the different speeds that Ethernet supports.

When IEEE started working on Ethernet standards, Ethernet used a 10-Mbps transmission rate. The actual transmission of data— the bits cross the wires at 10 Mbps. Over the years, the IEEE has created many enhancements to the Ethernet standards. Increase in speed is one of the important enhancements. The first time Ethernet speed was enhanced, the IEEE created a working group named 802.3u. A specification called Fast Ethernet was created which runs at 100 Mbps. It is abbreviated as FE. The IEEE decided that the rest of Fast Ethernet should work the same as plain old Ethernet i.e. same headers and trailers, same CSMA/CD logic and same 6-byte MAC addresses. Only differences come in how the bits are encoded and the speed at which they are encoded. Vendors only needed to make small changes to existing Ethernet products to create Fast Ethernet products. This made them to be easily accepted in the marketplace. Today's enterprise networks widely use Fast Ethernet. The older 10 Mbps Ethernet, when using twisted pair cabling, was called 10BaseT. FE was called 100BaseT. The IEEE working groups 802.3z and 802.3ab created 1000 Mbps or 1 gigabit per second (Gbps). This standard is called Gigabit Ethernet (GigE). The difference between Gigabit Ethernet and Fast Ethernet is speed. To support the speed, the standards call for better cabling, but everything else is the same. This makes it to be quickly accepted in the market. The 802.3z working group defined how to do GigE over optical cable, and the 802.3ab working group defined how to do GigE over copper cabling. Optical cabling uses glass fibers instead of copper wire. The devices that are attached to an optical cable send light across the cable, instead of electricity, to encode 0s and 1s. The physics behind optical signals over optical glass-fiber cabling allow for higher speeds, longer distances, fewer errors, and better security, but at a higher cost than copper wiring. **10 Gigabit Ethernet** is defined by the IEEE 802.3ae working group. 10 GigE is used for short.

Because all the more advanced forms of Ethernet use the same header, switch forwarding and learning logic do not change based on what speed Ethernet is used. The switch can behave the same way as always, with some ports using Ethernet, some Fast Ethernet, some Gigabit Ethernet, and so on. Imagine that you want to support a bunch of users on a building floor, some with NICs that only support 10 Mbps 802.3, and some that only support 100 Mbps 802.3u Fast Ethernet. You might buy a switch that has 24 ports; 12 10-Mbps ports and 12 100-Mbps ports. Next we show the basic setup, with Fred using a 10-Mbps NIC and Wilma using a 100-Mbps NIC.

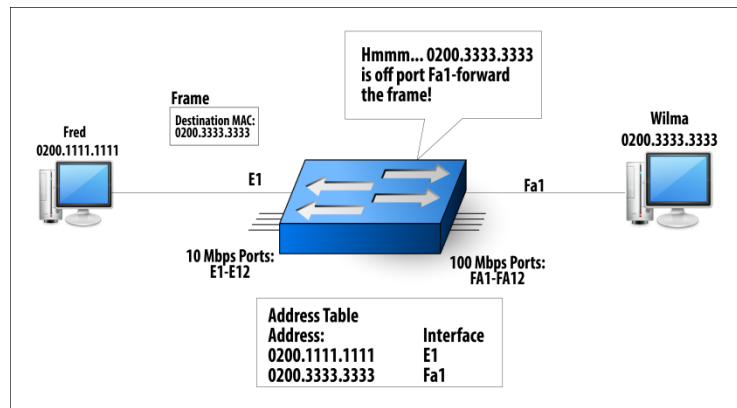


Figure 24.1: Supporting Multiple Speeds on a Single Switch

This type of switch does not provide an easy migration path from 10 Mbps to 100 Mbps. For the previous example, assume all FE ports are occupied. Can you add a new 100-Mbps NIC? Auto negotiation is an IEEE standard that allows the switch and NIC on either end of the cable to automatically negotiate to determine the speed. Full duplex or half duplex can also be negotiated. To perform auto negotiation, the switch and the NIC must support multiple speeds, as well as auto negotiation logic. "I want to use 100 Mbps," "Okay, me, too. Let's do it."

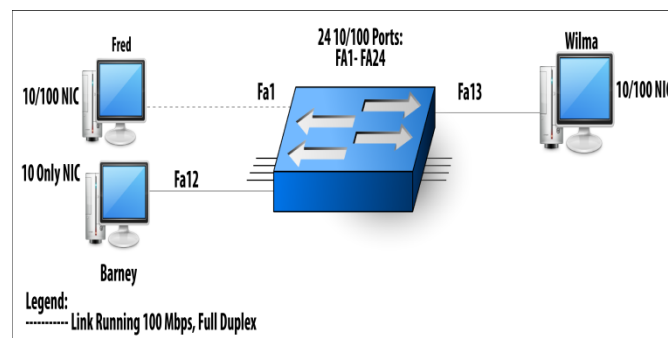


Figure 24.2: Auto negotiation with 10/100 Cards

Topic 25: Multiple Physical LANs

In this topic, we understand how physical LANs and multiple physical LANs handle broadcasts.

Hubs and switches behave differently. But process broadcast frames identically. A broadcast frame is an Ethernet frame that has a destination MAC address field set to FFFF.FFFF.FFFF. When a switch receives a broadcast frame, it forwards the frame out on all ports except to the incoming port. A hub works in the same fashion for all frames including broadcasts. A broadcast domain is a group of devices for which a broadcast frame sent by one device is received by all other devices in the same group. Next, we show three examples of broadcast domains. A single hub creates a single broadcast domain. A single switch also creates a single broadcast domain. A hub and a switch, connected together, also create a single broadcast domain.

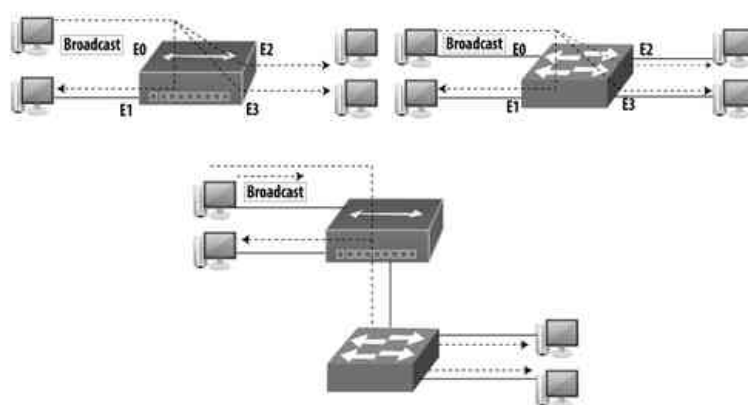


Figure 25.1: Three Broadcast Domains: A Hub, a Switch, and a Hub and Switch

A LAN consists of the devices inside a single broadcast domain. Let's imagine that you just took a job as a network engineer at a company with the small network shown next.

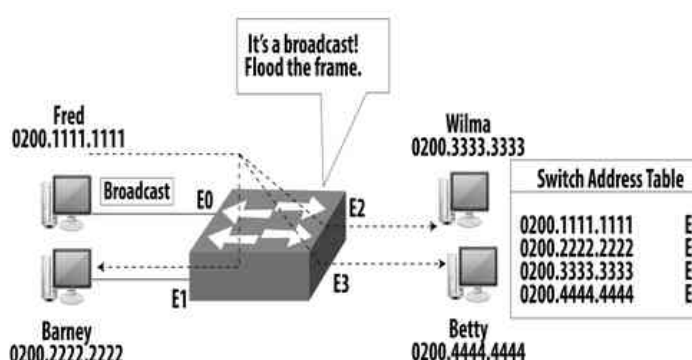


Figure 25.2: Small Physical LAN and You Are in Charge

Your boss told you that "We've got to get Fred and Barney on a different LAN than Betty and Wilma. They work with super-secret projects, and we can't meet our

security requirements if Betty and Wilma are on the same LAN". To put Betty and Wilma on a different LAN than the boys, you have to use two switches.

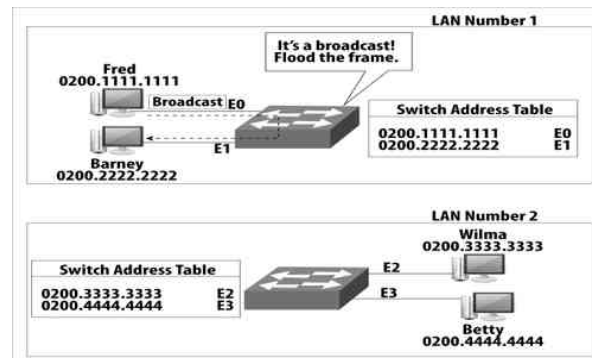


Figure 25.3: Two Physical LANs: Broadcasts Do Not Leave the Originating LAN

Both broadcast frames and unicast frames from the top LAN cannot be forwarded to the bottom LAN as there is no physical cable connecting the two switches in this case.

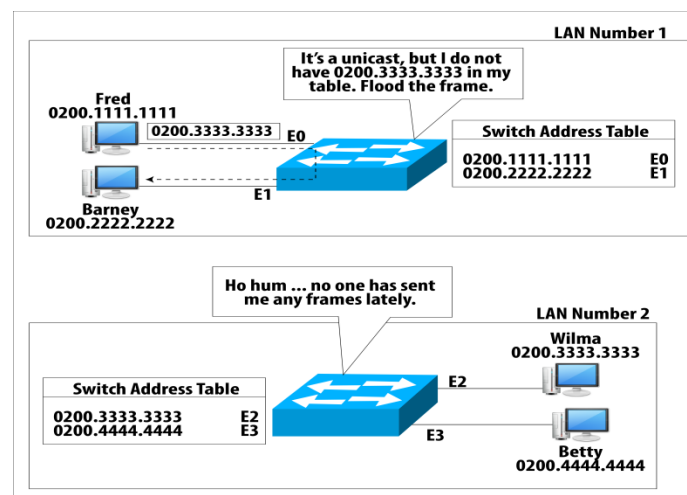


Figure 25.4: No Forwarding between the Two Physical LANs

Each LAN has an independent MAC address table as compared to the other LANs. Broadcasts originating in one LAN are flooded inside that LAN. Broadcasts originating in one LAN are not forwarded into the other LANs. Unicasts originating in one LAN are not forwarded into the other LANs.

Topic 26: Introduction to Virtual LANs

This topic explains how to create Virtual LANs.

A physical LAN or a broadcast domain is the group of devices for which a broadcast frame sent by one device is received by all devices in the group. To create multiple LANS, multiple LAN switches are required – expensive. VLANs allow you to create multiple LANs, but without requiring extra switch hardware.

A **Virtual LAN** is a broadcast domain, created by a switch, using a subset of the physical ports on the switch. LAN switch vendors include a feature in their products that allows you to create multiple broadcast domains in a single switch. A network engineer can configure some physical ports of a switch as if they are in one broadcast domain (one VLAN) and then configure other ports to be in a different broadcast domain (a different VLAN).

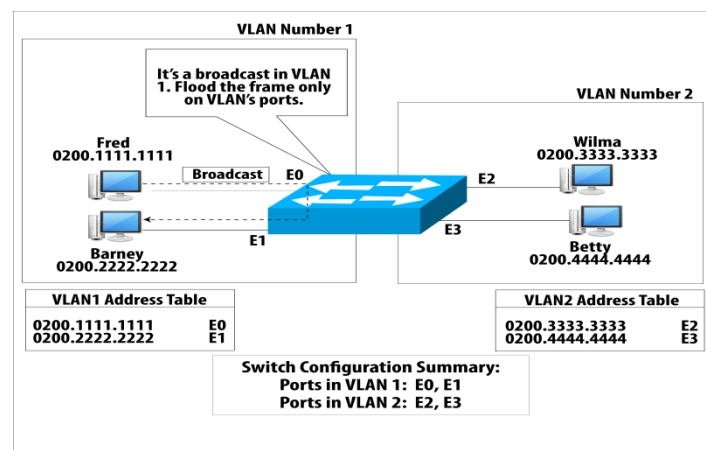


Figure 26.1: Two Virtual LANs: Broadcasts Do Not Leave the Originating VLAN

This network behaves just like it would with the two physical switches. However, you get the advantage of not having to buy another switch! In the previous example, the switch does learn all four MAC addresses, but the switch does not forward broadcasts or unicasts from one VLAN to the other. It keeps a separate address table for each VLAN.

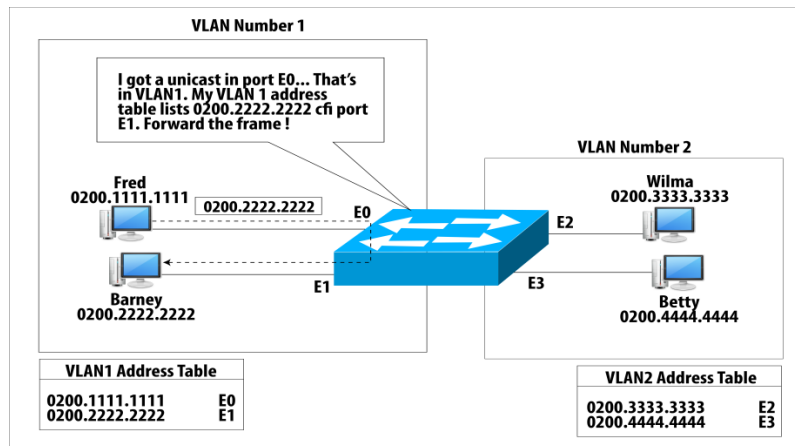


Figure 26.2: No Forwarding Between the Two VLANs

The switch knows that the frame came over port E0 and that E0 has been configured as part of VLAN1. The switch looks only at the VLAN1 address table and finds a match. So, the switch forwards the frame. Even if there had not been a match in the VLAN1 address table, the switch would have flooded the frame, but only out on ports in VLAN1. Therefore, neither Wilma nor Betty could get a copy of the frame. Broadcasts originating in one VLAN are not forwarded into the other VLANs. Unicasts originating in one VLAN are not forwarded into the other VLANs.

Advantages of VLAN

1. A PC must spend CPU cycles processing a received broadcast. As a LAN grows, every device has more broadcasts to process. Don't to put too many devices in a single VLAN.
2. Imagine payroll department of an organization moves to the same floor as the IT department. The payroll director is concerned that we will be on the same LAN as the IT group and sensitive payroll traffic can be disclosed.
3. Cost effective to just configure VLANs.

Topic 27: Packing VLAN's frames in a Trunk

In this topic, we discuss the need for VLAN trunking.

Let's assume 4 PCs owned by Fred, Betty, Barney and Wilma are connected via two switches with two PCs per switch. This allows devices on each switch to send frames to each other. The basic switch logic for each switch does not change when you use multiple switches. Both switches will populate their own address tables and each switch knows where to forward the frames.

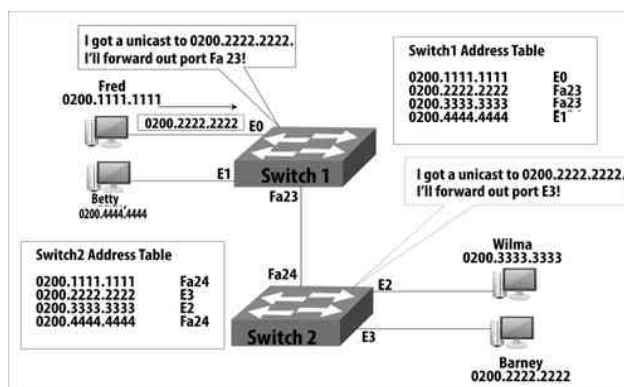


Figure 27.1: Forwarding Frames Between Two Switches

In the context of connecting multiple switches, the Ethernet cable segment between switch1 and switch2 is known as a trunk.

VLAN Trunking

Let's now add VLANs to the previous network such that Fred and Barney are connected to VLAN1 and Wilma and Betty are on VLAN2. Traffic must cross over the trunk between two switches. When a switch receives a frame over that trunk, it could be confused about which VLAN's MAC address table to use when deciding how to forward the frame. Should a switch look in both address tables?

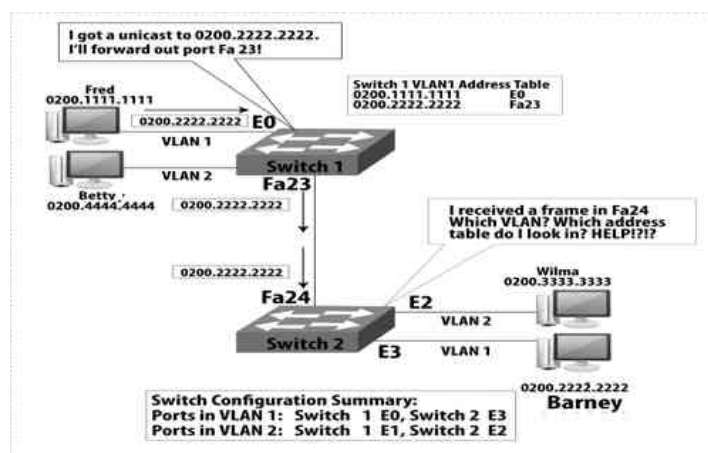


Figure 27.2: Switch2's Dilemma: Which VLAN?

To solve the problem, the switches use a mechanism called VLAN trunking. Before sending the frame over the Ethernet cable to the other switch, switch1 adds another header to the frame. An extra header identifies the frame as part of VLAN1. That tells switch2 which address table to use. Switch2 expects frames coming in over port Fa24 to have that extra header, so it can now process the frame.

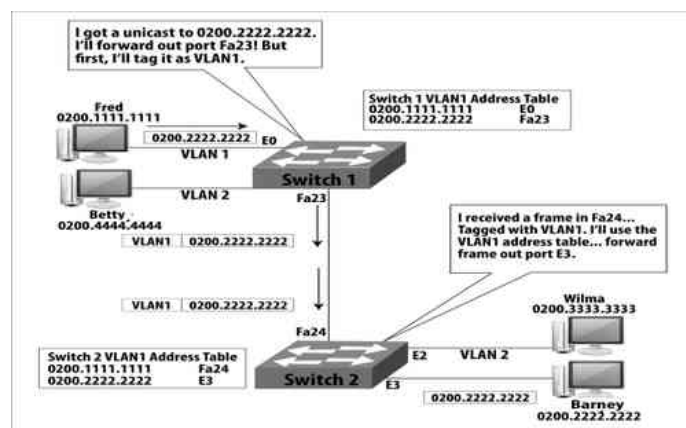


Figure 27.3: Trunking Header Tells Switch2 What to Do?

When switch2 needs to send frames out on port Fa24 to switch1, switch2 also adds a VLAN trunking header, knowing that switch1 expects to be able to find the header to identify the right VLAN. Initially, Cisco created its own standard and named it Inter-Switch Link (ISL). ISL defines the type of header that should be added to the frame, including the field in which the VLAN can be numbered. Later, the IEEE 802.1Q committee defined a standard for VLAN trunking called 802.1Q trunking, or simply "dot 1 Q." 802.1Q trunking differs from ISL. As a result, you need to pick between the two options. The switches on each end of the trunk must agree to which protocol to use, or trunking will not work.

Topic 28: Email: A Network Application

This topic describes how email works.

You can write a letter on paper and put it in an envelope. Write a correct name and address on the front of the envelope and put it in a mailbox. The postal service will try its best to deliver the mail to the right place. Put a return address on the envelope so that the recipient can reply by putting your address on the front of the envelope. To further elaborate the working of postal service, let's assume that you live in a house, and a postal worker typically comes by your place every working day. He leaves your mail in your mailbox. He also picks up any outgoing mail that you left either in your mailbox or in a centralized post office box (PO box) that was set up just for outgoing mail. This is shown next.

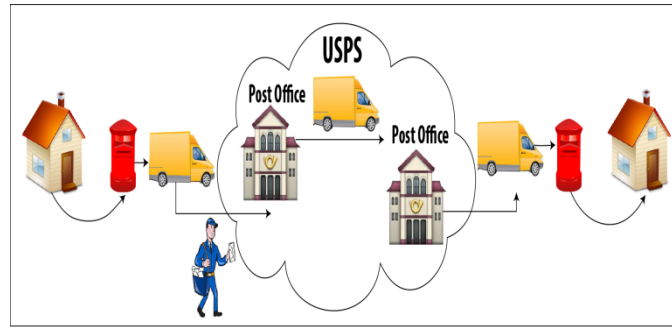


Figure 28.1: Postal Worker Picking Up and Dropping Off the Mail

Email allows you to do the same thing, but without the paper. You type some text, put that person's e-mail address at the top of the e-mail, and send the e-mail. By sending the e-mail, you do the equivalent of giving the e-mail to the postal service. The service delivers the e-mail, and the next time your friend checks his e-mail, he receives the message that you sent. Each e-mail includes the text you typed, the recipient's e-mail address, and your e-mail address. As the recipient now knows your e-mail address, he can easily respond to your e-mail. E-mail also enables you to send a message to multiple recipients at once. When you create and send an e-mail to your friend, your PC does not actually send the e-mail to your friend's PC. Instead, you would send the message to your e-mail server, which is the equivalent of dropping off a letter at the local post office. Your e-mail server would send the e-mail to your friend's e-mail server, which is the equivalent of the postal service delivering a paper letter to the post office near your friend. Then, at some point in the future, your friend would check his e-mail and retrieve the e-mail from his local e-mail server, which is the equivalent of retrieving his paper mail from his PO box.

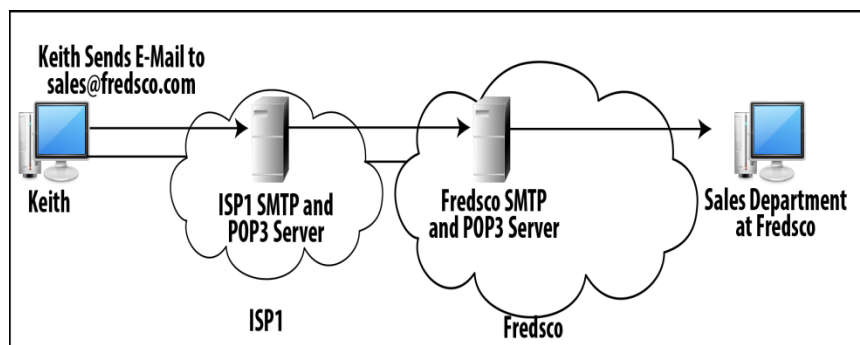


Figure 28.2: Sending E-Mail Using E-Mail Servers

Internet service providers (ISPs) have one (or more) e-mail servers. If you connect to an ISP from home, you would use that ISP's e-mail servers to drop off and pick up e-mail. If you use e-mail from your corporation's enterprise network, you typically use your company's e-mail servers to drop off and pick up e-mail. E-mail addresses have two parts: the name of the e-mail user and the name of the e-mail

server: barney@fredsco.com. Let's assume that Keith and Conner send e-mails to different people inside Fredsco. Next, we show why a two-part e-mail address is useful.

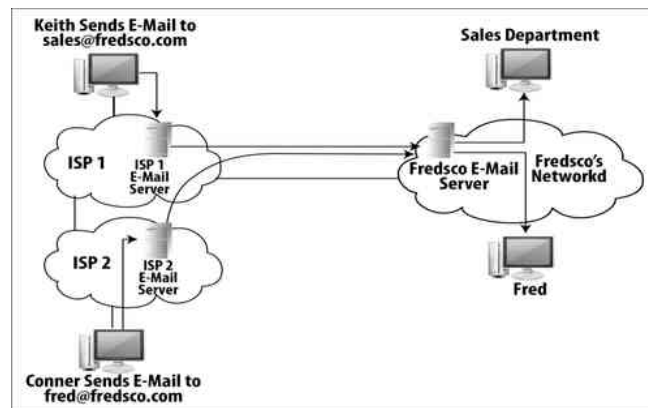


Figure 28.3: How a Two-Part E-Mail Address Is Used

The e-mail software on PCs does not think about the address at all. The PCs always send their e-mails to their respective local e-mail server. The servers look at the name after the @ sign – identifies the receiving side mail server. Local post office looks at the city and state on a letter, or the zip code, to figure out to which post office to send the letter.

Topic 29: TCP/IP Email Standards: SMTP, POP3

In this topic, we explain the use of Email standards such as SMTP and POP3.

Internet Message Format, Simple Mail Transport Protocol and Post Office Protocol Version 3 are the part of TCP/IP E-mail standard. Internet Message Format (IMF) is described in RFC 2822 and describes headers used to encapsulate the e-mail text, sender and receiver e-mail addresses. Simple Mail Transport Protocol (SMTP) is defined in 2821. It provides details about protocols for transmitting and receiving e-mails. Post Office Protocol Version 3 (POP3) and is defined in RFC 1939. It provides information about protocols for a client to retrieve e-mail from a server. A user uses e-mail client software to generate, send, receive, and read e-mails.

An **Email Client** software is a user interface, accepts text typed from the keyboard, understands what a user clicks on the screen, stores e-mails on the computer hard disk, and so on. E-mail TCP/IP protocols are implemented in the e-mail client software so that it can use the network. The e-mail client application is not the same thing as the application layer protocol.

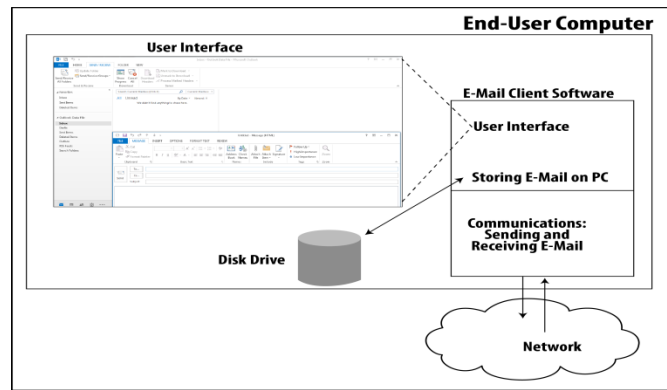


Figure 29.1: E-Mail Client Application and Its Use of Application Layer Protocols

An e-mail client sends not only the text of the e-mail message, but also a header. The header contains several fields, such as the recipient's and sender's e-mail addresses.

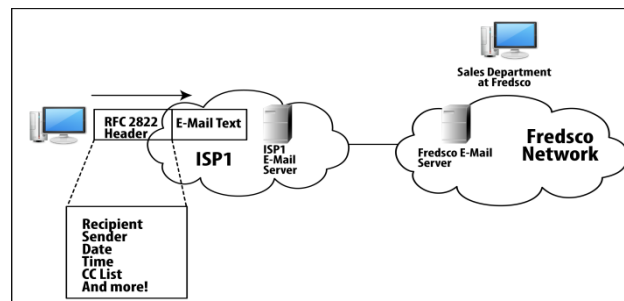


Figure 29.2: Sampling of the E-Mail Header Defined by RFC 2822

All the clients and servers know where to find all the information needed to forward the e-mail correctly. E-mail clients and servers use SMTP protocols to manage the process of sending and receiving e-mail. SMTP defines messages so that the e-mail clients and servers can manage the e-mail forwarding process. The Extended Hello (EHLO) command identifies the client. The Mail command tells the server that the client wants to either send or receive e-mail. The RCPT identifies recipient of e-mail. After the server replies with an acknowledgement (ACK) message to each of these first three commands, the e-mail can be transmitted.

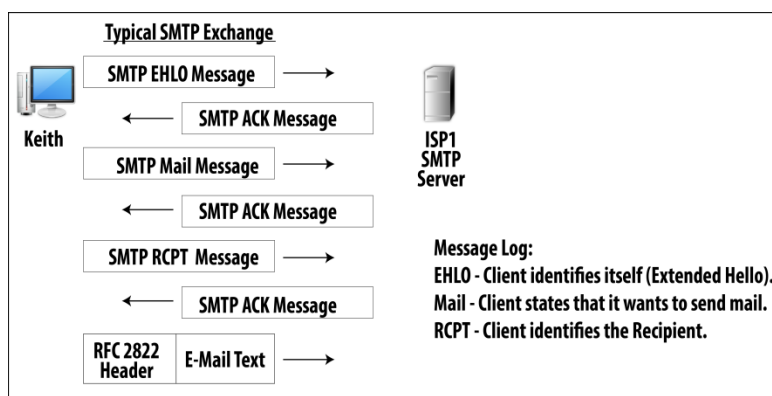


Figure 29.3: Simple SMTP Messages: Identifying the Client and the Recipient

The current version, POP3, was designed specifically for use between an e-mail client and its e-mail server. It cannot transfer e-mail between servers. POP3 allows for authentication, which refers to the process of one device identifying itself to another device via a name and password. The other device then decides whether the first device is allowed access.

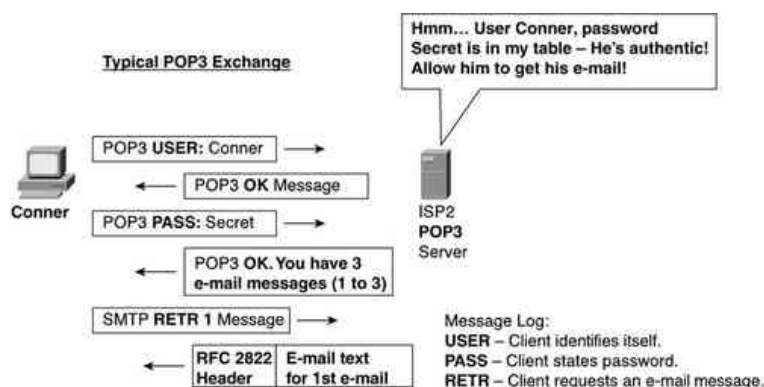


Figure 29.4: Basic Authentication with POP3

SMTP pushes emails between two devices. POP3 can be used to only pull the e-mail from the server. E-mail server (a single computer) needs to run both the server softwares. The SMTP server and the POP3 server on the same computer store and retrieve e-mails from the same message storage location.

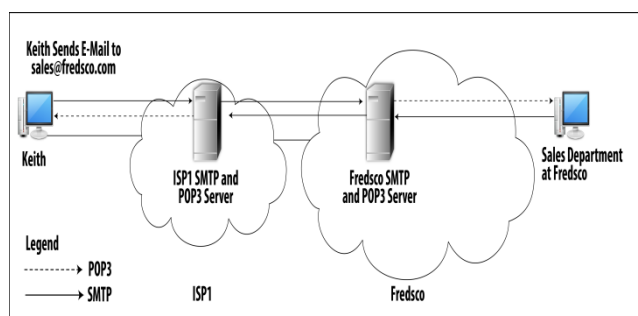


Figure 29.5 Typical Usages of SMTP and POP3

Topic 30: File Transfer Protocol (FTP)

This topic explains the working of file transfer protocol.

The working of FTP can be explained using a warehouse that a company might use to store goods. Imagine that a company has a bunch of stuff, and many people inside the company might need some of the stuff from time to time. Some of the stuff is big, and some of the stuff is small. The company leases a warehouse space and puts

its stuff in the warehouse. Then the company gives everyone a key, and anyone who needs some stuff can go get it when needed.

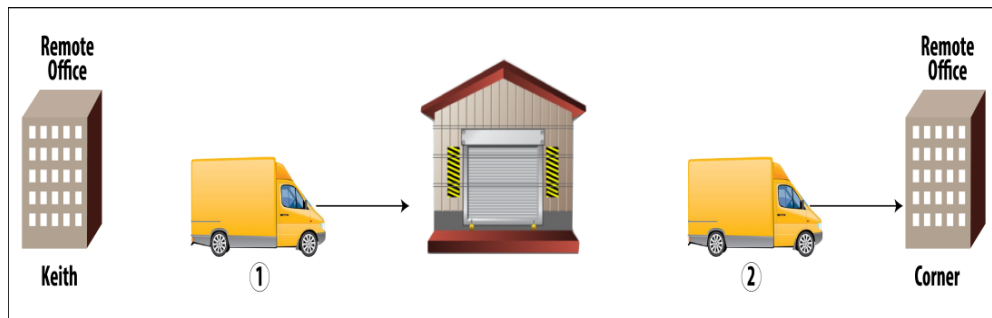


Figure 30.1: Warehousing Process for Transferring Stuff

Keith should put the widgets in the right place. Conner should know where to find Keith's widgets. Warehouse becomes useless if no mechanism to find widgets and keep track of inventory. A file transfer application called File Transfer Protocol (FTP) works by putting things on an FTP server and get them off the FTP server. Documentation is also needed as to where things are.

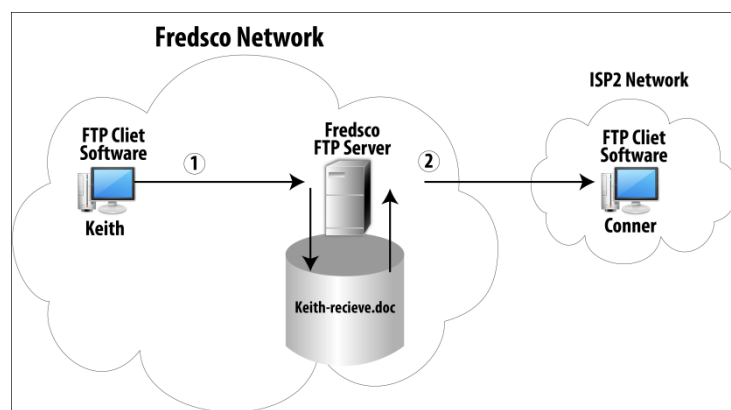


Figure 30.2: FTP Process for Transferring Files

FTP uses client software on the end user computer. It also implements application layer protocols. The FTP TCP/IP standard defines messages and headers to do something useful. FTP clients need to connect to a server to do anything useful. The client must identify itself to the server. FTP server software runs on a physical server.

An Example:

The sequence Keith needs to follow to store files on an FTP server is described next.

Authentication:

- 1- FTP client uses the FTP USER command to provide the username to the FTP server.
- 2- The FTP PASS command supplies the password to the server.

Data Transfer:

3- If authenticated, then FTP client uses the PUT command to inform the server that the file's name is file1 and will be available in the upcoming messages.

4- Send the actual contents of the file.

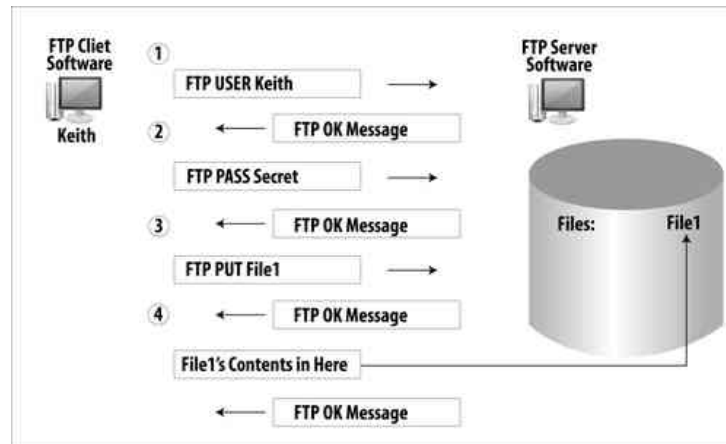


Figure 30.3: Stocking the FTP Warehouse

Assume Conner wants to get file1. He needs to follow the same sequence to connect to the FTP server and get file1 as followed by Keith except: Instead of issuing a PUT command, Conner issues a GET command, which refers to copying a file from the server to the client.

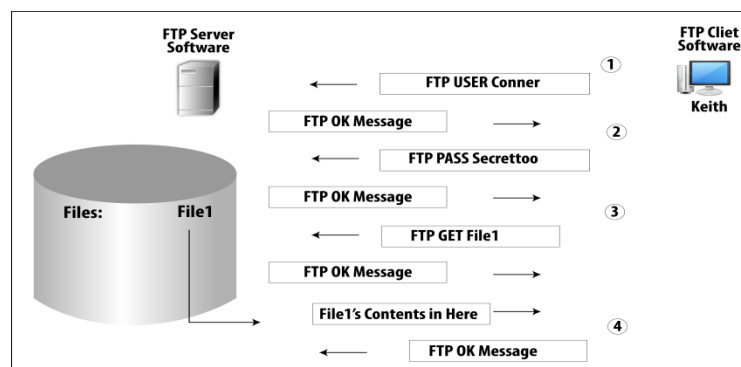


Figure 30.4: Copying Inventory (Files) from the FTP Warehouse

FTP is defined in RFC 959. FTP uses a different underlying TCP connection for the control messages, versus the actual file transfer.

Topic 31: World Wide Web and HTTP Protocol

This topic describes the working of the World Wide Web and the HTTP protocol.

The end user PC uses software called a web browser also known as a web client. Someone must create some content and put it on a web server before the end user can do anything useful. Content consists of individual web pages, the collection of which is called a website. As an example, let's imagine a company FredsCo wants to build a website so that clients can access it. Web server software must be installed on the computer that will be used as the web server. A number of files (content) created by someone should be put onto the web server and be displayed by the <http://www.fredsco.com> website. Now the web server is ready to respond to clients. A customer, Conner, opens his web browser and tries to connect to <http://www.fredsco.com>. The web server gets Conner's request and sends the web page back to Conner.

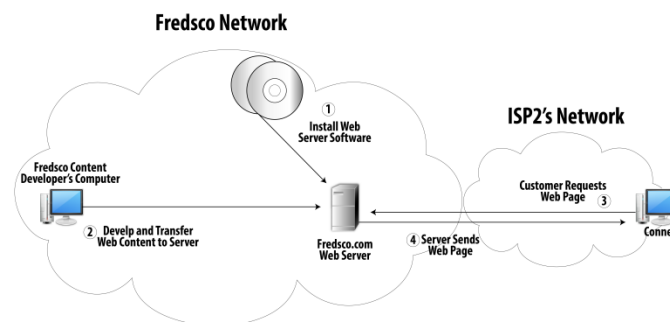


Figure 31.1: Building and Stocking the New Retail Store, AKA New Website

Rules for WWW:

When Conner types <http://www.fredsco.com> in his browser, the web page gets loaded. The above string is called a uniform resource locator (URL). A URL identifies the protocol in use and name of the server. HTTP stands for Hypertext Transfer Protocol and is the TCP/IP protocol you use to send the web page contents from the web server to the web browser. HTTP command GET requests the web server to send the server's default web page to the browser. When a web site is developed, one web page is defined as the default. The server then looks in its configuration and finds the name of the file that holds its home page. Using HTTP, the server sends the contents of a file called `home.html` back to the browser.

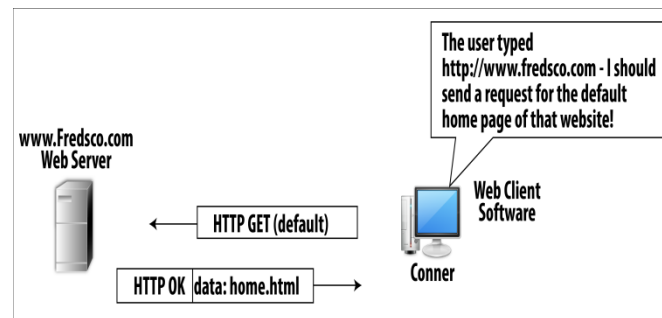


Figure 31.2: Connecting to the Fredsco Home Page

An .HTML file contains text that conforms to a specification called Hypertext Markup Language (HTML). HTML was the first language used to describe the contents of a web page. HTML tells the browser what to put in the browser window, what color to make it, what size, and so on, but it does not define anything about how to send and receive data. When a web page gets loaded, some parts show up right away while the rest of the page fills in slowly. A lot of files get transferred between a web client and server. The first individual file (known as object) downloaded by HTTP is an HTML file. HTML files include text that goes on the web page, formatting instructions, plus instructions to download other objects. When a web page gets loaded, some parts show up right away while the rest of the page fills in slowly. A lot of files get transferred between a web client and server. The first individual file (known as object) downloaded by HTTP is an HTML file. HTML files include text that goes on the web page, formatting instructions, plus instructions to download other objects.

An Example

Assume Conner types Fredsco's URL in his browser.

- 1- Conner's browser gets the HTML file using HTTP.
- 2- Conner's browser reads the file, displaying things on the screen as a result.
- 3 - When Conner's browser reads the file, it might have instructions that require Conner to download other objects. If so, it uses HTTP to get those objects.
- 4 - Conner's browser reads the contents of the new objects, displaying them on the screen.
- 5- Steps 3 and 4 are repeated until all objects are downloaded.

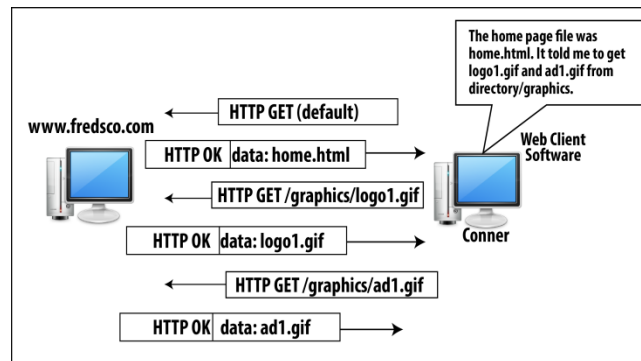


Figure 31.3: Transferring All the Files That Make Up a Web Page

Topic 32: Main Features of TCP

This topic discusses some of the main features of TCP.

The high points of TCP include ensuring delivery through error recovery, breaking large shipments into manageable sizes using segmentation, getting the data to the right individual program, not just the right computer by using port numbers, and simplifying the creation of applications by hiding the details of data delivery from the application. Assume Wilma's computer is sending three TCP segments to Fred. Sender numbers the segments so that if a receiver does not receive a segment or receives with some error, he can request sender to send again.

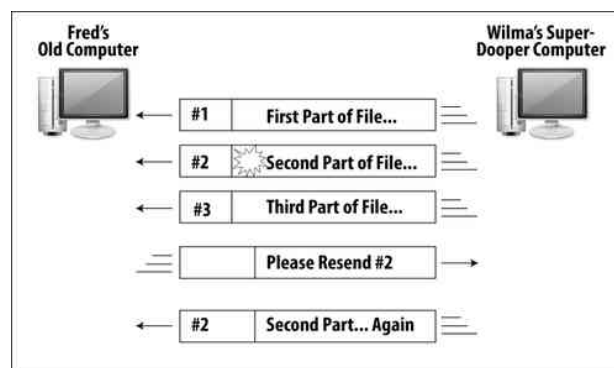


Figure 32.1: Assuring Data Delivery

TCP puts some interesting information into the TCP header. This is shown next.

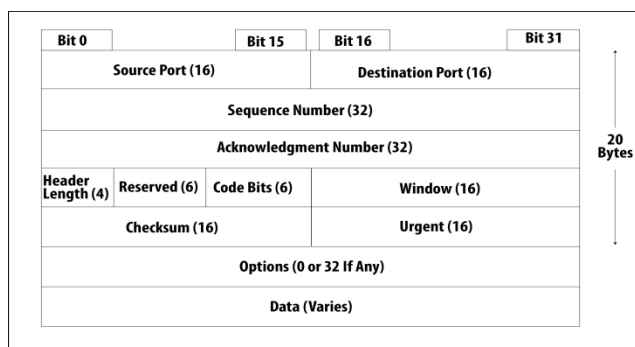


Figure 32.2: The Format of the Shipping Label: TCP Headers

For error recovery, it uses the sequence number and acknowledgment number fields. The sequence number identifies the segment, and the acknowledgement number is used when an error occurs. The ACK field means that this number is the number of the next segment receiver expects to receive. This procedure is called forward acknowledgment.

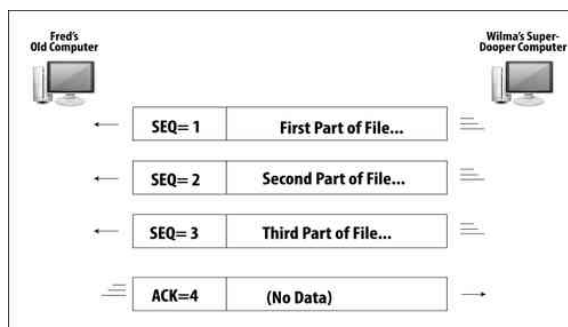


Figure 32.3: Delivery Confirmation, No Errors

TCP Error-Recovery Process

Assume while Wilma sent 3 segments, segment number 2 has some errors.

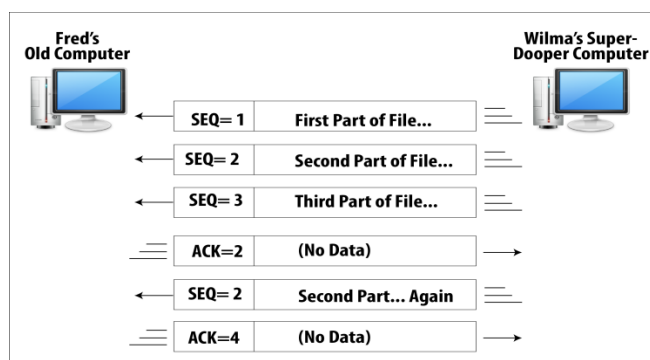


Figure 32.4: TCP Error-Recovery Process

Topic 33: Working of TCP

In this topic, we present the working of TCP protocol.

If you are using a TCP/IP application, the application layer protocol uses transport layer protocol to send their messages across the network. The application layer protocols are part of the application software. For example, HTTP is used for WWW and SMTP for email. The transport layer protocols are typically part of the operating system (OS) of the computer. The transport layer software hangs around, waiting on the application program specifically the part of the application program that implements the application layer protocols to ask it to do something.

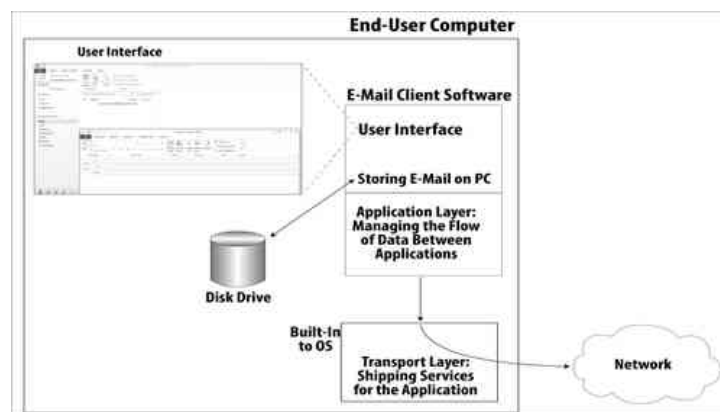


Figure 32.5: Segmenting Data Before Sending

An application layer protocol could avoid using a transport layer protocol. The transport layer protocols provide great services that many applications need. It takes a lot less time and effort for the application to use a transport layer protocol. It's better, faster, cheaper, and simpler for an application to use a transport layer protocol. A transport layer protocol provides a logical communication between application processes running on different hosts. Transport protocols run in end systems. On the sending side, transport layer breaks application messages into segments. On the receiving side, transport layer reassembles segments into messages, passes to application layer. For the **Internet** transport layer, there are two options. Transmission Control Protocol (TCP) and User Datagram Protocol (UDP). TCP provides a reliable delivery while UDP is unreliable. The **minimal** transport-layer services are process-to-process data delivery and error checking – error detection.

TCP's Encapsulation

Assume Keith's browser requests a home page. The browser does not actually send the request over the network. It asks the TCP software on Keith's computer to send it.

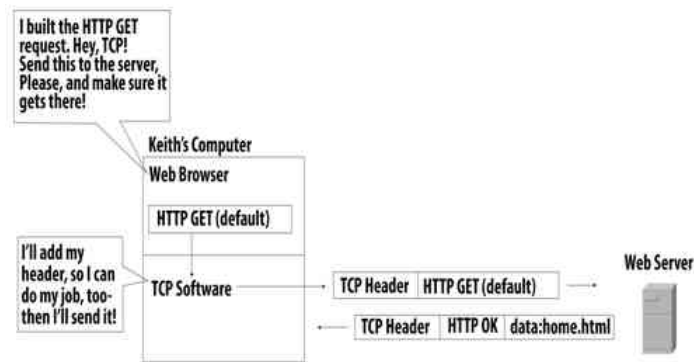


Figure 32.6: Acknowledging Each Byte

TCP provides several services, so it needs a place to record some information about those services. For this purpose, TCP defines a header. The TCP header and the data are called a TCP segment.

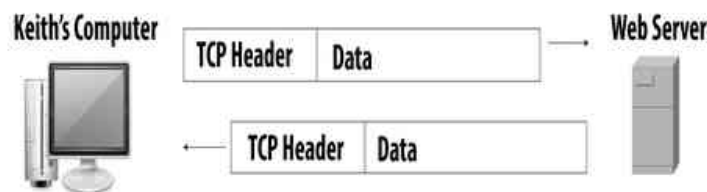


Figure 32.7: Keith's TCP Connections

TCP doesn't have to think about what HTTP is trying to do means that the TCP software does not need to know anything about how HTTP works. Software development becomes simpler.

Figures and Material used for this topic have been adapted from Kurose and Ross's book with the title "*Computer Networking: A Top-down Approach*", 6th Edition, 2013.

Topic 34: TCP Flow and Congestion Control

This topic describes the TCP flow and congestion control mechanisms.

Hosts on each side of a TCP connection set aside a receive buffer for the connection. When the TCP connection receives bytes that are correct and in sequence, it places the data in the receive buffer. The associated application process will read the data from this buffer, but not necessarily at the instant the data arrives. If the application is relatively slow at reading the data, the sender can very easily overflow the connection's receive buffer by sending too much data too quickly.



Figure 34.1 Receive Window

Flow Control does allow the sender to overflow receiver's buffer by transmitting too much, too fast. It is TCP's speed-matching service: matching the rate at which sender is sending against the rate at which receiving application is reading.

TCP Flow Control

Sending side of TCP maintains a variable, receive window. It gives the sender an idea about the free buffer space at receiver. TCP header contains a field named Receive Window. Receiver advertises spare room available in its receive buffer in the receive window field of every segment it sends to the sender.

Congestion means too many sources sending too much data too fast for network to handle. This can result into lost packets (buffer overflow at routers), long delays (queuing in router buffers).

TCP Congestion Control

Each sender limits the rate at which it sends traffic as a function of perceived congestion. In case the network is congestion-free, upon the arrival of ACKs, TCP sender indicates a congestion-free source-to-destination path. Occurrence of either timeout or receipt of three duplicate ACKs from receiver indicates congestion. If a sender perceives that there is little congestion on the path between itself and the destination, then it increases its send rate. In case there is congestion along the path, then it reduces its send rate.

Figures and Material used for this topic have been adapted from Kurose and Ross's book with the title "*Computer Networking: A Top-down Approach*", 6th Edition, 2013.

Topic 35: User Datagram Protocol (UDP)

This topic describes the User Datagram Protocol (UDP).

The **Internet** Transport Layer can either use Transmission Control Protocol (TCP) or User Datagram Protocol (UDP). TCP is a reliable service while UDP provides unreliable delivery.

Household Analogy:

Consider two houses, one in Lahore and other in Islamabad. Each house is a home to a dozen kids. The kids are cousins and each kid writes letter to each cousin every week (144 letters in total). Each letter is delivered by traditional postal service in a separate envelope. In each house, one kid is responsible for mail collection and mail distribution. Hassan does this job in Lahore while Umer does it in Islamabad. Each week Hassan collects the mail and gives it to the postal service mail carrier. He also distributes the received mail. Umer performs a similar job in Islamabad. The postal service provides logical communication between the two houses and not from person to person.

Hassan and Umer provide logical communication among the cousins. Application messages are like letters in envelopes. Processes resemble cousins. Hosts (also called end systems) are same as houses. Job of the transport-layer protocol is the same as that of Hassan and Umer. The network-layer protocol is like the postal service.

Assume Hassan and Umer go on vacation, and another cousin pair -- say Fahad and Ahmed replace them. Being younger, they pick up and drop off the mail less frequently and occasionally lose letters. They do not provide the same set of services as Hassan and Umer. Transmission Control Protocol (TCP), and User Datagram Protocol (UDP) offer different service models to applications.

In contrast to TCP:

UDP provides “best effort” service. Segments may be lost, delivered out of order to application. There is no connection between UDP sender, receiver. Each UDP segment is handled independently of others. UDP does not support error recovery, congestion control, and flow control.

Why is there a UDP?

No connection establishment (which can add delay) is required by UDP. No connection state at sender, and receiver, thus making it simpler. As no congestion control is supported, UDP can blast away as fast as desired. UDP is more suitable for real-time apps (online streaming video).

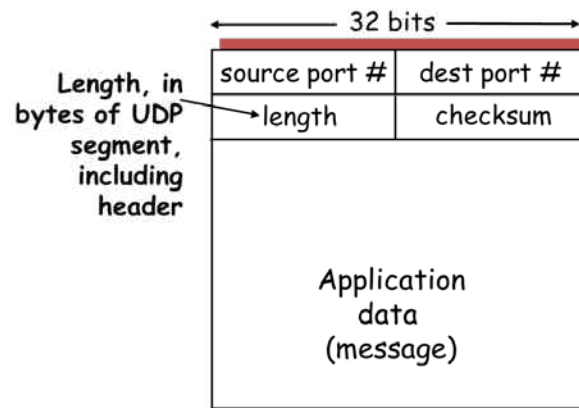


Figure 35.1: UDP segment structure

Usage of UDP

UDP is more suitable for streaming multimedia and Internet telephony (e.g. Skype).

Topic 36: TCP Ports Numbers

In this topic, we describe the importance of TCP port numbers.

Imagine that Keith has two browsers open: one to look at <http://www.fredsco.com>, and one to look at <http://www.espn.com>. Keith also has his e-mail client software up all the time. He also has an FTP client working, downloading some files. What happens when a new TCP segment arrives at Keith's computer? The segment gets to the right computer, but it might have data for one of the two browsers, the e-mail client, or the FTP client. Which one? This issue is depicted next.



Figure 36.1: TCP ports Number

There's a field in the TCP header called the TCP destination port that tells the receiving computer which application program needs to be given the data. Let's look at the TCP header.

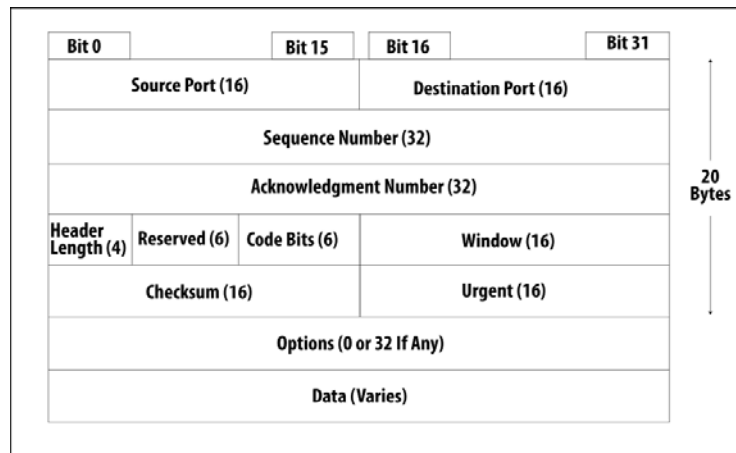


Figure 36.2: TCP Header

For each application program that is currently being run on a computer, a unique TCP port number is assigned to it.

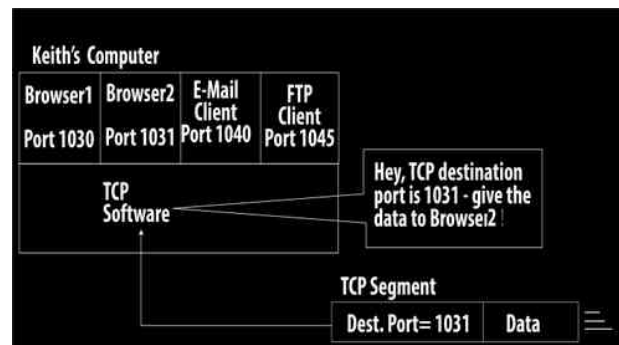


Figure 36.3: A unique TCP port number is assigned

Port Numbers range from 0-65,535 as port number field is of 16-bits. Well-known Port numbers are from 0 to 1023. These are used by well-known applications. For example, a web server uses port number 80. You can find the list of all well-known port numbers on website of www.iaina.org.

Dynamically assigned port numbers:

Typically, a client side of the application lets the transport layer automatically assign a port number.

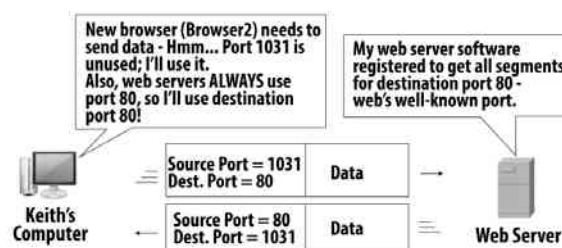


Figure 36.4: The transport layer automatically assigns a port number

Before useful data can be exchanged between a client and a server, a TCP connection is established with a three way handshaking phenomenon. A 3-way Handshaking takes place. Both endpoints agree to communicate with each other. Also, port numbers the client and server use, what values to start with in the sequence number field, and other details are also exchanged with each other.

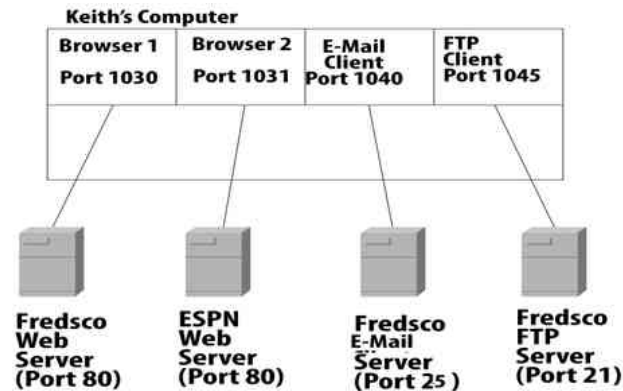


Figure 36.5: TCP ports

Topic 37: TCP Segmentation

This topic describes process and benefits of segmentation performed by TCP.

When a TCP/IP application such as email or web client has data to send, it gives it to transport layer protocol. The application layer protocol does not have a limit to the size of bytes it chooses to send. However, TCP segments data received from the applications to ensure that a TCP segment will fit into a single link-layer frame. The maximum amount of application-layer data that TCP can grab and place in a segment is called Maximum Segment Size (MSS). 1480 is a typical value for MSS. The process of breaking the application data into parts is called segmentation. The bytes that include the data and the TCP header are called TCP segments.

Example:

Let's assume a client contacts a web server. The home page has 3000 bytes of data. HTTP gives all 3000 bytes to TCP. MSS of TCP is 1480. Put 1460 bytes into data part of a segment.