
Exam : 640-802 (SG)

Title : Cisco Certified Network Associate

Ver : 09-16-08

## LIST OF ACRONYMS

| | |
|---|---|
| AAA | Authentication, Authorization, and Accounting |
| ABR | Area Border Router |
| ACF | Advanced Communications Function |
| ACK | Acknowledgment bit (in a TCP segment) |
| ACL | Access Control List |
| ACS | Access Control Server |
| AD | Advertised Distance |
| ADSL | Asymmetric Digital Subscriber Line |
| ANSI | American National Standards Institute |
| API | Application Programming Interface |
| APPC | Advanced Program-to-Program Communications |
| ARAP | AppleTalk Remote Access Protocol |
| ARE | All Routes Explorer |
| ARP | Address Resolution Protocol |
| ARPA | Advanced Research Projects Agency |
| ARPANET | Advanced Research Projects Agency Network |
| AS | Autonomous System |
| ASA | Adaptive Security Algorithm |
| ASBR | Autonomous System Boundary Router |
| ASCII | American Standard Code for Information Interchange |
| ASIC | Application Specific Integrated Circuits |
| ATM | Asynchronous Transfer Mode |
| AUI | Attachment Unit Interface |
| Bc | Committed burst (Frame Relay) |
| B channel | Bearer channel (ISDN) |
| BDR | Backup Designated Router |
| Be | Excess burst (Frame Relay) |
| BECN | Backward Explicit Congestion Notification (Frame Relay) |
| BGP | Border Gateway Protocol |
| BGP-4 | Border Gateway Protocol version 4 |
| BIA | Burned-in Address (another name for a MAC address) |
| BOD | Bandwidth on Demand. |
| BPDU | Bridge Protocol Data Unit |

| | |
|---|---|
| BRF | Bridge Relay Function |
| BRI | Basic Rate Interface (ISDN) |
| BSD | Berkeley Standard Distribution (UNIX) |
| CBT | Core Based Trees |
| CBWFQ | Class-Based Weighted Fair Queuing |
| CCITT | Consultative Committee for International Telegraph and Telephone |
| CCO | Cisco Connection Online |
| CDDI | Copper Distribution Data Interface |
| CEF | Cisco Express Forwarding |
| CHAP | Challenge Handshake Authentication Protocol |
| CIDR | Classless Interdomain Routing |
| CIR | Committed Information Rate. (Frame Relay) |
| CGMP | Cisco Group Management Protocol |
| CLI | Command-Line Interface |
| CLSC | Cisco LAN Switching Configuration |
| CPE | Customer Premises Equipment |
| CPU | Central Processing Unit |
| CR | Carriage Return. |
| CRC | Cyclic Redundancy Check (error) |
| CRF | Concentrator Relay Function |
| CST | Common Spanning Tree |
| CSU | Channel Service Unit |
| DB | Data Bus (connector) |
| DCE | Data Circuit-Terminating Equipment |
| dCEF | Distributed Cisco Express Forwarding |
| DDR | Dial-on-Demand Routing |
| DE | Discard Eligible Indicator |
| DECnet | Digital Equipment Corporation Protocols |
| DES | Data Encryption Standard |
| DHCP | Dynamic Host Control Protocol |
| DLCI | Data-Link Connection Identifier |
| DNIC | Data Network Identification Code. (X.121addressing) |
| DNS | Domain Name System |
| DoD | Department of Defense (US) |

| | |
|---|---|
| DR | Designated Router |
| DRiP | Duplicate Ring Protocol |
| DS | Digital Signal |
| DS0 | Digital Signal level 0 |
| DS1 | Digital Signal level 1 |
| DS3 | Digital Signal level 3 |
| DSL | Digital Subscriber Line |
| DSU | Data Service Unit |
| DTE | Data Terminal Equipment |
| DTP | Dynamic Trunking Protocol |
| DUAL | Diffusing Update Algorithm |
| DVMRP | Distance Vector Multicast Routing Protocol |
| EBC | Ethernet Bundling Controller |
| EGP | Exterior Gateway Protocol |
| EIA/TIA | Electronic Industries Association/Telecommunications Industry Association |
| EIGRP | Enhanced Interior Gateway Routing Protocol |
| ESI | End-System Identifier |
| FCC | Federal Communications Commission |
| FCS | Frame Check Sequence |
| FC | Feasible Condition (Routing) |
| FD | Feasible Distance (Routing) |
| FDDI | Fiber Distributed Data Interface |
| FEC | Fast EtherChannel |
| FECN | Forward Explicit Congestion Notification |
| FIB | Forwarding Information Base |
| FIFO | First-In, First-Out (Queuing) |
| FR | Frame Relay |
| FS | Feasible Successor (Routing) |
| FSSRP | Fast Simple Server Redundancy Protocol |
| FTP | File Transfer Protocol |
| GBIC | Gigabit Interface Converters |
| GEC | Gigabit EtherChannel |
| GSR | Gigabit Switch Router |
| HDLC | High-Level Data Link Control |

| | |
|---|---|
| HDSL | High data-rate digital subscriber line |
| HSRP | Hot Standby Router Protocol |
| HSSI | High-Speed Serial Interface |
| HTTP | Hypertext Transfer Protocol |
| I/O | Input/Output |
| IANA | Internet Assigned Numbers Authority |
| ICMP | Internet Control Message Protocol |
| IDN | International Data Number |
| IEEE | Institute of Electrical and Electronic Engineers |
| IETF | Internet Engineering Task Force |
| IGP | Interior Gateway Protocol |
| IGRP | Interior Gateway Routing Protocol |
| ILMI | Integrated Local Management Interface |
| IOS | Internetwork Operating System |
| IP | Internet Protocol |
| IPSec | IP Security |
| IPv6 | IP version 6 |
| IPX | Internetwork Packet Exchange (Novell) |
| IRDP | ICMP Router Discovery Protocol |
| IS | Information Systems |
| IS-IS | Intermediate System-to-Intermediate System |
| ISDN | Integrated Services Digital Network |
| ISL | Inter-Switch Link |
| ISO | International Organization for Standardization |
| ISOC | Internet Society |
| ISP | Internet Service Provider |
| ITU-T | International Telecommunication Union–Telecommunication Standardization Sector |
| kbps | kilobits per second (bandwidth) |
| LAN | Local Area Network |
| LANE | LAN Emulation |
| LAPB | Link Access Procedure, Balanced |
| LAPD | Link Access Procedure on the D channel |
| LEC | LAN Emulation Client |
| LECS | LAN Emulation Configuration Server |

| | |
|---|---|
| LED | Light Emitting Diode |
| LES | LAN Emulation Server |
| LLC | Logic Link Control (OSI Layer 2 sublayer) |
| LLQ | Low-Latency Queuing |
| LMI | Local Management Interface |
| LSA | Link-State Advertisement |
| MAC | Media Access Control (OSI Layer 2 sublayer) |
| MAN | Metropolitan-Area Network |
| MD5 | Message Digest Algorithm 5 |
| MLS | Multilayer Switching |
| MLS-RP | Multilayer Switching Route Processor |
| MLS-SE | Multilayer Switching Switch Engine |
| MLSP | Multilayer Switching Protocol |
| MOSPF | Multicast Open Shortest Path First |
| MSAU | Multistation Access Unit |
| MSFC | Multilayer Switch Feature Card |
| MTU | Maximum Transmission Unit |
| NAK | Negative Acknowledgment |
| NAS | Network Access Server |
| NAT | Network Address Translation |
| NBMA | Nonbroadcast Multiaccess |
| NetBEUI | NetBIOS Extended User Interface |
| NetBIOS | Network Basic Input/Output System |
| NFFC | NetFlow Feature Card |
| NMS | Network Management System |
| NNI | Network-to-Network Interface |
| NSAP | Network Service Access Point |
| NVRAM | Nonvolatile Random Access Memory |
| OC | Optical Carrier |
| ODBC | Open Database Connectivity |
| OLE | Object Linking and Embedding |
| OSI | Open Systems Interconnection (Model) |
| OSPF | Open Shortest Path First |
| OTDR | Optical Time Domain Reflectometer |
| OUI | Organizationally Unique Identifier |

| | |
|---|---|
| PAgP | Port Aggregation Protocol |
| PAP | Password Authentication Protocol |
| PAT | Port Address Translation |
| PDN | Public Data Network |
| PDU | Protocol Data Unit (i.e., a data packet) |
| PIM | Protocol Independent Multicast |
| PIM | SM Protocol Independent Multicast Sparse Mode |
| PIMDM | Protocol Independent Multicast Mode |
| PIX | Private Internet Exchange (Cisco Firewall) |
| PNNI | Private Network-to-Network Interface |
| POP | Point of Presence |
| POTS | Plain Old Telephone Service |
| PPP | Point-to-Point Protocol |
| PQ | Priority Queuing |
| PRI | Primary Rate Interface (ISDN) |
| PSTN | Public Switched Telephone Network |
| PTT | Poste, Telephone, Telegramme |
| PVC | Permanent Virtual Circuit (ATM) |
| PVST | Per-VLAN Spanning Tree |
| PVST+ | Per-VLAN Spanning Tree Plus |
| QoS | Quality of Service |
| RADIUS | Remote Authentication Dial-In User Service |
| RAS | Remote Access Service |
| RIF | Routing Information Field |
| RIP | Routing Information Protocol |
| RJ | Registered Jack (connector) |
| RMON | Embedded Remote Monitoring |
| RP | Rendezvous Point |
| RPF | Reverse Path Forwarding |
| RSFC | Route Switch Feature Card |
| RSM | Route Switch Module |
| RSP | Route Switch Processor |
| RSTP | Rapid Spanning Tree Protocol |
| RTP | Reliable Transport Protocol |
| RTO | Retransmission Timeout |

| | |
|---|---|
| SA | Source Address |
| SAID | Security Association Identifier |
| SAP | Service Access Point; also Service Advertising Protocol (Novell) |
| SAPI | Service Access Point Identifier |
| SAR | Segmentation and Reassembly |
| SDLC | Synchronous Data Link Control (SNA) |
| SIA | Stuck in Active (EIGRP) |
| SIN | Ships-in-the-Night (Routing) |
| SLIP | Serial Line Internet Protocol |
| SMDS | Switched Multimegabit Data Service |
| SMTP | Simple Mail Transfer Protocol |
| SNA | Systems Network Architecture (IBM) |
| SNAP | SubNetwork Access Protocol |
| SNMP | Simple Network Management Protocol |
| SOF | Start of Frame |
| SOHO | Small Office, Home Office |
| SONET | Synchronous Optical Network |
| SONET/SDH | Synchronous Optical Network/Synchronous Digital Hierarchy |
| SPAN | Switched Port Analyzer |
| SPF | Shortest Path First |
| SPID | Service Profile Identifier |
| SPP | Sequenced Packet Protocol (Vines) |
| SPX | Sequenced Packet Exchange (Novell) |
| SQL | Structured Query Language |
| SRAM | Static Random Access Memeory |
| SRB | Source-Route Bridge |
| SRT | Source-Route Transparent (Bridging) |
| SRTT | Smooth Round-Trip Timer (EIGRP) |
| SS7 | Signaling System 7 |
| SSAP | Source service access point (LLC) |
| SSE | Silicon Switching Engine. |
| SSP | Silicon Switch Processor |
| SSRP | Simple Server Redundancy Protocol |
| STA | Spanning-Tree Algorithm |

| | |
|---|---|
| STP | Spanning-Tree Protocol; also Shielded Twisted-Pair (cable) |
| SVC | Switched Virtual Circuit (ATM) |
| SYN | Synchronize (TCP segment) |
| TA | Terminal Adapter (ISDN) |
| TAC | Technical Assistance Center (Cisco) |
| TACACS | Terminal Access Controller Access Control System |
| TCI | Tag Control Information |
| TCP | Transmission Control Protocol |
| TCP/IP | Transmission Control Protocol/Internet Protocol |
| TCN | Topology Change Notification |
| TDM | Time-Division Multiplexing |
| TDR | Time Domain Reflectometers |
| TFTP | Trivial File Transfer Protocol |
| TIA | Telecommunications Industry Association |
| TLV | Type-Length-Value |
| ToS | Type of Service |
| TPID | Tag Protocol Identifier |
| TrBRF | Token Ring Bridge Relay Function |
| TrCRF | Token Ring Concentrator Relay Function |
| TTL | Time-To-Live |
| UDP | User Datagram Protocol |
| UNC | Universal Naming Convention or Uniform Naming Convention |
| UNI | User-Network Interface |
| URL | Uniform Resource Locator |
| UTC | Coordinated Universal Time (same as Greenwich Mean Time) |
| UTL | Utilization |
| UTP | Unshielded Twisted-Pair (cable) |
| VBR | Variable Bit Rate |
| VC | Virtual Circuit (ATM) |
| VID | VLAN Identifier |
| VIP | Versatile Interface Processor |
| VLAN | Virtual Local Aare Network |
| VLSM | Variable-Length Subnet Mask VLAN Membership Policy |
| VMPS | Server |
| VPN | Virtual Private Network |

| | |
|---|---|
| VTP | VLAN Trunking Protocol |
| vty | Virtual terminal line |
| WAIS | Wide Area Information Server |
| WAN | Wide Area Network |
| WFQ | Weighted Fair Queuing |
| WLAN | Wireless Local Area Network |
| WWW | World Wide Web |
| XNS | Xerox Network Systems |
| XOR | Exclusive-OR |
| XOT | X.25 over TCP |
| ZIP | Zone Information Protocol (AppleTalk) |

# INTRODUCTION

## Exam Code: 640-802

## Certifications:

## Cisco Certified Network Associate (CCNA)

## Prerequisites:
None

## About This Study Guide
This Study Guide is based on the current pool of exam questions for the Cisco CCNA 640-802 composite exam. As such it provides all the information required to pass the 640-802 exam and is organized around the specific skills that are tested in that exam. Thus, the information contained in this Study Guide is specific to the 640-802 exam and does not represent a complete reference work on the subject of Interconnecting Cisco Networking Devices. Topics covered in this Study Guide includes: Designing or Modifying a simple Local Area Network (LAN) using Cisco Products; Designing an IP Addressing Scheme; Selecting Appropriate Routing Protocols; Designing a simple Internetwork using Cisco products; Developing an Access List to Meet User Specifications; Choosing Wide Area Network (WAN) Services; Managing System Image and Device Configuration Files Performing an Initial Configuration on a Switch; Configuring Routing Protocols; Configuring IP Addresses, Subnet Masks, and Gateway Addresses on Routers and Hosts; Configuring a Router for Additional Administrative Functionality; Configuring a Switch with Virtual LANs (VLANs) and Inter-switch Communication; Implementing a LAN; Customizing a Switch Configuration; Implementing Access Lists; Implementing Simple WAN Protocols; Utilizing the OSI Reference Model as a Guide for Systematic Network Troubleshooting; Performing LAN and VLAN Troubleshooting; Troubleshooting Routing Protocols; Troubleshooting IP Addressing and Host Configuration; Troubleshooting a Device as Part of a Working Network; Troubleshooting an Access List; Performing Simple WAN Troubleshooting; Understanding Network Communications based on Layered Models; Understanding the Components of Network Devices; Understanding the Spanning Tree Process; Evaluating the Characteristics of LAN Environments; Evaluating the TCP/IP Communication Process and its Associated Protocols; Evaluating the

Characteristics of Routing Protocols; Evaluating Rules for Packet Control; and Evaluating Key Characteristics of WANs.

### Intended Audience

This Study Guide is targeted specifically at people who wish to take the Cisco CCNA 640-802 Composite exam. This information in this Study Guide is specific to the exam. It is not a complete reference work. Although our Study Guides are aimed at new comers to the world of IT, the concepts dealt with in this Study Guide are complex. Knowledge of CompTIA's A+ and Network+ courses would be advantageous.

**Note**: Because the 640-802 exam is a composite of the 640-822 and 640-816 exams, there is a fair amount of overlap between this Study Guide and the 640-822 and 640-816 Study Guides. However, this Study Guide does not combine the 640-822 and 640-816 Study Guides but addresses the 640-802 exam specifically. As such, we would not advise using this Study Guide for the 640-822 exam and/or the 640-816 exam.

### How To Use This Study Guide

To benefit from this Study Guide we recommend that you:

• Study each chapter carefully until you fully understand the information. This will require regular and disciplined work. Where possible, attempt to implement the information in a lab setup.

• Be sure that you have studied and understand the entire Study Guide before you take the exam.

**Note**: Remember to pay special attention to these note boxes as they contain important additional information that is specific to the exam.

Good luck!

# Topic 1: Networking Fundamentals

### Section 1.1: The OSI Reference Model

The OSI is the Open System Interconnection reference model for communications. As illustrated in Figure 1.1, the OSI reference model consists of seven layers, each of which can have several sublayers. The upper layers of the OSI reference model define functions focused on the application, while the lower three layers define functions focused on end-to-end delivery of the data.

• The Application Layer (Layer 7) refers to communications services to applications and is the interface

between the network and the application. Examples include: Telnet, HTTP, FTP, Internet browsers, NFS, SMTP gateways, SNMP, X.400 mail, and FTAM.

• The Presentation Layer (Layer 6) defining data formats, such as ASCII text, EBCDIC text, binary, BCD, and JPEG. Encryption also is defined as a presentation layer service. Examples include: JPEG, ASCII, EBCDIC, TIFF, GIF, PICT, encryption, MPEG, and MIDI.

• The Session Layer (Layer 5) defines how to start, control, and end communication sessions. This includes the control and management of multiple bidirectional messages so that the application can be notified if only some of a series of messages are completed. This allows the presentation layer to have a seamless view of an incoming stream of data. The presentation layer can be presented with data if all flows occur in some cases. Examples include: RPC, SQL, NFS, NetBios names, AppleTalk ASP, and DECnet SCP

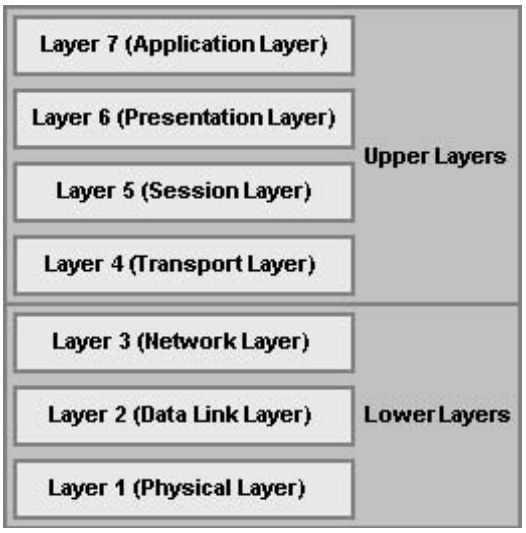

**Figure 1.1: The OSI Reference Model**

• The Transport Layer (Layer 4) defines several functions, including the choice of protocols. The most important Layer 4 functions are error recovery and flow control. The transport layer may provide for retransmission, i.e., error recovery, and may use flow control to prevent unnecessary congestion by attempting to send data at a rate that the network can accommodate, or it might not, depending on the choice of protocols. Multiplexing of incoming data for different flows to applications on the same host is also performed. Reordering of the incoming data stream when packets arrive out of order is included. Examples include: TCP, UDP, and SPX.

• The Network Layer (Layer 3) defines end-to-end delivery of packets and defines logical addressing to accomplish this. It also defines how routing works and how routes are learned; and how to fragment a packet into smaller packets to accommodate media with smaller maximum transmission unit sizes. Examples include: IP, IPX, AppleTalk DDP, and ICMP. Both IP and IPX define logical addressing, routing, the learning of routing information, and end-to-end delivery rules. The IP and IPX protocols most closely match the OSI network layer (Layer 3) and are called Layer 3 protocols because their functions most closely match OSI's Layer 3.

• The Data Link Layer (Layer 2) is concerned with getting data across one particular link or medium. The data link protocols define delivery across an individual link. These protocols are necessarily concerned with the type of media in use. Examples include: IEEE 802.3/802.2, HDLC, Frame Relay, PPP, ATM, and IEEE 802.5/802.2.

• The Physical Layer (Layer 1) deals with the physical characteristics of the transmission medium. Connectors, pins, use of pins, electrical currents, encoding, and light modulation are all part of different physical layer specifications. Examples includes: EIA/TIA-232, V.35, EIA/TIA-449, V.24, RJ-45, Ethernet, 802.3, 802.5, NRZI, NRZ, and B8ZS.

The upper layers of the OSI reference model, i.e., the Application Layer (Layer 7), the Presentation Layer

(Layer 6), and the Session Layer (Layer 5), define functions focused on the application. The lower four layers, i.e., the Transport Layer (Layer 4), the Network Layer (Layer 3), the Data Link Layer (Layer 2), and the Physical Layer (Layer 1), define functions focused on end-to-end delivery of the data. As a Cisco Certified Network Associate, you will deal mainly with the lower layers, particularly the data link layer (Layer 2) upon which switching is based, and the network layer (Layer 3) upon which routing is based.

### 1.1.1: Interaction Between OSI Layers

When a host receives a data transmission from another host on the network, that data is processed at each of the OSI layers to the next higher layer, in order to render the data transmission useful to the end-user. To facilitate this processing, headers and trailers are created by the sending host's software or hardware, that are placed before or after the data given to the next higher layer. Thus, each layer has a header and trailer, typically in each data packet that comprises the data flow. The sequence of processing at each OSI layer, i.e., the processing between adjacent OSI layers, is as follows:

• The Physical Layer (Layer 1) ensures bit synchronization and places the received binary pattern into a buffer. It notifies the Data Link Layer (Layer 2) that a frame has been received after decoding the incoming signal into a bit stream. Thus, Layer 1 provides delivery of a stream of bits across the medium.

• The Data Link Layer (Layer 2) examines the frame check sequence (FCS) in the trailer to determine whether errors occurred in transmission, providing error detection. If an error has occurred, the frame is discarded. The current host examines data link address is examined to determine if the data is addressed to it or whether to process the data further. If the data is addressed to the host, the data between the Layer 2 header and trailer is handed over to the Network Layer (Layer 3) software. Thus, the data link layer delivers data across the link.

• The Network Layer (Layer 3) examines the destination address. If the address is the current host's address, processing continues and the data after the Layer 3 header is handed over to the Transport Layer (Layer 4) software. Thus, Layer 3 provides end-to-end delivery.

• If error recovery was an option chosen for the Transport Layer (Layer 4), the counters identifying this piece of data are encoded in the Layer 4 header along with acknowledgment information, which is called error recovery. After error recovery and reordering of the incoming data, the data is given to the Session Layer (Layer 5).

• The Session Layer (Layer 5) ensures that a series of messages is completed. The Layer 5 header includes fields signifying sequence of the packet in the data stream, indicating the position of the data packet in the flow. After the session layer ensures that all flows are completed, it passes the data after the Layer 5 header to the Presentation Layer (Layer 6) software.

• The Presentation Layer (Layer 6) defines and manipulates the data format of the data transmission. It converts the data to the proper format specified in the Layer 6 header. Typically, this header is included only for initialization flows, not with every data packet being transmitted. After the data formats have been converted, the data after the Layer 6 header is passed to the Application Layer (Layer 7) software.

• The Application Layer (Layer 7) processes the final header and examines the end-user data. This header signifies agreement to operating parameters by the applications on the two hosts. The headers are

used to signal the values for all parameters; therefore, the header typically is sent and received at application initialization time only.

In addition to processing between adjacent OSI layers, the various layers must also interact with the same layer on another computer to successfully implement its functions. To interact with the same layer on another computer, each layer defines additional data bits in the header and, in some cases, trailer that is created by the sending host's software or hardware. The layer on the receiving host interprets the headers and trailers created by the corresponding layer on the sending host to determine how that layer's processing is being defined, and how to interact within that framework.

## Section 1.2: TCP/IP and the OSI Reference Model

As illustrated in Figure 1.2, the Transmission Control Protocol/Internet Protocol (TCP/IP) model consists of four layers, each of which can have several sublayers. These

layers correlate roughly to layers in the OSI reference model and define similar functions. Some of the TCP/IP layers correspond directly with layers in the OSI reference model while others span several OSI layers. The four TCP/IP layers are:

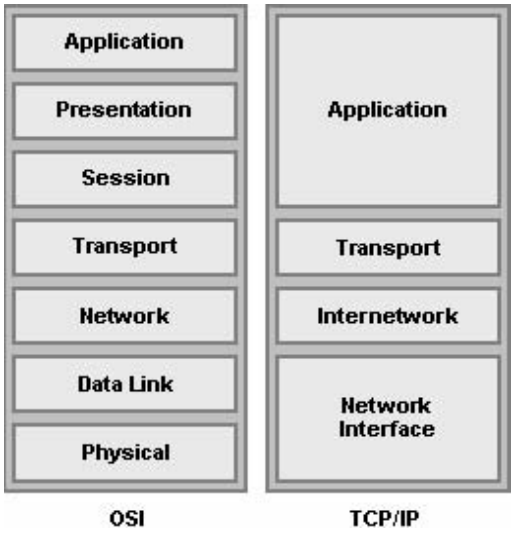

Figure 1.2: OSI and TCP/IP Models

• The TCP/IP Application Layer( refers to communications services to applications and is the interface between the network and the application. It is also responsible for presentation and controlling communication sessions. It spans the Application Layer, Presentation Layer and Session Layer of the OSI reference model. Examples include: HTTP, POP3, and SNMP.

• The TCP/IP Transport Layer defines several functions, including the choice of protocols, error recovery and flow control. The transport layer may provide for retransmission, i.e., error recovery, and may use flow control to prevent unnecessary congestion by attempting to send data at a rate

that the network can accommodate, or it might not, depending on the choice of protocols. Multiplexing of incoming data for different flows to applications on the same host is also performed. Reordering of the incoming data stream when packets arrive out of order is included. It correlates with the Transport Layer of the OSI reference model. Examples include: TCP and UDP, which are called Transport Layer, or Layer 4, protocols.

• The TCP/IP Internetwork Layer defines end-to-end delivery of packets and defines logical addressing to accomplish this. It also defines how routing works and how routes are learned; and how to fragment a packet into smaller packets to accommodate media with smaller maximum transmission unit sizes. It correlates with the Network Layer of the OSI reference model. Examples include: IP and ICMP.

• The TCP/IP Network Interface Layer is concerned with the physical characteristics of the transmission medium as well as getting data across one particular link or medium. This layer defines

delivery across an individual link as well as the physical layer specifications. It spans the Data Link Layer and Physical Layer of the OSI reference model. Examples include: Ethernet and Frame Relay.

1.2.1: The TCP/IP Protocol Architecture
TCP/IP defines a large collection of protocols that allow computers to communicate. Table 1.1 outlines the protocols and the TCP/IP architectural layer to which they belong. TCP/IP defines the details of each of these protocols in Requests For Comments (RFC) documents. By implementing the required protocols defined in TCP/IP RFCs, a computer that implements the standard networking protocols defined by TCP/IP can communicate with other computers that also use the TCP/IP standards.

**Table 1.1: The TCP/IP Architectural Model and Protocols**

| TCP/IP Architecture Layer | Protocols |
|---|---|
| Application | HTTP, POP3, SMTP |
| Transport | TCP, UDP |
| Internetwork | IP |
| Network interface | Ethernet, Frame Relay |

### 1.2.2: TCP/IP Data Encapsulation
The term encapsulation describes the process of putting headers and trailers around some data. A computer that needs to send data encapsulates the data in headers of the correct format so that the receiving computer will know how to interpret the received data. Data encapsulation with TCP/IP consists of five-steps:

**Step 1**: Create the application data and headers.

**Step 2**: Package the data for transport, which is performed by the transport layer (TCP or UDP). The Transport Layer creates the transport header and places the data behind it.

**Step 3:** Add the destination and source network layer addresses to the data, which is performed by the Internetwork Layer. The Internetwork Layer creates the network header, which includes the network layer addresses, and places the data behind it.

**Step 4**: Add the destination and source data link layer addresses to the data, which is performed by the Network Interface Layer. The Network Interface Layer creates the data link header, places the data behind it, and places the data link trailer at the end.

**Step 5**: Transmit the bits, which is performed by the Network Interface Layer. The Network Interface Layer encodes a signal onto the medium to transmit the frame.

## Section 1.3: Networks
A network is defined as a group of two or more computers linked together for the purpose of communicating and sharing information and other resources, such as printers and applications. Most networks are constructed around a cable connection that links the computers, however, modern wireless networks that use radio wave or infrared connections are also becoming quite prevalent. These connections permit the computers to communicate via the wires in the cable, radio wave or infrared signal. For a network to

function it must provide connections, communications, and services.

• Connections are defined by the hardware or physical components that are required to connect a computer to the network. This includes the network medium, which refers to the hardware that physically connects one computer to another, i.e., the network cable or a wireless connection; and the network interface, which refers to the hardware that attaches a computer to the network medium and is usually a network interface card (NIC).

• Communications refers to the network protocols that are used to establish the rules governing network communication between the networked computers. Network protocols allow computers running different operating systems and software to communicate with each.

• Services define the resources, such as files or printers, that a computer shares with the rest of the networked computers.

### 1.3.1: Network Definitions

Computer networks can be classified and defined according to geographical area that the network covers. There are four network definitions: a Local Area Network (LAN), a Campus Area Network (CAN), a Metropolitan Area Network (MAN) , and a Wide Area Network (WAN). There are three additional network definitions, namely the Internet, an intranet and an Internetwork. These network definitions are discussed in Table 1.2.

**Table 1.2: Network Definitions**

| Definition | Description |
|---|---|
| Local Area Network (LAN) | A LAN is defined as a network that is contained within a closed environment and does not exceed a distance of 1.25 mile (2 km). Computers and peripherals on a LAN are typically joined by a network cable or by a wireless network connection. A LAN that consists of wireless connections is referred to as a **Wireless LAN (WLAN)**. |
| Campus Area Network (CAN) | A CAN is limited to a single geographical area but may exceed the size of a LAN |
| Metropolitan Area Network (MAN) | A MAN is defined as a network that covers the geographical area of a city that is less than 100 miles. |
| Wide Area Network (WAN) | A WAN is defined as a network that exceeds 1.25 miles. A WAN often consists of a number of LANs that have been joined together. A CAN and a MAN is also a WAN. WANs typically connected numerous LANs through the internet via telephone lines, T1 lines, Integrated Services Digital Network (ISDN) lines, radio waves, cable or satellite links. |
| Internet | The Internet is a world wide web of networks that are based on the TCP/IP protocol and is not own by a single company or organization. |

| | |
|---|---|
| Intranet | An intranet uses that same technology as the Internet but is owned and managed by a company or organization. A LAN or a WAN s usually an intranet. |
| Internetwork | An internetwork consists of a number of networks that are joined by routers. The Internet is the largest example of an internetwork. |

Of these network definitions, the most common are the Internet, the LAN and the WAN.

### 1.3.2: Types of Networks

These network definitions can be divided into two types of networks, based on how information is stored on the network, how network security is handled, and how the computers on the network interact. These two types are: Peer-To-Peer (P2P) Networks and Server/Client Networks. The latter is often also called Server networks.

• On a Peer-To-Peer (P2P) Network, there is no hierarchy of computers; instead each computer acts as either a server which shares its data or services with other computers, or as a client which uses data or services on another computer. Furthermore, each user establishes the security on their own computers and determines which of their resources are made available to other users. These networks are typically limited to between 15 and 20 computers. Microsoft Windows for Workgroups, Windows 95, Windows 98, Windows ME, Windows NT Workstation, Windows 2000, Novell's NetWare, UNIX, and Linux are some operating systems that support peer-to-peer networking.

• A Server/Client Network consists of one or more dedicated computers configured as servers. This server manages access to all shared files and peripherals. The server runs the network operating system (NOS) manages security and administers access to resources. The client computers or workstations connect to the network and use the available resources. Among the most common network operating systems are Microsoft's Windows NT Server 4, Windows 2000 Server, and Novell's NetWare. Before the release of Windows NT, most dedicated servers worked only as hosts. Windows NT allows these servers to operate as an individual workstation as well.

### 1.3.3: Network Topologies

The layout of a LAN design is called its topology. There are three basic types of topologies: the star topology, the bus topology, and the ring topology. Hybrid combinations of these topologies also exist.

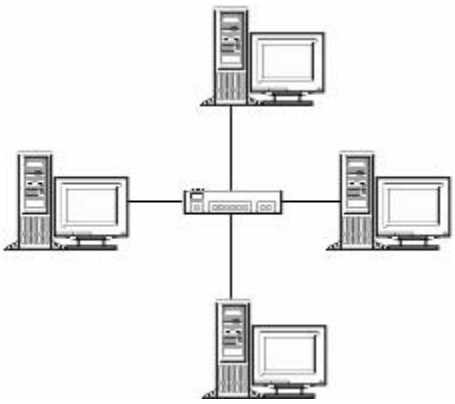

**Figure 1.3: The Star Topology**

• In a network based on the star topology, all computers and devices are connected to a centrally located hub or switch. The hub or switch collects and distributes the flow of data within the network. When a hub is used, data from the sending host are sent to the hub and are then transmitted to all hosts on the network except the sending host. Switches can be thought of as intelligent hubs. When switches are used rather than hubs, data from the sending host are sent to the switch which transmits the data to the intended recipient rather than to all hosts on the network

• In a network based on the bus topology, all computers and devices are connected in series to a single linear cable called a trunk. The trunk is also known as a backbone or a segment. Both ends of the trunk must be terminated to stop the signal from bouncing back up the cable. Because a bus network does not have a central point, it is more difficult to troubleshoot than a star network. Furthermore, a break or problem at any point along the bus can cause the entire network to go down.

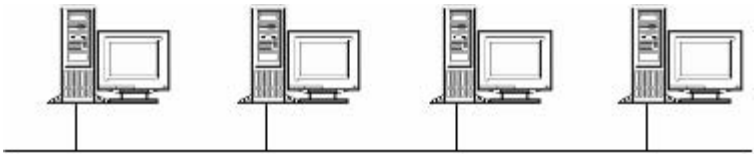

**Figure 1.4: The Bus Topology**

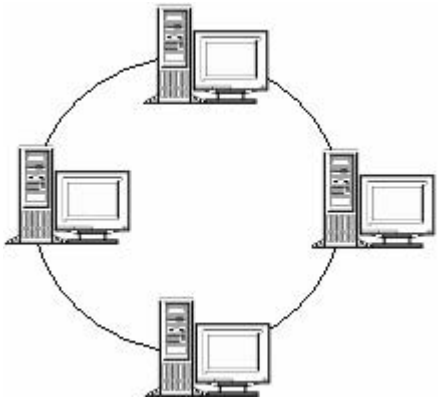

**Figure 1.5: The Ring Topology**

• In a network based on a ring topology, all computers and devices are connected to cable that forms a closed loop. On such networks there are no terminating ends; therefore, if one computer fails, the entire network will go down. Each computer on such a network acts like a repeater and boosts the signal before sending it to the next station. This type of network transmits data by passing a "token" around the network. If the token is free of data, a computer waiting to send data grabs it, attaches the data and the electronic address to the token, and sends it on its way. When the token reaches its destination computer, the data is removed and the token is sent on. Hence this type of network is commonly called a token ring network.
Of these three network topologies, the star topology is the most predominant network type and is based on the Ethernet standard.

### 1.3.4: Network Technologies

Various network technologies can be used to establish network connections, including Ethernet, Fiber Distribution Data Interface (FDDI), Copper Distribution Data Interface (CDDI), Token Ring, and Asynchronous Transfer Mode (ATM). Of these, Ethernet is the most popular choice in installed networks because of its low cost, availability, and scalability to higher bandwidths.

### 1.3.4.1: Ethernet

Ethernet is based on the Institute of Electrical and Electronics Engineers (IEEE) 802.3 standard and offers a bandwidth of 10 Mbps between end users. Ethernet is based on the carrier sense multiple access collision detect (CSMA/CD) technology, which requires that transmitting stations back off for a random period of time when a collision occurs.

Coaxial cable was the first media system specified in the Ethernet standard. Coaxial Ethernet cable comes in two major categories: Thicknet (10Base5) and Thinnet (10Base2). These cables differed in their size and their length limitation. Although Ethernet coaxial cable lengths can be quite long, they susceptible to electromagnetic interference (EMI) and eavesdropping.

**Table 1.3: Coaxial Cable for Ethernet**

| Cable | Diameter | Resistance | Bandwidth | Length |
|---|---|---|---|---|
| Thinnet (10Base2) | 10 mm | 50 ohms | 10 Mbps | 185 m |
| Thicknet (10Base5) | 5 mm | 50 ohms | 10 Mbps | 500 m |

Today most wired networks use twisted-pair media for connections to the desktop. Twisted-pair also comes in two major categories: Unshielded twisted-pair (UTP) and Shielded twisted-pair (STP). One pair of insulated copper wires twisted about each other forms a twisted-pair. The pairs are twisted top reduce interference and crosstalk. Both STP and UTP suffer from high attenuation, therefore these lines are usually restricted to an end-to-end distance of 100 meters between active devices. Furthermore, these cables are sensitive to EMI and eaves dropping. Most networks use 10BaseT UPT cable.

An alternative to twisted-pair cable is fiber optic cable (10BaseFL), which transmits light signals, generated either by light emitting diodes (LEDs) or laser diodes (LDs), instead of electrical signals. These cables support higher transmission speeds and longer distances but are more expensive. Because they do not carry electrical signals, fiber optic cables are immune to EMI and eavesdropping. They also have low attenuation which means they can be used to connect active devices that are up to 2 km apart. However, fiber optic devices are not cost effective while cable installation is complex.

**Table 1.4: Twisted-Pair and Fiber Optic Cable for Ethernet**

| Cable | Technology | Bandwidth | Cable Length |
|---|---|---|---|
| Twisted-Pair | (10BaseT) | 10 Mbps | 100 m |
| Fiber Optic | (10BaseFL) | 10 Mbps | 2,000 m |

## 1.3.4.2: Fast Ethernet

Fast Ethernet operates at 100 Mbps and is based on the IEEE 802.3u standard. The Ethernet cabling schemes, CSMA/CD operation, and all upper-layer protocol operations have been maintained with Fast Ethernet. Fast Ethernet is also backward compatible with 10 Mbps Ethernet. Compatibility is possible because the two devices at each end of a network connection can automatically negotiate link capabilities so that they both can operate at a common level. This negotiation involves the detection and selection of the highest available bandwidth and half-duplex or full-duplex operation. For this reason, Fast Ethernet is also referred to as 10/100 Mbps Ethernet.

Cabling for Fast Ethernet can be either UTP or fiber optic. Specifications for these cables are shown in Table 1.5.

**Table 1.5: Fast Ethernet Cabling and Distance Limitations**

| Technology | Wiring Type | Pairs | Cable Length |
|---|---|---|---|
| 100BaseTX | EIA/TIA Category 5 UTP | 2 | 100 m |
| 100BaseT2 | EIA/TIA Category 3,4,5 UTP | 2 | 100 m |
| 100BaseT4 | EIA/TIA Category 3,4,5 UTP | 4 | 100 m |
| 100BaseFX | Multimode fiber (MMF) with 62.5 micron core; 1300 nm laser | 1 | 400 m (half-duplex)<br>2,000 m (full-duplex) |
| | Single-mode fiber (SMF) with 62.5 micron core; 1300 nm laser | 1 | 10,000 m |

### 1.3.4.3: Gigabit Ethernet

Gigabit Ethernet is an escalation of the Fast Ethernet standard using the same IEEE 802.3 Ethernet frame format. Gigabit Ethernet offers a throughput of 1,000 Mbps (1 Gbps). Like Fast Ethernet, Gigabit Ethernet is compatible with earlier Ethernet standards. However, the physical layer has been modified to increase data transmission speeds: The IEEE 802.3 Ethernet standard and the American National Standards Institute (ANSI) X3T11 FibreChannel. IEEE 802.3 provided the foundation of frame format, CSMA/CD, full duplex, and other characteristics of Ethernet. FibreChannel provided a base of high-speed ASICs, optical components, and encoding/decoding and serialization mechanisms. The resulting protocol is termed IEEE 802.3z Gigabit Ethernet.
Gigabit Ethernet supports several cabling types, referred to as 1000BaseX. Table 1.6 lists the cabling specifications for each type.

Table 1.6: Gigabit Ethernet Cabling and Distance Limitations

| Technology | Wiring Type | Pairs | Cable Length |
|---|---|---|---|
| 1000BaseCX | Shielded Twisted Pair (STP) | 1 | 25 m |
| 1000BaseT | EIA/TIA Category 5 UTP | 4 | 100 m |
| 1000BaseSX | Multimode fiber (MMF) with 62.5 micron core; 850 nm laser | 1 | 275 m |
| | Multimode fiber (MMF) with 50 micron core; 1300 nm laser | 1 | 550 m |
| 1000BaseLX/LH | Multimode fiber (MMF) with 62.5 micron core; 1300 nm laser | 1 | 550 m |
| | Single-mode fiber (SMF) with 50 micron core; 1300 nm laser | 1 | 550 m |
| | Single-mode fiber (SMF) with 9 micron core; 1300 nm laser | 1 | 10 km |
| 1000BaseZX | Single-mode fiber (SMF) with 9 micron core; 1550 nm laser | 1 | 70 km |
| | Single-mode fiber (SMF) with 8 micron core; 1550 nm laser | 1 | 100 km |

## 1.3.5: Network Addressing

Network addressing identifies either individual devices or groups of devices on a LAN. A pair of network devices that transmit frames between each other use a source and destination address field to identify each other. These addresses are called unicast addresses, or individual addresses, because they identify an individual network interface card (NIC).

The IEEE defines the format and assignment of network addresses by requiring manufacturers to encode globally unique unicast Media Access Control (MAC) addresses on all NICs. The first half of the MAC address identifies the manufacturer of the card and is called the organizationally unique identifier (OUI).

### 1.3.6: Bridging

Bridging is used to connect two network segments. This alleviates congestion problems on a single Ethernet segment and extends allowed cabling distances because the segments on each side of the bridge conformed to the same distance limitation as a single segment. This bridge is called "transparent bridging" because the end-point devices do not need to know that the bridge exists.

Transparent bridges forward frames only when necessary and, thus, reduces network overhead. To accomplish this, transparent bridges learning MAC addresses by examining the source MAC address of each frame received by the bridge; decides when to forward a frame or when to filter a frame, based on the destination MAC address; and creates a loop-free environment with other bridges by using the Spanning-Tree Protocol.

Generally, broadcasts and multicast frames are forwarded by the bridge in networks that use bridges. In addition, transparent bridges perform switching of frames using Layer 2 headers and Layer 2 logic and are Layer 3 protocol-independent. Store-and-forward operation, which means that the entire frame is received before the first bit of the frame is forwarded, is also typical in transparent bridging devices. However, the transparent bridge must perform processing on the frame, which also can increase latency.

A transparent bridge operates in the following manner:

• The bridge has no initial knowledge of the location of any end device; therefore, the bridge must listen to frames coming into each of its ports to figure out on which network a device resides.

• The bridge constantly updates its bridging table upon detecting the presence of a new MAC address or upon detecting a MAC address that has changed location from one bridge port to another. The bridge is then able to forward frames by looking at the destination address, looking up the address in the bridge table, and sending the frame out the port where the destination device is located.

• If a frame arrives with the broadcast address as the destination address, the bridge must forward or flood the frame out all available ports. However, the frame is not forwarded out the port that initially received the frame. Hence, broadcasts are able to reach all available networks. A bridge only segments collision domains but does not segment broadcast domains.

• If a frame arrives with a destination address that is not found in the bridge table, the bridge is unable to determine which port to forward the frame to for transmission. This is known as an unknown unicast. In this case, the bridge treats the frame as if it was a broadcast and forwards it out all remaining ports.

After a reply to that frame is received, the bridge will learn the location of the unknown station and add it to the bridge table.

• Frames that are forwarded across the bridge cannot be modified.

### 1.3.7: LAN Switching

An Ethernet switch uses the same logic as a transparent bridge, but performs more functions, has more features, and has more physical ports. Switches use hardware to learn MAC addresses and to make forwarding and filtering decisions, whereas bridges use software.
A switch listens for frames that enter all its interfaces. After receiving a frame, a switch decides whether to forward a frame and out which port(s). To perform these functions, switches perform three tasks:

• Learning, which means that the switch learns MAC addresses by examining the source MAC address of each frame the bridge receives. Switches dynamically learn the MAC addresses in the network to build its MAC address table. With a full, accurate MAC address table, the switch can make accurate forwarding and filtering decisions. Switches build the MAC address table by listening to incoming frames and examining the frame's source MAC address. If a frame enters the switch, and the source MAC address is not in the address table, the switch creates an entry in the table. The MAC address is placed in the table, along with the interface in which the frame arrived. This allows the switch to make good forwarding choices in the future. Switches also forward unknown unicast frames, which are frames whose destination MAC addresses are not yet in the bridging table, out all ports, which is called flooding, with the hope that the unknown device will be on some other Ethernet segment and will reply. When the unknown device does reply, the switch will build an entry for that device in the address table.

• Forwarding or filtering, which means that the switch decides when to forward a frame or when to filter it, i.e., not to forward it, based on the destination MAC address. Switches reduce network overhead by forwarding traffic from one segment to another only when necessary. To decide whether to forward a frame, the switch uses a dynamically built table called a bridge table or MAC address table. The switch looks at the previously learned MAC addresses in an address table to decide where to forward the frames.

• Loop prevention, which means that the switch creates a loop-free environment with other bridges by using Spanning-Tree Protocol (STP). Having physically redundant links helps LAN availability, and STP prevents the switch logic from letting frames loop around the network indefinitely, congesting the LAN.

Frames sent to unicast addresses are destined for a single device; frames sent to a broadcast address are sent to all devices on the LAN. Frames sent to multicast addresses are meant for all devices that care to receive the frame. Thus, when a switch receives a frame, it checks if the address is a unicast address, a broadcast address or a multicast address. If the address is unicast, and the address is in the address table, and if the interface connecting the switch to the destination device is not the same interface on which the frame arrived, the switch forwards the frame to the destination device. If the address is not in the address table, the switch forwards the frame on all ports. If the address is a broadcast or multicast address, the switch also forwards the frame on all ports.

The internal processing on a switch can decrease latency for frames. Switches can use store-and-forward

processing as well as cut-through processing logic. With cut-through processing, the first bits of the frame are sent out the outbound port before the last bit of the incoming frame is received. However, because the frame check sequence (FCS) is in the Ethernet trailer, a cut-through forwarded frame might have bit errors that the switch will not notice before sending most of the frame..

### 1.3.8: Wireless Networks

Conventional Ethernet networks require cables connected computers via hubs and switches. This has the effect of restricting the computer's mobility and requires that even portable computers be physically connected to a hub or switch to access the network. An alternative to cabled networking is wireless networking. The first wireless network was developed at the University of Hawaii in 1971 to link computers on four islands without using telephone wires. Wireless networking entered the realm of personal computing in the 1980s, with the advent to networking computers. However, it was only in the early 1990s that wireless networks started to gain momentum when CPU processing power became sufficient to manage data transmitted and received over wireless connections.

Wireless networks use network cards, called Wireless Network Adapters, that rely radio signals or infrared (IR) signals to transmit and receive data via a Wireless Access Point (WAP). The WAP uses has an RJ-45 port that can be attached to attach to a 10BASE-T or 10/100BASE-T Ethernet hub or switch and contains a radio transceiver, encryption, and communications software. It translates conventional Ethernet signals into wireless Ethernet signals it broadcasts to wireless network adapters on the network and performs the same role in reverse to transfer signals from wireless network adapters to the conventional Ethernet network. WAP devices come in many variations, with some providing the Cable Modem Router and Switch functions in addition to the wireless connectivity.

**Note:** Access points are not necessary for direct peer-to-peer networking,
which is called ad hoc mode, but they are required for a shared Internet
connection or a connection with another network. When access points are
used, the network is operating in the infrastructure mode.

### 1.3.8.1: Wireless Network Standards

In the absence of an industry standard, the early forms of wireless networking were single-vendor proprietary solutions that could not communicate with wireless network products from other vendors. In 1997, the computer industry developed the IEE 802.11 wireless Ethernet standard. Wireless network products based on this standard are capable of multivendor interoperability.

The IEEE 802.11 wireless Ethernet standard consists of the IEEE 802.11b standard, the IEEE 802.11a standard, and the newer IEEE 802.11g standard.

**Note**: The Bluetooth standard for short-range wireless networking is designed
to complement, rather than rival, IEEE 802.11-based

wireless networks.

• IEEE 802.11 was the original standard for wireless networks that was ratified in 1997. It operated at a maximum speed of 2 Mbps and ensured interoperability been wireless products from various vendors. However, the standard had a few ambiguities allowed for potential problems with compatibility between devices. To ensure compatibility, a group of companies formed the Wireless Ethernet Compatibility Alliance (WECA), which has come to be known as the Wi-Fi Alliance, to ensure that their products would work together. The term Wi-Fi is now used to refer to any IEEE 802.11 wireless network products that have passed the Wi-Fi Alliance certification tests.

• IEEE 802.11b, which is also called 11 Mbps Wi-Fi, operates at a maximum speed of 11 Mbps and is thus slightly faster than 10BASE-T Ethernet. Most IEEE 802.11b hardware is designed to operate at four speeds, using three different data-encoding methods depending on the speed range. It operates at 11 Mbps using quatenery phase-shift keying/complimentary code keying (QPSK/CCK); at 5.5 Mbps also using QPSK/CCK; at 2 Mbps using differential quaternary phase-shift keying (DQPSK); and at 1 Mbps using differential binary phase-shift keying (DBPSK). As distances change and signal strength increases or decreases, IEEE 802.11b hardware switches to the most suitable data-encoding method.
Wireless networks running IEEE 802.11b hardware use the 2.4 GHz radio frequency band that many portable phones, wireless speakers, security devices, microwave ovens, and the Bluetooth short-range networking products use. Although the increasing use of these products is a potential source of interference, the short range of wireless networks (indoor ranges up to 300 feet and outdoor ranges up to 1,500 feet, varying by product) minimizes the practical risks. Many devices use a spread-spectrum method of connecting with other products to minimize potential interference.
IEEE 802.11b networks can connect to wired Ethernet networks or be used as independent networks.

• IEEE 802.11a uses the 5 GHz frequency band, which allows for much higher speeds, reaching a maximum speed of 54 Mbps. The 5 GHz frequency band also helps avoid interference from devices that cause interference with lower-frequency IEEE 802.11b networks. IEEE 802.11a hardware maintains relatively high speeds at both short and relatively long distances.

Because IEEE 802.11a uses the 5 GHz frequency band rather than the 2.4 GHz frequency band used by IEEE 802.11b, standard IEEE 802.11a hardware cannot communicate with 802.11b hardware. A solution to this compatibility problem is the use of dual-band hardware. Dual-band hardware can work with either IEEE 802.11a or IEEE 802.11b networks, enabling you to move from an IEEE 802.11b wireless network at home or at Starbucks to a faster IEEE 802.11a office network.

• IEEE 802.11g is also known as Wireless-G and combines compatibility with IEEE 802.11b with the speed of IEEE 802.11a at longer distances. This standard was ratified in mid-2003, however, many network vendors were already selling products based on the draft IEEE 802.11g standard before the final standard was approved. These early IEEE 802.11g hardware was slower and less compatible than the specification promises. In some cases, problems with early-release IEEE 802.11g hardware can be solved through firmware upgrades.

## 1.3.8.2: Wireless Network Modes

Wireless networks work in one of two modes that are also referred to as topologies. These two modes are ad-hoc mode and infrastructure mode. The mode you implement depends on whether you want your

computers to communicate directly with each other, or via a WAP.

• In ad-hoc mode, data is transferred to and from wireless network adapters connected to the computers. This cuts out the need to purchase a WAP. Throughput rates between two wireless network adapters are twice as fast as when you use a WAP. However, a network in ad-hoc mode cannot connect to a wired network as a WAP is required to provide connectivity to a wired network. An ad-hoc network is also called a peer-to-peer network.

• In infrastructure mode, data is transferred between computers via a WAP. Because a WAP is used in infrastructure mode, it provides connectivity with a wired network, allowing you to expand a wired network with wireless capability. Your wired and wirelessly networked computers can communicate with each other. In addition, a WAP can extend your wireless network's range as placing a WAP between two wireless network adapters doubles their range. Also, some WAPs have a built-in router and firewall. The router allows you to share Internet access between all your computers, and the firewall hides your network. Some of these multifunction access points include a hub with RJ-45 ports.

### 1.3.8.3: Security Features

Because wireless networks can be accessed by anyone with a compatible wireless network adapter, most models of wireless network adapters and WAPs provide for encryption options. Some devices with this feature enable you to set a security code known as an SSID on the wireless devices on your network. This seven-digit code prevents unauthorized users from accessing your network and acts as an additional layer of security along with your normal network authentication methods, such as user passwords. Other wireless network adapters and WAPs use a list of authorized MAC addresses to limit access to authorized devices only.

All Wi-Fi products support at least 40-bit encryption through the wired equivalent privacy (WEP) specification, but the minimum standard on newer products is 64-bit WEP encryption. Many vendors also offer 128-bit or 256-bit encryption on some of their products. However, the WEP specification is insecure. It is vulnerable to brute-force attacks at shorter key lengths, and it is also vulnerable to differential cryptanalysis attacks, which is the process of comparing an encrypted text with a known portion of the plain text and deriving the key by computing the difference between them. Because WEP encrypts TCP headers, hackers know what the headers should contain in many cases, and they can attempt to find patterns in a large body of collected WEP communications in order to decrypt the key. The attack is complex and difficult to automate, so it is unlikely to occur for most networks, especially at key lengths greater than 128 bits. Furthermore, WEP does not prevent an intruder from attaching a hidden WAP on the network and using it to exploit the network.

New network products introduced in 2003 and beyond now incorporate a new security standard known as Wi-Fi Protected Access (WPA). WPA is derived from the developing IEEE 802.11i security standard, which will not be completed until mid-decade. WPA-enabled hardware works with existing WEP-compliant devices, and software upgrades might be available for existing devices.

## Section 1.4: The Cisco IOS Software

Cisco routers run the Cisco Internetworking Operating System (IOS) with a command-line interface (CLI). The IOS also runs on some Cisco switch models, and it uses CLI. However, in some cases, the IOS CLI on a switch is slightly different than on a router. Furthermore, the IOS on the 1900 series switches is slightly

different than on some other Cisco IOS-based switches.

### 1.4.1: The Cisco IOS Software Command-Line Interface

The majority of Cisco routers run Cisco IOS Software with the command-line interface (CLI). The CLI is used to interface with the device and send commands to the device. This is achieved through the use of a terminal, a terminal emulator, or a Telnet connection. Some routing cards, such as the Multilayer Switch Feature Card (MSFC) daughter card for the Catalyst 6000 series LAN switches, also run Cisco IOS Software. Understanding the Cisco IOS Software CLI is as fundamental to supporting routers.
There are three ways in which you can access the CLI: you access the router through the console; through a dialup device through a modem attached to the auxiliary port; or by using a Telnet connection. Which ever method you use, you enter user exec mode first. User exec mode is one of three command exec modes in the IOS user interface. Enable mode, also known as privileged mode or Privileged exec mode, and command mode are the others. Enable mode is so named because the enable command is used to reach this mode. User mode allows commands that are not disruptive to be issued, with some information being displayed to the user. Privileged mode supports a superset of commands compared to user mode. However, none of the commands in user mode or privileged mode changes the configuration of the router.
Passwords are required for Telnet and auxiliary access as of Cisco IOS Release 12.x and later. However, there are no preconfigured passwords; therefore, you must configure passwords for Telnet and auxiliary access from the console first.

All Cisco routers have a console port, and most have an auxiliary port. The console port is intended for local administrative access from an ASCII terminal or a computer using a terminal emulator. The auxiliary port is intended for asynchronous dial access from an ASCII terminal or terminal emulator; the auxiliary port is often used for dial backup.

#### 1.4.1.1: The CLI Help Features

Typing ? in the console displays help for all commands supported by the CLI mode. In other words, the information supplied by using help depends on the CLI mode. If ? is typed in user mode, the commands allowed only in privileged exec mode are not displayed. Also, help is available in configuration mode; only configuration commands are displayed in that mode of operation. IOS stores the commands that you type in a history buffer. The last ten commands are stored by default. You can change the history size with the terminal history size size command, where size is the number of IOS commands for the CLI to store; this can be set to a value between 0 and 256. You can then retrieve commands so that you do not have to retype the commands.

#### 1.4.1.2: Syslog Messages and the debug Command

IOS creates messages, which are called syslog messages, when different events occur and, by default, sends them to the console. The router also generates messages that are treated like syslog messages in response to some troubleshooting tasks that you might perform. The debug command is one of the key diagnostic tools for troubleshooting problems on a Cisco router. It enables monitoring points in the IOS and generates messages that describe what the IOS is doing and seeing. When any debug command option is enabled, the router processes the messages with the same logic as other syslog messages.

The console port always receives syslog messages; however, when you Telnet to the router no syslog messages are seen unless you issue the terminal monitor command. Another alternative for viewing syslog messages is to have the IOS record the syslog messages in a buffer in RAM and then use the show

logging command to display the messages. For Telnet users, having the messages buffered using the global config command logging buffered is particularly useful. Finally, the logging synchronous lineconfiguration subcommand can be used for the console and vtys to tell the router to wait until the last command output is displayed before showing any syslog messages onscreen.
Syslog messages also can be sent to another device. Two alternatives exist: sending the messages to a syslog server, and sending the messages as SNMP traps to a management station. The logging host command, where host is the IP address or host name of the syslog server, is used to enable sending messages to the external server. After SNMP is configured, the snmp-server enable traps command tells the IOS to forward traps, including syslog messages.

## 1.4.2: Configuring Cisco IOS Software

Configuration mode is one of the modes for the Cisco CLI. It is similar to user mode and privileged mode. User mode allows commands that are not disruptive to be issued, with some information being displayed to the user. Privileged mode supports a superset of commands compared to user mode. However, none of the commands in user or privileged mode changes the configuration of the router. Configuration mode is another mode in which configuration commands are typed.

Commands typed in configuration mode update the active configuration file. These changes to the configuration occur immediately each time you press the Enter key at the end of a command. Configuration mode itself contains a multitude of subcommand modes. The type of command you enter moves you from one configuration subcommand mode to which ever subcommand mode is appropriate. For example, the interface command, which is the most commonly used configuration command, would move you to interface configuration mode.

Generally, when multiple instances of a parameter can be set on a single router, the command used to set the parameter is likely a configuration subcommand. Items that are set once for the entire router are likely global commands. For example, the hostname command is a global command because there is only one host name per router.

You can use CTRL + Z from any part of configuration mode, or use the exit command from global configuration mode, to exit configuration mode and return to privileged exec mode. The configuration mode end command also exits from any point in the configuration mode back to privileged exec mode. The exit commands from subcommand modes back up one level toward global configuration mode.

### 1.4.2.1: Managing Configuration Files

Your configuration commands, as well as some default configuration commands are stored in the configuration file. No hard disk or diskette storage exists on Cisco routers therefore; the configuration file is stored in memory. The configuration files can also be stored as ASCII text files anywhere exterior to the router using TFTP or FTP. Cisco routers support a number of types of memory. This includes:
• RAM, which is sometimes called DRAM for dynamic random-access memory, is used by the router in the same way it is used by any other computer: for storing data being used by the processor. The active configuration file, running-config, which is the configuration file that the router uses during operation, is stored in RAM.

• ROM, or read-only memory, stores a bootable IOS image, which is not typically used for normal operation. It contains the code that is used to boot the router and allows the router to access the IOS

image.

• Flash memory, which can be either an EEPROM or a PCMCIA card, stores fully functional IOS images and is the default location where the router accesses its IOS at boot time. Flash memory also can be used to store configuration files on some Cisco routers.

• NVRAM, which is nonvolatile RAM, stores the initial or startup configuration file, startup-config.
All these types of memory, except RAM, are permanent memory.

When the router first comes up, the router copies the stored configuration file from NVRAM into RAM, so the active and startup configuration files are identical at that point. The show running-config and show startup-config commands are used to verify the active and startup configuration files respectively. You can use the copy running-config startup-config command to overwrite the current startup configuration file with the current active configuration file. The copy command can be used to copy files in a router, most typically a configuration file, or a new version of the IOS Software. The most basic method for moving configuration files in and out of a router is by using a TFTP server. The copy command is used to copy configuration files among RAM, NVRAM, and a TFTP server. The syntax for copy command used to copy configuration files among RAM, NVRAM, and a TFTP server specifies the source location and the destination of the configuration file as in:

**copy source destination**

The source and the destination parameters can be running-config, startup-config, or tftp for RAM, NVRAM, and a TFTP server respectively. However, the source and the destination parameters cannot be the same. Thus, the following syntax copies the configuration from RAM to NVRAM, overwriting the current startup configuration file with the active configuration file:

**copy running-config startup-config**

The copy command does not always replace the existing file that it is copying. Any copy command option moving a file into NVRAM or a TFTP server replaces the existing file, however, any copy into RAM works by adding the commands to the active configuration file. Thus, if you change the active configuration file and then want to revert to the startup configuration file, you must use the reload command, which reboots the router

Two commands can be used to erase the contents of NVRAM. These are the write erase command, which is the older command, and the erase startup-config command, which is the newer command.

## 1.4.2.2: Upgrading Cisco IOS Software

Typically, a router has one IOS image and that is the IOS that is used. This IOS image is typically stored in Flash memory, which is a rewriteable, permanent form of storage. The IOS image can also be placed on an external TFTP server, but this is typically done for testing. In the IOS upgrade process you first must obtain the IOS image from Cisco. Then you must place the IOS image into the default directory of a TFTP server. Finally, you must use the copy tftp flash command from the router to copy the files into Flash memory.

During this process, the router will need to discover the IP address or host name of the TFTP server; the name of the file; the space available in Flash memory for this file; and whether you want to erase the old files. The router will prompt you for answers, as necessary. Afterward, the router erases Flash memory as needed, copies the file, and then verifies that the checksum for the file shows that no errors occurred in transmission. The show flash command then can be used to verify the contents of Flash memory. Before the new IOS is used, however, the router must be reloaded.

### 1.4.2.3: The Cisco IOS Software Boot Sequence

The basic boot sequence for a Cisco router is:

**Step 1**: The router performs a power-on self-test (POST) to discover and verify the hardware.

**Step 2**: The router loads and runs bootstrap code from ROM.

**Step 3**: The router finds the IOS or other software and loads it.

**Step 4**: The router finds the configuration file and loads it into running config.

All routers attempt all four steps each time that the router is powered on or reloaded. The POST code and functions cannot be changed by the router administrator. The location of the bootstrap code, the IOS to load, and the configuration file can be changed by the administrator-but you almost always use the default location for the bootstrap code (ROM) and for the initial configuration (NVRAM). So, the location of IOS or other software is the only part that typically is changed.

Three categories of operating systems can be loaded into the router:

• The full-function IOS image, which is typically located in Flash memory but can also be located on a TFTP server. This is the normal, full-feature IOS used in production;

• A limited-function IOS that resides in ROM; and provides basic IP connectivity when Flash memory is faulty and you need IP connectivity to copy a new IOS into Flash memory. This limited-function IOS is called RXBOOT mode.

• A different non-IOS operating system that is also stored in ROM. This operating system, called ROM Monitor (ROMMON) mode, is used for low-level debugging and for password recovery. Unless you are performing password recovery, you would seldom use ROMMON mode.

The configuration register tells the router whether to use a full-featured IOS, ROMMON, RXBOOT mode. The configuration register is a 16-bit software register in the router, and its value is set using the configregister global configuration command. The boot field is the name of the low-order 4 bits of the configuration register. This field can be considered a 4-bit value, represented as a single hexadecimal digit. If the boot field is hex 0, ROMMON is loaded. If the boot field is hex 1, RXBOOT mode is used. For anything else, it loads a full-featured IOS.

The second method used to determine where the router tries to obtain an IOS image is through the use of the boot system configuration command. If the configuration register calls for a full-featured IOS, the router

reads the configuration file for boot system commands.

If there are no boot system commands, the router takes the default action, which is to load the first file in Flash memory. Table 1.7 lists the configuration register and the boot system command.

**Table 1.7: The boot system Commands**

| Boot Filed Value | Function |
|---|---|
| **0x0 0x1** | Loads ROMMON and ignores **boot system**commands. Loads IOS from ROM and ignores **boot system**commands. This is also known as RXBOOT mode. |
| **0x2-0xF** | If used with the **no boot**command, the first IOS file in Flash memory is loaded; if that fails, the router broadcasts looking for an IOS on a TFTP server. If that fails, IOS from ROM is loaded. |
| **0x2-0xF** | If used with the **boot system ROM**command, IOS from ROM is loaded. |
| **0x2-0xF** | If used with the **boot system flash** command, the first file from Flash memory is loaded. |
| **0x2-0xF** | If used with the **boot system flash *file_name***command, IOS with the specified ***file_name***is loaded from Flash memory. |
| **0x2-0xF** | If used with the **boot system tftp *file_name*10.1.1.1** command, IOS with the specified ***file_name***is loaded from the TFTP server. |
| **0x2-0xF** | If used with multiple **boot system**commands, an attempt occurs to load IOS based on the first boot command in configuration. If that fails, the second boot command is used, etc., until an IOS is loaded successfully. |

## Section 1.5: Spanning-Tree Protocol (STP)

A Layer 2 switch, which functions as a transparent bridge, offers no additional links for redundancy purposes. To add redundancy, a second switch must be added. Now two switches offer the transparent bridging function in parallel. LAN designs with redundant links introduce the possibility that frames might loop around the network forever. These looping frames would cause network performance problems. For example, when the switches receive an unknown unicast, both will flood the frame out all their available ports, including the ports that link to the other switch, resulting in what is known as a bridging loop, as the frame is forwarded around and around between two switches. This occurs because parallel switches are unaware of each other. The Spanning-Tree Protocol (STP), which allows the redundant LAN links to be used while preventing frames from looping around the LAN indefinitely through those redundant links, was developed to overcome the possibility of bridging loops. It enables switches to become aware of each other so that they can negotiate a loop-free path through the network. Loops are discovered before they are opened for use, and redundant links are shut down to prevent the loops from forming. STP is communicated between all connected switches on a network. Each switch executes the Spanning-Tree Algorithm (STA) based on information received from other neighboring switches. The algorithm chooses a reference point in the network and calculates all the redundant paths to that reference point. When redundant paths are found,

STA picks one path to forward frames with and disables or blocks forwarding on the other redundant paths. STP computes a tree structure that spans all switches in a subnet or network. Redundant paths are placed in a blocking or standby state to prevent frame forwarding. The switched network is then in a loop-free condition. However, if a forwarding port fails or becomes disconnected, the STA will run again to recompute the Spanning-Tree topology so that blocked links can be reactivated.

By default, STP is enabled on all ports of a switch. STP should remain enabled in a network to prevent bridging loops from forming. However, if STP has been disabled on a CLI-based switch, it can be reenabled with the following command:

```
Switch (enable) set spantree enable [ all |
module_number/port_number ]
```

If STP has been disabled on an IOS-based switch, it can be re-enabled with the following command:

```
Switch (config)#
spantree vlan_list
```

You can use the show spantree [ vlan ] command to view the status of STP on either a CLI- or IOSbased switch.
The STA places each bridge/switch port in either a forwarding state or a blocking state. All the ports in forwarding state are considered to be in the current spanning tree. The collective set of forwarding ports creates a single path over which frames are sent between Ethernet segments. Switches can forward frames out ports and receive frames in ports that are in forwarding state; switches do not forward frames out ports and receive frames in ports that are in blocking state.

STP uses three criteria to choose whether to put an interface in forwarding state or a blocking state:

• STP elects a root bridge and puts all interfaces on the root bridge in forwarding state.

• Each nonroot bridge considers one of its ports to have the lowest administrative cost between itself and the root bridge. STP places this lowest-root-cost interface, called that bridge's root port, in forwarding state.

• Many bridges can attach to the same Ethernet segment. The bridge with the lowest administrative cost from itself to the root bridge, as compared with the other bridges attached to the same segment, is placed in forwarding state. The lowest-cost bridge on each segment is called the designated bridge, and that bridge's interface, attached to that segment, is called the designated port.
All other interfaces are placed in blocking state.

### 1.5.1: Root Bridge Election

For all switches in a network to agree on a loop-free topology, a common frame of reference must exist. This reference point is called the Root Bridge. The Root Bridge is chosen by an election process among all connected switches. Each switch has a unique Bridge ID that it uses to identify itself to other switches. The Bridge ID is an 8-byte value. 2 bytes of the Bridge ID is used for a Bridge Priority field, which is the priority or weight of a switch in relation to all other switches. The other 6 bytes of the Bridge ID is used for

the MAC Address field, which can come from the Supervisor module, the backplane, or a pool of 1024 addresses that are assigned to every Supervisor or backplane depending on the switch model. This address is hardcoded, unique, and cannot be changed.

The election process begins with every switch sending out BPDUs with a Root Bridge ID equal to its own Bridge ID as well as a Sender Bridge ID. The latter is used to identify the source of the BPDU message. Received BPDU messages are analyzed for a lower Root Bridge ID value. If the BPDU message has a Root Bridge ID of the lower value than the switch's own Root Bridge ID, it replaces its own Root Bridge ID with the Root Bridge ID announced in the BPDU. If two Bridge Priority values are equal, then the lower MAC address takes preference. The switch is then nominates the new Root Bridge ID in its own BPDU messages although it will still identify itself as the Sender Bridge ID. Once the process has converged, all switches will agree on the Root Bridge until a new switch is added.

The Root Bridge election is based on the idea that one switch is chosen as a common reference point, and all other switches choose ports that are closest to the Root. The Root Bridge election is also based on the idea that the Root Bridge can become a central hub that interconnects other legs of the network. Therefore, the Root Bridge can be faced with heavy switching loads in its central location. If heavy loads of traffic are expected to pass through the Root Bridge, the slowest switch is not the ideal candidate. Furthermore, only one Root Bridge is elected. This is thus not fault tolerant. To overcome these problems, you should set a Root Bridge in a determined fashion, and set a secondary Root Bridge in case of primary Root Bridge failure. The Root Bridge and the secondary Root Bridge should be placed near the center of the network. To configure a CLI-based Catalyst switch to become the Root Bridge, use the following command to modify the Bridge Priority value so that a switch can be given a lower Bridge ID value to win a Root Bridge election:

```
Switch (enable) set spantree priority
bridge_priority [ vlan ]
```

Alternatively, you can use the following command:

```
Switch (enable) set spantree root [
secondary ] [ vlan_list ]
[ dia diameter ] [ hello hello_time ]
```

This command is a macro that executes several other commands. The result is a more direct and automatic way to force one switch to become the Root Bridge. Actual Bridge Priorities are not given in the command. Rather, the switch will modify STP values according to the current values in use within the active network. To configure an IOS-based Catalyst switch to become the Root Bridge, use the following command to modify the Bridge Priority value so that a switch can be given a lower Bridge ID value to win a Root Bridge election:

```
Switch (config)# spanning-tree [ vlan
vlan_list ] priority
bridge_priority
```

### 1.5.2: Root Ports Election

Once a reference point has been nominated and elected for the entire switched network, each non-root switch must find its relation to the Root Bridge. This action can be performed by selecting only one Root Port on each non-root switch. STP uses the Root Path Cost to select a Root Port. The Root Path Cost is the cumulative cost of all the links leading to the Root Bridge. A particular switch link has a cost associated with it called the Port or Path Cost. This cost is inversely proportional to the port's bandwidth. As the Path Cost travels along, other switches can modify its value to make it cumulative. The Path Cost is known only to the local switch where the port or "path" to a neighboring switch resides as it is not contained in the BPDU. Only the Root Path Cost is contained in the BPDU. Path Costs are defined as a one-byte value.
The Root Bridge sends out a BPDU with a Root Path Cost value of zero because its ports sit directly on the Root Bridge. When the next closest neighbor receives the BPDU, it adds the Path Cost of its own port where the BPDU arrived. The neighbor then sends out BPDUs with this new cumulative value as the Root Path Cost. This value is incremented by subsequent switch port Path Costs as the BPDU is received by each switch on down the line. After incrementing the Root Path Cost, a switch also records the value in its memory. When a BPDU is received on another port and the new Root Path Cost is lower than the previously recorded value, this lower value becomes the new Root Path Cost. In addition, the lower cost tells the switch that the Root Bridge must be closer to this port than it was on other ports. The switch has now determined which of its ports is the closest to the root-the Root Port.
If desired, the cost of a port can be modified from the default value. However, changing one port's cost may influence STP to choose that port as a Root Port. Therefore careful calculation is required to ensure that the desired path will be elected. On a CLI-based switch, the port cost can be modified by using one of the following commands:

```
Switch (enable) set spantree portcost
module_number/port_number cost
```

or

```
Switch (enable) set spantree portvlancost
module_number/port_number
[ cost cost ] [ vlan_list ]
```

On an IOS-based switch, the port cost for individual VLANs can be modified by using the following command:

```
Switch (config-if)# spanning-tree [ vlan
vlan_list ] cost cost
```

### 1.5.3: Designated Ports Election

Once the Root Path Cost values have been computed, the Root Ports have been identified; however, all other links are still connected and could be active, leaving bridging loops. To remove the bridging loops, STP makes a final computation to identify one Designated Port on each network segment which would forward traffic to and from that segment. Switches choose a Designated Port based on the lowest cumulative Root Path Cost to the Root Bridge. All ports are still active and bridging loops are still possible. STP has a set of progressive states that each port must go through, regardless of the type or identification. These states will actively prevent loops from forming.

### 1.5.4: STP States

To participate in STP, each port of a switch must progress through several states. A port begins in a Disabled state moving through several passive states and finally into an active state if allowed to forward traffic. The STP port states are: Disabled, Blocking, Listening, Learning, and Forwarding.

• Ports that are administratively shut down by the network administrator or by the system due to a fault condition are in the Disabled state. This state is special and is not part of the normal STP progression for a port.

• After a port initializes, it begins in the Blocking state so that no bridging loops can form. In the Blocking state, a port cannot receive or transmit data and cannot add MAC addresses to its address table. Instead, a port is only allowed to receive BPDUs. Also, ports that are put into standby mode to remove a bridging loop enter the Blocking state.

• The port will be moved from the Blocking state to the Listening state if the switch thinks that the port can be selected as a Root Port or Designated Port. In the Listening state, the port still cannot send or receive data frames. However, the port is allowed to receive and send BPDUs so that it can actively participate in the Spanning-Tree topology process. Here the port is finally allowed to become a Root Port or Designated Port because the switch can advertise the port by sending BPDUs to other switches. Should the port lose its Root Port or Designated Port status, it is returned to the Blocking state.

• After a period of time called the Forward Delay in the Listening state, the port is allowed to move into the Learning state. The port still sends and receives BPDUs as before. In addition, the switch can now learn new MAC addresses to add into its address table.

• After another Forward Delay period in the Learning state, the port is allowed to move into the Forwarding state. The port can now send and receive data frames, collect MAC addresses into its address table, and send and receive BPDUs. The port is now a fully functioning switch port within the Spanning-Tree topology.

### 1.5.5: STP Timers

STP operates as switches send BPDUs to each other in an effort to form a loop-free topology. The BPDUs take a finite amount of time to travel from switch to switch. In addition, news of a topology change such as a link or Root Bridge failure can suffer from propagation delays as the announcement travels from one side of a network to the other. Because of the possibility of these delays, preventing the Spanning-Tree topology from converging until all switches have had time to receive accurate information is important. STP uses three timers for this purpose. There are three timers: Hello Time, Forward Delay, and Max Age.

• Hello Time is the time interval between Configuration BPDUs sent by the Root Bridge. The Hello Time value configured in the Root Bridge switch will determine the Hello Time for all non-root switches. However, all switches have a locally configured Hello Time that is used to time Topology Change Notification (TCN) BPDUs when they are retransmitted. The IEEE 802.1D standard specifies a default Hello Time value of two seconds.

• Forward Delay is the time interval that a switch port spends in both the Listening and Learning states.

The default value is 15 seconds.

• Max Age is the time interval that a switch stores a BPDU before discarding it. While executing the STP, each switch port keeps a copy of the "best" BPDU that it has heard. If the source of the BPDU loses contact with the switch port, the switch will notice that a topology change has occurred after the Max Age time elapses and the BPDU is aged out. The default Max Age value is 20 seconds.

To announce a change in the active network topology, switches send a Topology Change Notification (TCN) BPDU. This occurs when a switch either moves a port into the Forwarding state or moves a port from Forwarding or Learning into the Blocking state. The switch sends a TCN BPDU out its Designated Port. The TCN BPDU carries no data about the change, but only informs recipients that a change has occurred. However, the switch will not send TCN BPDUs if the port has been configured with PortFast enabled. The switch will continue sending TCN BPDUs every Hello Time interval until it gets an acknowledgement from an upstream neighbor. As the upstream neighbors receive the TCN BPDU, they will propagate it on toward the Root Bridge. When the Root Bridge receives the BPDU, the Root Bridge sends out an acknowledgement. The Root Bridge also sends out the Topology Change flag in a Configuration BPDU so that all other bridges will shorten their bridge table aging times down from the default 300 seconds to the Forward Delay value. This condition causes the learned locations of MAC addresses to be flushed out sooner than they normally would, easing the bridge table corruption that might occur due to the change in topology. However, any stations that are actively communicating during this time will be kept in the bridge table. This condition lasts for the sum of the Forward Delay and the Max Age.

The three STP timers can be adjusted. These timers need only be modified on the Root Bridge and any secondary or backup Root Bridges because the Root Bridge propagates all three timer values throughout the network in the Configuration BPDU.

### 1.5.6: Optional STP Features

Cisco has added several proprietary enhancements to STP and to the logic used by its switches. Also, the IEEE, which owns the STP specifications, has made other enhancements, some similar to Cisco's proprietary enhancements.

### 1.5.6.1: EtherChannel

EtherChannel combines from two to eight parallel Ethernet trunks between the same pair of switches, bundled into an EtherChannel. STP treats an EtherChannel as a single link, so if at least one of the links is up, STP convergence does not have to occur. With each pair of Ethernet links configured as an EtherChannel, STP treats each EtherChannel as a single link. Thus, both links to the same switch must fail for a switch to need to cause STP convergence. Without EtherChannel, if you have multiple parallel links between two switches, STP blocks all the links except one. With EtherChannel, all the parallel links can be up and working at the same time, while reducing the number of times STP must converge, which in turn makes the network more available.

EtherChannel also provides more network bandwidth. All trunks in an EtherChannel are either forwarding or blocking, because STP treats all the trunks in the same EtherChannel as one trunk. When an EtherChannel is in forwarding state, the switches forward traffic over all the trunks, providing more bandwidth.

### 1.5.6.2: PortFast

PortFast allows a switch to place a port in forwarding state immediately when the port becomes physically active. However, the only ports on which you can safely enable PortFast are ports on which you know that no bridges, switches, or other STP devices are connected. Thus, PortFast is most appropriate for connections to end-user devices. If you turn on PortFast for end-user devices, when an end-user PC boots, as soon as the Ethernet card is active, the switch port can forward traffic. Without PortFast, each port must wait MaxAge plus twice the Forward Delay, which is 50 seconds with the default MaxAge and Forward Delay settings.

### 1.5.6.3: Rapid Spanning Tree (IEEE 802.1w)

The IEEE has improved the 802.1d protocol, which defines STP, with the definition of Rapid Spanning Tree Protocol (RSTP), as defined in standard 802.1w. RSTP is similar to STP in that it elects the root switch using the same parameters and tiebreakers; elects the root port on nonroot switches with the same rules; elects designated ports on each LAN segment with the same rules; and places each port in either a forwarding state or a blocking state, with the latter being called the discarding state instead of the blocking state.

RSTP can be deployed alongside traditional STP bridges and switches, with RSTP features working in switches that support it, and STP features working in the switches that support only STP.
The advantage RSTP has over STP is improved network convergence when network topology changes occur. STP convergence has essentially wait periods: a switch must first cease to receive root BPDUs for MaxAge: seconds before it can begin to transition any interfaces from blocking to forwarding. For any interfaces that need to transition from blocking to forwarding, the interface must endure Forward Delay seconds in listening state and Forward Delay more seconds in learning state before being placed in forwarding state. By default, these three wait periods of are 20, 15, and 15 seconds.
RSTP convergence times typically take less than 10 seconds. In some cases, they can be as low as 1 to 2 seconds.

# Topic 2: Virtual LANs and Trunking

A fully Layer 2 switched network is referred to as a flat network topology. A flat network is a single broadcast domain in which every connected device sees every broadcast packet that is transmitted. As the number of hosts on the network increases, so does the number of broadcasts. Due to the Layer 2 foundation, flat networks cannot contain redundant paths for load balancing or fault tolerance. However, a switched network environment offers the technology to overcome flat network limitations. Switched networks can be subdivided into virtual LANs (VLANs), each of which is a single broadcast domain. All devices connected to the VLAN receive broadcasts from other VLAN members. However, devices connected to a different VLAN will not receive those same broadcasts because is made up of defined members communicating as a logical network segment. A VLAN can have connected members located

| Collision Domains |
| --- |
| A collision domain is a set of network interface cards (NICs) for which a frame sent by one NIC could result in a collision with a frame sent by any other NIC in the same collision domain. |

| Broadcast Domains |
| --- |
| A broadcast domain is a set of NICs for |

which a broadcast frame sent by one NIC is
received by all other NICs in the same
broadcast domain

anywhere in the campus network, as long as VLAN connectivity is provided between all members. Layer 2 switches are configured with a VLAN mapping and provide the logical connectivity between the VLAN members.

# Section 2.1: VLAN Membership

When a VLAN is provided at an access layer switch, an end user must be able to gain membership to it. Two membership methods exist on Cisco Catalyst switches: static VLANs and dynamic VLANs.

• Static VLANs offer port-based membership, where switch ports are assigned to specific VLANs. End user devices become members in a VLAN based on which physical switch port they are connected to. No handshaking or unique VLAN membership protocol is needed for the end devices; they automatically assume VLAN connectivity when they connect to a port. The static port-to-VLAN membership is normally handled in hardware with application specific integrated circuits (ASICs) in the switch. This membership provides good performance because all port mappings are done at the hardware level with no complex table lookups needed.

• Dynamic VLANs are used to provide membership based on the MAC address of an end user device. When a device is connected to a switch port, the switch must query a database to establish VLAN membership. A network administrator must assign the user's MAC address to a VLAN in the database of a VLAN Membership Policy Server (VMPS). With Cisco switches, dynamic VLANs are created and managed through the use of network management tools like CiscoWorks 2000 or CiscoWorks for Switched Internetworks (CWSI). Dynamic VLANs allow a great deal of flexibility and mobility for end users, but require more administrative overhead.

## Section 2.2: Extent of VLANs

The number of VLANs that will be implemented on a network is dependent on traffic patterns, application types, segmenting common workgroups, and network management requirements. However, consideration must be given to the relationship between VLANs and the IP addressing schemes. Cisco recommends a one-to-one correspondence between VLANs and IP subnets, which means that if a Class C network address is used for a VLAN, then no more than 254 devices should be in the VLAN. Cisco also recommends that VLANs not extend beyond the Layer 2 domain of the distribution switch, i.e., the VLAN should not reach across the core of a network and into another switch block. This is designed to keep broadcasts and unnecessary movement of traffic out of the core block. VLANs can be scaled in the switch block by using two basic methods: end-to-end VLANs and local VLANs.

• End-to-end VLANs span the entire switch fabric of a network and are also called campus-wide VLANs. They are positioned to support maximum flexibility and mobility of end devices. Users are assigned to VLANs regardless of their physical location. This means that each VLAN must be made available at the access layer in every switch block. End-to-end VLANs should group users according to common requirements, following the 80/20 rule. Although only 20 percent of the traffic in a VLAN is expected to cross the network core, end-to-end VLANs make it possible for all traffic within a single VLAN to cross

the core. Because all VLANs must be available at each access layer switch, VLAN trunking must be used to carry all VLANs between the access and distribution layer switches.

• In the modern network, end users require access to central resources outside their VLAN. Users must cross into the network core more frequently, making the end-to-end VLANs cumbersome and difficult to maintain. Most enterprise networks have adopted the 20/80 rule. Local VLANs deployed in this type of network. Local VLANs are designed to contain user communities based on geographic boundaries, with little regard to the amount of traffic leaving the VLAN. They range in size from a single switch in a wiring closet to an entire building. Local VLANs enables the Layer 3 function in the campus network to intelligently handle the inter-VLAN traffic loads. This provides maximum availability by using multiple paths to destinations, maximum scalability by keeping the VLAN within a switch block, and maximum manageability.

## Section 2.3: VLAN Trunking

When using VLANs in networks that have multiple interconnected switches, you need to use VLAN trunking between the switches. With VLAN trunking, the switches tag each frame sent between switches so that the receiving switch knows to what VLAN the frame belongs. End user devices connect to switch ports that provide simple connectivity to a single VLAN each. The attached devices are unaware of any VLAN structure.

A trunk link can transport more than one VLAN through a single switch port. A trunk link is not assigned to a specific VLAN. Instead, one or more active VLANs can be transported between switches using a single physical trunk link. Connecting two switches with separate physical links for each VLAN is also possible. In addition, trunking can support multiple VLANs that have members on more than one switch.
Cisco switches support two trunking protocols, namely, Inter-Switch Link (ISL) and IEEE 802.1Q.

### 2.3.1: Inter-Switch Link (ISL)

Cisco created ISL before the IEEE standardized a trunking protocol. Thus, ISL is a Cisco proprietary solution and can be used only between two Cisco switches. ISL fully encapsulates each original Ethernet frame in an ISL header and trailer. The original Ethernet frame inside the ISL header and trailer remains unchanged.

The ISL header includes a VLAN field that provides a place to encode the VLAN number. By tagging a frame with the correct VLAN number inside the header, the sending switch can ensure that the receiving switch knows to which VLAN the encapsulated frame belongs. Also, the source and destination addresses in the ISL header use MAC addresses of the sending and receiving switch, as opposed to the devices that actually sent the original frame.

### 2.3.2: IEEE 802.1Q

After Cisco created ISL, the IEEE completed work on the 802.1Q standard. IEEE 802.1Q uses a different style of header to tag frames with a VLAN number than the ISL. It does not encapsulate the original frame, but adds a 4-byte header to the original Ethernet header. This additional header includes a field with which to identify the VLAN number. Because the original header has been changed, IEEE 802.1Q encapsulation forces a recalculation of the original FCS field in the Ethernet trailer, because the FCS is based on the contents of the entire frame. IEEE 802.1Q also introduces the concept of a native VLAN on a trunk. Frames belonging to this VLAN are not encapsulated with tagging information. In the event that a host is connected

to an IEEE 802.1Q trunk link, that host will be able to receive and understand only the native VLAN frames.

## Section 2.4: VLAN Trunking Protocol (VTP)

Administration of network environments that consists of many interconnected switches is complicated. Cisco has developed a propriety solution to manage VLANs across such networks using the VLAN Trunking Protocol (VTP) to exchange VLAN configuration information between switches. VTP uses Layer 2 trunk frames to exchange VLAN information so that the VLAN configuration stays consistent throughout a network. VTP also manages the additions, deletions, and name changes of VLANs across multiple switches from a central point, minimizing misconfigurations and configuration inconsistencies that can cause problems, such as duplicate VLAN names or incorrect VLANtype settings.

VTP is organized into management domains or areas with common VLAN requirements. A switch can belong to only one VTP domain. Switches in different VTP domains do not share VTP information. Switches in a VTP domain advertise several attributes to their domain neighbors. Each advertisement contains information about the VTP management domain, VTP configuration revision number, known VLANs, and specific VLAN parameters.

**The VTP Configuration Revision Number**

Each time a VTP server modifies its VLAN information, it increments the configuration revision number that is sent with the VTP advertisement by 1. The VTP server then sends out a VTP advertisement that includes the new configuration revision number. When a switch receives a VTP advertisement with a larger configuration revision number, it updates its VLAN configuration.

The VTP process begins with VLAN creation on a switch called a VTP server. VTP floods advertisements throughout the VTP domain every 5 minutes, or whenever there is a change in VLAN configuration. The VTP advertisement includes a configuration revision number, VLAN names and numbers, and information about which switches have ports assigned to each VLAN. By configuring the details on one or more VTP server and propagating the information through advertisements, all switches know the names and numbers of all VLANs

### 2.4.1: VTP Modes

To participate in a VTP management domain, each switch must be configured to operate in one of three modes. These modes are: server mode, client mode, and transparent mode.

#### 2.4.1.1: Server Mode

Server mode is the default mode. In this mode, VTP servers have full control over VLAN creation and modification for their domains. All VTP information is advertised to other switches in the domain, while all

received VTP information is synchronized with the other switches. Because it is the default mode, server mode can be used on any switch in a management domain, even if other server and client switches are in use. This mode provides some redundancy in the event of a server failure in the domain.

### 2.4.1.2: Client Mode

Client mode is a passive listening mode. Switches listens to VTP advertisements from other switches and modify their VLAN configurations accordingly. Thus the administrator is not allowed to create, change, or delete any VLANs. If other switches are in the management domain, a new switch should be configured for client mode operation. In this way, the switch will learn any existing VTP information from a server. If this switch will be used as a redundant server, it should start out in client mode to learn all VTP information from reliable sources. If the switch was initially configured for server mode instead, it might propagate incorrect information to the other domain switches. Once the switch has learned the current VTP information, it can be reconfigured for server mode.

### 2.4.1.3: Transparent Mode

Transparent mode does not allow the switch to participate in VTP negotiations. Thus, a switch does not advertise its own VLAN configuration, and a switch does not synchronize its VLAN database with received advertisements. VLANs can still be created, deleted, and renamed on the transparent switch. However, they will not be advertised to other neighboring switches. VTP advertisements received by a transparent switch will be forwarded on to other switches on trunk links.

### 2.4.2: VTP Pruning

A switch must forward broadcast frames out all available ports in the broadcast domain because broadcasts are destined everywhere there is a listener. Multicast frames, unless forwarded by more intelligent means, follow the same pattern. In addition, frames destined for an address that the switch has not yet learned or has forgotten must be forwarded out all ports in an attempt to find the destination. When forwarding frames out all ports in a broadcast domain or VLAN, trunk ports are included. By default, a trunk link transports traffic from all VLANs, unless specific VLANs are removed from the trunk with the clear trunk command. In a network with several switches, trunk links are enabled between switches and VTP is used to manage the propagation of VLAN information. This causes the trunk links between switches to carry traffic from all VLANs.

VTP pruning makes more efficient use of trunk bandwidth by reducing unnecessary flooded traffic. Broadcast and unknown unicast frames on a VLAN are forwarded over a trunk link only if the switch on the receiving end of the trunk has ports in that VLAN. In other words, VTP pruning allows switches to prevent broadcasts and unknown unicasts from flowing to switches that do not have any ports in that VLAN. VTP pruning occurs as an extension to VTP version 1. When a Catalyst switch has a port associated with a VLAN, the switch sends an advertisement to its neighbor switches that it has active ports on that VLAN. The neighbors keep this information, enabling them to decide if flooded traffic from a VLAN should use a trunk port or not.

By default, VTP pruning is disabled on IOS-based and CLI-based switches. On IOS-based switches, the vtp pruning command in the VLAN database configuration mode, the can be used to enable pruning while the set vtp pruning enable command can be used to enabled VTP pruning on CLI-based switches. .

### 2.4.3: VTP Configuration

Before VLANs can be configured, VTP must be configured. By default, every switch will operate in VTP server mode for the management domain NULL, with no password or secure mode. The following sections discuss the commands and considerations that should be used to configure a switch for VTP operation.

## 2.4.3.1: Configuring a VTP Management Domain

Before a switch is added into a network, the VTP management domain should be identified. If this switch is the first one on the network, the management domain will need to be created. Otherwise, the switch may have to join an existing management domain with other existing switches.

The following command can be used to assign a switch to a management domain on an IOS-based switch:

```
Switch# vlan database
Switch(vlan)# vtp domain
domain_name
```

To assign a switch to a management domain on a CLI-based switch, use the following command:

```
Switch(enable) set vtp [ domain
domain_name ]
```

## 2.4.3.2: Configuring the VTP Mode

Once you have assigned the switch to a VTP management domain, you need to select the VTP mode for the new switch. There are three VTP modes that can be selected: server mode, client mode and transparent mode. These VTP modes were discussed in Section 2.4.1.

On an IOS-based switch, the following commands can be used to configure the VTP mode:

```
Switch# vlan database
Switch(vlan)# vtp domain
domain_name
Switch(vlan)# vtp { server | client |
transparent }
Switch(vlan)# vtp password password
```

On a CLI-based switch, the following command can be used to configure the VTP mode:

```
Switch(enable) set vtp [ domain
domain_name ]
[ mode{ server | client | transparent }] [
password password ]
```

If the domain is operating in secure mode, a password can be included in the command line. The password can have 8 to 64 characters.

### 2.4.3.3: Configuring the VTP Version

Two versions of VTP, VTP version 1 and VTP version 2, are available for use in a management domain.

Although VTP version 1 is the default protocol on a Catalyst switch, Catalyst switches are capable of running both versions; however, the two versions are not interoperable within a management domain. Thus, the same VTP version must be configured on each switch in a domain. However, a switch running VTP version 2 may coexist with other version 1 switches, if its VTP version 2 is not enabled. This situation becomes important if you want to use version 2 in a domain. Then, only one server mode switch needs to have VTP version 2 enabled. The new version number is propagated to all other version 2-capable switches in the domain, causing them to enable version 2 for use. By default, VTP version 1 is enabled. Version 2 can be enabled or disabled using the v2 option. The two versions of VTP differ in the features they support. VTP version 2 offers the following additional features over version 1:

• In transparent mode VTP version 1 matches the VTP version and domain name before forwarding the information to other switches using VTP. On the other hand, VTP version 2 in transparent mode forwards the VTP messages without checking the version number.

• VTP version 2 performs consistency checks on the VTP and VLAN parameters entered from the CLI or by Simple Network Management Protocol (SNMP). This checking helps prevent errors in such things as VLAN names and numbers from being propagated to other switches in the domain. However, no consistency checks are performed on VTP messages that are received on trunk links or on configuration and database data that is read from NVRAM.

• VTP version 2 supports the use of Token Ring switching and Token Ring VLANs.

• VTP version 2 has Unrecognized Type-Length-Value (TLV) support, which means that VTP version 2 switches will propagate received configuration change messages out other trunk links, even if the switch supervisor is not able to parse or understand the message.

On an IOS-based switch, the VTP version number is configured using the following commands:

```
Switch# vlan database
Switch(vlan)# vtp v2-mode
```

On a CLI-based switch, the VTP version number is configured using the following command:

```
Switch(enable) set vtp v2 enable
```

# Topic 3: IP Addressing and Subnetting

## Section 3.1: IP Addressing

IP addressing and subnetting is probably the single most important topic you need to know for the CCNA exam. An IP address is a network layer (Layer 3) address that uniquely identifies a host, including network components and devices, on a TCP/IP network. An IP address is composed of 32 binary bits and consists of two parts: a network ID and a host ID.

• The Network ID identifies the TCP/IP hosts that are located on the same physical network. All hosts on the same physical network must be assigned the same network ID to communicate with each other. If routers connect your networks, a unique network ID is required for each wide area connection.

• The Host ID identifies the individual hosts within a network. The host ID must be unique to the network designated by the network ID.

The boundary between the network ID and the host ID of the IP address is defined by the subnet mask, which is another 32-bit field. There is a bit-for-bit alignment between the IP address and the subnet mask. The subnet mask contains a continuous field of 1s followed by a continuous field of 0s. The contiguous 1s stop at the boundary between the network ID and the host ID of the IP address. The network boundary can occur at any place after the eighth bit position from the left. Once the boundary between the network part and the host part of the IP address is known, all devices addressed in that network will have a common binary pattern in the network part that identifies the device as belonging to the specified network.
There are a number of formats for referencing an IP address. These include binary, dotted decimal notation and Classless Interdomain Routing (CIDR) Notation..

**3.1.1: Binary Format**
Binary is a numeral system that is 2 based, i.e., it uses only 0s and 1s, to denote a value. Because binary is 2 based, each successive bit is twice the value of the preceding bit, read from right to left. This is illustrated in Appendix
A. A 0 denotes that the bit does not carry a value and a 1 denotes that the bit does carry a value.
When binary value has more than one 1, as in 000001001 the decimal values for the 1s are added to produce the decimal value. In this example 000000001 is 1 and 000001000 is 8. Therefore the decimal value for 000001001 is 9 (8+1). The maximum binary value for an octet would contain all 1s, as in 111101111, and would have a decimal value 255 (128+64+32+16+8+4+2+1), as illustrated in Figure 3.1.

| **Binary Code** | **1** | **1** | **1** | **1** | **1** | **1** | **1** | **1** |
|---|---|---|---|---|---|---|---|---|
| **Decimal Value** | 128 | 64 | 32 | 16 | 8 | 4 | 2 | 1 |

**Figure 3.1: Binary Code 1111 1111**

The decimal value of the binary code is the sum of decimal value of each bit. Therefore the decimal value for a binary code of 111101111 is 128+64+32+16+8+4+2+1=255

**Note**: The corresponding decimal value of the binary code is calculated from
right to left and not left to right.

A 0 in the binary code indicates that the corresponding bit has no value. Figure 3.2 illustrates a byte with a binary code of 111001101 and the value of each of its eight bits.

| **Binary Code** | **1** | **1** | **1** | **0** | **1** | **1** | **0** | **1** |
|---|---|---|---|---|---|---|---|---|
| **Decimal Value** | 128 | 64 | 32 | 16 | 8 | 4 | 2 | 1 |

**Figure 3.2: Binary Code 1110 1101**

The decimal value for this binary code is 128+64+32+0+8+4+0+1=237

**Note**: Each bit in the binary code that is marked with a 0 has no value.
Therefore the corresponding decimal value of these bits are also 0.

## 3.1.2: Dotted Decimal Format

Both the IP address and its associated subnet mask contain 32 bits. However, the 32-bit IP address can be represented in other formats. The common formats include decimal (base 10) and hexadecimal (base 16) notation. The generally accepted format for representing

IP addresses and subnet masks the doted decimal notation in which the 32-bit field is divided into four groups of eight bits, also called a byte, that are translated to decimal value and separated by dots. Each group of eight bits is called an octet. Thus, an IP Address expressed as 11000000001010010000101000110001110 in binary format can be broken into its four octets: 110000000.101001000.101000110.011101110. These octets are converted to decimal value in Figure 3.3.

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **First Octet** | **Binary Code** | **1** | **1** | **0** | **0** | **0** | **0** | **0** | **0** |
| | **Decimal Value** | 128 | 64 | 32 | 16 | 8 | 4 | 2 | 1 |
| **Second Octet** | **Binary Code** | **1** | **0** | **1** | **0** | **1** | **0** | **0** | **0** |
| | **Decimal Value** | 128 | 64 | 32 | 16 | 8 | 4 | 2 | 1 |
| **Third Octet** | **Binary Code** | **1** | **0** | **1** | **0** | **0** | **1** | **1** | **0** |
| | **Decimal Value** | 128 | 64 | 32 | 16 | 8 | 4 | 2 | 1 |
| **Fourth Octet** | **Binary Code** | **0** | **1** | **1** | **1** | **1** | **1** | **1** | **0** |
| | **Decimal Value** | 128 | 64 | 32 | 16 | 8 | 4 | 2 | 1 |

**Figure 3.3: Binary Code 1100 0000.1010 1000.0111 1011**

The decimal value of the first octet is: 128+64+0+0+0+0+0+0 = 192
The decimal value of the second octet is: 128+0+32+0+8+0+0+0 = 168
The decimal value of the third octet is: 128+0+32+0+0+4+2+0 = 166
The decimal value of the fourth octet is: 0+64+32+16+8+4+2+0 = 126
In dotted decimal format this IP Address would be expressed as: 192.168.166.126

## 3.1.3: IP Address Classes

IP addresses are divided in to 'classes', based on the decimal value represented in the first octet. This class definition is referred to as the First Octet Rule. There are five classes of IP addresses: classes A, class B, class C, class D; and class E, but only class A, B and C addresses are used to identify devices connected to the Internet. Class D addresses are used for multicasting, and Class E addresses are reserved for experimental use. The subnet mask is related to the IP address class. Thus, once the IP address class is known, the default routing mask is also known. The IP address classes and their related subnet masks are:

• Class A addresses range from 0.0.0.0 through 126.255.255.255 and use a default subnet mask of 255.0.0.0. In Class A addresses, the first octet is used as for the network ID while the last three octets are used for the host ID. In other words, the first 8 bits of the subnet mask are all 1s, hence a subnet mask of 255.0.0.0. As a result, networks that use Class A addresses can theoretically support a maximum of 256 networks and 16,581,375 (255x255x255) hosts, however, the first and the last address cannot be used. The first address is the network address and the last address is the broadcast address. For example, a network with an IP address of 10.10.11.12 has a network ID of 10.0.0.0, the fist address, and a broadcast address of 10.255.255.255, the last address. Thus networks with a Class A IP address space can support a maximum of 254 networks (28-2) and 16,777,214 hosts (224-2). Consequently, Class A addresses are used for a few networks with a very large number of hosts on each network.

• Class B addresses range from 128.0.0.0 through 168.255.255.255 and 170.0.0.0 through 191.255.255.255. These addresses use a default subnet mask of 255.255.0.0. In Class B addresses, the first two bits are used as for the network ID while the last two bits are used for the host ID. As a result, networks that use Class B addresses can support a maximum of 65,534 networks (216-2) and 65,534 hosts. Consequently, Class B addresses are used for a reasonable number of medium sized networks.

**Note**: IP addresses with a first octet of 127, i.e. 127.0.0.0 through 127.255.255.255 do not fall in either the Class A address range or the Class B address range. IP addresses that have a first octet of 127 are reserved for diagnostics purposes.

• Class C addresses range from 192.0.0.1 through 223.225.225.225 and default subnet mask of 255.255.255.0. In Class C addresses, the first three bits are used as for the network ID while only the last bit is used for the host ID. As a result, networks that use Class C addresses can support a maximum of 16,777,214 networks and 254 hosts. Consequently, Class C addresses are used for a large number of networks with a relatively small number of hosts on each network.

• Class D addresses are in the range 224.0.0.0 through 239.255.255.255. These addresses are reserved for multicast transmissions.

• Class E addresses are in the range 240.0.0.0 through 254.255.255.255. These addresses are reserved for experimental use.

**Note**: InterNIC has reserved a number of IP address range, including 169.0.0.1 through 169.253.255.254, which has been reserved by for future use;
0.0.0.0, which was originally defined for use as a broadcast address; and
127.0.0.0, which is used as the loopback address.
128.0.0.0, 191.255.0.0, 192.0.0.0, and 223.255.255.0 also are reserved.

### 3.1.4: Classless Interdomain Routing (CIDR) Notation

Class-based IP addressing is fairly rigid. Thus, a small company with 50 hosts that wants to connect to the Internet would need a Class C address. However, a Class C address range supports 253 hosts; therefore 203 addresses would be wasted. Similarly, a company with 4,000 hosts would require a Class B address to connect to the Internet. A Class B address can support up to 65,023 hosts, resulting in 61,023 addresses being wasted. This problem can be overcome by extending the default subnet mask by adding more continuous 1s to it. The result is that the network can support less hosts. Thus, the company that has 4,000 hosts would use a Class B address with a subnet mask of 255.255.240.0. This is achieved by extending the subnet mask by 4 bits so that the first 20 bits represent the network ID and 12 bits only represent the host ID. Thus the address range now supports only 4,094 hosts, representing a loss of only 94 addresses. We can calculate the number of hosts supported by using the formula: 2n-2 where n is the number of bits used for the host ID. We need to subtract 2 addresses: the network address and the broadcast address. In this example, 12 bits are used for the host ID. Thus using the formula we can see that this subnet mask supports 4,094 hosts (212-2).

We can calculate the number of subnets supported by a subnet mask by using the same formula: 2n-2. However, this time n is the number of bits used for the network ID. We again need to subtract 2 addresses: the network address and the broadcast address. Thus, in the example 255.255.240.0, 20 bits represent the network ID therefore this subnet mask supports 1048,574 subnets (220-2).

This solves the problem of IP address allocation on the internet but presents a problem for routing tables, as the routing table cannot determine the subnet mask on the basses of the IP address class. Hence a different format of representing the IP address and its subnet mask is required. This format is called the Classless Interdomain Routing (CIDR) notation, or prefix notation. CIDR is in essence an adaptation of the Dotted Decimal Format and represents the subnet mask as a number of bits used for the network ID. This number of bits is indicated after the IP address by the number that follows the slash (/) symbol. For example, the CIDR notation IP address 140.12.2.128/20 indicates that the first 20 bits of the IP address is used for the subnet mask, i.e., the first 20 bits are all 1s. Thus, the subnet mask expressed in binary format is 111101111.111101111.111100000.000000000, being represented in dotted decimal format as 255.255.240.0. In addition, the routing protocols must send the mask with the routing update.

### 3.1.5: Variable-Length Subnet Masks

CIDR is used within the Internet. Its counterpart within an organization is the Variable-length subnet mask (VLSM). Like CIDR, VLSM allows you to allocate the required host bits on a granular basis. In other words, it allows you to provide only the bits required to address the number of hosts on a particular subnetwork. Like CIDR, VLSM requires a routing protocol that supports the sending of the subnet mask in its updates. The routing protocols that support VLSM are: RIPv2; OSPF; IS-IS; EIGRP; and BGP-4. The routing protocols do not support VLSM are: RIPv1; IGRP; and EGP.

## Section 3.2: Subneting

The process of extending the default subnet mask creates a counting range in the octet that the subnet was extended into, which can be used to represent subnetworks. This allows a single Class A, B, or C network to be subdivided into many smaller groups with each group, or subdivision treated as if it were a network itself. Thus, when we extend the default Class B subnet mask to 255.255.240.0, we do so by extending the subnet mask by 4 bits into the third octet. The number of bits that the subnet mask is extended by represents a counting range for counting the number of subnetworks that new subnet mask can support, using the 2n-2

formula. Thus, the subnet mask 255.255.240.0 subnet mask can support 14 subnets (24-2). In other words, the 65,534 hosts supported by the default subnet mask can now be divided among 14 subnetworks. The number of IP addresses supported by each subnet is called an address range. To calculate the range of addresses for each subnet, we would take the decimal value for the last bit used for the subnet mask as the starting point for the first address in our subnetwork, and then increment that number for each subsequent subnet. In this octet the bit range would be 1111 0000. The last bit in the subnet mask would thus have a decimal value of 16 (0001 0000). Therefore the first IP address in the first subnet address range would be 140.12.16.1. The address ranges for the 14 subnets would be:

- 140.12.16.1 to 140.12.31.254
- 140.12.32.1 to 140.12.47.254
- 140.12.48.1 to 140.12.63.254
- 140.12.64.1 to 140.12.79.254
- 140.12.80.1 to 140.12.95.254
- 140.12.96.1 to 140.12.111.254
- 140.12.112.1 to 140.12.127.254
- 140.12.128.1 to 140.12.143.254
- 140.12.144.1 to 140.12.159.254
- 140.12.160.1 to 140.12.175.254
- 140.12.176.1 to 140.12.191.254
- 140.12.192.1 to 140.12.207.254
- 140.12.208.1 to 140.12.223.254
- 140.12.224.1 to 140.12.239.254

**Note**: The IP address range for each subnet begins with a 1, as in 140.12.16.1
or 140.12.32.1 and not 140.12.16.0 or 140.12.32.0 as this would be the first
address in the subnetwork, and would therefore be the network address.
Similarly, the last address in the range ends in 254 and not 255 as the last
address would be the broadcast address.

## Section 3.3: Summarization

Summarization allows the representation of a series of networks in a single summary address. At the top of the hierarchical design, the subnets in the routing table are more generalized. The subnet masks are shorter because they have aggregated the subnets lower in the network hierarchy. These summarized networks are often referred to as supernets, particularly when seen in the Internet as an aggregation of class addresses. They are also known as aggregated routes. The summarization of multiple subnets within a few subnets has several advantages. These include: reducing the size of the routing table; simplifying the recalculation of the network as the routing tables are smaller; network overhead scalability; and hiding network changes.

### 3.3.1: Automatic Summarization

All routing protocols employ some a type of summarization. RIP and IGRP automatically summarize at the NIC or natural class boundary as the subnet mask is not sent in the routing updates. When a routing update is received, the router checks if it has an interface in the same class network. If it has one, it applies the mask configured on the interface to the incoming routing update. With no interface configured in the same NIC network, there is insufficient information and the routing protocol uses the first octet rule to determine the

default subnet mask for the routing update.

### 3.3.2: Manual Summarization

Both EIGRP and Open Shortest Path First (OSPF) send the subnet mask along with the routing update. This feature allows the use of VLSM and summarization. When the routing update is received, it assigns the subnet mask to the particular subnet. When the routing process performs a lookup, it searches the entire database and acts on the longest match, which is important because it allows for the granularity of the hierarchical design, summarization, and discontiguous networks.

A discontiguous network is a network in which a different NIC number separates two instances of the same NIC number. This can happen either through intentional design or through a break in the network topology. If the network is not using a routing protocol that supports VLSM, this will create a routing problem because the router will not know where to send the traffic. Without a subnet mask, a routing protocol that supports VLSM resolves the address down to the NIC number, which appears as if there is a duplicate address. This will incorrectly lead to the appearance of intermittent connectivity symptoms.

If there are discontiguous networks in the organization, it is important that summarization is turned off or not configured. Summarization may not provide enough information to the routing table on the other side of the intervening NIC number to be capable of appropriately routing to the destination subnets, especially with EIGRP, which automatically summarizes at the NIC boundary. In OSPF and EIGRP, manual configuration is required for any sophistication in the network design. However, because EIGRP can perform summarization at the interface level, it is possible to select interfaces that do not feed discontiguous networks for summarization.

If summarization is not possible, you can either turn summarization off and understand the scaling limitations that have now been set on the network, or you can readdress the network.

## Section 3.4: Determining the Network ID using the Logical AND Operation

When an IP address is assigned to an interface, it is configured with the subnet mask. Although represented in a dotted decimal format, the router converts the IP address and the subnet mask into binary and performs a logical AND operation to find the network portion of the address, i.e., the network ID. To perform a logical AND, the IP address is written out in binary, with the subnet or Internet mask written beneath it in binary. Each binary digit of the address is then ANDed with the corresponding binary digit of the mask. The AND operation has two rules: 1 AND 1 is 1; and 0 AND 1 or 0 remains 0. Essentially, the logical AND operation removes the host ID from the IP address, as illustrated in Figure 3.4.

| | |
|---|---|
| IP address: | 140.12.26.128 |
| IP subnet mask: | 255.255.240.0 |
| IP address in binary: | **10001100.00001100.00011010.10000000** |
| IP subnet mask in binary: | **11111111.11111111.11110000.00000000** |
| The result of the logical **AND**in binary: | **10001100.00001100.00010000.00000000** |
| The result of the logical **AND**in dotted decimal format: | 140.12.16.0 |

**Figure 3.4: The Logical AND Operation**

In the above example, the network to which the host 140.12.26.128 belongs has the network ID of 140.12.16.0. Once the network ID is determined, the router can perform a search on the routing table to see whether it can route to the remote network. Therefore, the correct mask is essential to ensure that traffic can

be directed through the overall network.

## Section 3.5: IP Version 6

IPv4 has a number of disadvantages. The two most important disadvantages is the limited address space, with public IPv4 addresses are becoming scarce; and the lack of built-in security. Instead, security for IPv4 is provided by the use of IPSec. However, IPSec is optional for IPv4 implementations. Because an application cannot rely on IPSec being present to secure traffic, an application might resort to other security standards or a proprietary security scheme.

These and other issues prompted the Internet Engineering Task Force (IETF) to begin the development of IPv6 that would replace IPv4, solve the problems of IPv4, and be extensible to solve additional problems in the future. The IPv6 specification is defined in RFC 2460. Other RFCs that describe IPv6 specifications are 2373, 2374, 2461, 2462, and 2463.

IPv6 offers a number of advantages over IPv4.

• The IPv6 address field is 128 bits long - a significant increase from 32-bits IPv4 address field, and thus provides a larger address space.

• IPv6 has built-in support for IPSec and thus offers better security.

• IPv6 provides a new header format that is streamlined to minimize overhead and provide more efficient processing while crossing intermediate routers.

• All the option fields and any other fields in the header that are not required for routing are placed after the IPv6 header.

• The IPv6 header also added more Quality of Service (QoS) support by adding Flow Label fields that provide special handling for a series of packets that travel between a source and destination.

• IPv6 also provides Neighbor Discovery (ND), which is a set of process and messages that are used in an IPv6 environment to identify relationships between neighboring nodes. This allows hosts to discover routers on the same segment, addresses, and address prefixes. With ND, hosts can also resolve neighboring nodes and determine when the MAC address of a neighbor changes. This is similar to ARP in IPv4.

• ND provides the process for address autoconfiguration, which provides for the dynamic assignment of IPv6 addresses and is referred to as stateless address configuration. In the absence of a stateful address configuration server, such as a DHCP version 6 (DHCPv6) protocol server, ND provides a complex process that allows each interface to use router advertisement messages to define an IPv6 address, and then subsequently ensure the uniqueness of the selected address. However, the standards for DHCPv6 and IPv6 stateful addressing are still under development.

• The new routing structure provides a hierarchical addressing and routing structure that includes a global addressing scheme. Global addresses are the equivalent of public IPv4 addresses and are accessible over the Internet.

• The global addressing scheme defines new ways to summarize global addresses to facilitate smaller routing tables on the Internet backbone, thus improving the efficiency and performance on the Internet.

### 3.5.1 IPv6 Address Representation

The IPv6 addressing architecture is defined in RFC 2373. IPv6 addresses are 128-bits long with the first 64 bits defining the network address and the last 64 bits defining the host address. An IPv6 address consists of eight 16-bit sections and is represented in hexadecimal format. Each 16-bit section is separated by a colon. An example of a full IPv6 address is FE36:0000:0000:36F0:0000:0000:004B:04B0. All leading 0s do not need to be represented while all 0 16-bit sections can be compressed to 0. Multiple 16-bit sections of 0s can be represented with a :: symbol, which can appear only once in the number. Thus FE36:0000:0000:36F0:0000:0000:004B:04B0 can be shortened to FE36:0:0:36F0::4B:4B0 or FE36::36F0:0:0:4B:4B0

In a mixed IPv4 and IPv6 environment, addresses can be represented by six hexadecimal 16-bit sections that are concatenated with the dotted-decimal format.

### 3.5.2 Allocated IPv6 Addresses

The leading bits of an IPv6 address can define the address type. These leading bits are of variable length and are called the format prefix (FP). Table 3.2 shows some allocations of some prefixes.

TABLE 3.1: IPv6 Prefix Allocations

| Prefix | Allocation |
|---|---|
| 00 | Unspecified, looback, IPv4-compatible |
| 2 or 3 | Aggregatable global unicast address |
| FE8 | Link-local unicast addresses |
| FEC | Site-local unicast addresses |
| FF | Multicast addresses |

**Note**: IPv6 supports unicast addressing, which identifies a single IP host;
anycast addressing, which identifies a set of IP hosts and delivers the
transmitted packet to nearest of these hosts; and multicast addressing,
which identities a set of hosts who can choose to receive the packet or not.
IPv6 does not support broadcast addressing and has no broadcast address.
IPv6 uses "all-nodes" multicast instead.

To convert a hexadecimal number to a decimal number, you must multiply the decimal value of each digit by 16 to the power n-1 where n is the position of the digit from right to left, and add the resultant decimal values. Thus, to convert 3E8h to decimal, we would take the third right most digit, 3h, and convert it to its

decimal equivalent, which is 3. We would then multiply 3 by 162 which would give us 768. Next, we would take the second right most digit, Eh, and convert it to its decimal equivalent, which is 14. We would then multiply 14 by 161 which would give us 224. Next, we would take the right most digit, 8h, and convert it to its decimal equivalent, which is 8. We would then multiply 8 by 160 which would give us 8.
Finally, we would add the three resultant values: 768, 224 and 8, which would give us 1000.

# Topic 4: Routing

Routing is a relay system by which packets are forwarded from one device to another. Each device in the network as well as the network itself has a logical address so it can be identified and reached individually or as part of a larger group of devices. For a router to act as an effective relay device, it must be able to understand the logical topology of the network and to communicate with its neighboring devices. The router understands several different logical addressing schemes and regularly exchanges topology information with other devices in the network. The mechanism of learning and maintaining awareness of the network topology is considered to be the routing function while the movement of traffic through the router is a separate function and is considered to be the switching function. Routing devices must perform both a routing and a switching function to be an effective relay device. A router receiving a packet from a host, the router will need to make a routing decision based on the protocol in use; the existence of the destination network address in its routing table; and the interface that is connected to the destination network. After the decision has been made the router will switch the packet to the appropriate interface on the router to forward it out. If the destination logical network does not exist in the routing table, routing devices will discard the packet and to generate an Internet Control Message Protocol (ICMP) message to notify the sender of the event.

## Section 4.1: Routing Tables

A routing table is a database repository that holds the router's routing information that represents each possible logical destination network that is known to the router. The entries for major networks are listed in ascending order and, most commonly, within each major network the subnetworks are listed in descending order. If the routing table entry points to an IP address, the router will perform a recursive lookup on that next-hop address until the router finds an interface to use. The router will switch the packet to the outbound interfaces buffer. The router will then determine the Layer 2 address that maps to the Layer 3 address. The packet will then be encapsulated in a Layer 2 frame appropriate for the type of encapsulation used by the outbound interface. The outbound interface will then place the packet on the medium and forward it to the next hop. The packet will continue this process until it reaches its destination.
There are two ways in which a routing table can be populated: a route can be entered manually, this is called static routing, or a router can dynamically learning a route. Once a router learns a route, it is added to its route table.

### 4.1.1: Static Routing

A statically defined route is a route is manually entered into the router. The purpose of this is to add routes to a router's routing table. Thus, static routing consists of individual configuration commands that define a route to a router. A router can forward packets only to subnets in its routing table. The router always knows about directly connected routes. By adding static routes, a router can be told how to forward packets to subnets that are not attached to it.

A static route can be entered into the router in global configuration mode with the following command:

```
ip route destination_ip_address
subnet_mask
{ ip-address | interface } [
distance ]
```

In the ip route command, the destination_ip_address and subnet_mask is the IP address and subnet mask for the destination host. The ip-address parameter is the IP address of the next hop that can be used to reach the destination and interface is the router interface to use. The optional distance parameter specifies the administrative distance.

The advantages to using static routes in an internetwork are the administrator has total control of what is in the routers routing table and there is no network overhead for a routing protocol. The disadvantage of using only static routes is they do not scale well.

### 4.1.2: Dynamic Routing

Dynamic routing is a process in which a routing protocol will find the best path in a network and maintain that route. Once a route fails, the routing protocol will automatically find an alternate route to the destination. Routing protocols are easier to use than static routes. However, a routing protocol will consume more CPU cycles and network bandwidth than a static route.

### 4.1.3: Routing Updates

Routing updates can occur using the distance vector approach or the link-state approach.

• Distance-vector protocols use a routine, periodic announcement that contains the entire contents of the routing table. These announcements are usually broadcasts and are propagated only to directlyconnected, next-hop, devices. This allows the router to view the network from the neighbor's perspective and facilitates the addition of the router's metric to the 'distance' already stated by the neighboring router. However, this approach uses considerable bandwidth at regular intervals on each link even if no topology changes have occurred.

• Link-state protocols use a triggered-update type of announcement that is generated only when there is a topology change within the network. The link-state announcements only contain information about the link that changed and are propagated or flooded to all devices in the network. This approach saves bandwidth on each link because the announcements contain less information and is only sent when there is a topology change. In some link-state protocols, a periodic announcement is required to ensure that the topology database is synchronized among all routing devices.

### 4.1.4: Verifying Routing Tables

You can use the show ip route privileged exec command to view an IP routing table. If the information that is displayed is not correct, you can force an update from the neighboring devices with the clear ip route command. An optional keyword specifying an ip_address and subnet_mask, or the * (wildcard) character, can be used to further identify the routes to be refreshed.

## Section 4.2: Routing Protocols

There are two types of dynamic routing protocols: Interior Gateway Protocols (IGP) and External Gateway Protocols (EGP). IGPs are used to exchange routing information within an autonomous system

(AS), which is a collection of routing domains under the same administrative control the same routing domain. An EGP, on the other hand, is used to exchange routing information between different ASs. IGPs can be broken into two classes: distance-vector and link-state, and can also be broken into two categories: classful routing protocols and classless routing protocols..

### 4.2.1: Distance-Vector Routing

Distance-vector routing is consists of two parts: distance and vector. Distance is the measure of how far it is to reach the destination and vector is the direction the packet must travel to reach that destination. The latter is determined by the next hop of the path. Distance-vector routing protocols will learn routes from its neighbors. This is called routing by rumor. Examples of distance-vector routing protocols are: Routing Information Protocol (RIP), Interior Gateway Routing Protocol (IGRP), and Enhanced Interior Gateway Routing Protocol (EIGRP).

#### 4.2.1.1: Route Poisoning

Route poisoning is a feature that distance vector protocols use to reduce the chance of routing loops. Route poisoning begins when a router notices that a connected route is no longer valid. The router then advertises that route out all its interfaces and with a very large metric so that other routers consider the metric infinite and the route invalid. However, route poisoning does not solve the counting-to-infinity problem.

#### 4.2.1.2: Split Horizon

As mentioned earlier, route poisoning does not solve the counting-to-infinity problem. Counting-to-infinity can occur when one router has a valid metric that points to an address that is reachable through an intermediate router while the intermediate router has an infinite-distance route to the same address (see Figure 4.1). If routing table updates are sent by both routers at the same in time, the intermediate router will advertise that the route to the destination address is an infinite-distance route while the other router will advertise that the route has a valid metric. Because the two routers use the same update interval between updates, this process repeats itself with the next routing update, with the difference that the valid metric will be incremented by 1 each time until an infinite metric is reached, hence this phenomenon is called counting to infinity.

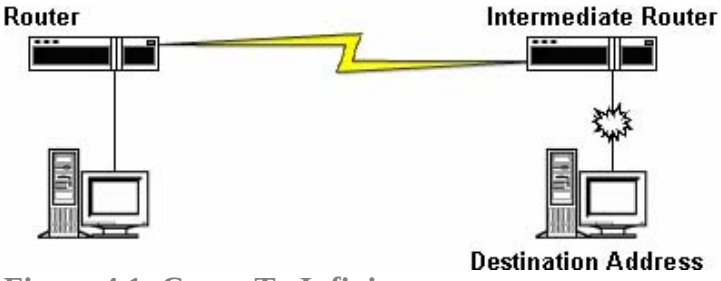

Figure 4.1: Count To Infinity

Split horizon solves the counting-to-infinity problem by preventing a router from sending routing updates out the same interface on which it learnt the route. Thus, in Figure 4.1, the router would have learnt the route to the destination address across the link from the intermediate router. With split horizon, that router cannot then send advertisements about the route to the destination address out across the same link. Therefore the intermediate router will not receive the valid metric from the route to the destination address from the other router ad the count to infinity problem will not occur, solving the count-to-infinity problem on a single link.

#### 4.2.1.3: Split Horizon with Poison Reverse

Split horizon with poison reverse, or simply poison reverse combines the two features. When a route fails

the router uses route poisoning, i.e., the router advertises an infinite-metric route about that subnet out all interfaces, including interfaces previously prevented by split horizon. This ensures that all routers know for sure that the route has failed, while split horizon prevents counting to infinity.

### 4.2.1.4: Hold-Down Timer

Split horizon solves the counting-to-infinity problem over a single link but the counting to infinity problem can also occur in networks with multiple or redundant paths because there are more than one path to a router. In such networks, the hold-down timer feature prevents the counting-to-infinity problem.
With the Hold-down timer feature, a router ignores any information about an alternative route to a destination address for a time equal to the hold-down timer once it has learnt that a route to the destination address has failed.

### 4.2.1.5: Triggered Updates

Distance vector protocols typically send updates based on a regular update interval. However, most looping problems occur when a route fails. Therefore, some distance vector protocols send triggered updates as soon as a route fails. This causes the information about the route whose status has changed to be forwarded more quickly and also starts the hold-down timers more quickly on the neighboring routers.

### 4.2.2: Link-State Routing

Link-state routing differs from distance-vector routing in that each router knows the exact topology of the network. This reduces the number of bad routing decisions that can be made because every router in the process has an identical view of the network. Each router in the network will report on its state, the directly connected links, and the state of each link. The router will then propagate this information to all routers in the network. Each router that receives this information will take a snapshot of the information. This ensures all routers in the process have the same view of the network, allowing each router to make its own routing decisions based upon the same information.

In addition, link-state routing protocols generate routing updates only when there is a change in the network topology. When a link, i.e., a point on a route, changes state, a link-state advertisement (LSA) concerning that link is created by the device that detected the change and propagated to all neighboring devices using a multicast address. Each routing device takes a copy of the LSA, updates its topological database and forwards the LSA to all neighboring devices. An LSA is generated for each link on a router. Each LSA will include an identifier for the link, the state of the link, and a metric for the link. With the use of LSAs, linkstate protocols reduces routing bandwidth usage.

Examples of link-state routing protocols are: Open Shortest Path First (OSPF) and Integrated Intermediate System to Intermediate System (IS-IS). Another protocol, Enhanced Interior Gateway Routing Protocol (EIGRP) is considered a hybrid protocol because it contains traits of both distance-vector and link-state routing protocols. Most link-state routing protocols require a hierarchical design, especially to support proper address summarization. The hierarchical approach, such as creating multiple logical areas for OSPF, reduces the need to flood an LSA to all devices in the routing domain. The use of areas restricts the flooding to the logical boundary of the area rather than to all devices in the OSPF domain. In other words, a change in one area should only cause routing table recalculation in that area, not in the entire domain.
OSPF is discussed in more detail in Section 5 and EIGRP is discussed in more detail in Section 6. IS-IS is not covered in the CCNA 640-822 exam and is thus not discussed in this Study Guide.

### 4.2.3: Classful Routing

Classful routing is used in routing packets based upon the class of IP address. IP addresses are divided into five classes: Class A, Class B, Class C, Class D, and Class E. Class A, Class B and Class C are used to private and public network addressing; Class D is used for multicast broadcasting; and Class E is reserved by the Internet Assigned Numbers Authority (IANA) for future use. IP Address classes were discussed in detail in Section 3.1.3.

Classful routing is a consequence of the fact that routing masks are not advertised in the periodic, routine, routing advertisements generated by distance vector routing protocols. In a classful environment, the receiving device must know the routing mask associated with any advertised subnets or those subnets cannot be advertised to it. There are two ways this information can be gained:

• Share the same routing mask as the advertising device

• If the routing mask does not match, this device must summarize the received route a classful boundary and send the default routing mask in its own advertisements.

Classful routing protocols, such as Routing Information Protocol version 1 (RIPv1) and Interior Gateway Routing Protocol (IGRP), exchange routes to subnetworks within the same network if network administrator configured all of the subntworks in the major network have the same routing mask. When routes are exchanged with foreign networks, subnetwork information from this network cannot be included because the routing mask of the other network is not known. As a result, the subnetwork information from this network must be summarized to a classful boundary using a default routing mask prior to inclusion in the routing update. The creation of a classful summary route at major network boundaries is handled automatically by classful routing protocols. However, summarization at other points within the major network address is not allowed by classful routing protocols.

#### 4.2.4: Classless Routing

One of the most serious limitations in a classful network environment is that the routing mask is not exchanged during the routing update process. This requires the same routing mask be used on all subnetworks. The classless approach advertises the routing mask for each route and therefore a more precise lookup can be performed in the routing table. Classless routing, which is also known as Classless Interdomain Routing (CIDR), is thus not dependent on IP address classes but, instead, allows a variablelength subnet mask (VLSM), which extends IP addressing beyond the limitations of using fixed-length subnet masks (FLSM),to be sent in the routing update with the route. This allows you to conserve IP addresses, extending the use of IP addresses. Classless routing protocols also addressed the need to summarize to a classful network with a default routing mask at major network boundaries. In the classless environment, the summarization process is manually controlled and can be invoked at any point within the network. VLSM was discussed in Section 3.1.5.

The routing protocols that support classless routing protocols are: Routing Information Protocol version 2 (RIPv2) ; Enhanced Interior Gateway Routing Protocol (EIGRP); Open Shortest Path First (OSPF) ; and Integrated Intermediate System to Intermediate System (IS-IS).

## Section 4.3: Basic Switching Functions

In order to forward a packet that has arrived at a router interface, the router must perform a switching

function. This switching function has four steps:

• A packet transiting the router will be accepted into the router if the frame header contains the MAC address of one of the router's NIC cards. If properly addressed, the frame and its content will be buffered occurs in memory pending further processing.

• The switching process checks the destination logical network portion of the packet header against the network/subnetwork entries in the routing table. If the search is successful, the switching process associates the destination network with a next-hop logical device and an outbound interface.

• Once the next-hop logical device address is known, a lookup is performed to locate a physical address for the next device in the relay chain. The lookup is performed in an Address Resolution Protocol (ARP) table for LAN interfaces or a map table for WAN interfaces.

• Once the physical address of the next-hop device is known, the frame header is overwritten, and the frame is then moved to the outbound interface for transmission onto the media. As the frame is placed on the media, the outbound interface adds the CRC character and ending delimiters to the frame. These characters will need to be validated at the arriving interface on the next-hop relay device.

## Section 4.4: Convergence

In a routed network, the routing process in each router must maintain a loop-free, single path to each possible destination logical network. When all of the routing tables are synchronized and each contains a usable route to each destination network, the network is described as being 'converged'. Convergence is the time it takes for all routers to agree on the network topology after a change in the network. Convergence efforts are different within different routing protocols. There are at least two different detection methods used by all routing protocols. The first method is used by the Physical Layer (Layer 1) and the Data Link Layer (Layer2) protocols. When the network interface on the router does not receive three consecutive keepalives, the link will be considered down. The second method is that when the routing protocol at the Network/Transport Layer (Layer 3) fails to receive three consecutive Hello messages, the link will be considered down.

Routing protocols have timers that are used to stop network loops from occurring on a network when a link failure has been detected. Hold-down timers are used to give the network stability while new route calculations are being performed. They also allow all the routers a chance to learn about the failed route to avoid routing loops and counting to infinity problems. Since a network cannot converge during this holddown period, this can cause a delay in the routing process of the network. Because of this slow convergence time, link-state routing protocols do not use hold-down timers.

### 4.4.1: Distance-Vector Routing Convergence

#### 4.4.1.1: RIP and IGRP Convergence

Convergence time is one of the problems associated with distance-vector protocols, such as RIPv1 and IGRP. When a router detects a link failure between itself and a neighbor, it sends a flash update with a poisoned route to it other neighbors. These neighbors in turn create a new flash update and send it to all of its neighbors, and so on. The router that detected the link failure purges the entry for the failed link and removes all routes associated with that link from the routing table. The router then sends a query to its neighbors for the routs that have been removed. If a neighbor responds with a route, it is immediately

installed in the routing table. The router does not go into hold-down because the entry was already purged. However, its neighbors are in hold-down for the failed route, thus ignoring periodic advertisement for that route. As the other routers come out of hold-down, the new route announced by the router that detected the failed link will cause their routing table entries to be updated.

### 4.4.1.2: EIGRP Convergence

Enhanced IGRP (EIGRP) convergence differs slightly. If a router detects a link failure between itself and a neighbor, it checks the network topology table for a feasible alternate route. If it does not find a qualifying alternate route, it enters in an active convergence state and sends a Query out all interfaces for other routes to the failed link. If a neighbor replies to the Query with a route to the failed link, the router accepts the new path and metric information, places it in the topology table, and creates an entry for the routing table. It then sends an update about the new route out all interfaces. All neighbors acknowledge the update and send updates of their own back to the sender. These bi-directional updates ensure the routing tables are synchronized and validate the neighbor's awareness of the new topology. Convergence time in this event is the total of detection time, plus Query and Reply times and Update times.

### 4.4.2: Link-State Convergence

The convergence cycle used in Link-State Routing Protocols, such as OSFP and IS-IS, differs from that of the distance-vector protocols. When a router detects a link failure between itself and a neighbor, it tries to perform a Designated Router (DR) election process on the LAN interface, but fails to reach any neighbors. It then deletes the route from the routing table, builds a link-state advertisement (LSA) for OSFP or a link-state PDU (LSP) for IS-IS, and sends it out all other interfaces. Upon receipt of the LSA, the other neighbors that are up copy the advertisement and forward the LSA packet out all interfaces other than the one upon which it arrived. All routers, including the router that detected the failure, wait five seconds after receiving the LSA and run the shortest path first (SPF) algorithm. There after the router that detected the failure adds the new route to the routing table, while its neighbors update the metric in their routing table. After approximately 30 seconds, the failed router sends an LSA after aging out the topology entry from router that detected the failure. After five seconds, all routers run the SPF algorithm again and update their routing tables to the path to the failed link. Convergence time is the total of detection time, plus LSA flooding time, plus the five seconds wait before the second SPF algorithm is run.

## Section 4.5: Testing and Troubleshooting Routes

There are two tools that can be used for testing and troubleshooting routes or reachability. These are: ping and traceroute.

### 4.5.1: The ping Command

The ping command, which is included as a part of the TCP/IP protocol suite, is supported at the user and privileged exec modes. In user mode, you must specify an IP address or a host name, if the host name can be resolved to an IP address, with the ping command. The ping command tests the round-trip path to and from a target. In privileged mode, you must enter a protocol, a target IP address, a repeat count, datagram size, and a timeout in seconds.

Cisco IOS makes ping available for a number of protocols including IPX and AppleTalk. Cisco introduced ping for IPX in IOS version 8.2. This is, however, a Cisco proprietary tool. Therefore non-Cisco devices such as Novell servers do not respond to it. If you want the Cisco router to generate Novell-compliant pings, you must use the global configuration command ipx ping-default novell. Ping for AppleTalk sends

AppleTalk Echo Protocol (AEP) packets to the destination node and waits for replies. Generally, the syntax for the ping command is:

```
ping -s ip_address [ packet_size] [ packet_count]
```

**Table 4.1: Parameters for the ping Command**

| Parameter | Purpose |
|---|---|
| `-s` | Causes **ping**to send one datagram per second, printing one line of output for every response received. The **ping** command does not return any output when no response is received. |
| *ip_address* | The IP address or IP alias of the host. |
| *packet_size* | This optional parameter represents the number of bytes in a packet, from **1**to **2000**bytes, with a default of **56**bytes. The actual packet size is eight bytes larger because the switch adds header information. |
| *packet_count* | This optional parameter represents the number of packets to send. |

## 4.5.2: The traceroute Command

The traceroute command was introduced with the release 10.0 of Cisco IOS and can be used to find the route between IP devices. The traceroute command can be executed in user and privileged exec modes, but in privileged exec mode, you can use the extended traceroute, which is more flexible and informative. Initially, traceroute was available only for the IP protocol but since release 12.0 of Cisco IOS, traceroute is also available for IPX. This command can be very useful in troubleshooting by determining where along a particular network path a particular problem might be as the traceroute command displays a hop-by-hop path through an IP network from the switch to a specific destination host. The syntax for the traceroute command is:

```
traceroute [ -n ] [- w wait_time ] [ -i initial_ttl ] [
-m max_ttl ]
[ -p dest_port ] [ -q nqueries ] [ -t tos ]
ip_address [ data_size ]
```

**Table 4.2: Parameters for the traceroute Command**

| Parameter | Description |
|---|---|
| `-n` | Prevents traceroute from performing a DNS lookup for each hop on the path. Only numerical IP addresses are printed. |
| `-w wait_time` | Specifies the amount of time that **traceroute** will wait for an ICMP response message. The allowed range for wait time is **1**to **300** seconds; the default is **5**. |

| | |
|---|---|
| **-i initial_ttl** | Causes traceroute to send ICMP datagrams with a TTL value equal to **initial_ttl**instead of the default TTL of 1. This causes **traceroute**to skip processing for hosts that are less than **initial_ttl**hops away. |
| **-m max_ttl** | Specifies the maximum TTL value for outgoing ICMP datagrams. The allowed range is **1**to **255**; the default value is **30**. |
| **-p dest_port** | Specifies the base UDP destination port number used in traceroute datagrams. This value is incremented each time a datagram is sent. The allowed range is **1**to **65535**; the default base port is **33434**. |
| **-q nqueries** | Specifies the number of datagrams to send for each TTL value. The allowed range is **1**to **1000**; the default is **3**. |
| **-t tos** | Specifies the TOS to be set in the IP header of the outgoing datagrams. The allowed range is **0**to **255**; the default is **0**. |
| ***ip_address*** | IP alias or IP address in dot notation of the destination host. |
| ***Data_size*** | Number of bytes, in addition to the default of 40 bytes, of the outgoing datagrams. The allowed range is **0**to **1420**; the default is **0**. |

# Topic 5: Link-State Protocols

Like distance vector protocols, link-state protocols use routing tables that are populated with the currentlybest routes. Link-state protocols, however, differ from Distance vector protocols in the methods they use to build their routing tables. The biggest difference between the two is that distance vector protocols advertise little information.

Unlike distance vector protocols, link-state protocols do not receive metrics in the routing table updates. Instead they must calculate the metric form the topology information learned by a router, which includes a cost associated with each link in the network. A router totals the cost associated with each link in each route to find the metric associated with that route. Link-state protocols use the Shortest Path First (SPF) algorithm, which is also called the Dijkstra SPF algorithm, to calculate route metrics.

When a new router that is configured with a link-state protocol is booted for the first time, it does not start broadcasting topology information out every interface. Instead, the router uses the Hello protocol to send and receive a small Hello packet to discover neighbors, i.e., other routers that use the same link-state protocol and share a common subnet. It has a source address of the router and a multicast destination address set to AllSPFRouters (224.0.0.5). All routers running OSPF or the SPF algorithm listen for the hello packet and send their own hello packets periodically.

Once a router identifies a neighbor, the two routers exchange routing information, which is called the topology database, and then run the SPF algorithm to calculate new routes. When their

**Topology and Routing Databases**

The topology database, which sometimes referred to as the link-state database, is the router's view of the network within the area. It includes every OSPF router within the area and all the connected networks. This database is a routing table for which no path decisions have been made. The topology database is updated by

topology databases are synchronized, the neighbors are said to be fully adjacent. The Hello protocol continues to transmit the Hello packets periodically. The transmitting router and its networks reside in the topology database for as long as the other routers receive the Hello protocol. This provides another mechanism for determining that a router has gone down, i.e., when the neighbor no longer sends Hello packets.

link-state advertisements (LSAs). Each router within the area has exactly the same topology database. The synchronization of the topology maps is ensured by the use of sequence numbers in the LSA headers.

A routing database is constructed from the topology map. This database is unique to each router, which creates a routing database by running the shortest path first (SPF) algorithm to determine the best path to each network and creates an SPF tree on which it places itself at the top. If there are equal metrics for a remote network, OSPF includes all the paths and load balances the routed data traffic among them.

The routing updates sent by an OSPF router are called link-state updates (LSUs), and the items sent in an LSU include individual link-state advertisements (LSAs). OSPF uses a reliable protocol to exchange routing information, ensuring that lost LSU packets are retransmitted. OSPF routers can, thus, determine whether a neighbor has received all the LSAs.

## Section 5.1: Building Routing Table on New OSPF-Configured Routers

Five packets are used to build the routing table on a new OSPF-configured router. These are the Hello protocol; the database descriptor, which is used to send summary information to neighbors to synchronize topology databases; the link-state request, which works as a request for more detailed information that is sent when the router receives a database descriptor that contains new information; the link-state update, which works as the link-state advertisement (LSA) packet issued in response to the request for database information in the link-state request packet; and the link-state acknowledgement, which acknowledges the link-state update.

When the new OSPF-configured router router is connected to the network, it must learn the network from the routers that are up and running. The router goes through three stages while exchanging information: the down state, the init stage, and the two-way state. You can check what stage an interface running OSPF is in by using the show ip ospf neighbor command or the debug ip ospf adjacency command.

• The new router starts in a down state. It transmits its own Hello packets to introduce itself to the segment and to find any other OSPF-configured routers. This is sent out as a Hello to the multicast address 224.0.0.5 (AllSPFRouters). It sets the designated router (DR) router and the backup designated router (BDR) routerin the Hello to 0.0.0.0.

**The DR and the BDR**

The designated router (DR) is a router on broadcast multi-access network that is responsible for maintaining the topology table for its segment. This router can be dynamically elected through use of the Hello protocol, or can be designated by the network administrator. Redundancy is provided by the Backup Designated Router (BDR).

• While the new router waits for a reply, which usually is four times the length of the Hello timer, the router is in the init state. Within the wait time, the new router hears a Hello from another router and learns the DR and the BDR. If there is no DR or BDR stated in the incoming Hello, an election takes place.

• Once the new router sees its own router ID in the list of neighbors, and a neighbor relationship is established, it changes its status to the two-way state

The new router and the DR have now established a neighbor relationship and need to ensure that the new router has all the relevant information about the network. The DR must update and synchronize the topology database of the new router. This is achieved by using the exchange protocol with the database description packets (DDPs). There are four different stages that the router goes through while exchanging routing information with a neighbor: the exstart state, the exchange state, the loading state, and the full state.

• During the exstart state, one of the routers will take seniority and become the master router, based on highest IP interface address.

• Both routers will send out database description packets, changing the state to the exchange state. At this stage, the new router has no knowledge and can inform the DR only of the networks or links to which it is directly connected. The DR sends out a series of DDPs containing the networks, referred to as links that are held in the topology database. Most of these links have been received from other routers via link-state advertisements (LSAs). The source of the link information is referred to by the router ID. Each link will have an interface ID for the outgoing interface, a link ID, and a metric to state the value of the path. The DDPs will contain a summary rather than all the necessary information. When the router has received the DDPs from the neighboring router, it compares the received network information with that

in its topology table. In the case of a new router, all the DDPs are new.

• If the new router requires more information, it will request that particular link in more detail using the link-state request packet (LSR). The LSR will prompt the master router to send the link-state update packet (LSU) . This is the same as a LSA used to flood the network with routing information. While the new router is awaiting the LSUs from its neighbor, it is in the loading state.

• When these LSRs are received and the databases are updated and synchronized, the neighbors are fully adjacent. This is the full state.

## Section 5.2: Steady-State Operation

Link-state protocols keep in touch with their neighbors by periodically exchanging small packets rather than complete routing updates. In OSPF, these packets are called Hello packets, which identify the subnet, the router sending the packets and a few other details. These Hello packets serve the same purpose as timed, regular full routing updates serve for distance vector protocols. When a router fails to hear Hellos from a neighbor for an interval called the dead interval, the router assumes that the silent router has failed. OSPF then marks the silent router as "down" in its topology database. The other router then runs the SPF algorithm to calculate new routes, based on the fact that one of the network's routers is now unavailable. In addition, the router that notices the failure immediately floods the new router or link status to its neighbors, with those routers forwarding the updated status to their neighbors, eventually flooding the new status information to all the routers in the network. This quick convergence of link-state protocols prevents the occurrence of loops.

## Section 5.3: OSPF Areas

There are a number of problems associated with using OSPF. These problems are related to the network size. The larger the network, the greater the probability of a network change, which would require a recalculation of the whole area. This increases the frequency with which the SPF algorithm is being run. In addition, each recalculation will take longer. As the network grows, the size of the routing table will increase. Although the complete routing table is not sent out as in a distance vector routing protocol, the greater the size of the table, the longer each lookup becomes. The memory requirements on the router will also increase. Furthermore, the topological database will increase in size and will eventually become unmanageable. As the various databases increase in size and the calculations become increasingly frequent, the CPU utilization will increase as more of the available memory I consumed. This will have a negative impact on network response time, not because of congestion on the line but because of congestion within the router itself.

Using multiple OSPF areas solves most of the common problems with running OSPF in larger networks. The division of a large single area network into multiple areas allows routers in each area to maintain their own topological databases. This limits the size of the topological databases within an area, which results in routers requiring less memory and processing time to run SPF, and a decrease in convergence time. Summary and external links ensure connectivity between areas and networks outside the autonomous area (AS). This is achieved by creating areas from groups of subnets. Each area is treated internally as a small entity on its own. It communicates with the other areas, exchanging routing information which is kept to a minimum by allowing only that information that is required for connectivity.
There are two approaches to implementing multiple area networks. The first approach is to grow a single area until it becomes unmanageable. This approach requires less initial work and configuration but care

should be put into the design of the network because this may cause problems in the future, particularly in addressing. The second approach is to design the network with multiple areas, which are very small, in anticipation that the networks will grow to fit comfortably into their areas. In practice, many companies convert their networks into OSPF from a distance vector routing protocol when they realize that they have outgrown the existing routing protocol. This allows the planned implementation of the second approach.

## 5.3.1: OSPF Area Types

Regardless of which approach is used, a multiple area OSP network has a hierarchical structure and consists a number of distinct areas. These areas are:

• The backbone area, which is also referred to as Area 0. All other areas must connect to the backbone area. Hence, this area is obligatory.

• An ordinary or standard area, which is an area that connects to the backbone (Area 0) and is treated as a separate entity. All routers in a standard area have the same topological database, but their routing tables will be based on the routers position in the standard area and will thus be unique to the router.

• A stub area, which is an area that does not accept external summary routes. A router within a stub area can only see outside the autonomous system if a default route has been configuration for it.

• A totally stubby area, which is similar to a stub area. In this area, the default route must be configured as 0.0.0.0. This type of area is useful for remote sites that have few networks and limited connectivity with the rest of the network and is a Cisco proprietary solution.

• A not so stubby area (NSSA) , which is a stub area that can receive external routes but will not propagate those external routes into the backbone area.

### 5.3.2: Router Responsibilities

Because of the hierarchical nature of a multiple area OSPF network, routers have different responsibilities, depending on their position and functionality within the hierarchical design. These routers have different designations such as internal routers, backbone routers, area border routers (ABR), and autonomous system boundary routers (ASBR).

• The Internal Router exists within an area. It is responsible for maintaining a current and accurate database of every subnet within the area. It is also responsible for forwarding data to other networks by the shortest path. Flooding of routing updates is confined to the area. All interfaces on this router are within the same area.

• The Backbone Router exists within the backbone area, which is also called Area 0. The design rules for OSPF require that all the areas be connected through a single area, known as Area 0. Area 0 is also known as Area 0.0.0.0 on other routers. A router within this area is referred to as a backbone router. It may also be an internal router or an Area Border Router.

• The Area Border Router (ABR) is responsible for connecting two or more areas. It holds a full topological database for each area to which it is connected and sends LSA updates between the areas. These LSA updates are summary updates of the subnets within an area. It is at the area border that

summarization should be configured for OSPF because this is where the LSAs make use of the reduced routing updates to minimize the routing overhead on both the network and the routers.

• The Autonomous System Boundary Router (ASBR) is used to connect to a network or routing protocol outside the OSPF domain. OSPF is an interior routing protocol or Interior Gateway Protocol (IGP); gateway is an older term for a router. If there is any redistribution between other protocols to OSPF on a router, it will be an ASBR. This router should reside in the backbone area but you can place it anywhere in the OSPF hierarchical design.

## Section 5.4: Balanced Hybrid Routing Protocol and EIGRP

Cisco supports two distance vector IP routing protocols, namely RIP and IGRP; two link-state IP routing protocols, namely OSPF and Intermediate System-to-Intermediate System (IS-IS); and a single balanced hybrid IP routing protocol, namely Enhance IGRP (EIGRP).

EIGRP is called a balanced hybrid protocol because it has some features that act like distance vector protocols and some features that act like link-state protocols. EIGRP uses neighbor discover and exchange full routing information. Like OSPF, EIGRP sends and receives hello packets to ensure that the neighbor is still available but uses a different Hello packet than OSPF. When link status changes or new subnets are discovered, reliable routing updates are sent, but only with the new information.
EIGRP uses a formula based on bandwidth and delay to calculate the metric associated with a route. It uses the same formula used by IGRP, but the number is multiplied by 256 to accommodate calculations when very high bandwidth values are used.

### 5.4.1: EIGRP Loop Avoidance

EIGRP avoids loops by keeping some basic topological information but not full information. When a router learns multiple routes to the same subnet, it puts the best route in the routing table, following the same rules about adding multiple equal-metric routes as IGRP. In EIGRP, the best route, i.e., the route with the lowest metric is called the successor. EIGRP also runs an algorithm to identify which backup routes could be used in case of a route failure, without causing a loop. These routes are called feasible successors.
Should the best route (successor) fail and there are no feasible successors for that route, EIGRP uses a distributed algorithm called Diffusing Update Algorithm (DUAL). DUAL sends queries looking for a loop-free route to the subnet in question. When the new route is found, DUAL adds it to the routing table.

**Table 5.1: EIGRP, IGRP and OSPF Compared**

| Feature | EIGRP | IGRP | OSPF |
|---|---|---|---|
| Discovers neighbors before exchanging routing table | ✓ | ✗ | ✓ |
| Builds topology table | ✓ | ✗ | ✓ |
| Quick convergence | ✓ | ✗ | ✓ |
| Metrics based on bandwidth and delay by default | ✓ | ✓ | ✗ |
| Exchanges full routing information | ✗ | ✓ | ✗ |
| Requires distance vector loop-avoidance features | ✗ | ✓ | ✗ |
| Public standard | ✗ | ✗ | ✓ |

## Section 5.5: Router Configuration

### 5.5.1: Configuring OSPF

There are a few simple commands that are used to configure and troubleshoot a Cisco router configured to use OSPF in a single area and in a multiple area network. The commands used to configure OSPF are:

• router ospf < process_number > where process_number is a number local to the router. This command configures OSPF as the routing protocol on the router.

• network network_number wildcard_mask defines the networks that are to participate in the OSPF updates and the area that they reside in.

• interface loopback < interface_number > ip address < ip_address > < subnet_mask > defines a loopback interface, which is a virtual interface, on the router.

• ip ospf cost < cost > sets the default cost for the router.

• auto-cost reference-bandwidth changes the OSPF cost formula.

**Note**: The ip ospf cost command overrides the auto-cost referencebandwidth
command.

### 5.5.2: Verifying the OSPF Configuration

There are a number of show ip commands that can be used when troubleshooting an OSPF network. These commands are:

• show ip ospf, which provides information about the OSPF process and its details.

• show ip ospf database, which provides information about the contents of the topological database.

• show ip ospf interface, which provides information on how OSPF has been configured on each interface.

• show ip ospf neighbor, which displays all the information about the relationship that the router has with its neighbors.

• show ip protocols, which displays the IP configuration on the router, including the interfaces and the configuration of the IP routing protocols.

• show ip route [ip-address [mask] [longer-prefixes]] | [protocol [process-id]], which provides detailed information on the networks that the router is aware of and the preferred paths to those networks. It also gives the next logical hop as the next step in the path.

• debug ip ospf events, which issues log messages for each OSPF packet.

• debug ip ospf packet, which issues log messages describing the contents of all OSPF packets.

### 5.5.3: Configuring EIGRP

The commands used to configure EIGRP on a Cisco router are consistent with the other IP routing protocol commands. The EIGRP commands are:

• router eigrp autonomous_system_number configures EIGRP as the routing protocol on the router.

• network network_number [ wildcard_mask ] defines the networks that are to participate in the EIGRP updates. The [ wildcard_mask ] optional parameter identifies which interfaces are running EIGRP.

• no network network_number [ wildcard_mask ] disables EIGRP.

• no autosummary turns off automatic summarization.

• ip summary address eigrp autonomous_system_number ip_address subnet_mask configures summarization at the interface level.

• variance multiplier configures EIGRP to load-balance across unequal paths.

• bandwidth line_speed overrides the default bandwidth settings on the links.

## 5.5.4: Verifying the EIGRP Configuration

There are a number of show and debug commands that can be used to configure, maintain, and troubleshoot a live EIGP network. The show commands are:

• show ip eigrp neighbors, which provides detailed information on the neighbors.

• show ip eigrp topology, which provides details about the routes held in the topology table and for detailed information on the networks that the router is aware of and the preferred paths to those networks, as well as the next logical hop as the first step in the path.

• show ip eigrp topology all, which provides details about all the routes and alternative paths held in the topology table.

• show ip eigrp traffic, which provides information on the aggregate traffic sent to and from the EIGRP process.

• show ipx route, which shows the routing table for IPX and is the source of the information on how to reach the remote destination network.

• show ip route, which provides detailed information on the networks that the router is aware of and the preferred paths to those networks.

• show ip protocols, which displays the IP configuration on the router, including the interfaces and the configuration of the IP routing protocols.

## Topic 6: Advanced TCP/IP

The original design for the Internet required every organization to have one or more unique IP network numbers. In the early to mid-1990s, it became apparent that the Internet was growing so fast that all IP network numbers would be used. One solution to this problem was to increase the size of the IP address by developing IP Version 6 (IPv6). IPv6 has a much larger address structure than IPv4, allowing for trillions of IPv6 networks.

Three other IP functions have been introduced to reduce the need for IPv4 registered network numbers. These include Network Address Translation (NAT), along with a feature called private IP addressing, which allows organizations to use unregistered IP network numbers internally and still communicate well with the Internet; and Classless Interdomain Routing (CIDR), which allows Internet service providers (ISPs) to reduce the wasting of IP addresses by assigning a company a subset of a network number rather than the entire network. CIDR has been discussed in Section 3.1.4.

## Section 6.1: Private IP Addressing

Most organizations have a number of computers that will never be connected to the Internet. These computers do not need globally unique IP addresses but must be unique within the organization's network. Thus, an organization could use any network number(s) it wanted, regardless of whether those network number(s) are in use on the Internet or not. However, a set of IP addresses from Class A, Class B and Class C has been set aside for use in private networks and has been defined in RFC 1918. This RFC defines a set of networks that not be assigned to any organization as a registered network number to be used on the Internet. These network numbers allow organizations to use unregistered network numbers that are not used by anyone else in the public Internet. However, no organization is allowed to advertise these networks using a routing protocol on the Internet.

**TABLE 6.1: The Private IP Address Space defined by RFC 1918**

| Range of IP Addresses | Number of Networks | Class |
|---|---|---|
| 10.0.0.0 to 10.255.255.255 | 1 | A |
| 172.16.0.0 to 172.31.255.255 | 16 | B |
| 192.168.0.0 to 192.168.255.255 | 256 | C |

## Section 6.2: Network Address Translation (NAT)

The advantage of using private IP addresses is that it allows an organization to use private addressing in a network, and use the Internet at the same time, by implementing Network Address Translation (NAT). NAT is defined in RFC 1631 and allows a host that does not have a valid registered IP address to communicate with other hosts through the Internet. Essentially, NAT allows hosts that use private addresses or addresses assigned to another organization, i.e. addresses that are not Internet-ready, to continue to be used and still allows communication with hosts across the Internet. NAT accomplishes this by using a valid registered IP address to represent the private address to the rest of the Internet. The NAT function changes the private IP addresses to publicly registered IP addresses inside each IP packet that is transmitted to a host on the Internet.

### 6.2.1: Variations of NAT

The Cisco IOS software supports several variations of NAT.

**Cisco Terminology**

Cisco uses the term inside local

These include Static NAT; Dynamic NAT; and Overloading NAT with Port Address Translation (PAT) .

### 6.2.1.1: Static NAT

In Static NAT, the IP addresses are statically mapped to each other. Thus, the NAT router simply configures a one-to-one mapping between the private address and the registered address that is used on its behalf. Supporting two IP hosts in the private network requires a second static one-to-one mapping using a second IP address in the public address range, depending on the number of addresses supported by the registered IP address.

for the private IP addresses and inside global for the public IP addresses. The enterprise network that uses private addresses, and therefore that needs NAT, is the "inside" part of the network. The Internet side of the NAT function is the "outside" part of the network. A host that needs NAT has the IP address it uses inside the network, and it needs an IP address to represent it in the outside network

### 6.2.1.2: Dynamic NAT

Dynamic NAT is similar to static NAT in that the NAT router creates a one-to-one mapping between an inside local and inside global address and changes the IP addresses in packets as they exit and enter the inside network. However, the mapping of an inside local address to an inside global address happens dynamically. Dynamic NAT accomplishes this by setting up a pool of possible inside global addresses and defining criteria for the set of inside local IP addresses whose traffic should be translated with NAT. With dynamic NAT, you can configure the NAT router with more IP addresses in the inside local address list than in the inside global address pool. When the number of registered public IP addresses is defined in the inside global address pool, the router allocates addresses from the pool until all are allocated. If a new packet arrives, and it needs a NAT entry, but all the pooled IP addresses are already allocated, the router discards the packet. The user must try again until a NAT entry times out, at which point the NAT function works for the next host that sends a packet. This can be overcome through the use of Port Address Translation (PAT).

### 6.2.1.3: Overloading NAT with Port Address Translation (PAT)

In some networks, most, if not all, IP hosts need to reach the Internet. If that network uses private IP addresses, the NAT router needs a very large set of registered IP addresses. If you use static NAT, each private IP host that needs Internet access needs a publicly registered IP address. Dynamic NAT lessens the problem, but if a large percentage of the IP hosts in the network need Internet access throughout normal business hours, a large number of registered IP addresses would also be required. These problems can be overcome through overloading with port address translation. Overloading allows NAT to scale to support many clients with only a few public IP addresses.

To support lots of inside local IP addresses with only a few inside global, publicly registered IP addresses, NAT overload uses Port Address Translation (PAT) , translating the IP address as well as translating the port number. When NAT creates the dynamic mapping, it selects not only an inside global IP address but also a unique port number to use with that address. The NAT router keeps a NAT table entry for every unique combination of inside local IP address and port, with translation to the inside global address and a unique port number associated with the inside global address. And because the port number field has 16 bits, NAT overload can use more than 65,000 port numbers, allowing it to scale well without needing many registered IP addresses..

### 6.2.1.4: Translating Overlapping Addresses

NAT can also be used in organizations that do not use private addressing but use a network number registered to another company. If one organization uses a network number that is registered to another organization, and both organizations are connected to the Internet, NAT can be used to translate both the source and the destination IP addresses. However, both the source and the destination addresses must be changed as the packet passes through the NAT router.

### 6.2.2: Configuring NAT

There are a number of commands that can be used to configure the different variations of NAT.

### 6.2.2.1: Configuring Static NAT

Static NAT configuration requires that each static mapping between a local, or private, address and a global, or public, address must be configured. Then, each interface needs to be identified as either an inside or outside interface.

The ip nat inside source static command is used to create a static mapping. The inside keyword indicates that NAT translates addresses for hosts on the inside part of the network. The source keyword indicates that NAT translates the source IP address of packets coming into its inside interfaces. The static keyword indicates that the parameters define a static entry. If two hosts require Internet access, two ip nat inside commands must be used.

The ip nat inside and ip nat outside interface subcommands identify which interfaces are "inside" and which are "outside" respectively.
Two show commands list the most important information about static NAT. These commands are:

• show ip nat translations, which lists the static NAT entries; and the

• show ip nat statistics, which lists statistics, including the number of currently active translation table entries and the number of hits, which increments for every packet for which NAT must translate addresses.

### 6.2.2.2: Configuring Dynamic NAT

Dynamic NAT configuration differs from static NAT but it also has some similarities. It requires that each interface be identified as either an inside or outside interface but the static mapping is not required. In addition, a pool of inside global addresses needs to be defined.

The ip nat inside source command is used to identify which inside local IP addresses need to have their addresses translated.

The ip nat pool command defines the set of IP addresses to be used as inside global addresses.
The two show commands used to trouble shoot static NAT can also be used to troubleshoot dynamic NAT. In addition to these you can use the debug ip nat command. This command causes the router to issue a message every time a packet has its address translated for NAT.

### 6.2.2.3: Configuring NAT Overload and PAT

The ip nat inside source overload command is used to configure NAT overload. The overload parameter is required to enable overload. Without this parameter, the router does not perform overload, but dynamic NAT.

You can use the show ip nat translations to troubleshoot NAT overload.

## Section 6.3: Internet Control Message Protocol (ICMP)

Internet Control Message Protocol (ICMP) is a TCP/IP protocol designed to help manage and control the operation of a TCP/IP network. The ICMP protocol provides a wide variety of information about a network's status and is considered part of TCP/IP's network layer. ICMP can provide useful information for troubleshooting TCP/IP.
ICMP uses messages to accomplish its tasks. Many of these messages are used in even the smallest IP network. Table 6.2 lists some of the ICMP messages.

**Table 6.1: ICMP Messages**

| Message | Description |
|---|---|
| Destination Unreachable | Informs the source host that there is a problem delivering a packet. |
| Time Exceeded | Indicates that the time that it takes a packet to be delivered has expired and that the packet has been discarded. |
| Redirect | Indicates that the packet has been redirected to another router that has a better route to the destination address. The message informs the sender to use the better route. |
| Echo | Used by the **ping**command to verify connectivity. |

## Section 6.4: FTP and TFTP

File Transfer Protocol (FTP) and Trivial File Transfer Protocol (TFTP) are two popular file transfer protocols used in TCP/IP networks. Most end users use FTP, but Cisco router and switch administrators often use TFTP. FTP is a TCP-based application that has many options and features, including the capability to change directories, list files using wildcard characters, transfer multiple files with a single command, and use a variety of character sets or file formats. When a user, called a FTP client, attempts to connect to an FTP server, a TCP connection is established to the FTP server's well-known port 21. The user is required to enter a username and password, which the server uses to authenticate the files available to that user for read and write permissions. This security is based on the file security on the server's platform. All the commands used to control the transfer of a file are sent across this connection. At this point, the user has a variety of

commands available to enable settings for transfer, change directories, list files, etc. However, whenever a get (mget for multiple files) or put (or mput for multiple files) command is entered, or the equivalent button is clicked on the user interface, a file is transferred. The data is transferred over a separate FTP data connection, another TCP connection, established to well-known port 20. This prevents a file transfer from impacting on the control connection.

Trivial File Transfer Protocol (TFTP) is a more basic file transfer protocol that use a small set of features, takes little memory to load, and little time to program. TFTP uses User Datagram Protocol (UDP), so there is no connection establishment and no error recovery by the transport layer. However, TFTP uses application layer recovery by embedding a small header between the UDP header and the data. This header includes codes along with a numbering scheme that numbers 512-byte blocks of data. The TFTP application uses these block numbers to acknowledge receipt and resend the data. TFTP sends one block and waits on an acknowledgment before sending another block.

### Section 6.5: MTU and Fragmentation

TCP/IP defines a maximum length for an IP packet. The term used to describe that maximum length is maximum transmission unit (MTU). The MTU varies based on configuration and the interface's characteristics. By default, a computer calculates an interface's MTU based on the maximum size of the data portion of the data-link frame. IP hosts, including routers, cannot forward a packet out an interface if the packet is longer than the MTU. Therefore, if a router's interface MTU is smaller than a packet that must be forwarded, the router fragments the packet into smaller packets, each of which is less than or equal to the MTU value.

The fragmented packets are reassembled by the endpoint host. The IP header contains fields that are used for reassembling the fragments. This includes an ID value that is the same in each fragmented packet, as well as an offset value that defines which part of the original packet is held in each fragment. Therefore fragmented packets can be reassembled in the correct order.

Two configuration commands can be used to change the IP MTU size on an interface. These are:

• mtu, which sets the MTU for all Layer 3 protocols; and

• ip mtu, which sets the MTU for IP only.

If both mtu and ip mtu are configured on an interface, the ip mtu setting takes precedence. However, if the mtu command is configured after ip mtu is configured, the ip mtu value is reset to the mtu value.

## Topic 7: Wide Area Networks (WANs)

When designing networks, you need to know about the various Wide Area Network (WAN) connectivity options. There are three main categories of WAN connectivity options. These are:

• Leased point-to-point lines;

• Dial lines, which are also called circuit-switched lines; and

• Packet-switched networks.

This chapter discusses these three WAN connectivity options.

## Section 7.1: Point-to-Point Leased Lines

### 7.1.1: Overview

Point-to-point leased lines are established across synchronous point-to-point serial links. These synchronous point-to-point links include a cable from a service provider, with the service including the capability to send and receive bits across that cable at a predetermined speed. The physical connection includes a CSU/DSU on each end of the link. After the CSU/DSUs are configured and the lines are installed, only a small amount of configuration is required on the routers to get the two routers working. You only need to configure IP addresses on each router and run a no shutdown command on each interface to enable them to ping each other across the link. The IP addresses of the two routers at either end of the synchronous point-to-point serial link must be in the same subnet because the two routers' interfaces are not separated by some other IP router.

Generally, the no shutdown command is not required but if a Cisco router comes up, and the physical WAN link is not working, the router might place a shutdown command on the interface configuration. So the no shutdown interface subcommand would be needed to put the interface in service.

**Synchronicity**

Synchronous WAN links require that the CSU/DSUs on each end of the link operate at the exact same speed. The CSU/DSUs on each side of the WAN link agree to use a certain clock rate, or speed, to send and receive bits. After they agree to a particular speed, both CSU/DSUs try to operate at that speed. One CSU/DSU is responsible for monitoring the clock rates between itself and the other CSU/DSU and makes small adjustments to match the clock rate of the other CSU/DSU. The CSU/DSU that does not adjust its clock is called the clock source.

### 7.1.2: Data-Link Protocols

There are a number of different data link layer protocols that can be implemented on a point-to-point WAN. WAN data-link protocols used on point-to-point serial links provide the basic function of data delivery across that one link. The two most popular WAN data-link protocols are High-Level Data Link Control (HDLC) and PPP. Both of these protocols provide for the delivery of data across a single point-to-point serial link and deliver data on synchronous serial links. In addition, PPP also supports asynchronous serial links.

Each synchronous serial data-link protocol is frame-oriented, with each data-link protocol defining the beginning and end of the frame, the information and format of a header and trailer, and the location of the packet between the header and trailer. Data-link protocols also send idle frames. This is because synchronous WAN links require that the CSU/DSUs on each end of the link operate at the exact same speed. To accomplish this, the CSU/DSUs on each side of the WAN link agree to use a certain clock rate, or speed, to send and receive bits. After they agree to a particular speed, both CSU/DSUs try to operate at that speed. One CSU/DSU is responsible for monitoring the clock rates between itself and the other CSU/DSU by noticing changes in the electrical signal received on the physical line. When a change occurs, the CSU/DSU monitoring the clock rates responds by adjusting its clock speed. If no traffic was sent across the link, there would be no electrical signal and clock synchronization would be lost. Therefore synchronous data-link protocols send idle frames when there is no end-user data to be sent over the link. The idle frames are called Receiver Ready. This need to monitor and adjust the clock rates for synchronous protocols requires more expensive hardware than asynchronous protocols. However, synchronous protocols allow more throughput over a serial link than asynchronous protocols. For links between routers, synchronous links are typically desired and used.

Almost all data-link protocols, including PPP and HDLC, perform error detection. These protocols use a field in the trailer called the frame check sequence (FCS) for this purpose. The FCS is used to verify whether bit errors occurred during transmission of the frame. If bit errors occurred, the frame is discarded. However, error recovery, which is the process that causes retransmission of the lost or errored frame, is not guaranteed. Error recovery can be performed by the data-link protocol or a higher-layer protocol, or it might not be performed at all.

PPP was defined much later than the original HDLC specifications. As a result, PPP includes many new features that are not implemented in HDLC. For this reason, PPP has become the most popular WAN data link layer protocol.

PPP uses a protocol that offers features regardless of the Layer 3 protocol used, and a protocol to support each Layer 3 protocol supported on the link. The PPP Link Control Protocol (LCP) provides the core features for PPP that operate regardless of the Layer 3 protocol used, while a series of PPP control protocols, such as IP Control Protocol (IPCP), provide features related to a specific Layer 3 protocol. Thus, PPP uses one LCP per link and one Control Protocol for each Layer 3 protocol defined on the link. If a router is configured for IPX, AppleTalk, and IP on a PPP serial link, the router configured for PPP encapsulation automatically tries to bring up the appropriate control protocols for each Layer 3 protocol. Cisco routers also use a PPP CP for supporting CDP traffic, called CDPCP.

LCP provides a variety of optional features for PPP. These are:

• Error detection, which is provided by Link Quality Monitoring (LQM). The router can be configured to take down the link after a configured error rate has been exceeded. By taking down a link that has many errors, you can cause packets to use an alternative path that might not have as many errors but this is only useful when you have redundant routes in the network.

• Looped link detection, which is provided by magic numbers. Using different magic numbers, routers send messages to each other. If a router receives its own magic number, it knows that the frame it sent

has been looped back. If configured to do so, the router can take down the interface through which the frame was sent, and effectively close the loop. This will improve convergence.

• Multilink support, which is provided by Multilink PPP and allows PPP to load-balance fragments of packets across multiple links.

• Authentication, which can be provided by Password Authentication Protocol (PAP) or Challenge Handshake Authentication Protocol (CHAP) and allows for the exchange of names and passwords so that each device can verify the identity of the device on the other end of the link. CHAP is the preferred method because it uses a Message Digest 5 (MD5) one-way hash to encode the password while PAP sends passwords in clear-text.

### 7.1.3: Configuring HDLC and PPP

HDLC and PPP configuration is straightforward. You only need to be sure that the same WAN data-link protocol is configured on each end of the serial link because each WAN data-link protocol uses a different frame format. The command used to configure which protocol to use is: encapsulation {hdlc | ppp}.
The compress [predictor | stac | mppc [ignore-pfc]] command can be used to configure compression. The predictor, stac or mppc options specify which compression algorithm must be used. These are predictor for predictor, Stacker (LZS) for stac and MPPC for mppc. The ignore-pfc option specifies that the protocol field compression flag negotiated through LCP will be ignored.

There are also a few show commands that can be used to troubleshoot HDLC and PPP. These are:

• show interfaces[type number] , which lists statistics and details of interface configuration, including the encapsulation type;

• show compress, which lists compression ratios; and

• show processes[cpu], which lists processor and task utilization. This is useful for monitoring the impact of compression.

## Section 7.2: Frame Relay

Frame Relay is a connection-oriented, Layer 2 WAN connection technology. It operates at speeds from 56 kpbs to 45 Mbps. It is very flexible and offers a wide array of deployment options. Frame Relay operates by statistically multiplexing multiple data streams over a single physical link. Each data stream is known as a virtual circuit (VC). Frame Relay VCs come in two types: Permanent Virtual Circuits (PVCs) and Switched Virtual Circuits (SVCs).

Each VC is tagged with an identifier to keep it unique. The identifier, known as a Data Link Connection Identifier (DLCI), is determined on a per-leg basis during the transmission. It must be unique and agreed upon by two adjacent Frame Relay devices. As long as the two agree, the value can be any valid number, and the number does not have to be the same end to end. Valid DLCI numbers are 16-1007. For DLCI purposes, 0-15 and 1008-1023 are reserved. The DLCI also defines the logical connection between the Frame Relay (FR) switch and the customer premises equipment (CPE).
Frame Relay devices fall into two possible roles, data terminal equipment (DTE) or data circuitterminating equipment (DCE). The DTE/DCE relationship is a Layer 2 (data link) layer relationship. DTE

and DCE relationships are normally electrical. The DTE/DCE relationship at Layer 1 is independent of that at Layer 2.

• DTEs are generally considered to be terminating equipment for a specific network and are located at the customer premises.

• DCEs are carrier-owned internetworking devices. DCE equipment provides clocking and switching services in a network; they are the devices that actually transmit data through the WAN. In most cases, the devices are packet switches.

Local Management Interface (LMI) is the means by which Frame Relay edge devices maintain keepalive messages. The Frame Relay switch is responsible for maintaining the status of the CPE device(s) to which it is attached. LMI is the communication by which the switch monitors status. LMI implements a keepalive mechanism that verifies connectivity between DCE and DTE and the fact that data can flow. A LMI multicast capability, in conjunction with an LMI multicast addressing mechanism, enables attached devices to learn local DLCIs as well as provide global, rather than local, significance to those DLCIs. Finally, LMI provides a status indicator that is constantly exchanged between router and switch. The LMI setting is configurable.

Frame Relay networks provide more features and benefits than simple point-to-point WAN links, but to do that, Frame Relay protocols are more detailed. Frame Relay networks are multiaccess networks, which means that more than two devices can attach to the network, similar to LANs. However, unlike LANs, you cannot send a data link layer broadcast over Frame Relay. Therefore, Frame Relay networks are called nonbroadcast multi-access (NBMA) networks. Also, because Frame Relay is multiaccess, it requires the use of an address that identifies to which remote router each frame is addressed.

### 7.2.1: Virtual Circuits

Frame Relay provides significant advantages over simply using point-to-point leased lines. The primary advantage has to do with virtual circuits (VCs) which define a logical path between two Frame Relay DTEs. The VC acts like a point-to-point circuit, providing the ability to send data between two endpoints over a WAN. However, there is no physical circuit directly between the two endpoints. VCs share the access link and the Frame Relay network. Each VC has a committed information rate (CIR), which is a guarantee by the provider that a particular VC gets at least that much bandwidth. Service providers can build their Frame Relay networks more cost-effectively than for leased lines. Therefore, Frame Relay is more cost-effective than leased lines for connecting many WAN sites.

Two types of VCs are allowed-permanent (PVC) and switched (SVC). PVCs are predefined by the provider, while SVCs are created dynamically. A Frame Relay network which includes PVCs between each pair of sites is called a full mesh Frame Relay network. In such a network, any two sites are connected by a PVC. When not all pairs have a direct PVC, it is called a partial mesh network. In such networks packets must be forwarded through other sites when packets are to be transmitted between two sites that are not directly connected by a PVC. This is the major disadvantage of partial mesh networks, however, partial mesh networks are cheaper because the provider charges per VC.
Frame Relay uses an address to differentiate one PVC from another. This address is called a data-link connection identifier (DLCI). The name is descriptive: The address is for an OSI Layer 2 (data link) protocol, and it identifies a VC, which is sometimes called a virtual connection. DCLI addressing is

discussed in more detail in Section 7.3.3. (SVC)

### 7.2.2: LMI and Encapsulation Types

The LMI is a definition of the messages used between the DTE and the DCE. The encapsulation defines the headers used by a DTE to communicate some information to the DTE on the other end of a VC. The switch and its connected router care about using the same LMI; the switch does not care about the encapsulation. The endpoint routers (DTEs) do care about the encapsulation. The most important LMI message relating to topics on the exam is the LMI status inquiry message. Status messages perform two key functions:

• Perform a keepalive function between the DTE and DCE. If the access link has a problem, the absence of keepalive messages implies that the link is down.

• Signal whether a PVC is active or inactive. Even though each PVC is predefined, its status can change. An access link might be up, but one or more VCs could be down. The router needs to know which VCs are up and which are down. It learns that information from the switch using LMI status messages.

There are three LMI protocol options are available in Cisco IOS software:

• Cisco LMI, which is a Cisco propriety solution and uses DLCI 1023;

• ANSI LMI, which is also known as Annex D and uses DLCI 0; and

• Q933a, which is defined by the ITU and is also known as Annex A.

Although the difference between these three LMI protocol options is slight; they are incompatible with one another. Therefore both the DTE and DCE on each end of an access link must use the same LMI standard. Configuring the LMI type is easy and includes a default LMI setting, which uses the LMI autosense feature, in which the router figures out which LMI type the switch is using. If you choose to configure the LMI type, it disables the autosense feature. You can use the frame-relay { cisco | ansi | itu } interface subcommand to configure LMI type.

A Frame Relay-connected router encapsulates each Layer 3 packet inside a Frame Relay header and trailer before it is sent out an access link. The header and trailer are defined by the Link Access Procedure Frame Bearer Services (LAPF) specification, ITU Q.922-
A. The sparse LAPF framing provides error detection
with an FCS in the trailer, as well as the DLCI, DE, FECN, and BECN fields in the header.
However, the LAPF header and trailer do not identify the type of protocol, which is needed by routers. As discussed in Section 2 and Section 3, a field in the data-link header must define the type of packet that follows the data-link header. If Frame Relay is using only the LAPF header, DTEs (including routers) cannot support multiprotocol traffic, because there is no way to identify the type of protocol in the Information field. Two solutions were created to compensate for the lack of a protocol type field in the standard Frame Relay header:

• Cisco and three other companies created an additional header, which comes between the LAPF header and the Layer 3 packet. It includes a 2-byte Protocol Type field, with values matching the same field used for HDLC by Cisco.

• RFC 1490, which was superceded by RFC 2427, defines a similar header, also placed between the LAPF header and Layer 3 packet, and includes a Protocol Type field as well as many other options. ITU and ANSI later incorporated RFC 1490 headers into their Q.933 Annex E and T1.617 Annex F specifications, respectively.

DTEs use and react to the fields specified by these two specifications, but Frame Relay switches ignore them. In the configuration, the encapsulation created by Cisco is called cisco, and the other is called ietf.

### 7.2.3: DLCI Addressing

The DLCI is an addressing mechanism used to identify a VC so that when multiple VCs use the same access link the Frame Relay switches know how to forward the frames to the correct remote sites. Two important features of the DLCI are:

• The Frame Relay headers, which have a single DLCI field, not both Source and Destination DLCI fields; and

• The local significance of the DLCI, which means that the addresses need to be unique only on the local access link. This is called local addressing.

Because there is only a single DLCI field in the Frame Relay header, Global addressing can be used, making DLCI addressing look like LAN addressing in concept. Global addressing is a way of choosing DLCI numbers when planning a Frame Relay network so that working with DLCIs is much easier.

### 7.2.4: Frame Relay Configuration

In many cases, the configuration of Frame Relay can be as simple as setting the encapsulation and putting an IP address on the interface. This enables inverse-ARP to dynamically configure the DLCI and discover neighboring routers across the cloud. Although basic functionality can be achieved in this manner, more complex procedures are necessary for hub and spoke subinterface configurations dealing with point-tomultipoint
implementations. The configuration of Frame Relay can be accomplished in a four steps, and entails determining the interface to be configured, configuring Frame Relay encapsulation, configuring protocol specific parameters, and configure Frame Relay characteristics.

### 7.2.4.1: Determining the Interface

The interface that interfaces the Frame Relay network is the one that should be configured. Once the interface has been selected, you should change to the appropriate interface configuration mode in the router. You should decide whether subinterfaces should be implemented. For a single point-to-point implementation, it might not be necessary to use subinterfaces; however, this implementation does not scale. If future sites are planned, it is best to use subinterfaces from the beginning.
To create a subinterface, use the following command to change to the desired interface:

```
interface interface_type
interface_number.subinterface_number
```

For example, to create subinterface 1 on Serial 0, use the command interface serial 0.1. You must also determine the nature, or cast type, of the subinterface to be created, i.e., decide whether the subinterface will act as a point-to-point connection or a point-to-multipoint connection. If not specified, the subinterface defaults to a multipoint connection. To specify the cast type, add the keywords point-to-point or multipoint to the end of the previous command.

### 7.2.4.2: Configuring Frame Relay Encapsulation

To enable Frame Relay on the interface, issue the encapsulation frame relay command. The encapsulation of the interface determines the way it should act because each encapsulation is technologyspecific.
The encapsulation specified at this point dictates the Layer 2 framing characteristics of the packet passed to this specific interface from Layer 3. Once the Layer 2 framing is established, the resulting frame can be passed down to the physical layer for transmission.

### 7.2.4.3: Configuring Protocol-Specific Parameters

For each protocol to be passed across the Frame Relay connection, you must configure appropriate addressing. This addressing must be planned in advance. For point-to-point connections, each individual circuit should have its own subnetwork addressing and two available host addresses. For IP, each subinterface is assigned a separate and unique IP subnet. For IPX, each subinterface must have a unique IPX network number, and so on. As with any other addressing scheme, each side of the link must have a unique host address. For point-to-multipoint connections, each subinterface also must have unique addressing. However, a point-to-multipoint connection can connect to multiple remote sites. Thus, all sites sharing the point-to-multipoint connection are members of the same subnetwork, no matter the number of connections or the protocol. The cast type of the interface also dictates the manner in which DLCIs are assigned to the Frame Relay interface. The next section covers this topic in detail.

### 7.2.4.4: Configuring Frame Relay Characteristics

You must define specific parameters for Frame Relay operation. The parameters include LMI and DLCI configuration. If you use a pre 11.2 release of IOS Software, you must specify the LMI type that is being implemented. The Frame Relay service provider, or service provider, should provide the LMI information. For IOS Software Release 11.2 and later, you need not configure the LMI type. To disable LMI completely, use the no keepalive command to cease to transmit and receive LMI. However, keepalives must also be disabled at the switch.

You can now configure address mapping, if necessary. In the case of point-to-point connections, mapping of protocol addresses to DLCIs is dynamic and requires no intervention. However, if point-to-multipoint connections are in use, manual mapping is necessary. Mapping is the same from protocol to protocol and uses the frame-relay map protocol next_hop dlci [ broadcast ][ ietf | cisco ]command.
Protocols supported in the frame-relay map command include IP, IPX, AppleTalk, CLNS, DECnet, XNS, and Vines. The next_hop argument in the command represents the next hop logical address for the router on the remote end of the connection. The dlci argument represents the local DLCI, not that of the remote end. The broadcast keyword specifies that routing updates traverse the network through this circuit. The final option in the command specifies which Frame Relay implementation to utilize in communications with the remote router. When communicating with a Cisco device on the remote side, the default value (cisco) can be utilized. However, when communicating with non-Cisco gear on the remote end, it can be necessary to specify that the IETF implementation of Frame Relay be used.

### 7.2.4.5: Verifying Frame Relay Configuration

The most useful method of verifying the Frame Relay configurations is through the use of the show and debug commands. Some of the more useful show and debug commands are:

• show frame-relay pvc is useful for viewing the status of statically or dynamically defined PVCs. The output for each PVC is detailed. The output also gives information on each circuit that specifies the receipt or transmission of FECN/BECN packets. FECN and BECN deal with congestion in the Frame Relay network, so obviously, a low number is preferable. The output also details the number of discard eligible (DE) packets received for which a low number is better.

• show frame-relay lmi allows you to check the status of individual PVCs and to monitor the communication status between the router and the switch. This command show shows the number of LMI messages sent and received across the link between the router and the switch router. The LMI type can be specified differently for each interface, so the type is specified in the output. This command also shows the LMI input/output information.

• debug frame-relay lmi is probably the most useful tool in verifying and troubleshooting Frame Relay problems. This command makes it possible to watch the real-time communication between the router and the switch. Each request sent from the router to the switch is noted, and the counter is incremented by 1 with each request. LMIs sent from the switch to the router by the service provider are noted and also are incremented by 1 with each request. As long as both numbers are greater than 0, the router should be functioning normally at Layers 1 and 3.

• show frame-relay map is used to view the DLCI mappings that have been created. They can be static or dynamic and are noted as such in the command output.

# Topic 8: IP Access List Security

Network security is a crucial element of any network strategy. Cisco routers can be used as part of your network security strategy. The most important tool in Cisco IOS software used as part of that strategy are IP Access Lists or Access Control Lists (ACLs). IP access lists define rules that can be used to prevent some packets from flowing through the network and should be part of an organization's security policy.
IP access lists cause a router to discard some packets based on criteria the network engineer defines by means of filters. The goal of these filters is to prevent unwanted traffic in the network. Access lists.

There are two main categories of IOS IP ACLs:

• Standard Access Lists, which use simpler logic; and

• Extended Access Lists, which use more-complex logic.

## Section 8.1: Standard IP Access Lists

Filtering logic could be configured on any router and on any of its interfaces. Cisco IOS software applies the filtering logic of an IP access list either as a packet enters an interface or as it exits the interface. In other words, IOS associates an IP access list with an interface, and specifically for traffic either entering or exiting

the interface. After you have chosen the router on which you want to place the access list, you must choose the interface on which to apply the access logic, as well as whether to apply the logic for inbound or outbound packets.

The key features of Cisco IP access list are:

• Packets can be filtered as they enter an interface, before the routing decision.

• Packets can be filtered before they exit an interface, after the routing decision.

• Deny is the term used in Cisco IOS software to imply that the packet will be filtered.

• Permit is the term used in Cisco IOS software to imply that the packet will not be filtered.

• The filtering logic is configured in the access list.

• If a packet does not match any of your access list statements, it is blocked.

Access lists have two major steps in their logic: matching, which determines whether it matches the access-list statement; and action, which can be either deny or permit. Deny means to discard the packet, and permit implies that the packet should be allowed. However, the logic that IOS uses with a multiple-entry ACL can be much more complex. Generally, the logic can be summarized as follows:

**Step 1**: The matching parameters of the access-list statement are compared to the packet.

**Step 2:** If a match is made, the action defined in this access-list statement (permit or deny) is performed.

**Step 3**: If a match is not made in Step 2, repeat Steps 1 and 2 using each successive statement in the IP access list until a match is made.

**Step 4**: If no match is made with an entry in the access list, the deny action is performed.

### 8.1.1: Wildcard Masks

IOS IP access lists match packets by looking at the IP, TCP, and UDP headers in the packet. Standard IP access lists can also examine only the source IP address. You can configure the router to match the entire IP address or just a part of the IP address. When defining the ACL statements you can define a wildcard mask along with the IP address. The wildcard mask tells the router which part of the IP address in the configuration statement must be compared with the packet header. The wildcard masks look similar to subnet masks, in that they represent a 32-bit number. However, the wildcard mask's 0 bits tell the router that those corresponding bits in the address must be compared when performing the matching logic. The binary 1s in the wildcard mask tell the router that those bits do not need to be compared. Thus, wildcard mask 0.0.0.0, which in binary form is 00000000.00000000.00000000.00000000, indicates that the entire IP address must be matched, while wildcard mask 0.0.0.255, which in binary form is 00000000.00000000.00000000.11111111, indicates that the first 24 bits of the IP address must be

matched, and wildcard mask 0.0.31.255, which in binary form is 00000000.00000000.00011111.11111111, indicates that the first 24 bits of the IP address must be matched.

### 8.1.2: Standard IP Access List Configuration

A standard access list is used to match a packet and then take the directed action. Each standard IP access list can match all, or only part, of the packet's source IP address. The only two actions taken when an access-list statement is matched are to either deny or permit the packet. The configuration commands required are:

• ip access-group {number | action [in | out]}, in which action can be either permit of deny and is used to enable access lists; and

• access-class number | action [in | out], which can be used to enable either standard or extended access lists.

The standard access list configuration can be verified using the following show commands:

• show ip interface[type number], which includes a reference to the access lists enabled on the interface;

• show access-lists [access-list-number | access-list-name], which shows details of configured access lists for all protocols; and

• show ip access-list [access-list-number | access-list-name], which shows the access lists.

## Section 8.2: Extended IP Access Lists

Extended IP access lists are similar to standard IP access lists in that you enable extended access lists on interfaces for packets either entering or exiting the interface. IOS then searches the list sequentially. The first statement matched stops the search through the list and defines the action to be taken. The key difference between the extended IP access lists and standard IP access lists is the variety of fields in the packet that can be compared for matching by extended access lists. A single extended IP access list statement can examine multiple parts of the packet headers, requiring that all the parameters be matched correctly in order to match that one IP access list statement. That matching logic is what makes extended access lists both much more useful and much more complex than standard IP access lists. You can configure extended IP access list to match the IP protocol type, which identifies what header follows the IP header. You can specify all IP packets, or those with TCP headers, UDP headers, ICMP, etc, by checking the Protocol field. You can also check the source and destination IP addresses, as well as the TCP source and destination port numbers.
An extended access list is more complex than standard access lists. Therefore the configuration commands are more complex. The configuration command for extended access lists is:

• access-list access-list-number action protocol source source-wildcard destination destination-wildcard [log | log-input], which can be used to enable access lists;

## Section 8.3: Named IP Access Lists

Named IP access lists can be used to match the same packets, with the same parameters, you can match with

standard and extended IP access lists. Named IP access lists do have some differences, however. The most obvious difference is that IOS identifies named IP access lists using names you assign them as opposed to numbers. Named IP access lists also have another key feature that numbered IP access lists do not: You can delete individual lines in a named IP access list.

In addition, two important configuration differences exist between numbered and named access lists. One key difference is that named access lists use a global command that places the user in a named IP access list submode, under which the matching and permit or deny logic is configured. The other key difference is that when a named matching statement is deleted, only that one statement is deleted. With numbered lists, the deletion of any statement in the list deletes all the statements in the list

## Section 8.4: Controlling Telnet Access with IP Access Lists

Access into and out of the virtual terminal line (vty) ports of the Cisco IOS software can also be controlled by IP access lists. IOS uses vtys to represent a user who has Telnetted to a router, as well as for Telnet sessions a user of a router has created to other devices. You can use ACLs to limit the IP hosts that can Telnet into the router, and you can also limit the hosts to which a user of the router can Telnet.
Telnet into the router, and you can also limit the hosts to which a user of the router can Telnet.