CCNAX

Interconnecting Cisco Networking Devices: Accelerated

Student Guide

Volume 1

Version 3.0

Part Number: 97-3636-01

Americas Headquarters
Cisco Systems, Inc.
San Jose, CA

Asia Pacific Headquarters
Cisco Systems (USA) Pte. Ltd.
Singapore

Europe Headquarters
Cisco Systems International BV
Amsterdam,
The Netherlands

Cisco has more than 200 offices worldwide. Addresses, phone numbers, and fax numbers are listed on the Cisco Website at www.cisco.com/go/offices.

DISCLAIMER WARRANTY: THIS CONTENT IS BEING PROVIDED "AS IS" AND AS SUCH MAY INCLUDE TYPOGRAPHICAL, GRAPHICS, OR FORMATTING ERRORS. CISCO MAKES AND YOU RECEIVE NO WARRANTIES IN CONNECTION WITH THE CONTENT PROVIDED HEREUNDER, EXPRESS, IMPLIED, STATUTORY OR IN ANY OTHER PROVISION OF THIS CONTENT OR COMMUNICATION BETWEEN CISCO AND YOU. CISCO SPECIFICALLY DISCLAIMS ALL IMPLIED WARRANTIES, INCLUDING WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT AND FITNESS FOR A PARTICULAR PURPOSE, OR ARISING FROM A COURSE OF DEALING, USAGE OR TRADE PRACTICE. This learning product may contain early release content, and while Cisco believes it to be accurate, it falls subject to the disclaimer above.

Welcome Students

Students, this letter describes important course evaluation access information!

Welcome to Cisco Systems Learning. Through the Cisco Learning Partner Program, Cisco is committed to bringing you the highest-quality training in the industry. Cisco learning products are designed to advance your professional goals and give you the expertise that you need to build and maintain strategic networks.

Cisco relies on customer feedback to guide business decisions. Therefore, your valuable input will help shape future Cisco course curricula, products, and training offerings. Please complete a brief Cisco online course evaluation of your instructor and the course materials in this student kit. On the final day of class, your instructor will provide you with a URL, directing you to a short post-course evaluation. If there is no Internet access in the classroom, please complete the evaluation within the next 48 hours or as soon as you can access the web.

On behalf of Cisco, thank you for choosing Cisco Learning Partners for your Internet technology training.

Sincerely,
Cisco Systems Learning

Table of Contents

Course Introduction

Introduction

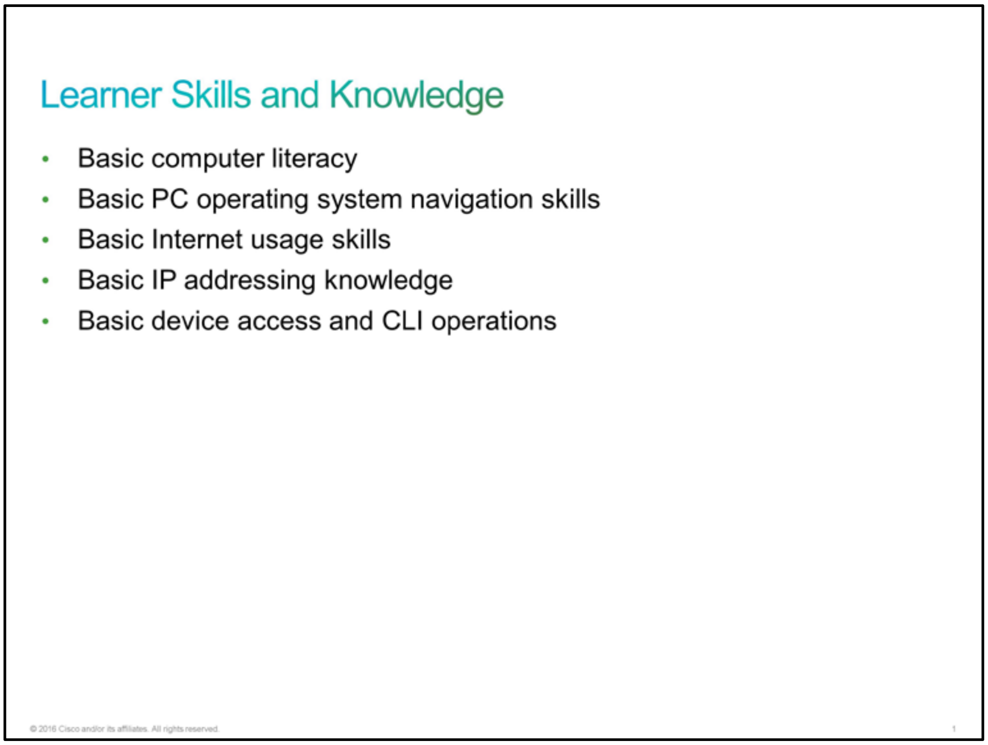
You may ask yourself, "What do I need to know to support my network?” The answer to this question depends on the size and complexity of your network. Fortunately, regardless of the size and complexity of the network, the starting point for learning to support a network is the same. This course is intended to be that starting point. It focuses on providing the skills and knowledge necessary to implement, configure, operate, and troubleshoot small and medium-sized networks. However, because of the accelerated nature of this course you will be expected to have foundation of basic network concepts and IPv4 address before starting this course.

Interconnecting Cisco Networking Devices: Accelerated (CCNAX) v3.0 is an instructor-led course presented by Cisco training partners to their customers. This extended-hours course retains all of the major content of *Interconnecting Cisco Networking Devices, Part 1* (ICND1) and *Interconnecting Cisco Networking Devices, Part 2* (ICND2), but merges the content into a single course.

Upon completing the CCNAX course, the student will have the same knowledge and skills as those obtained by attending the ICND1 and ICND2 courses independently.

Learner Skills and Knowledge

This subtopic lists the skills and knowledge that learners must possess to benefit fully from the course. The subtopic also includes recommended Cisco learning offerings that learners should first complete to benefit fully from this course.



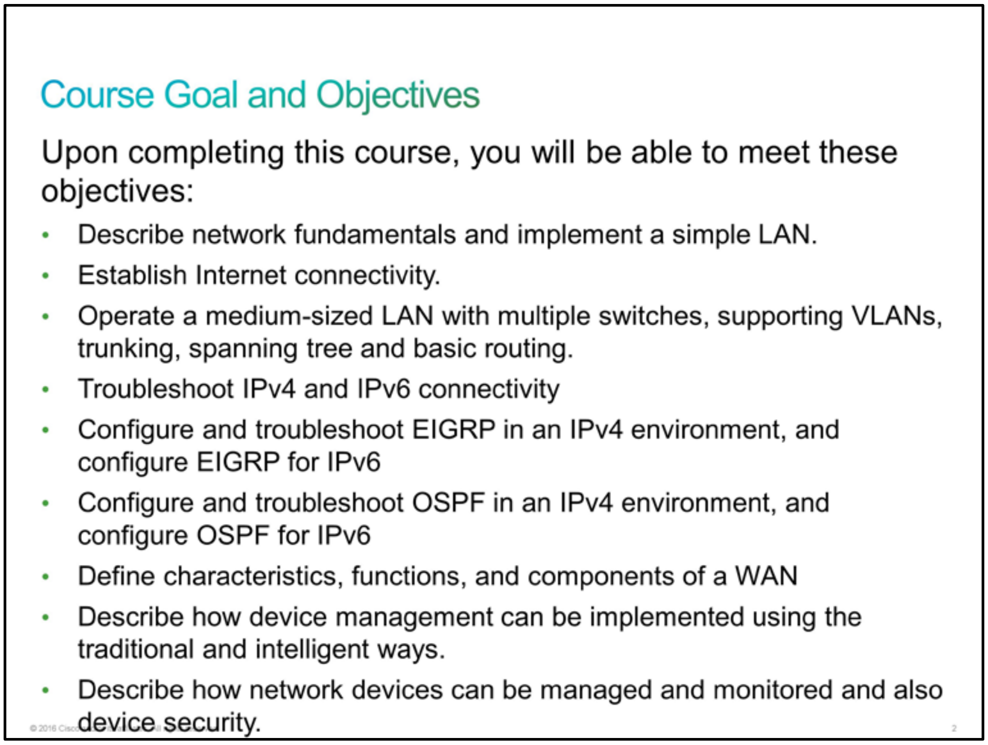
Learner Skills and Knowledge

- Basic computer literacy
- Basic PC operating system navigation skills
- Basic Internet usage skills
- Basic IP addressing knowledge
- Basic device access and CLI operations

1

Course Goal and Objectives

This topic describes the course goal and objectives.



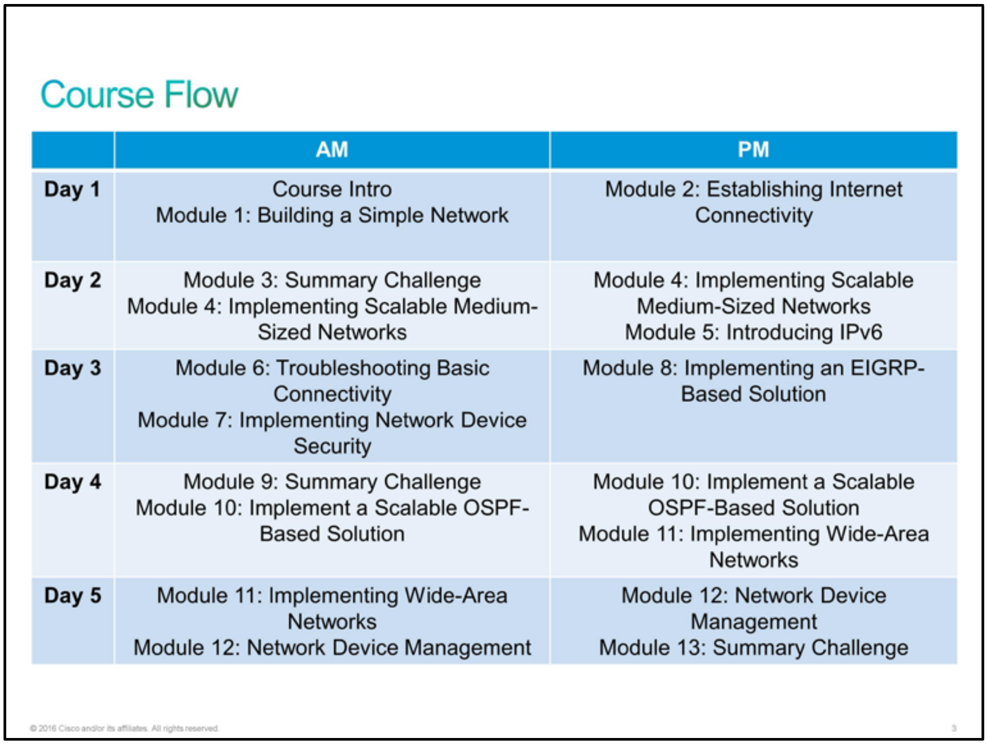
Course Goal and Objectives

Upon completing this course, you will be able to meet these objectives:

- Describe network fundamentals and implement a simple LAN.
- Establish Internet connectivity.
- Operate a medium-sized LAN with multiple switches, supporting VLANs, trunking, spanning tree and basic routing.
- Troubleshoot IPv4 and IPv6 connectivity
- Configure and troubleshoot EIGRP in an IPv4 environment, and configure EIGRP for IPv6
- Configure and troubleshoot OSPF in an IPv4 environment, and configure OSPF for IPv6
- Define characteristics, functions, and components of a WAN
- Describe how device management can be implemented using the traditional and intelligent ways.
- Describe how network devices can be managed and monitored and also device security.

Course Flow

This topic presents the suggested flow of the course materials.



Course Flow

	AM	PM
Day 1	Course Intro Module 1: Building a Simple Network	Module 2: Establishing Internet Connectivity
Day 2	Module 3: Summary Challenge Module 4: Implementing Scalable Medium-Sized Networks	Module 4: Implementing Scalable Medium-Sized Networks Module 5: Introducing IPv6
Day 3	Module 6: Troubleshooting Basic Connectivity Module 7: Implementing Network Device Security	Module 8: Implementing an EIGRP-Based Solution
Day 4	Module 9: Summary Challenge Module 10: Implement a Scalable OSPF-Based Solution	Module 10: Implement a Scalable OSPF-Based Solution Module 11: Implementing Wide-Area Networks
Day 5	Module 11: Implementing Wide-Area Networks Module 12: Network Device Management	Module 12: Network Device Management Module 13: Summary Challenge

© 2016 Cisco and/or its affiliates. All rights reserved.

3

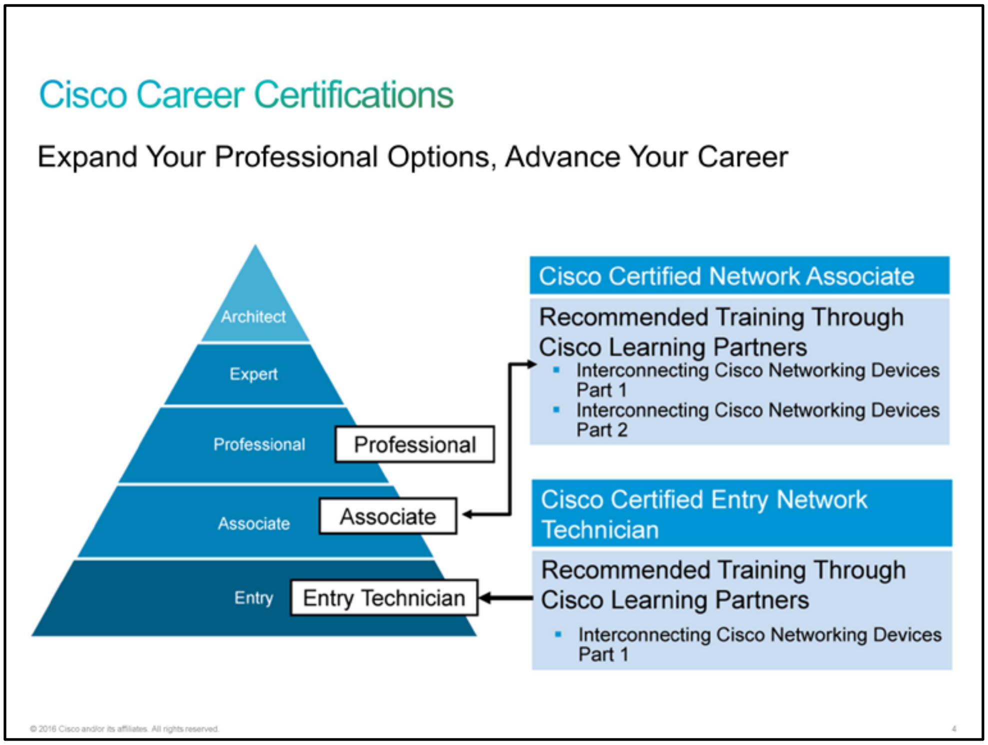
The schedule reflects the recommended structure for this course. This structure allows enough time for the instructor to present the course information and for you to work through the lab activities. The exact timing of the subject materials and labs depends on the pace of your specific class.

Your Training Curriculum

You are encouraged to join the Cisco Certification Community, a discussion forum open to anyone holding a valid Cisco Career Certification (such as Cisco CCIE®, CCNA®, CCDA®, CCNP®, CCDP®, CCIP®, CCNP® Security and CCNP® Voice, or CCSP™). It provides a gathering place for Cisco certified professionals to share questions, suggestions, and information about Cisco Career Certification programs and other certification-related topics. For more information, visit http://www.cisco.com/web/learning/training-index.html.

Cisco Career Certifications

Cisco provides three levels of general certifications for IT professionals with several different tracks to meet individual needs.



Cisco Career Certifications

Expand Your Professional Options, Advance Your Career

Cisco Certified Network Associate

Recommended Training Through Cisco Learning Partners

- Interconnecting Cisco Networking Devices Part 1
- Interconnecting Cisco Networking Devices Part 2

Cisco Certified Entry Network Technician

Recommended Training Through Cisco Learning Partners

- Interconnecting Cisco Networking Devices Part 1

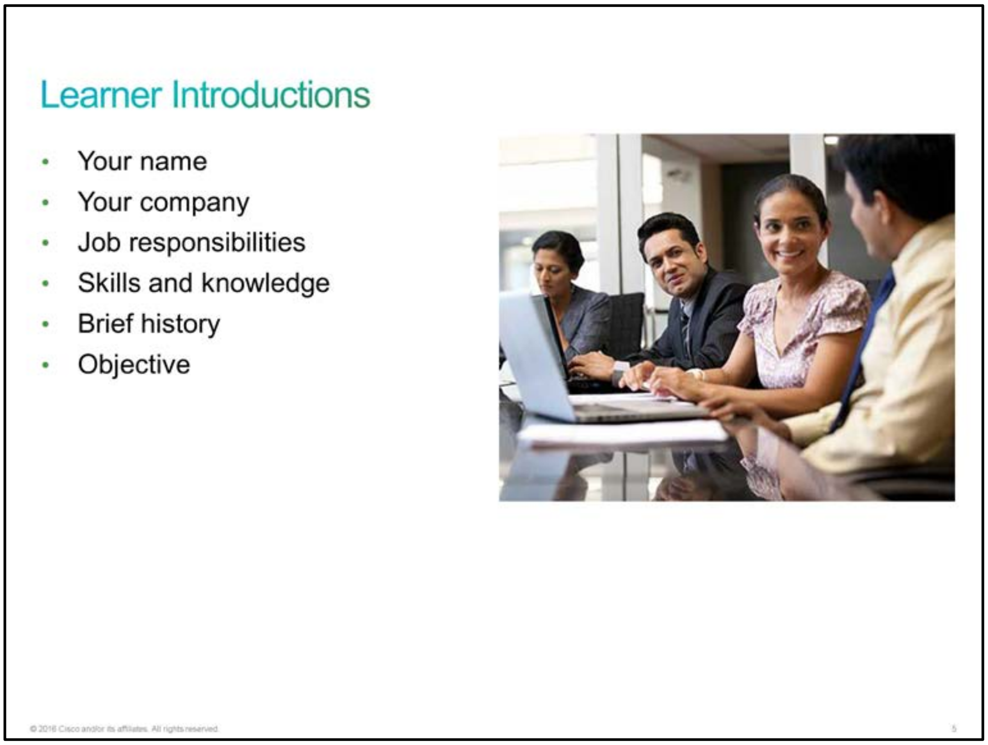
Cisco also provides focused certifications for designated areas such as cable communications and security. There are many paths to Cisco certification, but only one requirement—passing one or more exams demonstrating knowledge and skill. For details, go to http://www.cisco.com/web/learning/training-index.html.

Course-Specific Training Resources

These are URLs for course-specific training resources:

- https://learningnetwork.cisco.com/community/certifications/ccna
- https://learningnetwork.cisco.com/community/learning_center

Learner Introductions



Learner Introductions

- Your name
- Your company
- Job responsibilities
- Skills and knowledge
- Brief history
- Objective

Module 1: Building a Simple Network

Introduction

This module provides a high-level introduction to basic networking components and their functions. The need for a communication module is explained, followed by an introduction to the TCP/IP protocol stack. Cisco IOS Software is introduced, and its basic functions and features are described. Basic switch configuration is described, with configuration examples so that learners can perform switch startup and initial configuration in the associated lab. LANs are introduced, as well as the Ethernet standard. The operation and role of switches within LANs is described. Finally, the module provides an introduction to common switch media issues and lists recommended troubleshooting steps.

Lesson 1: Exploring the Functions of Networking

Introduction

You have succeeded in getting a job interview with a company named CCS. It is an IT services firm that specializes in providing managed IT and software services to law firms, among other companies. CCS provides networking design, implementation, and support services. You are interviewing for an entry-level network engineer position on the implementation team.

The hiring manager, Maya, explains the interview process with you. The first phase is a two-part phone interview. The first part focuses on your work history, and the second part is more technical. She wants to conduct the first part of the interview right away.

You can either schedule the second part later or move forward with it immediately. She suggests that before you decide, she can provide you with more information about the technical portion of the interview. You understand that if you do well, you will be scheduled for a more in-depth technical interview, so you want to make sure that you are fully prepared. She explains what the interview will require you to do:

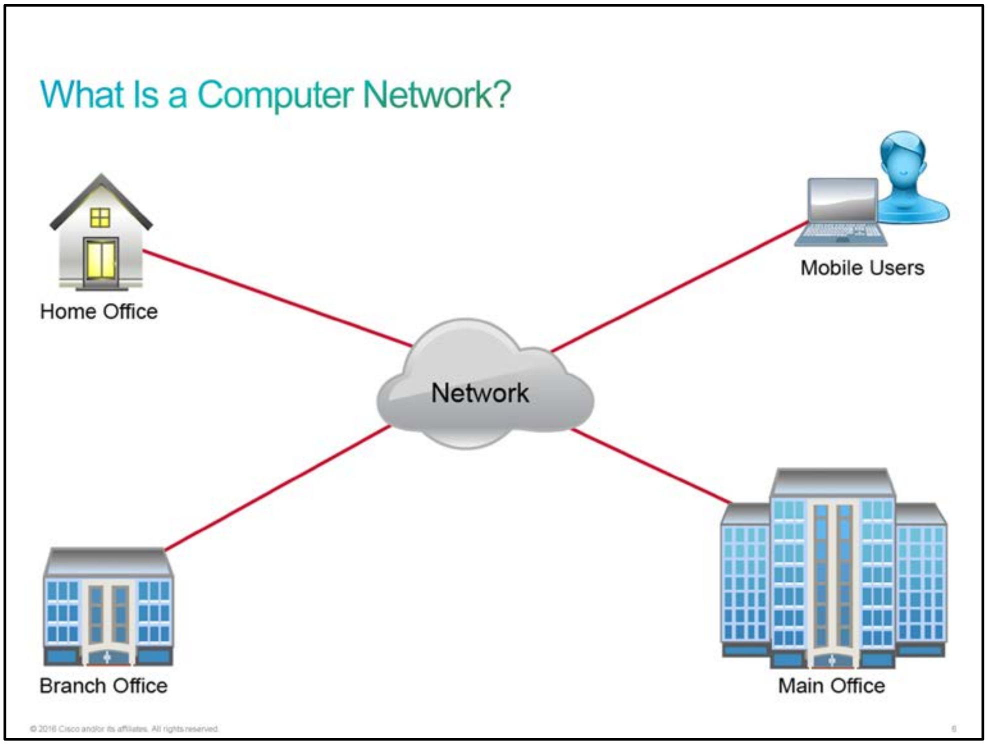
- Explain the functions, characteristics, and common components of a network
- Read a network diagram, including comparing and contrasting the logical and physical topologies
- Describe the impact of user applications on the network

Maya asks if you would like to continue the interview now or take some time to prepare.

What Is a Computer Network?

The term *network* is used in many different arenas. There are social networks, phone networks, television networks, and, of course, computer networks. In all cases, a network is a means of connecting various components together. A computer network connects PCs, printers, servers, phones, cameras, and other types of devices so that they can communicate with each other.

Networks carry data in many types of environments, including homes, small businesses, and large enterprises. Large enterprise networks may have a number of locations that need to communicate with each other. Based on where workers are situated, these locations are as follows:

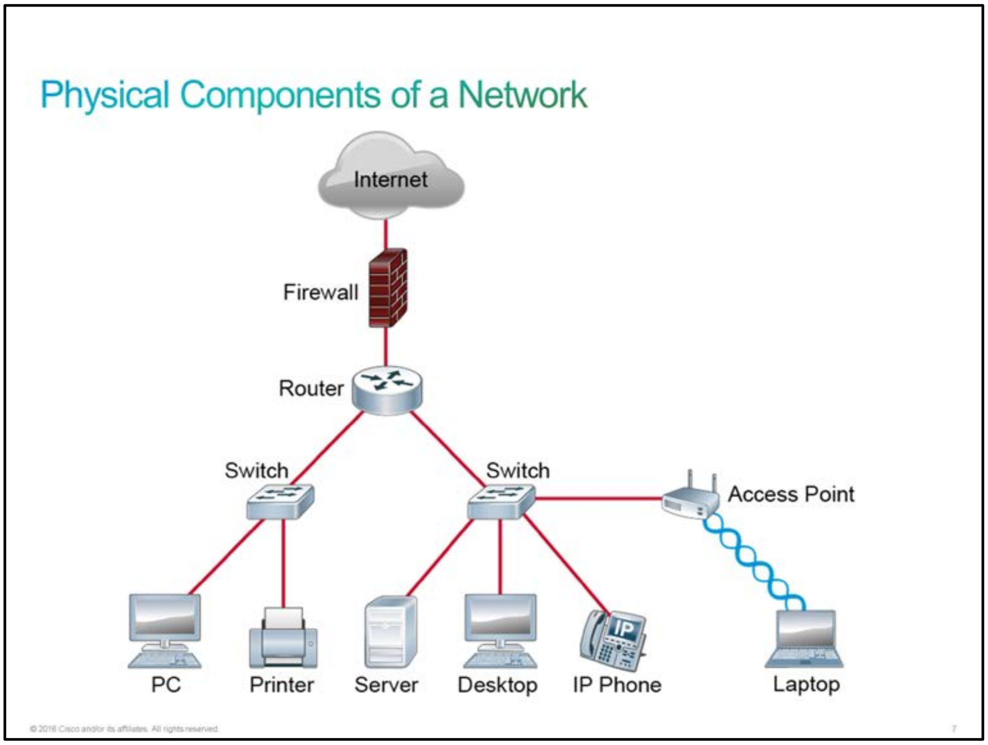

What Is a Computer Network?

- **Main office:** A main office is a site where everyone is connected via a network and where most corporate information is located. A main office can have hundreds or even thousands of people who depend on network access to do their jobs. A main office may use several connected networks that can span many floors in an office building or cover a campus that contains several buildings.
- **Remote locations:** Various remote access locations use networks to connect to the main office or to each other.
  - **Branch offices:** In branch offices, smaller groups of people work and communicate with each other via a network. Although some corporate information may be stored at a branch office, it is more likely that branch offices have local network resources, such as printers, but must access information directly from the main office.
  - **Home offices:** When individuals work from home, their location is called a home office. Home-office workers often require on-demand connections to the main office or branch offices to access information or to use network resources such as file servers.
  - **Mobile users:** Mobile users connect to the main office network while at the main office, at the branch office, or while traveling. The location of mobile users determines their network access requirements.

You can use a network in your home office to communicate via the Internet in order to locate information, place orders for merchandise, and send messages to friends. You can also have a small office that is set up with a network that connects other computers and printers in the office. Similarly, you can work in a large enterprise with many computers, printers, storage devices, and servers that are used to communicate and store information from many departments over large geographic areas.

Physical Components of a Network

Man different types of devices can be part of a network. A network can be as simple as two PCs that are connected by a wire or as complex as several thousands of devices that are connected through many different types of media.



Physical Components of a Network

© 2016 Cisco and/or its affiliates. All rights reserved.

7

Take a look at the five major components that you may find in a network:

- **Endpoints:** These elements include devices such as PCs, file servers, printers, tablets, sensors, cameras, and manufacturing robots.
- **Interconnections:** These are the components that connect the devices on the network. They provide a means for data to travel from one point to another in the network. This category includes the following components:
  - NICs translate computer data into a format that can be transmitted over the network. A NIC is the device on a PC into which you plug a network cable.
  - Network media, such as cables or wireless media, provide the means by which signals are transmitted from one network device to another.
  - Connectors provide connection points for the media. The most common type of connector is the plug on the end of a network cable that looks like an analog phone connector. This is called an RJ-45 connector. Compared to an analog phone connector the RJ-45 connector is slightly larger and contains eight wires instead of four.
- **Switches:** These are devices that endpoints such as PCs, file servers, printers, sensors, cameras, and manufacturing robots typically connect to. Usually, all the computers that are connected to the same switch can communicate directly with one another. They share what can be called a *common network*. If a computer wants to communicate with a device that is on a separate network, then it needs a device that is known as a *router*, which connects the two networks together.

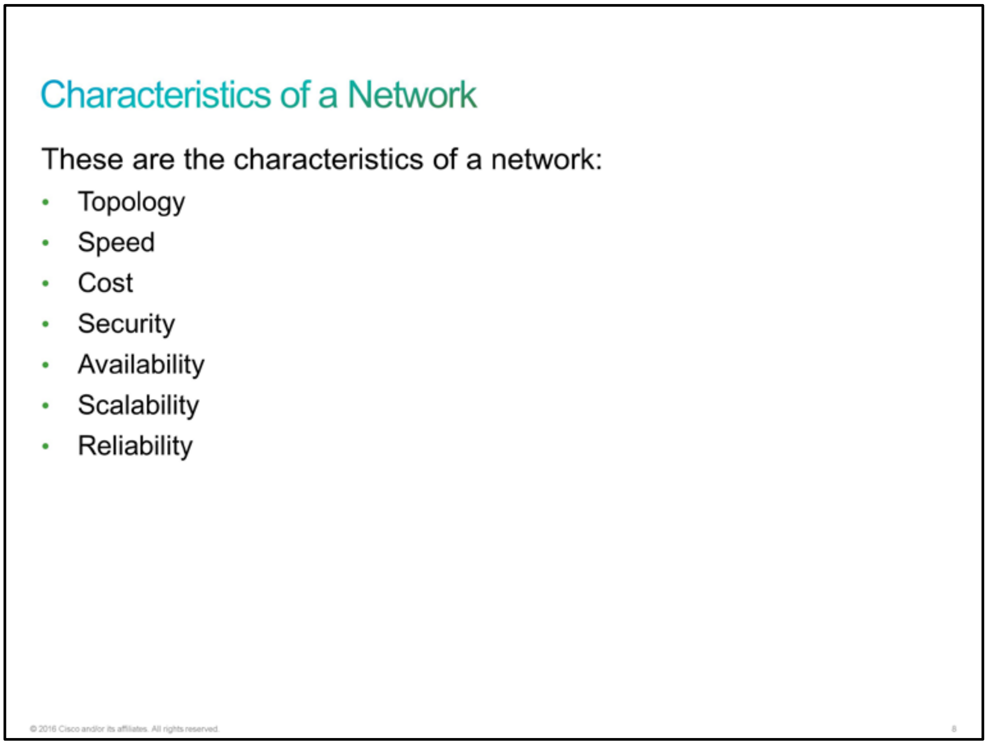
Note Before switches became affordable, many networks used devices that were called hubs. Hubs serve functions similar to switches, but they have a number of limitations. These limitations include lower speed and poor performance.

- **Routers:** These devices connect networks and intelligently choose the best paths between networks. Their main function is to route traffic from one network to another. For example, you need a router to connect your office network to the Internet. An analogy that may help you understand the function of switches and routers is that of a neighborhood. Think of the devices that are plugged into a switch as the houses on a city block. From a house on that block, you can go to any other house on the block without having to cross a street. However, if you want to travel to a house that is on another block, you must cross a street intersection. In networking, when you want to connect to a device on a different network from your own, you need to cross a router. Just like an intersection connects blocks in a neighborhood, a router connects computer networks.
- **WLAN devices:** These devices connect wireless devices such as computers, printers, and tablets to the network. The minimum requirement for wireless access to the network is a device with a WLAN NIC and a wireless AP that is connected to a traditional wired network.
- **Access Points or APs**: These devices allow wireless devices to connect to a wired network. An AP usually connects to a router as a standalone device, but it can also be an integral component of the router itself.
- **WLAN Controllers**: These are the devices that network administrators or network operations centers use to manage access points in large quantities. The WLAN controller automatically handles the configuration of wireless access points.
- **Firewalls:** These devices are network security systems that monitor and control the incoming and outgoing network traffic based on predetermined security rules. A firewall typically establishes a barrier between a trusted, secure internal network and another outside network, such as the Internet, that is assumed not to be secure or trusted.

Homes may have a single device that provides connectivity for wired and wireless devices as well as providing access to the Internet. You may be wondering which kind of device that is. It seems to have characteristics of a switch, a router, and a WLAN AP. The answer is that it is actually all three of those devices in a single package. It acts as a switch by providing physical ports to plug local devices into. It acts as a WLAN AP by allowing wireless devices to connect to it. And it acts as a router by connecting the local network to the Internet.

Characteristics of a Network

When you purchase a PC, one of the things that you examine is the specifications list. It tells you the important characteristics of the PC. As with the specifications list for a PC, the specific characteristics of a network help to describe its performance and structure. Understanding what each of the characteristics of a network means, enables you to better understand how the network is designed and what type of performance you should expect from it.



Characteristics of a Network

These are the characteristics of a network:

- Topology
- Speed
- Cost
- Security
- Availability
- Scalability
- Reliability

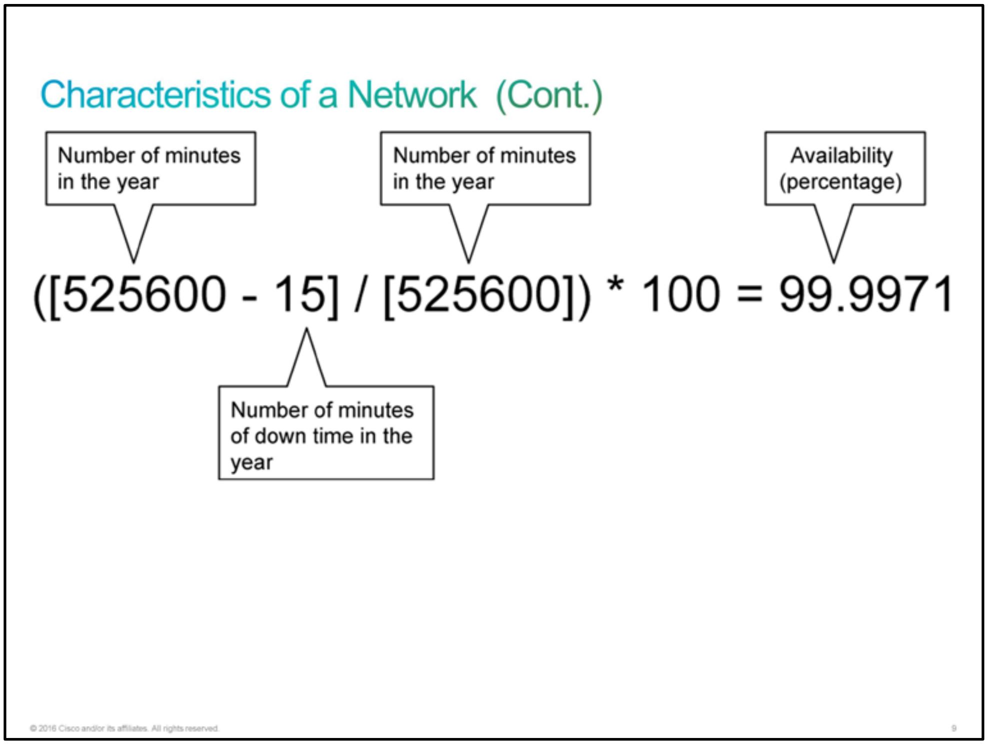
You can describe a network according to performance and structure:

- **Topology:** In networks, there are physical and logical topologies. The physical topology is the arrangement of cables, network devices, and end systems. The logical topology is the path over which the data is transferred in a network. For example, a physical topology describes how the network devices are actually interconnected with wires and cables. A logical topology describes how the network devices appear connected to network users.
- **Speed:** Speed is a measure of the data rate in bits per second of a given link in the network.
- **Cost:** Cost indicates the general expense for the purchasing of network components and the installation and maintenance of the network.
- **Security:** Security indicates how protected the network is, including the information that is transmitted over the network. The subject of security is important, and techniques and practices are constantly evolving. You should consider security whenever you take actions that affect the network.
- **Availability:** Availability is a measure of the probability that the network will be available for use when it is required. For networks that are meant to be used 24 hours per day, 7 days per week, 365 days per year, availability is calculated by dividing the time that it is actually available by the total time in a year and then multiplying by 100 to get a percentage.

For example, if a network is unavailable for 15 minutes per year because of network outages, you can calculate its percentage availability as follows:

([Number of minutes in a year – down time] / [number of minutes in a year]) * 100 = percentage availability

([525600 – 15] / [525600]) * 100 = 99.9971



Characteristics of a Network (Cont.)

Number of minutes in the year

Number of minutes in the year

Availability (percentage)

([525600 - 15] / [525600]) * 100 = 99.9971

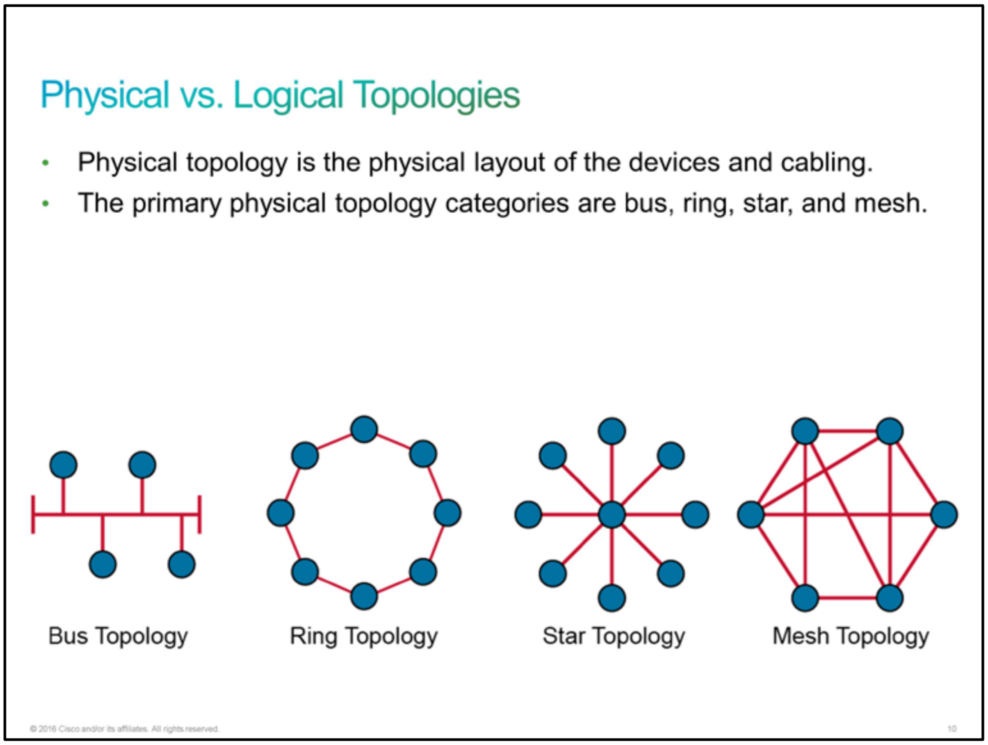
Number of minutes of down time in the year

- **Scalability:** Scalability indicates how easily the network can accommodate more users and data transmission requirements. If you design and optimize a network for only the current requirements, it can be very expensive and difficult to meet new needs when the network grows.
- **Reliability:** Reliability indicates the dependability of the components that make up the network, such as routers, switches, PCs, and servers. Reliability is often measured as a probability of failure or as MTBF.

These characteristics and attributes provide a means to compare various networking solutions.

Physical vs. Logical Topologies

Each type of network has both a physical and a logical topology. The physical topology of a network refers to the physical layout of the devices and cabling. You must match the appropriate physical topology to the type of cabling that you will install, such as twisted pair, coaxial, or fiber. Therefore, understanding the type of cabling that is used is important in understanding each type of physical topology. The logical topology defines the logical path on which data will travel from one point to another. First, take a look at some of the types of physical topologies that you may encounter.

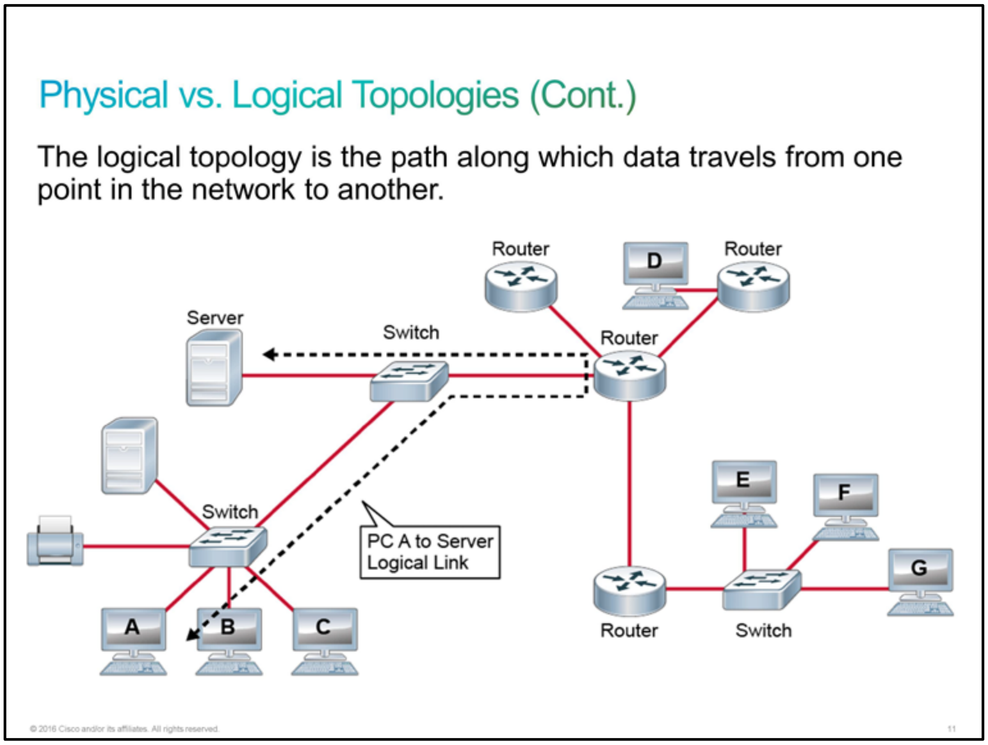


Physical vs. Logical Topologies

- Physical topology is the physical layout of the devices and cabling.
- The primary physical topology categories are bus, ring, star, and mesh.

10

- **Bus:** In a bus topology, every workstation is connected to the main cable. Therefore, each workstation is directly connected to every other workstation in the network. In early bus topologies, computers and other network devices were cabled together in a line using coaxial cable. Modern bus topologies establish the bus in a hardware device and connect the host devices to the bus using twisted-pair wiring.
- **Ring:** In a ring topology, computers and other network devices are cabled together in a way that the last device is connected to the first to form a circle or ring. Each device is connected to exactly two neighbors and has no direct connection to a third. Physical connection can be made using either coaxial or fiber wiring.
- **Star:** The most common physical topology is a star topology. In this topology, a central cabling device connects the computers and other network devices. This category includes star and extended star topologies. Physical connection is commonly made using twisted-pair wiring.
- **Mesh:** In a mesh topology, every network device is cabled together with many others. Redundant links increase reliability and self-healing. The physical connection is commonly made using fiber or twisted-pair wiring.



Physical vs. Logical Topologies (Cont.)

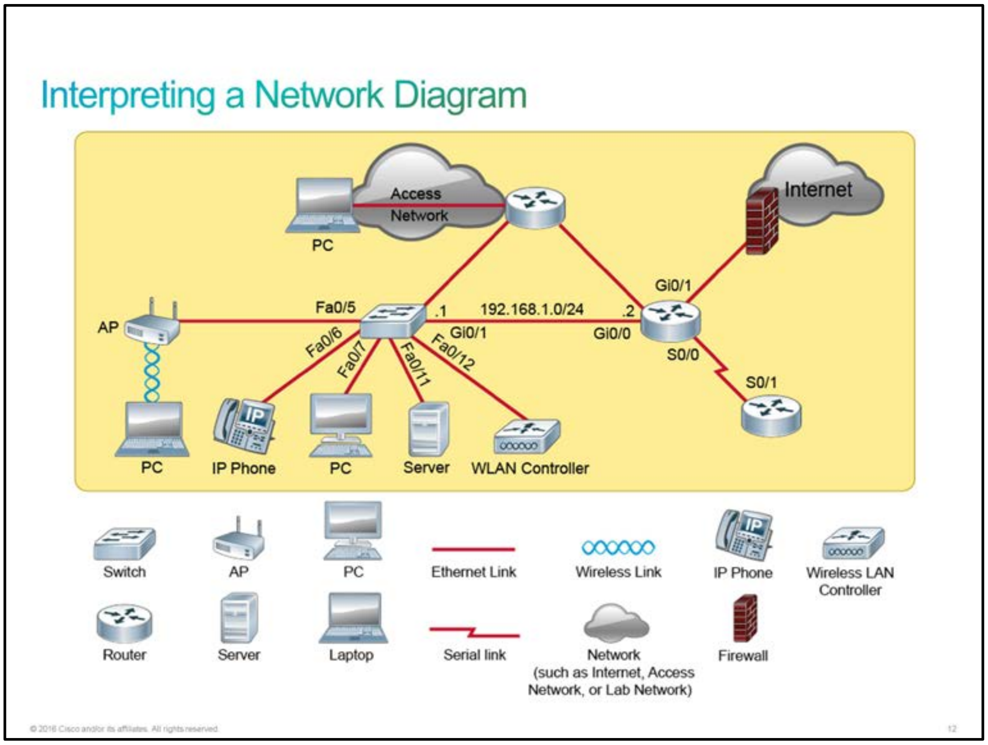
The logical topology is the path along which data travels from one point in the network to another.

PC A to Server Logical Link

It is possible for the logical and physical topology of a network to be of the same type. However, physical and logical topologies often differ. For example, an Ethernet hub is an example of a physical star topology with a logical bus topology. A physical star topology is by far the most common implementation of LANs today. Ethernet uses a logical bus topology in either a physical bus or a physical star topology.

Interpreting a Network Diagram

One of the most important tasks that you must complete when designing a network is to create a network diagram. It is basically a map of the network. It illustrates the logical representation of all devices in the network and clarifies how they are interconnected. In addition, a proper diagram provides information such as the interface IDs and network addressing. The figure shows a network diagram and Cisco icons that are commonly used to represent network devices in network diagrams.

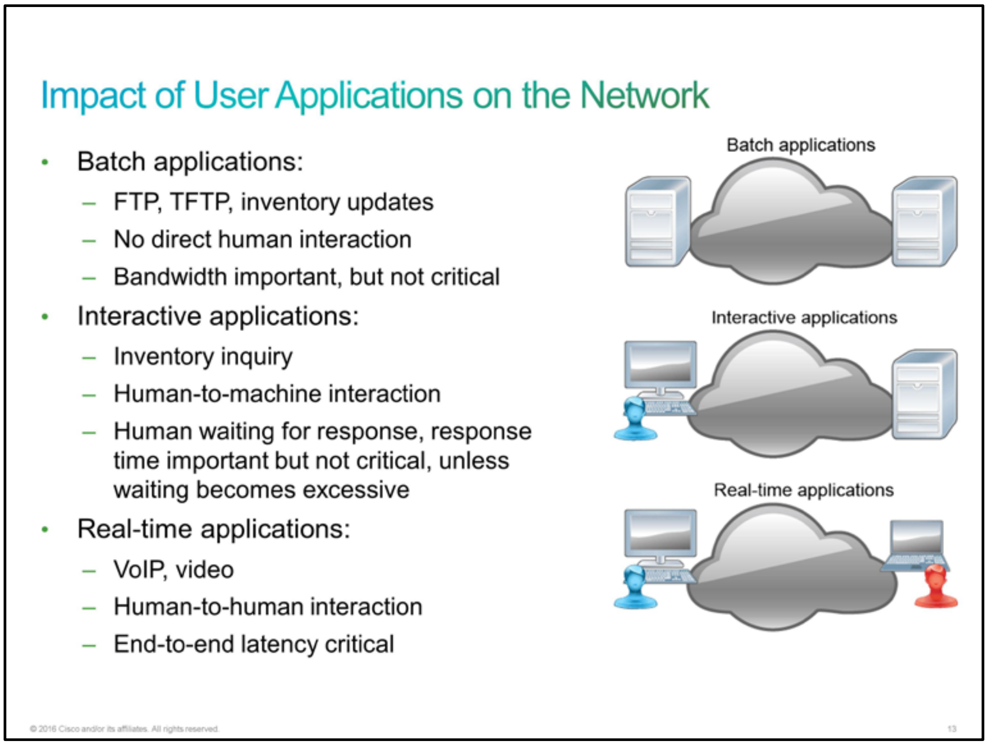


Interpreting a Network Diagram

Other information may be included in the network diagram as space allows. For example, it is common to identify the interface on a device in the S0/0/0 format for a serial interface. For Ethernet interfaces, Fa0/0 identifies a Fast Ethernet interface and Gi0/1 identifies a Gigabit Ethernet interface. It is also common to include the network address of the segment in the 192.168.1.0/24 format. In the example that is shown in the figure, 192.168.1.0 indicates the network address, /24 indicates the subnet mask, and .1 and .2 at the device ends indicate IP addresses of the interfaces (.1 corresponds to 192.168.1.1).

Impact of User Applications on the Network

Applications can affect network performance and, conversely, network performance can affect applications. Here you will learn about common interactions between user applications and the network.



Impact of User Applications on the Network

- Batch applications:
  - FTP, TFTP, inventory updates
  - No direct human interaction
  - Bandwidth important, but not critical
- Interactive applications:
  - Inventory inquiry
  - Human-to-machine interaction
  - Human waiting for response, response time important but not critical, unless waiting becomes excessive
- Real-time applications:
  - VoIP, video
  - Human-to-human interaction
  - End-to-end latency critical

13

Batch Applications

Applications such as FTP and TFTP are considered batch applications. Both are used to send and receive files. Typically, a user selects a group of files that need to be retrieved and then starts the transfer. Once the download starts, no additional human interaction is required. The amount of available bandwidth determines the speed at which the download occurs. While bandwidth is important for batch applications, it is not critical. Even with low bandwidth, the download is completed eventually.

Interactive Applications

Interactive applications are applications in which the user waits for a response—for example, inventory lookup. Because these applications require human interaction, response times are more important than for batch applications, but they are still not critical. The transaction will still be completed if the appropriate amount of bandwidth is not available, but it may take longer. Examples of the interactive applications are used in plant automation, physical security, sport, and retail.

Real-Time Applications

Real-time applications such as voice and video applications also involve human interaction. Because of the amount of information that is transmitted, bandwidth is critical. In addition, because these applications are time-critical, a delay on the network can cause a problem. Timely delivery of the data is crucial. It is also important that data is not lost during transmission, because real-time applications, unlike other applications, do not retransmit lost data. Therefore, sufficient bandwidth is mandatory and the quality of the transmission must be ensured by implementing QoS. QoS is a way of granting higher priority to certain types of data, such as VoIP.

Challenge

1. What are the varieties of remote access locations? (Choose three.)

   A. branch offices
   B. head office
   C. mobile users
   D. main office
   E. home offices
   F. Internet

2. What is a function of the WLAN Controller?

   A. to monitor and control the incoming and outgoing network traffic
   B. to automatically handle the configuration of wireless access point
   C. to allow wireless devices to connect to a wired network
   D. to connect networks and intelligently choose the best paths between networks

3. What is a function of the firewall?

   A. to automatically handle the configuration of wireless access points
   B. to allow wireless devices to connect to a wired network
   C. to connect networks and intelligently choose the best paths between networks
   D. to monitor and control the incoming and outgoing network traffic

4. What is the percentage availability of the network that is not available for 15 minutes each month?

   A. 99.9657%
   B. 0.9996%
   C. 99.8457%
   D. 0.0342%

5. What network characteristic indicates the general expense for the purchasing of network components and installation and maintenance?

   A. speed
   B. security
   C. cost
   D. availability

6. What network characteristic indicates the dependability of the components that make up the network?

   A. reliability
   B. scalability
   C. security
   D. availability

7. In what type of applications does the user need to wait for a response when performing actions such as inventory lookup or a database update?

   A. batch applications
   B. real-time applications
   C. interactive applications
   D. human-to-human applications

Answer Key

Challenge

1. A, C, E
2. B
3. D
4. A
5. C
6. A
7. C

Lesson 2: Understanding the Host-to-Host Communications Model

Introduction

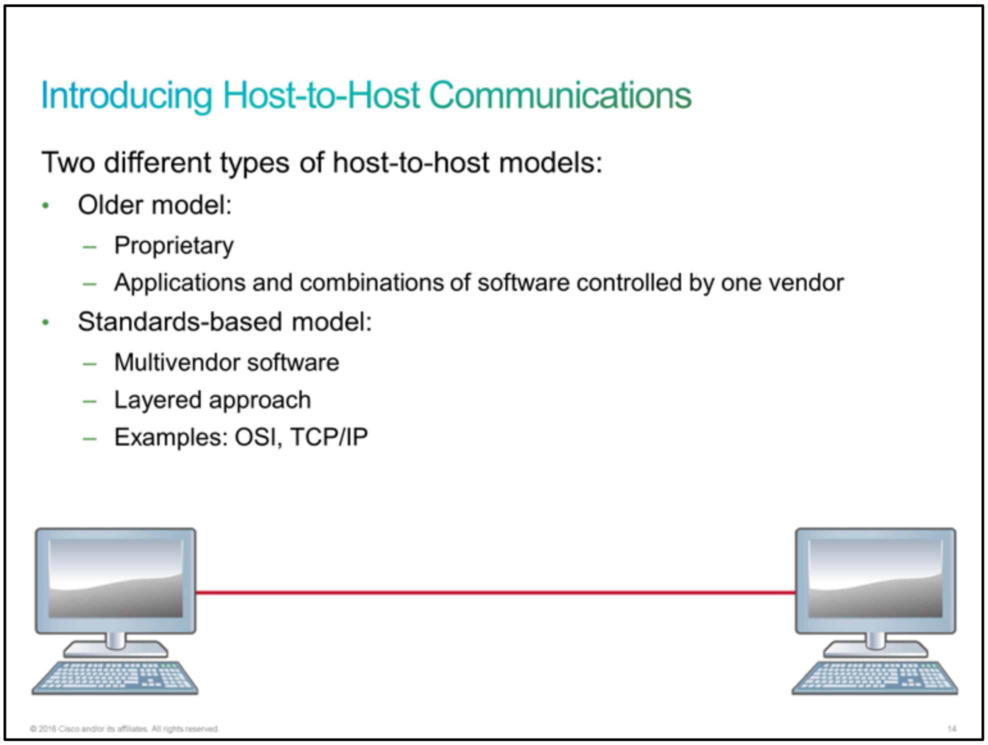
You just received a phone call from Maya. She has reviewed the technical portion of your phone interview with the lead engineer, Bob, and she was very impressed. Bob would like to interview you. Maya tells you that, in addition to the typical interview questions, Bob will also want to test your knowledge of topics that are more technical than the ones that were covered in the phone interview.

Maya says that Bob has time now for the interview. Otherwise, you can set up an appointment for later this week after you have had time to prepare.

You will need to demonstrate that you are familiar with the host-to-host communications model. Are you familiar with the TCP/IP model, its layers, and functions? Can you explain the terms bit, frame, packet, segment, and so on? You should also review the OSI reference model, as it is an alternative to the TCP/IP stack. Lastly, to show your understanding of the operation of the communications model, you should review the encapsulation and de-encapsulation processes.

Introducing Host-to-Host Communications

Host-to-host communications require a consistent model. The model addresses hardware, software, and data transmission.



Introducing Host-to-Host Communications

Two different types of host-to-host models:

- Older model:
  - Proprietary
  - Applications and combinations of software controlled by one vendor
- Standards-based model:
  - Multivendor software
  - Layered approach
  - Examples: OSI, TCP/IP

14

The network devices that people are most familiar with are called *end devices*. End devices form the interface between the human network and the underlying communications network. In the context of a network, end devices are called *hosts*. A host device is either the source or the destination of a message that is transmitted over the network. Communication begins with a message, or information, that must be sent from one device to another device. The message then flows through the network and arrives at the end device.

Successful communication between hosts on a network requires the interaction of many different protocols. A protocol is a set of rules that govern communications. Networking protocols describe the functions that occur during network communications. Protocols are implemented in the software and hardware of each host and other devices.

Original host-to-host communications models were proprietary. Each vendor controlled its own application and embedded communications software. An application that was written by one vendor would not function on a network that was developed by another vendor. In the computer industry, *proprietary* is the opposite of *open*. Proprietary means that one company or a small group of companies control all use of the technology. Open means that the use of the technology is available and is free to the public.

Business drivers and technology advances led to a multivendor solution. The first step was to separate application software from communications software, which allowed new communications technologies to be implemented without requiring new applications. However, it still requires a single-vendor solution for communications software and hardware.

It became apparent that a multivendor solution for communications software and hardware would require a layered approach with clearly defined rules for interlayer interaction. Within a layered model, various vendors provide solutions for separate layers. Hardware vendors could design hardware and software to support emerging physical-level technologies (that is, Ethernet, Token Ring, Frame Relay, and so on). Other vendors could write software that network operating systems that control host communications would use.

Standards-based layered models provide several benefits:

- Reducing complexity by breaking network communications into smaller, simpler parts
- Standardizing network components to allow different vendors to provide solutions for separate layers
- Facilitating modular engineering, allowing different types of network hardware and software to communicate with one another
- Ensuring interoperable technology and preventing changes in one layer from affecting the other layers
- Accelerating evolution, providing for effective updates and improvements to individual components without affecting other components or having to rewrite the entire protocol
- Simplifying teaching and learning

Examples of such standards-based models are TCP/IP and OSI.

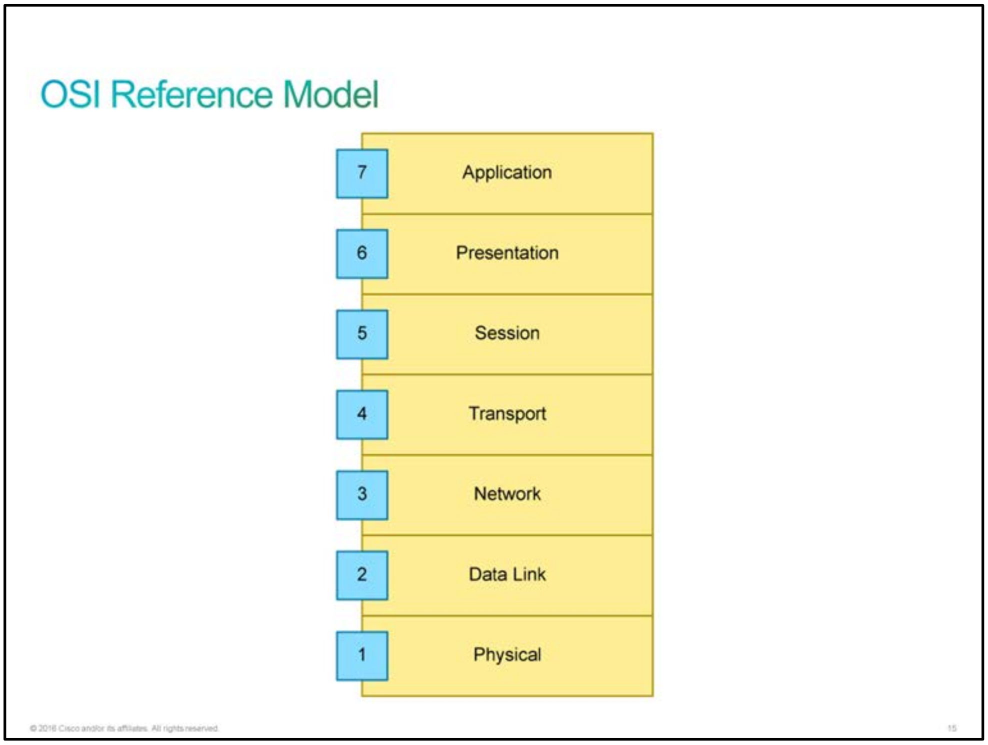
OSI Reference Model

To address the problem of networks being incompatible and unable to communicate with each other, the ISO researched different network schemes. As a result of this research, the ISO created a model to serve as a framework on which to build a suite of open systems protocols. The vision was that this set of protocols would be used to develop an international network that would not depend on proprietary systems.

Note ISO, the International Organization for Standardization, is an independent, nongovernmental organization. It is the world's largest developer of voluntary international standards. Those standards help businesses to increase productivity while minimizing errors and waste.

The OSIreference model provides a means of describing how data is transmitted over a network. The model addresses hardware, software, and data transmission.

As a reference, the OSI model provides an extensive list of functions and services that can occur at each layer. It also describes the interaction of each layer with the layers directly above and below it. More importantly, the OSI model facilitates an understanding of how information travels throughout the network. It provides vendors with a set of standards that ensure compatibility and interoperability between the various types of network technologies that companies produce around the world. You also use the OSI model for data network design, operation specifications, and troubleshooting.



OSI Reference Model

7	Application
6	Presentation
5	Session
4	Transport
3	Network
2	Data Link
1	Physical

The OSI reference model separates network functions into seven categories. This separation of networking functions is called layering. The OSI reference model has seven numbered layers, each one illustrating a particular network function.

Physical Layer (Layer 1)

The physical layer defines certain specifications (for example, electrical, mechanical, procedural, and functional). These specifications are needed for activating, maintaining, and deactivating the physical link between end devices. This physical link enables bit transmission between end devices. Physical layer specifications are defining characteristics—for example, voltage levels, timing of voltage changes, physical data rates, maximum transmission distances, physical connectors, and other similar attributes.

Data Link Layer (Layer 2)

The data link layer defines how data is formatted for transmission and how access to physical media is controlled. This layer also typically includes error detection and correction to ensure a reliable delivery of the data.

Network Layer (Layer 3)

The network layer provides connectivity and path selection between two host systems that may be located on geographically separated networks. With the growth of the Internet, the number of users that access information from sites around the world has increased. The network layer is the layer that manages the connectivity of these users by providing logical addressing.

Transport Layer (Layer 4)

The transport layer defines services to segment, transfer, and reassemble the data for individual communications between the end devices. For example, business users in large corporations often transfer large files from field locations to a corporate site. Reliable delivery of the files is important, so the transport layer breaks down large files into smaller segments that are less likely to incur transmission problems. TCP and UDP are the protocols that operate at this layer. TCP is used when data loss cannot be tolerated (file transfer), and UDP is used when some data loss is acceptable (when speed is more important than accuracy, for example in video streaming).

Session Layer (Layer 5)

The session layer establishes, manages, and terminates sessions between two communicating hosts. The session layer also synchronizes dialog between the presentation layers of the two hosts and manages their data exchange. For example, web servers have many users, so there are many communication processes open at a given time. As a result, it is important to keep track of which user communicates on which path. In addition to session regulation, the session layer offers provisions for efficient data transfer, CoS, and exception reporting of session layer, presentation layer, and application layer problems.

Presentation Layer (Layer 6)

The presentation layer ensures that the information that is sent at the application layer of one system is readable by the application layer of another system. For example, when a PC program is communicating with another computer, each computer might be using a different encoding scheme. The presentation layer has to translate among multiple data formats using a common format.

Application Layer (Layer 7)

The application layer is the OSI layer that is closest to the user. This layer provides network services to the applications of the user, such as email, file transfer, and terminal emulation. The application layer differs from the other layers in that it does not provide services to any other OSI layer. It provides services only to applications outside the OSI model. The application layer establishes the availability of intended communication partners. It then synchronizes and establishes agreement on procedures for error recovery and control of data integrity.

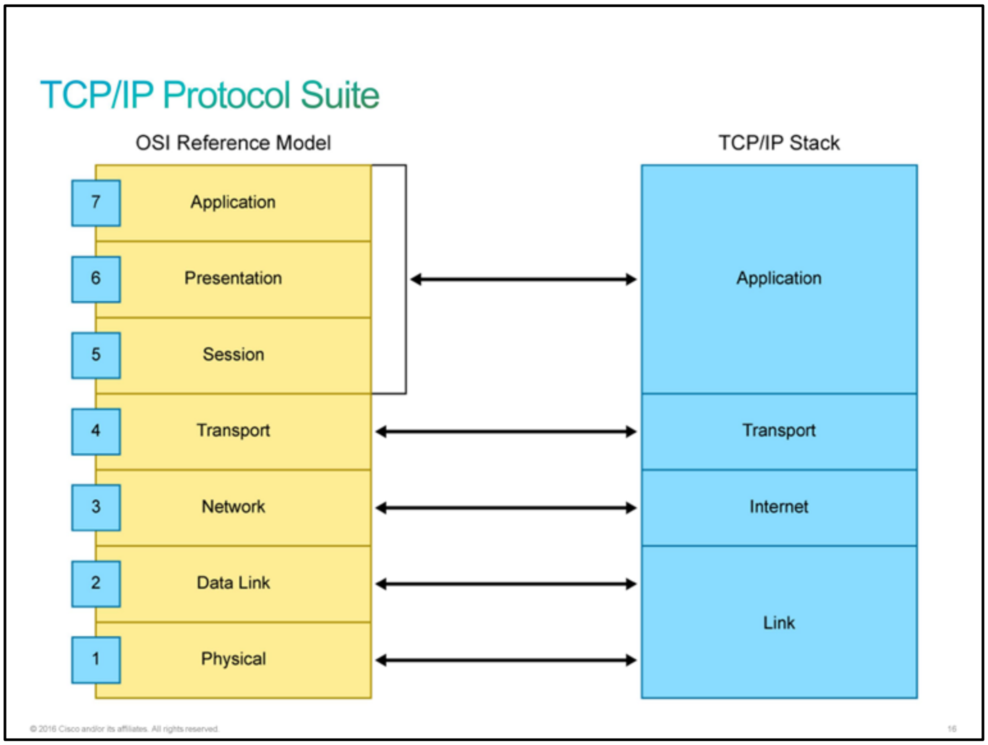
TCP/IP Protocol Suite

TCP/IP stands for Transmission Control Protocol/Internet Protocol. It defines how devices should be connected over the Internet, and how data should be transmitted between those devices.

TCP/IP is actually two protocols, but they are used together so often that many people think of them as a single protocol. TCP operates at Layer 4 and is responsible for making sure that the data that the source device sends arrives at its destination. IP operates at Layer 3 and is responsible for the transmission of data. It does not do any error correction itself.

The TCP/IP reference model is similar to the OSImodel. It also separates data communication into layers. However, it predates the OSI model and consists of only four layers. The TCP/IP model provides a common reference for maintaining consistency within all types of network protocols and services. It is not intended to be an implementation specification or to provide sufficient detail to precisely define the services of the network architecture. The primary purpose of this reference model is to help you understand the functions and processes that are involved in data communication.

Note Although this course refers to the TCP/IP stack, it has become common in the industry to shorten this term to "IP stack."



TCP/IP Protocol Suite

© 2016 Cisco and/or its affiliates. All rights reserved.

16

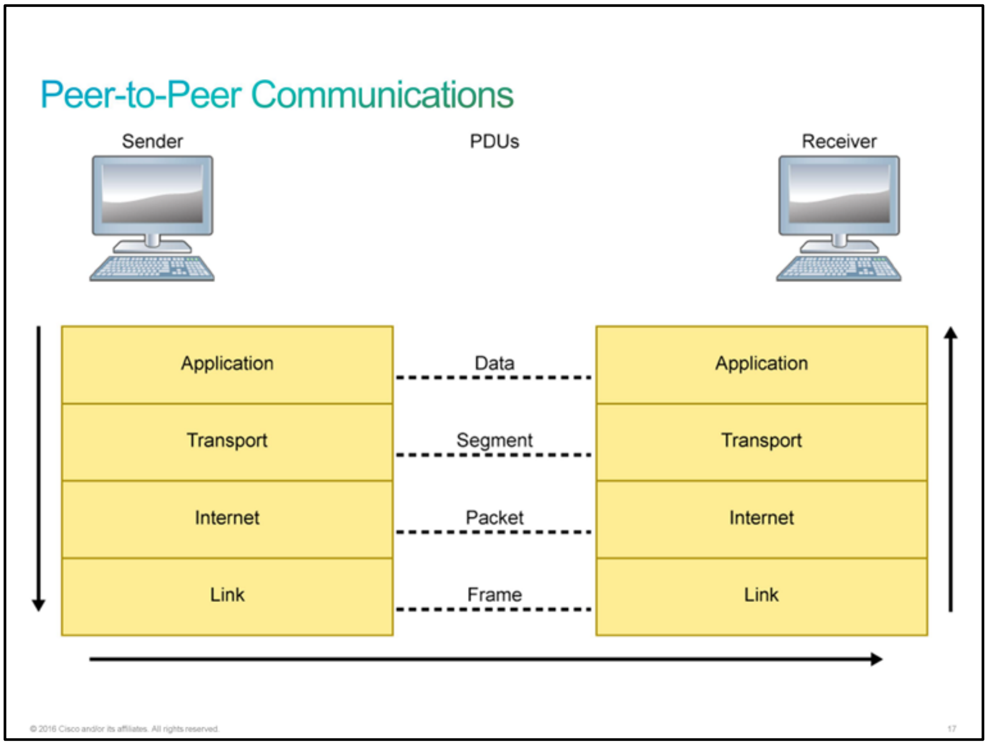
Take a look at the four layers of the TCP/IP model:

- **Link layer:** This layer is also known as the network access layer and is the equivalent of both the physical and data link layers of the OSI model. It deals with components such as cables, connectors, and network cards, like OSI Layer 1. Like Layer 2 of the OSI model, the link layer of the TCP/IP model is concerned with hardware addresses.
- **Internet layer—aligns directly with Layer 3 of the OSI model:** You may also know this layer as the Internet layer. It routes data from the source to the destination by defining the packet and the addressing scheme, moving data between the link and transport layers, routing packets of data to remote hosts, and performing fragmentation and reassembly of data packets. This is the layer where IP operates.
- **Transport layer—directly aligned with Layer 4 of the OSI model:** This layer is the core of the TCP/IP architecture. It is the layer where TCP and UDP operate. This layer provides communication services directly to the application processes that are running on network hosts.
- **Application layer—corresponds to Layers 5, 6, and 7 of the OSI model:** It provides applications for file transfer, network troubleshooting, and Internet activities. It also supports network APIs, which allow programs that have been created for a particular operating system to access the network.

Peer-to-Peer Communications

The term *peer* means the equal of a person or object. Therefore, peer-to-peer communication means communications between equals. In other words, each layer must be able to communicate with its equal (peer) on the other side. So each source layer must be able to communicate with its corresponding destination layer. One way that this understanding is accomplished is by packing data in a format that the peer layer will understand.

During the process of peer-to-peer communication, the protocols at each layer exchange packets between peer layers. These packets of information are called PDUs. At each stage of the process, a PDU has a different name to reflect its new appearance.


Peer-to-Peer Communications

Although there is no universal naming convention for PDUs, here, the PDUs are named as follows:

- **Data:** The general term for the PDU used at the application layer
- **Segment:** A transport layer PDU
- **Packet:** An Internet layer PDU
- **Frame:** A link layer PDU

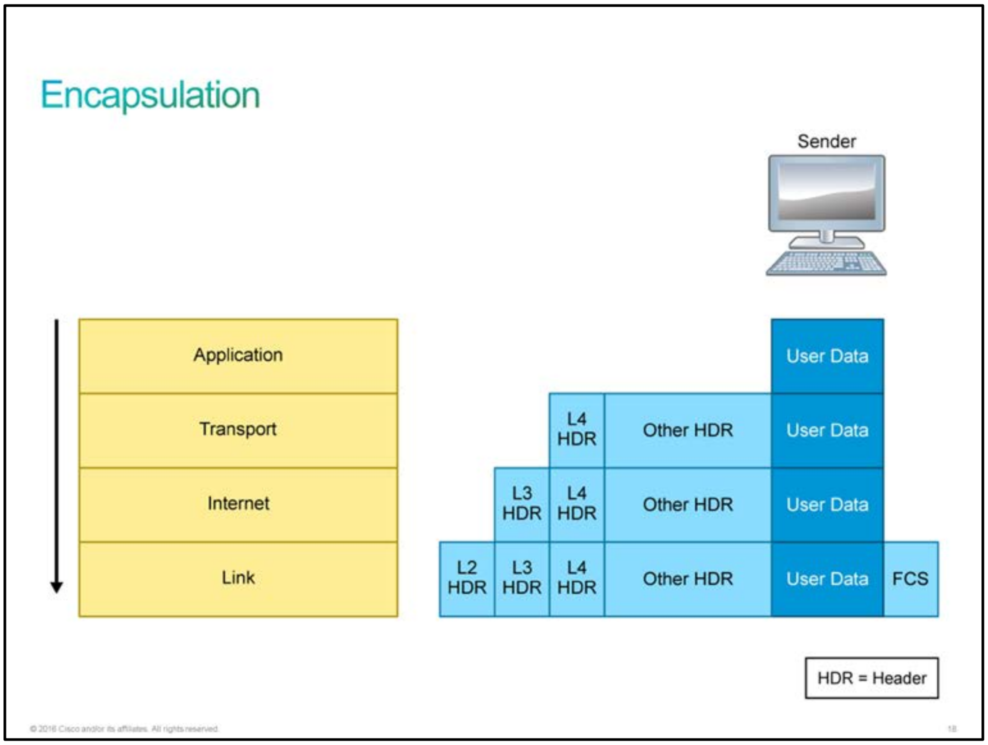
To look into PDUs from peer-to-peer communication, you can use a program for analyzing packets—for example, Wireshark. Wireshark is a free and open-source packet analyzer. It is used for network troubleshooting, analysis, software and communications protocol development, and education.

Encapsulation and De-Encapsulation

Information that is transmitted over a network must undergo a process of conversion at the sending and receiving ends of the communication. The conversion process is known as encapsulation and de-encapsulation of data.

Encapsulation

Have you ever opened a very large present and found a smaller box inside? And then an even smaller box inside that one, until you get to the smallest box and, finally, to your present? Encapsulation operates similarly in the OSI model. The application layer receives the user data and adds a header before sending it to the presentation layer, like putting it into a box. The presentation layer then adds its own header before sending to the session layer, placing the small box into a larger one. This process continues at each layer. At the data link layer, a trailer is also added. The data is then sent across the physical layer. When it reaches the destination, each layer removes the header that its peer layer added, equivalent to unpacking the boxes. The layer reads the information in the header to determine what to do with the data and then hands the PDU up to the next layer for processing.

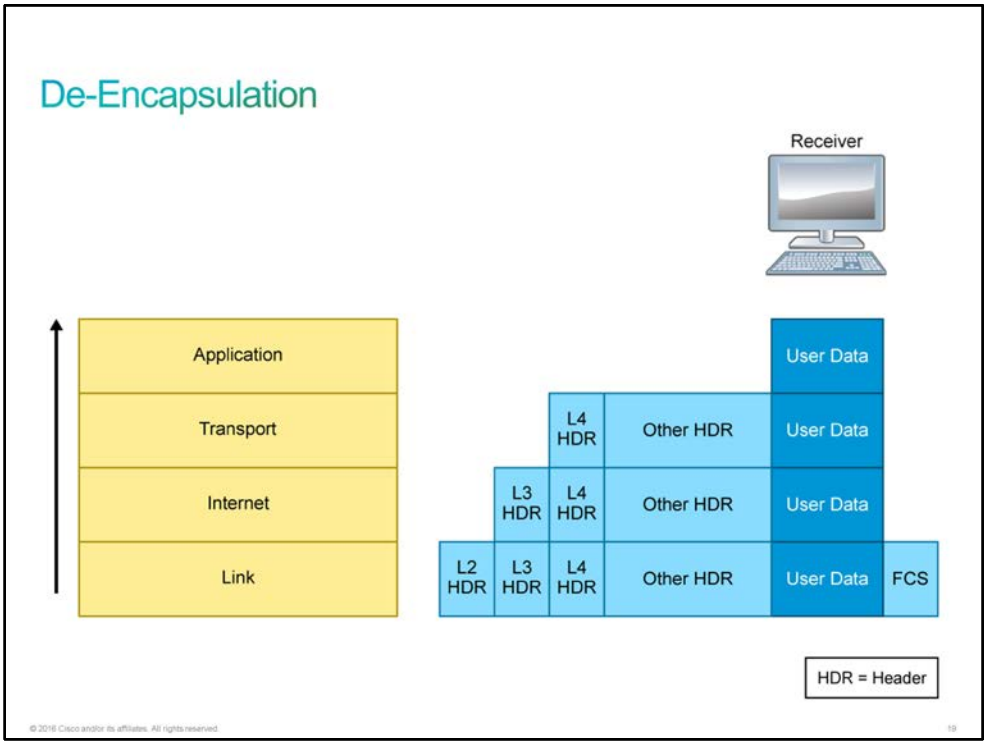

Encapsulation

The information that is sent on a network is referred to as data or data packets. As application data is passed down the protocol stack on its way to be transmitted across the network media, various protocols add information to it at each level. This process is commonly known as the encapsulation process. Each layer adds a header (and a trailer, if applicable) to the data before passing it down to a lower layer. The headers and trailers contain control information for the network devices and the receiver. This information ensures that the data is properly delivered and that the receiver can correctly interpret the data.

The figure shows how encapsulation occurs. It shows how data travels through the layers. The data is encapsulated as follows:

1. The user data is sent from an application to the application layer.
2. The transport layer adds the transport layer header (Layer 4 header) to the data. The Layer 4 header and the previous data become the data that is passed down to the Internet layer.
3. The Internet layer adds the Internet layer header (Layer 3 header) to the data. The Layer 3 header and the previous data become the data that is passed down to the link layer.
4. The link layer adds the Layer 2 header and trailer to the data. A Layer 2 trailer is usually the FCS, which the receiver uses to detect whether the data is in error.

De-Encapsulation



When receiving messages on a network, the protocol stack on a host operates from the bottom to the top. The process of encapsulation is reversed at the receiving host. When the data reaches the destination, each layer removes the header that its peer layer added, equivalent to unpacking the boxes. The layer reads the information in the header to determine what to do with the data and then hands the PDU up to the next layer for processing. The data is de-encapsulated as it moves up the stack toward the end-user application.

When the remote device receives a sequence of bits, the data is de-encapsulated as follows:

1. The link layer checks the trailer (the FCS) to see if the data is in error. The frame may be discarded or the link layer may ask for the data to be retransmitted.
2. If the data is not in error, the link layer reads and interprets the control information in the Layer 2 header.
3. The link layer strips the Layer 2 header and trailer and then passes the remaining data up to the Internet layer, which is based on the control information in the link layer header.

Each subsequent layer performs a similar de-encapsulation process.

The de-encapsulation process is like reading the address on a package to see if it is addressed to you and then removing the contents of the package.

Challenge

1. Which OSI layer defines services to segment the data?

   A. presentation layer
   B. session layer
   C. transport layer
   D. network layer

2. Which OSI layer manages sessions between two communicating hosts?

   A. transport layer
   B. presentation layer
   C. session layer
   D. network layer

3. Which OSI layer ensures that the application layer of the receiving system will be able to read the information that the application layer of another system has sent?

   A. transport layer
   B. presentation layer
   C. session layer
   D. network layer

4. To which OSI model layer or layers is the TCP/IP model transport layer aligned?

   A. network
   B. transport
   C. session
   D. session, presentation, and application

5. To which OSI model layer or layers does the TCP/IP model application layer correspond?

   A. network
   B. transport
   C. session
   D. session, presentation, and application

6. Match the PDUs with the correct descriptions.

frame	the general term for the PDU used at the application layer
packet	a transport layer PDU
segment	an Internet layer PDU
data	a link layer PDU

7. Align TCP/IP layers with the corresponding PDUs.

packet	application layer
frame	transport layer
data	Internet layer
segment	link layer

Answer Key

Challenge

1. C
2. C
3. B
4. B
5. D
6.

data	the general term for the PDU used at the application layer
segment	a transport layer PDU
packet	an Internet layer PDU
frame	a link layer PDU

7.

data	application layer
segment	transport layer
packet	Internet layer
frame	link layer

Lesson 3: Introducing LANs

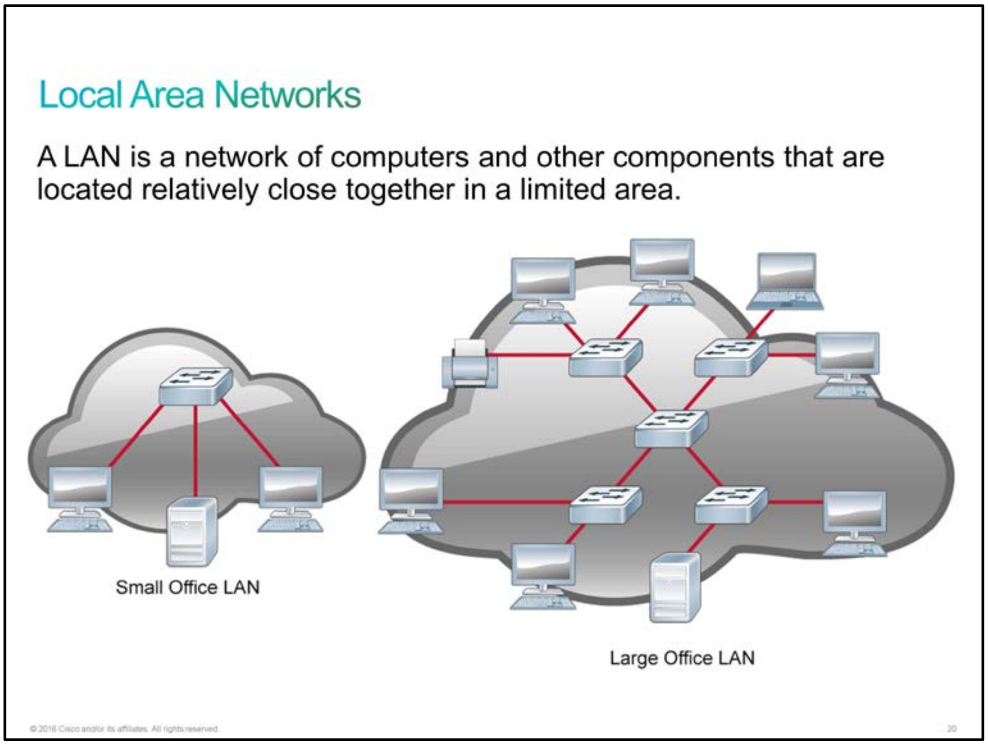
Introduction

Your first interview with Bob was a success. Maya calls the following afternoon to set up a second interview and provide you information about what you will need to know. She says that this time, you will be asked about LANs and their components, especially switches. She asks if you would like to come in for the interview now, because Bob is available all afternoon.

You need to know what a LAN is and be able to identify LAN components. You should also be able to explain why you need switches and list important switch features and characteristics.

Local Area Networks

The LAN emerged to serve the needs of a disconnected collective. While there have been many types of LAN transports, Ethernet became the favorite of businesses starting in the early 1990s. Since its introduction, Ethernet bandwidth has scaled from the original shared-media 10 Mb/s to 100 Gb/s in Cisco Nexus 7000 Series Switches for the data center.



Local Area Networks

A LAN is a network of computers and other components that are located relatively close together in a limited area.

© 2016 Cisco and/or its affiliates. All rights reserved. 20

LANs can vary widely in size. A LAN may consist of only two computers in a home office or small business, or it may include hundreds of computers in a large corporate office or multiple buildings.

The defining characteristics of LANs, in contrast to WANs, include their typically higher data transfer rates, smaller geographic area, and the lack of need for leased telecommunication lines.

Examples: A Small-Office LAN and a Large-Office LAN

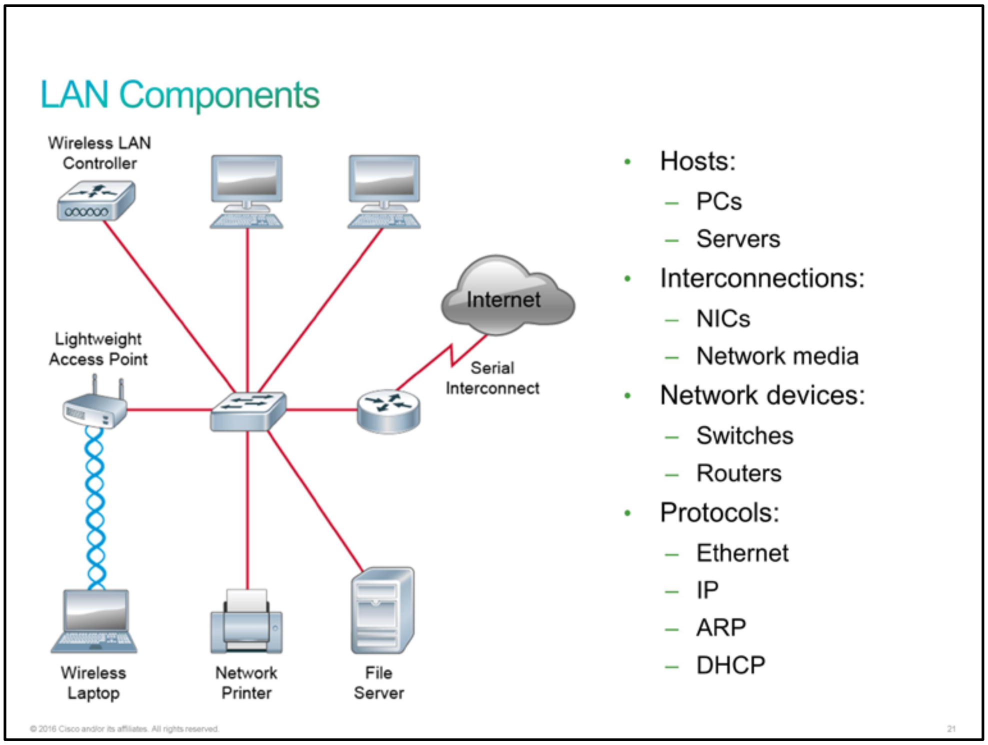
A small home business or a small-office environment can use a small LAN to connect two or more computers and to connect the computers to one or more shared peripheral devices, such as printers.

A large corporate office can use multiple LANs to accommodate hundreds of computers and shared peripheral devices, spanning many floors in an office complex.

LAN Components

On the first LANs, devices with Ethernet connectivity were mostly limited to PCs, file servers, print servers, hubs, and bridges.

Today, a typical small office will include a router for Internet connectivity, wireless capabilities, network printers, file servers, PCs, and laptops.



LAN Components

- Hosts:
  - PCs
  - Servers
- Interconnections:
  - NICs
  - Network media
- Network devices:
  - Switches
  - Routers
- Protocols:
  - Ethernet
  - IP
  - ARP
  - DHCP

21

Regardless of its size, a LAN requires these fundamental components for its operation:

- **Hosts:** Hosts include any device that can send or receive data on the LAN.
- **Interconnections:** Interconnections allow data to travel from one point to another in the network. Interconnections include these components:
  - **NICs:** NICs translate the data that is produced by the computer into a format that can be transmitted over the LAN. NICs connect a station to the LAN over copper cable, fiber-optic cable, or wireless communication.
  - **Network media:** In traditional LANs, data was transmitted mostly over copper and fiber-optic cables. Modern LANs (even small home LANs) generally include wireless connectivity.
- **Network devices:** Network devices, like switches and routers, are responsible for data delivery between hosts.
- **Protocols:** Protocols are rules that govern how data is transmitted over a LAN. Here are some commonly used LAN protocols:
  - Ethernet protocols (IEEE 802.2 and IEEE 802.3)
  - IP
  - TCP
  - UDP

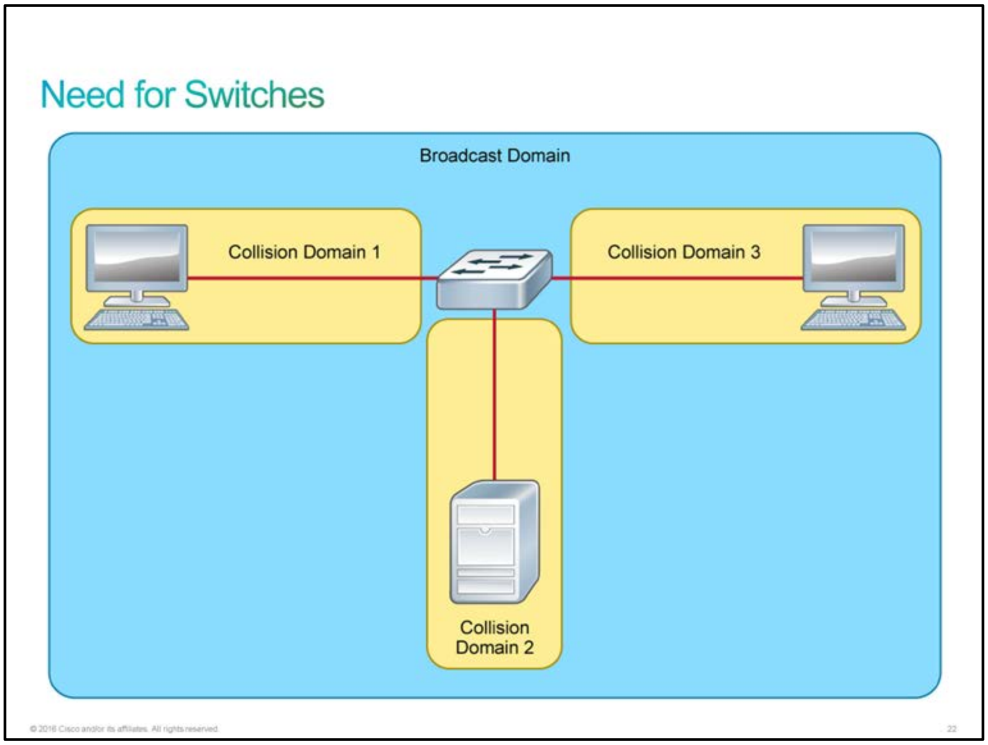
- ARP and RARP
- CIFS
- DHCP

Need for Switches

When you connect three or more devices together, you need a dedicated network device to enable communication between these hosts.

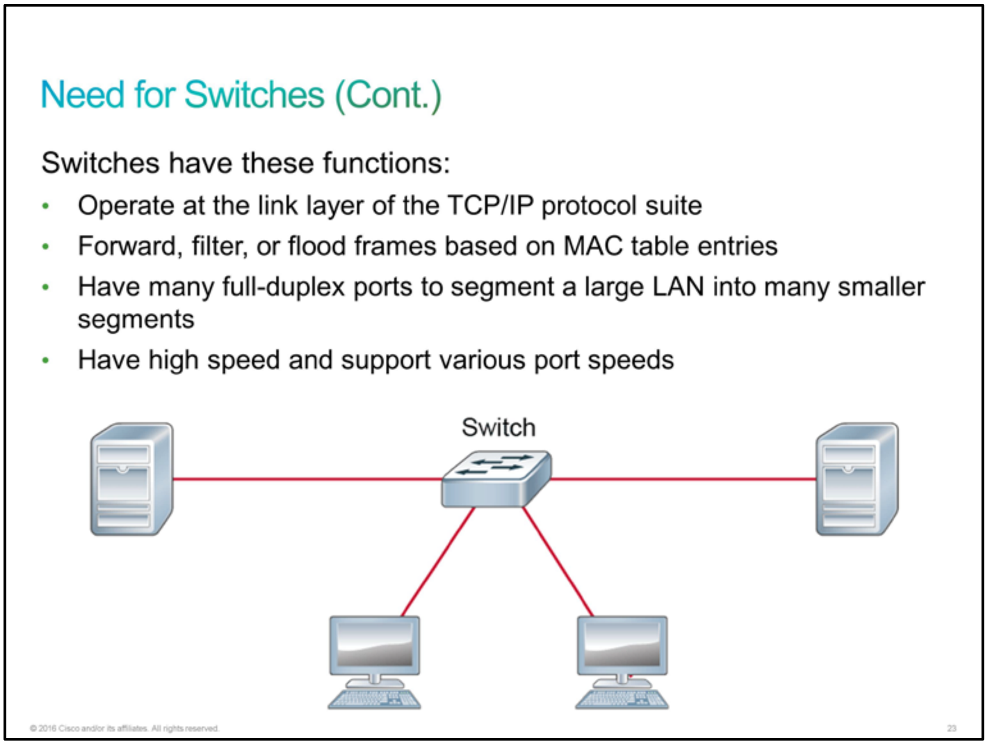
Historically, when network devices had few Ethernet segments, end host devices had to compete for the same bandwidth, and only one device was able to transmit data at a time. Network segments that share the same bandwidth are known as *collision domains*, because when two or more devices within that segment try to communicate at the same time, collisions may occur.

Today, it is common to use switches as network devices, operating at the link layer of the TCP/IP protocol suite, to divide a network into segments and reduce the number of devices that compete for bandwidth. Each new segment, then, results in a new collision domain. More bandwidth is available to the devices on a segment, and collisions in one collision domain do not interfere with the working of the other segments.


Need for Switches

As shown in the figure, each switch port connects to a single PC or server. Each switch port represents a unique collision domain.

The *broadcast domain* is another key concept. The filtering of frames by switches, which is done based on their MAC addresses, does not extend to filtering broadcast frames. By their nature, broadcast frames must be forwarded. Therefore, a port on a switch forms a single broadcast domain. It takes a Layer 3 entity, such as a router, to terminate a Layer 2 broadcast domain.



Need for Switches (Cont.)

Switches have these functions:

- Operate at the link layer of the TCP/IP protocol suite
- Forward, filter, or flood frames based on MAC table entries
- Have many full-duplex ports to segment a large LAN into many smaller segments
- Have high speed and support various port speeds

23

Ethernet switches selectively forward individual frames from a receiving port to the port where the destination node is connected. This selective forwarding process can be thought of as establishing a momentary point-to-point connection between the transmitting and receiving nodes. The connection is made only long enough to forward a single frame. During this instant, the two nodes have a full-bandwidth connection between them and represent a logical point-to-point connection.

The switch builds and maintains a table, which is called a MAC table. This table matches a destination MAC address with the port that is used to connect to a node. For each incoming frame, the destination MAC address in the frame header is compared to the list of addresses in the MAC table. Switches then use MAC addresses as they decide whether to filter, forward, or flood frames.

The table shows how switches process unicast frames.

How Switches Process Unicast Frames on an Ethernet LAN

Step	Action
1	When a unicast frame is received on a port, the switch compares the destination MAC address to the MAC addresses that it has listed in its table.
2	If the switch determines that the destination MAC address of the frame resides on the same network segment as the source, it does not forward the frame. This process is called filtering. By performing this process, switches can significantly reduce the amount of traffic going between network segments by eliminating the unnecessary frames.
3	If the switch determines that the destination MAC address of the frame is not in the same network segment as the source, it forwards the frame to the appropriate segment.

Step	Action
4	If the switch does not have an entry in its table for the destination address, it transmits the frame out of all ports except the port on which it received the frame. This process is called *flooding*.

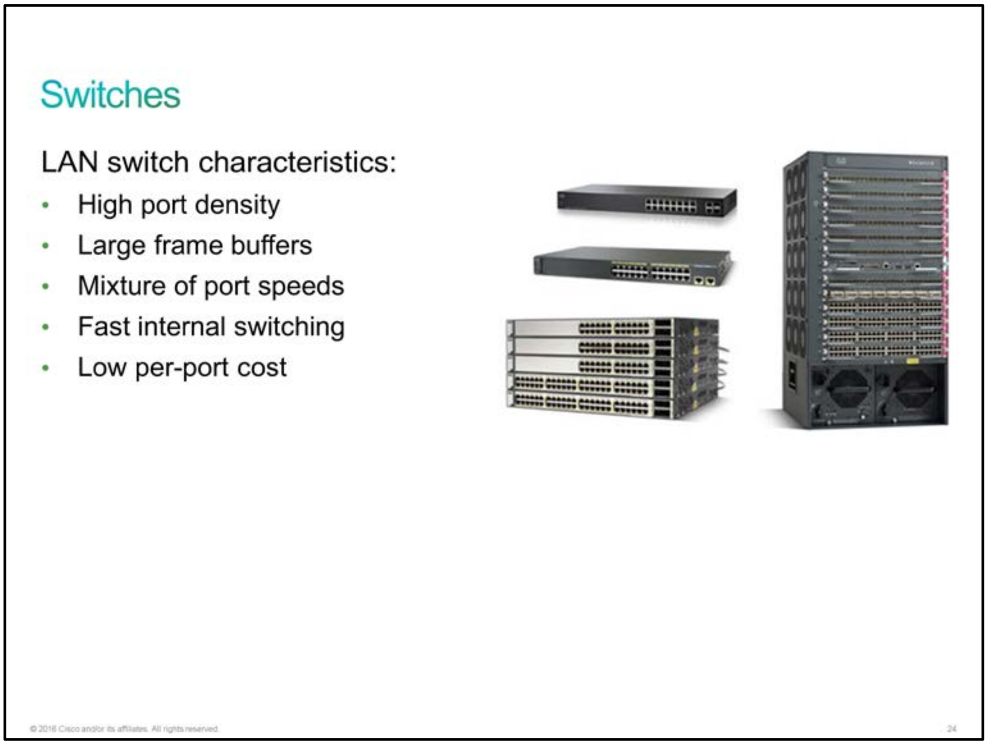
Switches

Switches have become a fundamental part of most networks. They allow the segmentation of a LAN into separate collision domains. Each port of the switch represents a separate collision domain and provides the full media to the node or nodes that are connected on that port. The introduction of full-duplex communications (a connection that can carry transmitted and received signals at the same time) has enabled 1 Gbps Ethernet and beyond.

Instead of Ethernet stations connecting to shared media, stations connect directly to a port on the LAN switch, providing two features that are unavailable in shared Ethernet LANs:

- **Dedicated bandwidth:** Because only one station is connected to a LAN switch port, the station does not compete for access to the media with other Ethernet stations. So, it receives the full bandwidth that is configured on the port.
- **Full-duplex operation:** In classical shared Ethernet, a station can either transmit or receive at a given time, which is referred to as half-duplex operation. Because the Ethernet station now directly connects its transmit and receive wires to the switch port, it can simultaneously transmit and receive. This mode is termed full duplex. Half-duplex mode is available for legacy 10BASE-T NICs that do not support full-duplex operation.

LAN switches have special characteristics that make them effective in alleviating network congestion.



Switches

LAN switch characteristics:

- High port density
- Large frame buffers
- Mixture of port speeds
- Fast internal switching
- Low per-port cost

© 2016 Cisco and/or its affiliates. All rights reserved. 24

Switches connect LAN segments, use a table of MAC addresses to determine the segment to which it will send the data, and reduce network traffic. The following are some important characteristics of switches:

- **High port density:** Switches have high port densities: 24- and 48-port switches operate at speeds of 100 Mbps, 1 Gbps, and 10 Gbps. Large enterprise switches may support hundreds of ports.
- **Large frame buffers:** The ability to store more received frames before having to start dropping them is useful, particularly when there may be congested ports to servers or other parts of the network.
- **Port speed:** Depending on the cost of a switch, they may support a mixture of speeds. You would expect ports of 100 Mbps, but switches offering ports that support 1 or 10 Gbps are more common.
- **Fast internal switching:** Having fast internal switching allows many speeds: 100 Mbps, 1 Gbps, and 10 Gbps. The method that is used may be a fast internal bus, shared memory, or integrated crossbar switch fabric, which affects the overall performance of the switch.
- **Low per-port cost:** Switches provide high port density at a lower cost. For this reason, LAN switches can accommodate network designs which feature fewer users per segment. This feature, therefore, increases the average available bandwidth per user.

Challenge

1. What are the two defining characteristics of a LAN? (Choose two.)

   A. higher data transfer rates in contrast to WAN
   B. lower data transfer rates in contrast to WAN
   C. smaller geographic area in contrast to WAN
   D. larger geographic area in contrast to WAN
   E. need for leased telecommunication lines

2. What are typical LAN host components? (Choose two.)

   A. switches
   B. routers
   C. servers
   D. hosts
   E. firewalls

3. What are the two typical LAN network components? (Choose two.)

   A. switches
   B. routers
   C. servers
   D. hosts
   E. firewalls

4. Which devices or protocols are responsible for sending or receiving data on the LAN?

   A. hosts
   B. interconnections
   C. network devices
   D. TCP

5. Which devices or protocols allow data to travel from one point to another in the network?

   A. hosts
   B. interconnections
   C. network devices
   D. TCP

6. What are the three functions of switches? (Choose three.)

   A. have high speed and support single port speeds
   B. operate at the link layer of the TCP/IP protocol suite
   C. operate at the network layer of the TCP/IP protocol suite
   D. forward, filter, or flood frames based on MAC table entries
   E. forward, filter, or flood packets based on IP routing table entries
   F. have many full-duplex ports to segment a large LAN into many smaller segments

7. Which switch characteristic can accommodate network designs that feature fewer users per segment?

   A. high port density
   B. large frame buffers
   C. port speed
   D. fast internal switching
   E. low per-port cost

Answer Key

Challenge

1. A, C
2. C, D
3. A, B
4. A
5. B
6. B, D, F
7. E

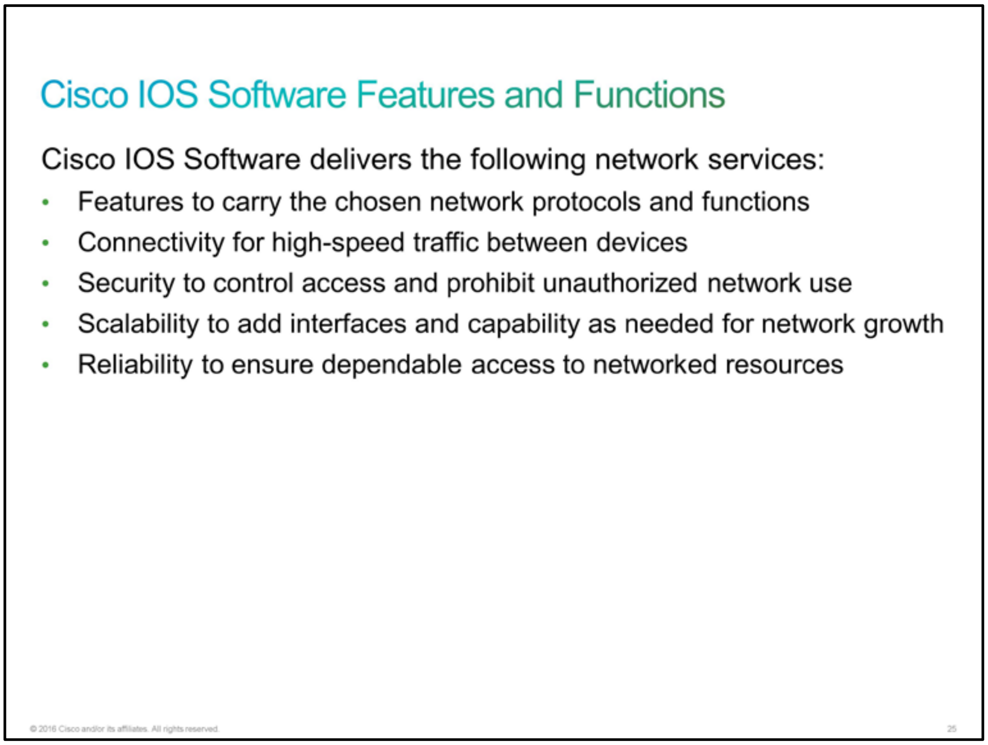
Lesson 4: Operating Cisco IOS Software

Introduction

Maya calls to set up the final technical interview with Bob. She says that, in the final phase, you will need to demonstrate your ability to perform essential tasks in Cisco IOS Software, such as managing Cisco IOS configurations, using the built-in help functionality, and improving the user experience in the CLI. She asks if you would like some time to practice.

Cisco IOS Software Features and Functions

Like a computer, a switch and a router also need an operating system to function. An operating system is the software that manages how various hardware components of a device function together. Just as an office needs a manager to supervise workers, a computing device needs an operating system. Many Cisco devices use Cisco IOS Software as their operating system. It is the core technology that extends across most Cisco devices, regardless of the size and type of the device. Many devices use the Cisco IOS Software—for example, routers, LAN switches, small wireless APs, large routers with numerous interfaces, and so on.



Cisco IOS Software Features and Functions

Cisco IOS Software delivers the following network services:

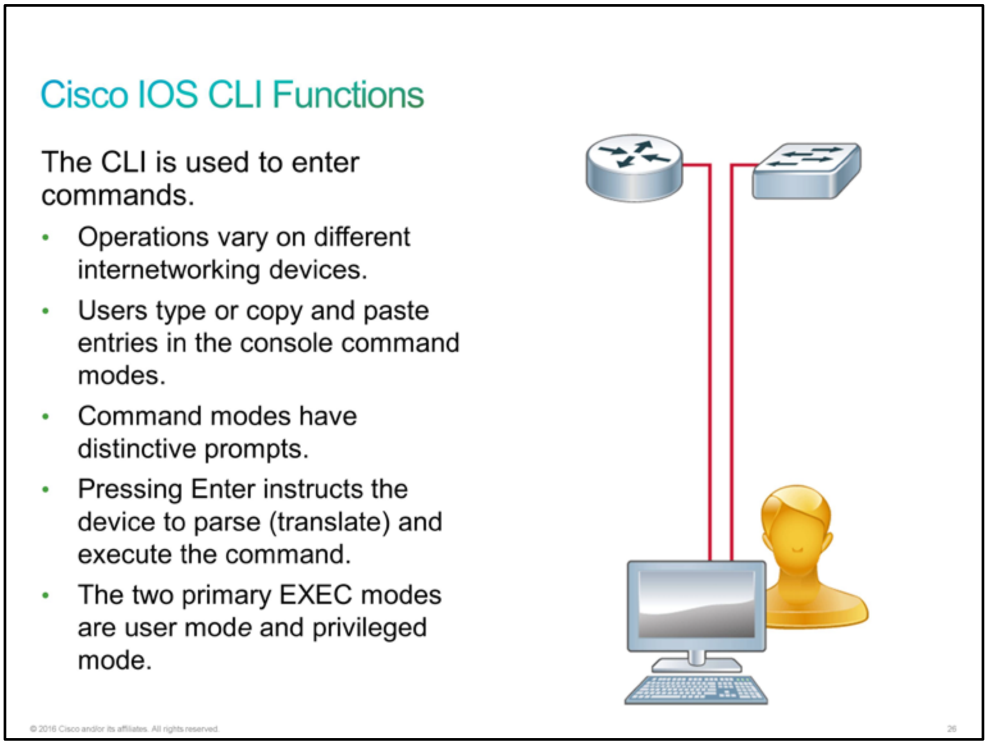
- Features to carry the chosen network protocols and functions
- Connectivity for high-speed traffic between devices
- Security to control access and prohibit unauthorized network use
- Scalability to add interfaces and capability as needed for network growth
- Reliability to ensure dependable access to networked resources

© 2016 Cisco and/or its affiliates. All rights reserved. 25

The services that are provided by Cisco IOS Software are generally accessed using a CLI. The CLI is a text-based interface that is similar to the old Microsoft operating system that is called MS-DOS. The CLI is accessed through a direct cabled connection that is called the *console connection*, a modem connection, or a Telnet or SSH session. Regardless of which connection method you use, access to the Cisco IOS CLI is generally referred to as an EXEC session. The features that you can access via the CLI vary—it depends on the version of the Cisco IOS Software and the type of device.

Cisco IOS CLI Functions

Cisco IOS Software uses a CLI via the console as its traditional environment to enter commands. While Cisco IOS Software is a core technology that extends across many products, the details of its operation vary on different internetworking devices.



Cisco IOS CLI Functions

The CLI is used to enter commands.

- Operations vary on different internetworking devices.
- Users type or copy and paste entries in the console command modes.
- Command modes have distinctive prompts.
- Pressing Enter instructs the device to parse (translate) and execute the command.
- The two primary EXEC modes are user mode and privileged mode.

26

Cisco IOS Software is designed as a modal operating system. The term *modal* describes a system which has various modes of operation, each having its own domain of operation. The CLI uses a hierarchical structure for the modes.

To enter commands into the CLI, type or copy and paste the entries within one of the several console command modes. Each command mode is indicated with a distinctive prompt. The term *prompt* is used because the system is prompting you to make an entry. By default, every prompt begins with the device name. Following the name, the remainder of the prompt indicates the mode. As you use commands and change mode, the prompt changes to reflect the current context. Pressing Enter instructs the device to parse and execute the command.

Note	It is important to remember that as soon as you enter a command, the command is executed. If you enter an incorrect command in a production router, it can negatively affect the network.

Cisco IOS Software uses a hierarchy of commands in its command-mode structure. Each command mode supports specific Cisco IOS commands that are related to the type of operation on the device.

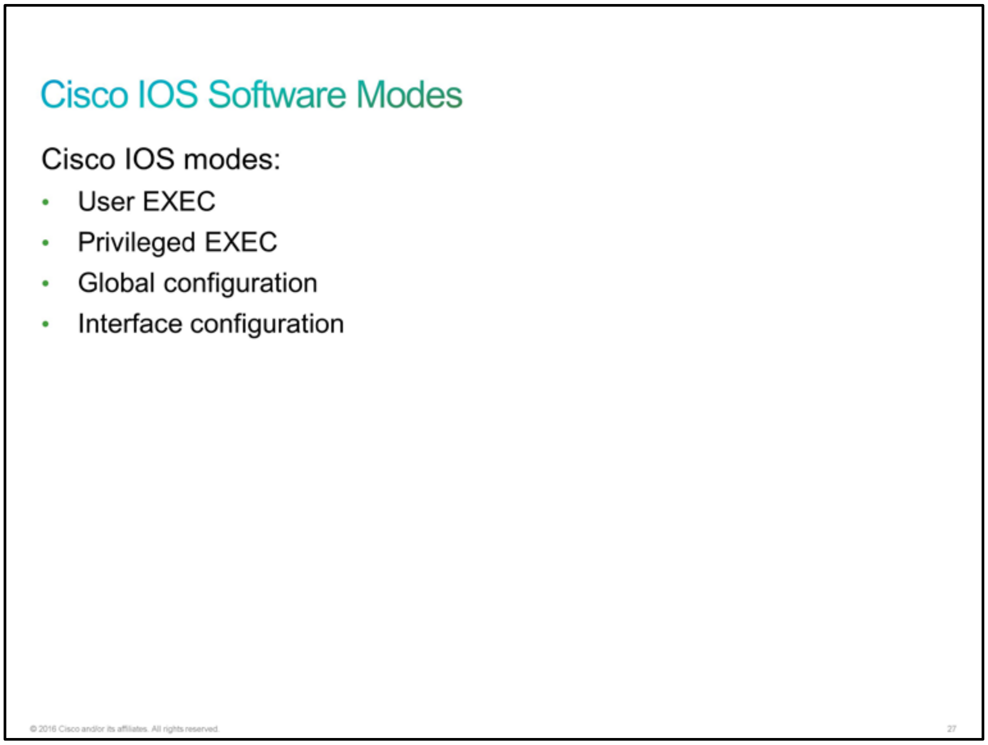
As a security feature, Cisco IOS Software separates EXEC sessions into two access levels:

- **User EXEC:** Allows a person to access only a limited number of basic monitoring commands.
- **Privileged EXEC:** Allows a person to access all device commands, such as those that you use for configuration and management. This level can be password-protected to allow only authorized users to access the device.

Cisco IOS Software Modes

Cisco IOS Software on a Cisco switch has various configuration modes that are hierarchically structured. The major modes are user EXEC, privileged EXEC, global configuration, and interface configuration.

Because these modes have a hierarchy, you can only access a lower-level mode from a higher-level mode. For example, in order to access global configuration mode, you must be in the privileged EXEC mode. Each mode is used to accomplish particular tasks and has a specific set of commands that are available in that mode. For example, to configure a switch interface, you must be in interface configuration mode. All configurations that you enter in interface configuration mode apply only to that interface.



Cisco IOS Software Modes

Cisco IOS modes:

- User EXEC
- Privileged EXEC
- Global configuration
- Interface configuration

© 2016 Cisco and/or its affiliates. All rights reserved. 27

The following table offers more detail on each mode:

Mode	Access Method	Prompt	Exit Method	About This Mode
User EXEC	Begin a session with your switch.	Switch>	Enter **logout** or **quit**.	Use this mode to change terminal settings, perform basic tests, or display system information.
Privileged EXEC	While in user EXEC mode, enter the **enable** command.	Switch#	Enter **disable** or **exit**.	Use this mode to verify commands that you have entered. Use a password to protect access to this mode.

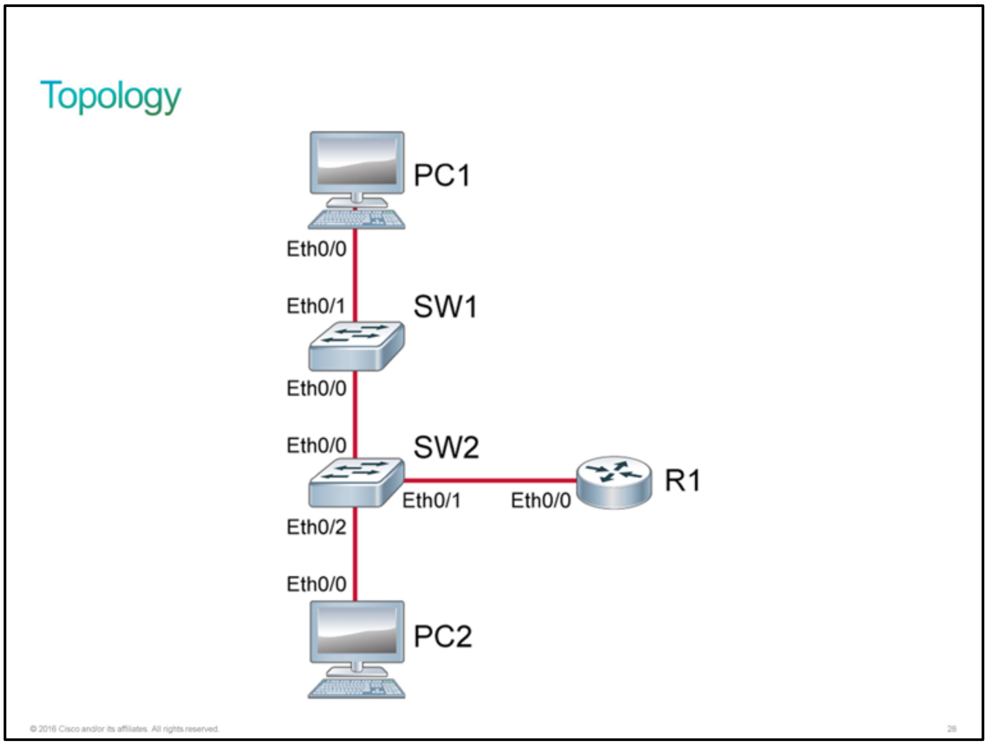
Mode	Access Method	Prompt	Exit Method	About This Mode
Global configuration	While in privileged EXEC mode, enter the **configure** command.	Switch(config)#	To exit to privileged EXEC mode, enter **exit** or **end**, or press **Ctrl-Z**.	Use this mode to configure parameters that apply to the entire switch.
Interface configuration	While in global configuration mode, enter the **interface** command (with a specific interface).	Switch(config-if)#	To exit to global configuration mode, enter **exit**. To return to privileged EXEC mode, press **Ctrl-Z** or enter **end**.	Use this mode to configure parameters for the Ethernet interfaces.

Discovery 1: Get Started with Cisco CLI

Introduction

In this discovery lab, you will learn about EXEC modes, CLI help, and the CLI error message. You will also learn how to manage Cisco IOS configuration and how to improve user experience in CLI.

Topology



Job Aid

Device Information

Device Information Table

Device	Characteristic	Value
PC1	Hostname	PC1
PC1	IP address	10.10.1.10/24
PC1	Default gateway	10.10.1.1
PC2	Hostname	PC2
PC2	IP address	10.10.1.20/24

Device	Characteristic	Value
PC2	Default gateway	10.10.1.1
SW1	Hostname	SW1
SW1	VLAN 1 IP address	10.10.1.2/24
SW1	Default gateway	10.10.1.1
SW1	Ethernet0/0 description	Link to SW2
SW1	Ethernet0/1 description	Link to PC1
SW2	Hostname	SW2
SW2	VLAN 1 IP address	10.10.1.3/24
SW2	Default gateway	10.10.1.1
SW2	Ethernet0/0 description	Link to SW1
SW2	Ethernet0/1 description	Link to R1
SW2	Ethernet0/2 description	Link to PC2
R1	Hostname	R1
R1	Ethernet0/0 description	Link to SW2
R1	Ethernet0/0 IP address	10.10.1.1/24
R1	Loopback 0 IP	10.10.3.1/24

PCs in the virtual lab environment are simulated as routers, so you should use Cisco IOS commands to configure them or make verifications.

Task 1: Navigate Between EXEC Modes

This session will guide you through the navigation between user EXEC and privileged EXEC on the Cisco IOS command line. The lab is prepared with the devices that are represented in the topology, but for this session you will only be using SW2.

Activity

Step 1 Access the console of SW2.

The greater than symbol (>) at the end of the prompt is an indication that you are accessing the user EXEC.

```
SW2>
```

Step 2 Use the question mark (**?**) to view the list of commands that are available in user EXEC.

When the display output pauses with the --More-- prompt, you can use the space bar to display the next page of the output.

```
SW2> ?
Exec commands:
  access-enable    Create a temporary Access-List entry
  access-profile   Apply user-profile to interface
  clear            Reset functions
  connect          Open a terminal connection
  crypto           Encryption related commands.
  disable          Turn off privileged commands
  <... output omitted ...>
  mtrace           Trace reverse multicast path from destination to source
  name-connection  Name an existing network connection
 --More-- <space bar>
  pad              Open a X.29 PAD connection
  ping             Send echo messages
  <... output omitted ...>
  where            List active connections
  x3               Set X.3 parameters on PAD
```

Note You have to press the space bar twice to scroll through the complete command list under the user EXEC. Have this information in mind, because you will soon contrast it to what you will be able to see in the privileged EXEC mode.

Note The commands are listed in alphabetical order. Note that the **configure** command is not available under user EXEC.

Note In the outputs, like in the previous one, many lines are omitted, due to space preservation. Omitted lines are indicated with <... output omitted ...> string.

Step 3 As you just saw, when you are presented with the --More-- prompt, you can use the space bar to scroll through the output page by page.

You can also use the Enter key to scroll forward just one line. You can also cancel the remaining output. The method to cancel the remaining output is device and operating system-version dependent. Sometimes you need to press Ctrl-C and sometimes you need to press "Q." On SW2, you can press any key other than the space bar or the Enter key. Give it a try!

```
SW2> ?
Exec commands:
  access-enable    Create a temporary Access-List entry
  access-profile   Apply user-profile to interface
  clear            Reset functions
  connect          Open a terminal connection
  crypto           Encryption related commands.
  disable          Turn off privileged commands
  <... output omitted ...>
  mtrace           Trace reverse multicast path from destination to source
  name-connection  Name an existing network connection
--More--  <Enter>
  pad              Open a X.29 PAD connection
--More--  <Enter>
  ping             Send echo messages
--More--  <Ctrl-C>
SW2>
```

Entering EXEC Mode

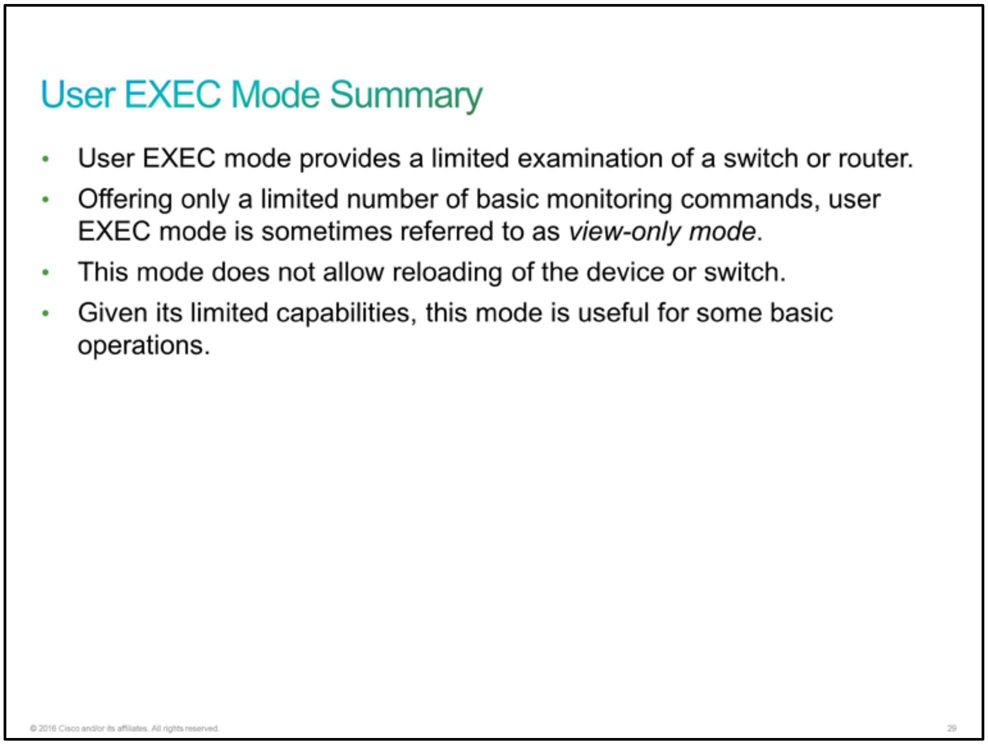
As a security feature, Cisco IOS Software separates EXEC sessions into the following two access levels:

- **User EXEC:** Allows you to access only a limited number of basic monitoring commands. When in EXEC mode, the prompt ends with the greater than or right angle bracket (>) symbol. For example, when you are in EXEC mode on a device with the hostname DTW_Switch, the prompt would be DTW_Switch>.
- **Privileged EXEC:** Allows you to access all device commands, such as those that you would use for configuration and management. It can be password-protected to allow only authorized users to access the device. When in this mode, the prompt ends with the octothorpe or pound (#) symbol. For example, when you are in privileged EXEC mode on a device with the hostname DTW_Switch, the prompt would look like DTW_Switch#. To change from user EXEC mode to privileged EXEC mode, enter the **enable** command at the hostname> prompt. To return to the user EXEC level, enter the **disable** command at the hostname# prompt.

By default, no authentication is required to access user EXEC mode from the console. You can enter the EXEC mode by simply pressing the Enter key. However, if login is configured, you must enter a username and password to enter the EXEC mode. It is a good practice to ensure that authentication is configured during the initial configuration.

Entering the question mark (**?**) in privileged EXEC mode reveals many more command options than entering the command at the user EXEC level. This feature is referred to as *context-sensitive help*.

User EXEC Mode Summary

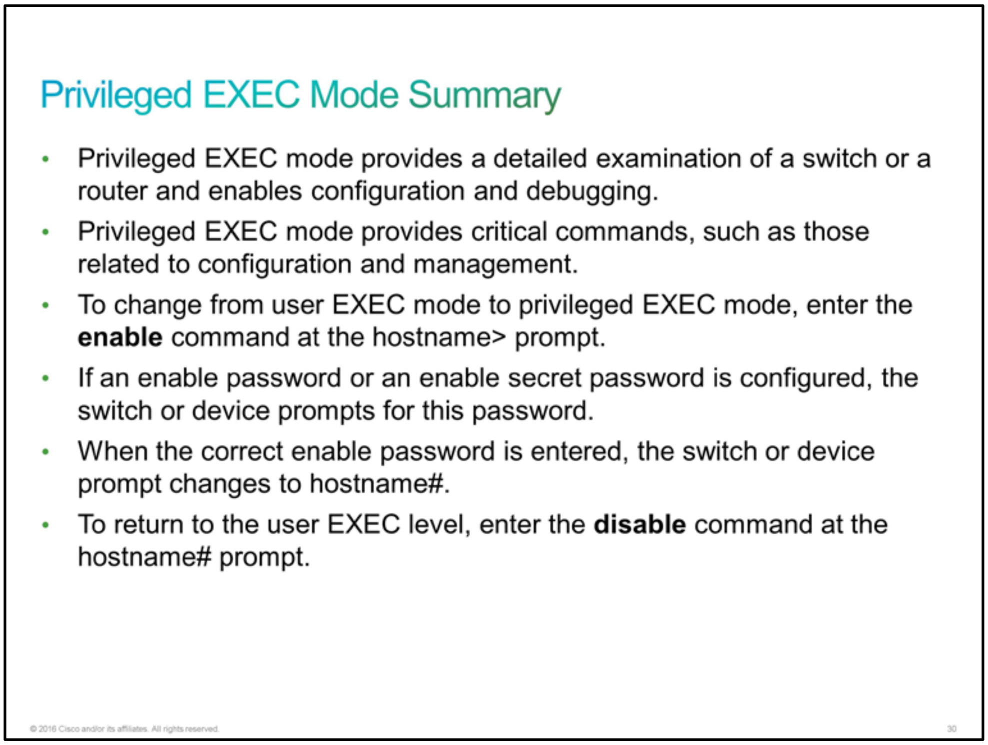


User EXEC Mode Summary

- User EXEC mode provides a limited examination of a switch or router.
- Offering only a limited number of basic monitoring commands, user EXEC mode is sometimes referred to as *view-only mode*.
- This mode does not allow reloading of the device or switch.
- Given its limited capabilities, this mode is useful for some basic operations.

29

Privileged EXEC Mode Summary



Privileged EXEC Mode Summary

- Privileged EXEC mode provides a detailed examination of a switch or a router and enables configuration and debugging.
- Privileged EXEC mode provides critical commands, such as those related to configuration and management.
- To change from user EXEC mode to privileged EXEC mode, enter the **enable** command at the hostname> prompt.
- If an enable password or an enable secret password is configured, the switch or device prompts for this password.
- When the correct enable password is entered, the switch or device prompt changes to hostname#.
- To return to the user EXEC level, enter the **disable** command at the hostname# prompt.

30

Step 4 Use the **enable** command to access the privileged EXEC.

The last character in the prompt has changed to the octothorpe (#) symbol. This symbol indicates to you that you are in privileged EXEC.

```
SW2> enable
SW2#
```

Step 5 Use the **?** command again to display the commands that you can use under privileged EXEC. Use the space bar to scroll through the entire list of the output.

```
SW2# ?
Exec commands:
  access-enable    Create a temporary Access-List entry
  access-profile   Apply user-profile to interface
  access-template  Create a temporary Access-List entry
  archive          manage archive files
  beep             Blocks Extensible Exchange Protocol commands
  calendar         Manage the hardware calendar
  cd               Change current directory
  clear            Reset functions
  clock            Manage the system clock
  cns              CNS agents
  configure        Enter configuration mode
  connect          Open a terminal connection
  <... output omitted ...>
  enable           Turn on privileged commands
  eou              EAPoUDP
--More--  <Enter>
  erase            Erase a filesystem
  <... output omitted ...>
```

Note Under privileged EXEC, you needed to press the space bar four times to get through the entire list of commands. Under user EXEC, you only needed to hit the space bar twice.

Note Under privileged EXEC, you can use the configuration command. You cannot proceed to the configuration mode from the user EXEC—you must traverse through the privileged EXEC first.

Step 6 Use the **disable** command to return to user EXEC.

```
SW2# disable
SW2>
```

Note The last character in the system prompt has returned to the greater than sign (>).

Task 2: Explore CLI Help

This session will guide you through using the question mark (**?**) command for help on the IOS CLI. It will also demonstrate how you can take advantage of the tab completion feature of the IOS CLI. The lab is prepared with the devices that are represented in the topology, but for this session you will only be using SW2. You will also take a look at CLI error messages.

Activity

Step 1 On SW2, use the **enable** command to access privileged EXEC.

```
SW2> enable
SW2#
```

Step 2 Use the question mark (**?**) to display all the commands that you can use under privileged EXEC.

Use the space bar to scroll through the entire list of the output.

```
SW2# ?
Exec commands:
  access-enable    Create a temporary Access-List entry
  access-profile   Apply user-profile to interface
  access-template  Create a temporary Access-List entry
  archive          manage archive files
  beep             Blocks Extensible Exchange Protocol commands
  calendar         Manage the hardware calendar
  cd               Change current directory
  clear            Reset functions
  clock            Manage the system clock
  cns              CNS agents
  configure        Enter configuration mode
  connect          Open a terminal connection
  <... output omitted ...>
  enable           Turn on privileged commands
  eou              EAPoUDP
--More--  <space bar>
  <... output omitted ...>
```

Note The list is quite long. You have to use the space bar four times to get through the entire list.

CLI Help

When you are learning a new program or interface, you usually depend on the Help features the program offers. Cisco IOS Software includes extensive command-line help functions, including context-sensitive help. There are two basic types of CLI keyboard help that the Cisco IOS devices enable. The first is context-sensitive help, which offers assistance when you are trying to determine the proper command and syntax. To use it, press the question mark (**?**) key. For example, you know that the command that you want to use starts with **sh**, but you are not sure what the rest of the command is. Enter **sh?** and you are presented with every command that starts with sh and that you can use in the current privilege mode. You can also use context-sensitive help to figure out the syntax for a command.

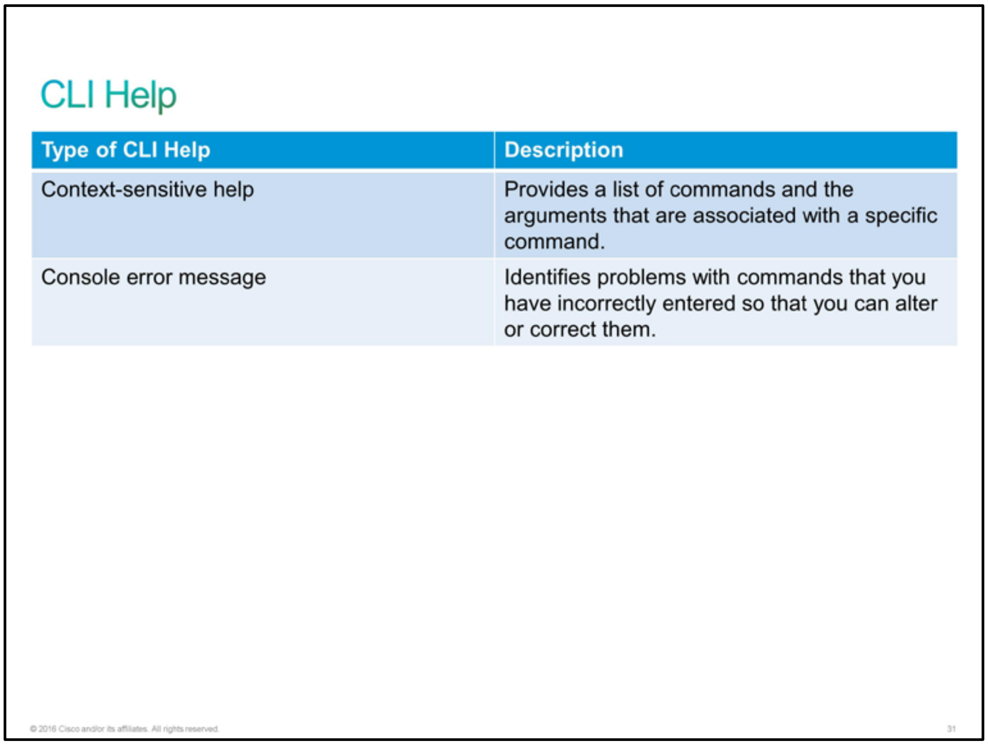
Another use of the context-sensitive help is to get a list of available commands for the current CLI mode. This list can be used when you are unsure of the name of a command or you want to see if Cisco IOS Software supports a particular command in a particular mode. To use context-sensitive help in this way, enter the question mark (**?**) at any prompt.

The other type of the CLI keyboard help is the error messages. When you enter a command in the CLI, the syntax is checked. If it is not correct, you receive an error that states "Invalid input detected at '^' marker." In addition to the message, the caret symbol (^) is added below the place in the command at which the error was detected. Basically, Cisco IOS Software is saying "I understood what you typed up to this point."

You may also receive an error message for an ambiguous command. This type of error occurs when you use an abbreviation for a command and the abbreviation results in multiple matches. In Cisco IOS Software, when you type enough letters that match only one command, you may press the Enter key. Because there is no other command that starts with those letters, Cisco IOS Software executes the command. For example, assume that there are several commands that start with the letter "c" but only one command that begins with "clo." If you press Enter after entering only **c**, you receive an "ambiguous command" error message. However, if you enter **clo** and press Enter, you do not receive a message that the command is ambiguous because the **clock** command is the only command that starts with those three letters. However, you would receive an "incomplete command" error message because clock is not a complete command. This message means that you did not enter enough information for Cisco IOS Software to understand what you were requesting.

Note This functionality may vary across Cisco IOS platforms.

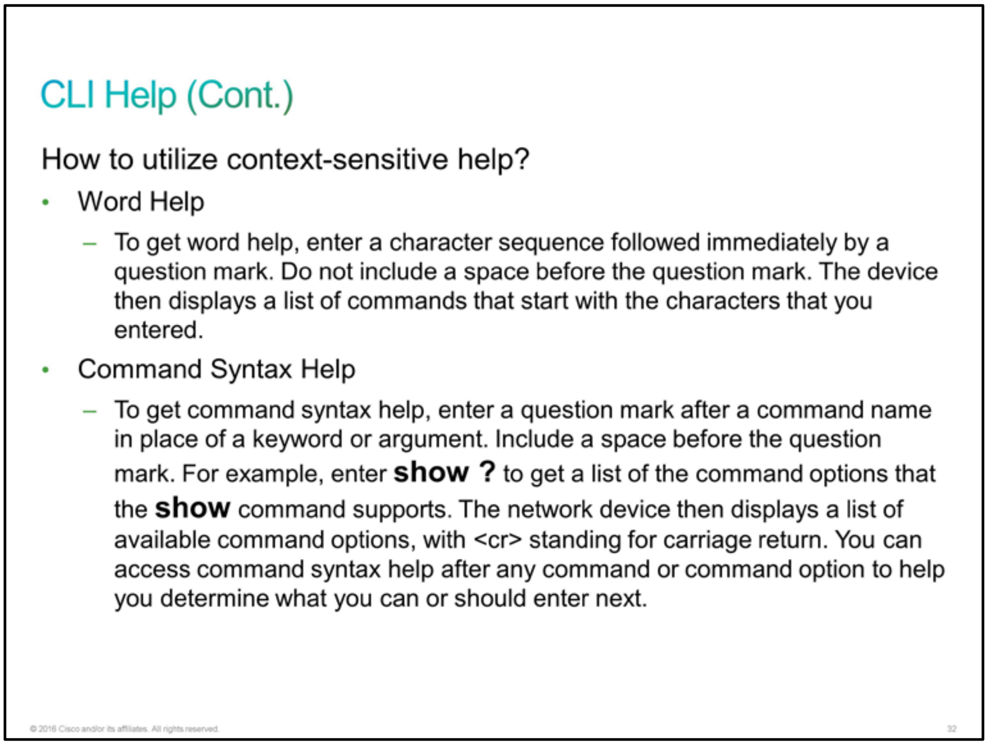
Note	The question mark is your friend when you are using the CLI. It is impossible to remember the syntax for every Cisco IOS command, so be sure to use this tool.



CLI Help

Type of CLI Help	Description
Context-sensitive help	Provides a list of commands and the arguments that are associated with a specific command.
Console error message	Identifies problems with commands that you have incorrectly entered so that you can alter or correct them.

31



CLI Help (Cont.)

How to utilize context-sensitive help?

- Word Help
  - To get word help, enter a character sequence followed immediately by a question mark. Do not include a space before the question mark. The device then displays a list of commands that start with the characters that you entered.
- Command Syntax Help
  - To get command syntax help, enter a question mark after a command name in place of a keyword or argument. Include a space before the question mark. For example, enter **show ?** to get a list of the command options that the **show** command supports. The network device then displays a list of available command options, with <cr> standing for carriage return. You can access command syntax help after any command or command option to help you determine what you can or should enter next.

32

Step 3 List only the commands that start with the letter "s" by entering **s?** on the command line.

```
SW2# s?
*s=show
sdlc        send    set        setup
show        slip    spec-file  ssh
start-chat  systat
```

The output shows that there is an exception to normal command parsing rules. The CLI will interpret the letter "s" all by itself as "show". This feature is specific to the device and operating system version. While it will work on SW2, it may not work on all devices. Abbreviating "show" with the characters "sh" is going to be effective more consistently across IOS devices.

Step 4 Try out the tab completion feature.

Like command abbreviation, tab completion works as long as you have entered enough characters to remove ambiguity. Type **sh** and then press the **tab** key.

The CLI parser expands the unambiguous abbreviation into the full command.

```
SW2# sh<tab>
SW2# show
```

Step 5 You might find tab completion helpful because it prevents you from attempting to use command abbreviation and accidentally abbreviate too much.

If there are multiple matches for the abbreviation, tab completion will not work. If you are not sure why, you can always use the question mark (**?**) at that point. Demonstrate this example by attempting to abbreviate the configure command with "con".

When you tried to use the tab to complete the abbreviation "con," it did not work. The command parser simply redisplayed "con". Using the question mark (**?**) at that point shows that there are two commands that begin with "con". To be unambiguous, you must use at least "conf" as your abbreviation for *configure*.

```
SW2# con<tab>
SW2# con?
configure  connect
SW2# con
```

Step 6 You will not go into the configuration mode during this session. Use the **Backspace** key to delete the "con" that is currently on the CLI input line.

Step 7 You have just demonstrated that the question mark (**?**) and tab completion work for commands.

They are also helpful for arguments to commands. For example, if you want to display all the arguments that you can use with the **show** command, use the question mark (**?**) and separate it from the **show** command by a space.

```
SW2# show ?
  aaa                Show AAA values
  access-expression  List access expression
  <... output omitted ...>
 --More—  <space bar>
  <... output omitted ...>
```

Note There are a lot of **show** commands. To scroll through the entire list you have to press the space bar nine times.

Step 8 Just like with commands, you can combine some explicit characters followed by the question mark to display a subset of the argument options.

For example, use **show r?** to display all the show command options that start with the letter "r".

```
SW2# show r?
radius     region     registry        reload
resource   rhosts     rib             rif
route-map  route-tag  running-config
```

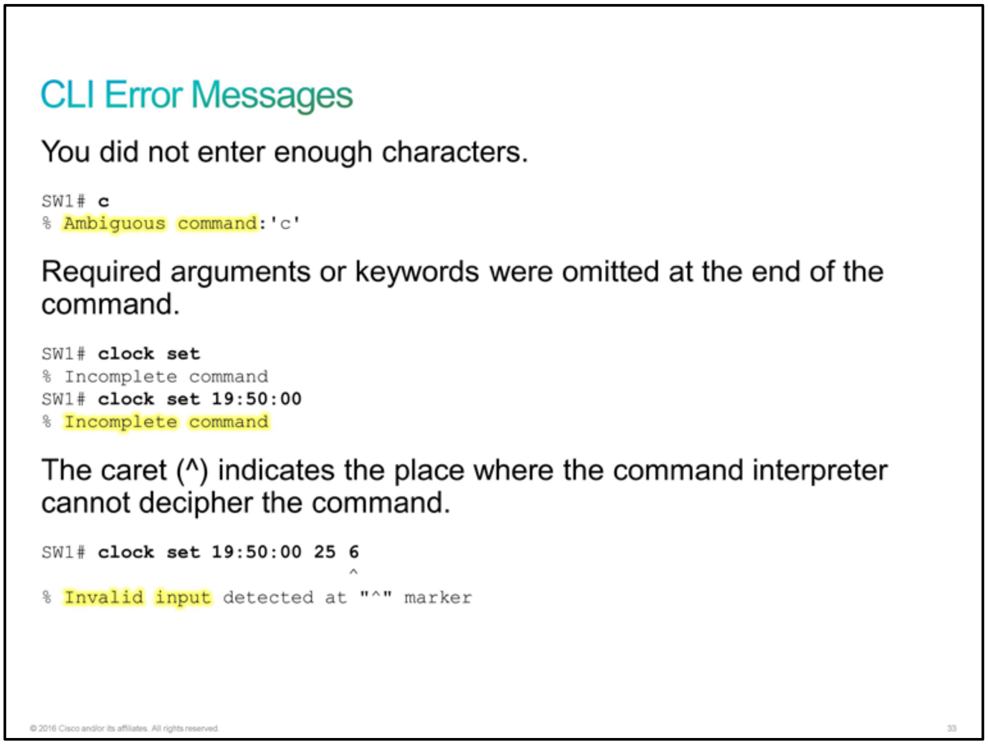
Step 9 Experiment with command abbreviation and tab completion in creative ways, until you feel you are comfortable using them.

You can see one example for **show running-config**, but still, feel free to experiment independently.

```
SW2# sh<tab>
SW2# show run<tab>
SW2# show running-config
Building configuration...

Current configuration : 865 bytes
!
version 15.1
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
service compress-config
<... output omitted ...>
```

Note You may also find tab completion useful if you are working with someone else. If you are typing at the CLI, you may understand some command abbreviations that your partner does not. Using command completion allows your partner to see the entire command verbiage.

CLI Error Messages



CLI Error Messages

You did not enter enough characters.

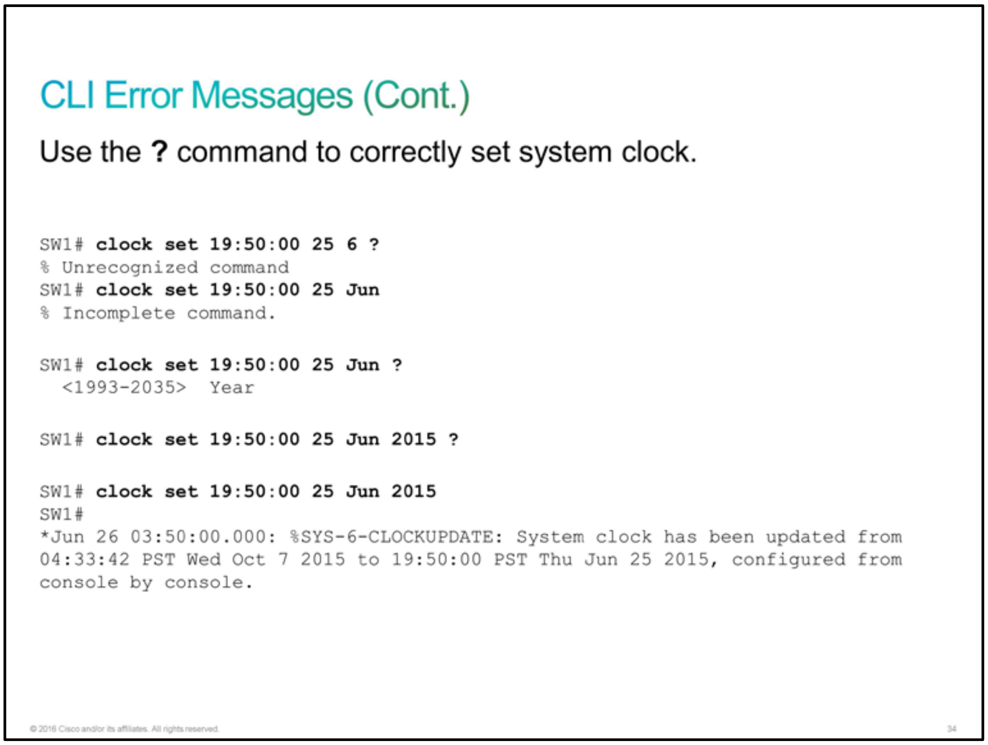
```
SW1# c
% Ambiguous command:'c'
```

Required arguments or keywords were omitted at the end of the command.

```
SW1# clock set
% Incomplete command
SW1# clock set 19:50:00
% Incomplete command
```

The caret (^) indicates the place where the command interpreter cannot decipher the command.

```
SW1# clock set 19:50:00 25 6
                           ^
% Invalid input detected at "^" marker
```



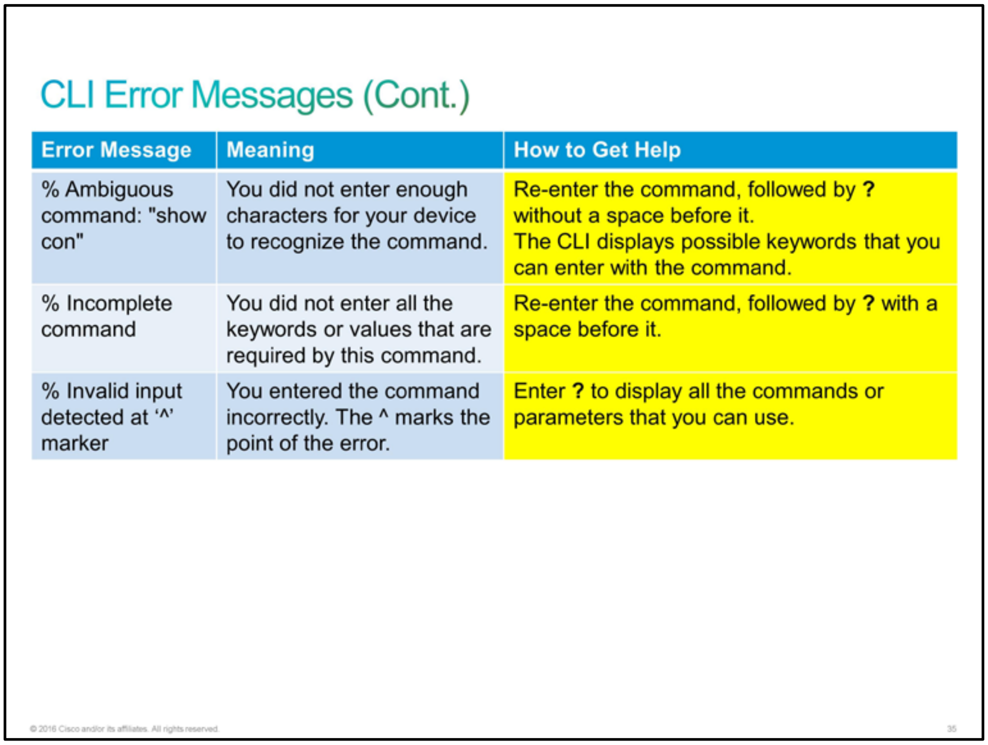
CLI Error Messages (Cont.)

Use the **?** command to correctly set system clock.

```
SW1# clock set 19:50:00 25 6 ?
% Unrecognized command
SW1# clock set 19:50:00 25 Jun
% Incomplete command.

SW1# clock set 19:50:00 25 Jun ?
  <1993-2035>  Year

SW1# clock set 19:50:00 25 Jun 2015 ?

SW1# clock set 19:50:00 25 Jun 2015
SW1#
*Jun 26 03:50:00.000: %SYS-6-CLOCKUPDATE: System clock has been updated from
04:33:42 PST Wed Oct 7 2015 to 19:50:00 PST Thu Jun 25 2015, configured from
console by console.
```

There are three types of console error messages:

- Ambiguous command
- Incomplete command
- Incorrect command



CLI Error Messages (Cont.)

Error Message	Meaning	How to Get Help
% Ambiguous command: "show con"	You did not enter enough characters for your device to recognize the command.	Re-enter the command, followed by **?** without a space before it. The CLI displays possible keywords that you can enter with the command.
% Incomplete command	You did not enter all the keywords or values that are required by this command.	Re-enter the command, followed by **?** with a space before it.
% Invalid input detected at '^' marker	You entered the command incorrectly. The ^ marks the point of the error.	Enter **?** to display all the commands or parameters that you can use.

The command history buffer stores the commands that have been most recently entered. To see these commands, enter the Cisco IOS **show history** command.

You can use context-sensitive help to determine the syntax of a particular command. For example, if the device clock needs to be set but you are not sure of the **clock** command syntax, the context-sensitive help provides a means to check the syntax.

Context-sensitive help supplies the whole command even if you enter just the first part of the command, such as **cl?**.

If you enter the command **clock** but an error message is displayed, indicating that the command is incomplete, enter the **?** command (preceded by a space) to determine which arguments are required for the command. In the **clock ?** example, the help output shows that the keyword **set** is required after **clock**.

If you now enter the command **clock set** but another error message appears, indicating that the command is still incomplete, press the **Up Arrow** key to repeat the command entry. Then, add a space and enter the question mark (**?**) to display a list of arguments that you can use for the command.

The example shows that after the last command recall, the administrator used the **?** to reveal additional arguments, which involve entering the current time using correct form of month and year..

The figure continues to illustrate how to set the device clock.

If after entering the current time you still see the Cisco IOS Software error message indicating that the command that you have entered is incomplete, recall the command, add a space, and enter the **?** command to display a list of arguments that are available for the command. In this example, enter the day, month, and year using the correct syntax. Then press **Enter** to execute the command.

Syntax checking uses the caret symbol (^) as an error-location indicator. It appears at the point in the command string where the user has entered an incorrect command, keyword, or argument. The error-location indicator and interactive help system provide a way to easily find and correct syntax errors. In the clock example, the caret symbol indicates that the month was entered incorrectly as a number. The parser is expecting the month to be spelled out.

Task 3: Manage Cisco IOS Configuration

Now you will go through the startup and running configurations on a Cisco IOS device. The lab is prepared with the devices that are represented in the topology, but for this session you will only be using SW2. In the end, you will erase the configuration on SW2. Do not worry, though. The lab system will return the configurations the next time the lab is initialized.

The prompt displays the hostname that is configured on the device. You will modify this component of the switch configuration as you experiment with the startup and running configurations.

Activity

Step 1 On SW2, enter the global configuration mode and change the hostname of the switch to "Temp" and return to privileged EXEC.

Immediately after you change the hostname setting on the switch, the system prompt reflects the new name.

```
SW2# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
SW2(config)# hostname Temp
Temp(config)# end
Temp#
```

You just modified the running configuration on the switch. The startup configuration has not changed.

Step 2 Display the running configuration that is parsed through the include filter, showing only the lines that include the string "hostname".

The use of tab completion and the question mark is intended to remind you that these options are always available to you. They will not be demonstrated any further in this session, but feel free to take advantage of them at any time.

```
Temp# sh<tab>
Temp# show r?
radius     region     registry        reload
resource   rhosts     rib             rif
route-map  route-tag  running-config

Temp# show run<tab>
Temp# show running-config | inc<tab>
Temp# show running-config | include hostname
hostname Temp
```

Tab completion was available for **show** and **running-config** and **include**, but not for **hostname**, because hostname is a freeform variable. It can be any string. There is no way for the Cisco IOS parser to guess what you want that string to be.

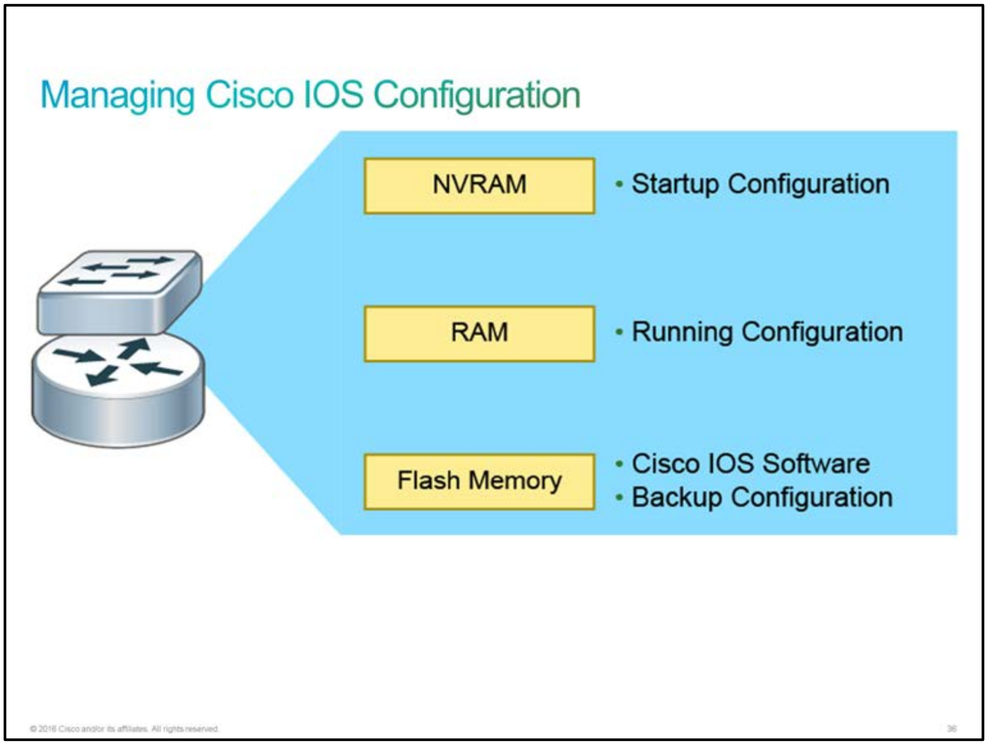
The one line in the configuration that includes the string hostname is the **hostname** command setting the hostname to "Temp."

Managing Cisco IOS Configuration

When a switch or a router starts, it looks for a configuration file in the NVRAM of the device. NVRAM is the memory in the device that retains information even when the device is powered down. The configuration file that is stored in NVRAM is called the startup-config file. If there is no startup-config file in NVRAM, the router or switch enters the setup utility and loads a blank configuration. The setup utility prompts you at the console for specific configuration information to create a basic initial configuration on the router or switch. You can also interrupt the setup utility and start configuring the device manually.

Once the device has started, the system copies the startup configuration to RAM. The configuration file in RAM is called the running-config file. As you make additional configurations, the system stores them in the running configuration. It is important to understand that RAM does not retain its information when the device is powered off or rebooted. If a change is made to the running configuration, it must be copied to the startup configuration, which is stored in the NVRAM, for it to be retained after a reboot.

In addition to NVRAM and RAM, Cisco devices have a third type of memory, called *flash memory*. Flash memory is similar to a hard drive in that the information that the system stores there is retained even when the device is powered off. Cisco IOS Software is stored in flash memory. Flash memory may also store backup configuration files and additional device-supported files.



Managing Cisco IOS Configuration

NVRAM
- Startup Configuration

RAM
- Running Configuration

Flash Memory
- Cisco IOS Software
- Backup Configuration

© 2016 Cisco and/or its affiliates. All rights reserved.

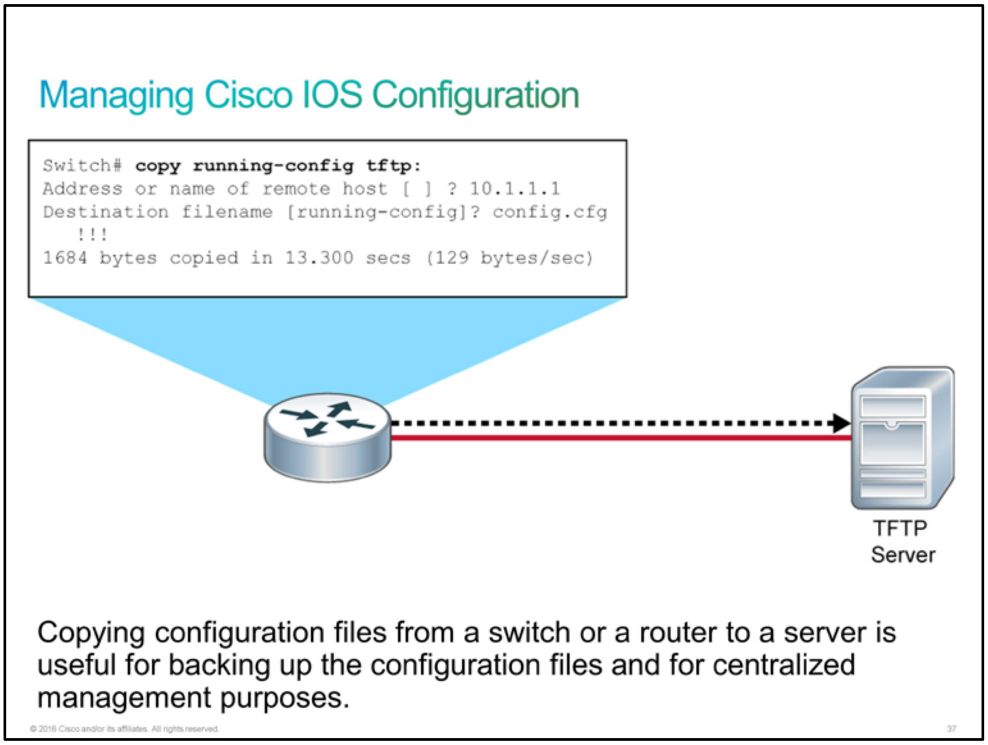
36

To view the configuration files, use the **show** command followed by the name of the file. For example, if you want to view the configuration that is stored in RAM, type **show running-config**. To save the running configuration, copy it to NVRAM. To do so, use the **copy** command followed by the names of the source and destination files. The complete command is **copy running-config startup-config**. Review the table for additional commonly used Cisco IOS commands.

Common IOS Management

Command	Function
show running-config	Displays the current running configuration. You can also use filters. For example, you can use the **show running-config interface GigabitEthernet0/1** command to display only the interface GigabitEthernet0/1 running configuration.
show startup-config	Displays the saved configuration in NVRAM.
configure terminal	Enters the configuration mode, where you can interactively create configurations in RAM from the console or remote terminal.
copy running-config startup-config	Saves the running configuration to NVRAM.
copy startup-config running-config	Startup configuration in NVRAM is merged into running configuration.
erase startup-config	Deletes the saved startup-config file in NVRAM.

You can also use the **copy** command to copy configuration files and Cisco IOS Software files from a switch or a router to a server (or vice versa) using FTP, SCP, HTTP, TFTP, and other protocols. For example, in the **copy running-config tftp:** command, the system copies the running configuration in RAM to a TFTP server. You must supply the IP address or name of the TFTP server and a destination filename. During the copying process, a series of exclamation marks show the progress of the upload.



Managing Cisco IOS Configuration

```
Switch# copy running-config tftp:
Address or name of remote host [ ] ? 10.1.1.1
Destination filename [running-config]? config.cfg
   !!!
1684 bytes copied in 13.300 secs (129 bytes/sec)
```

Copying configuration files from a switch or a router to a server is useful for backing up the configuration files and for centralized management purposes.

37

Note Regardless of the size of the network, there should always be a copy of the current running configuration online as a backup.

Copying configuration files from an external server to the running configuration in RAM or to the startup configuration file in NVRAM of the router or switch is useful for restoring backups. You should copy the files to a device other than the one that they were created on.

When you copy a configuration into RAM from any source, the configuration merges with the existing configuration in RAM. New configuration parameters are added, and changes to existing parameters overwrite the old parameters. Configuration commands in RAM for which there is no corresponding command in NVRAM remain unaffected.

Step 3 Now display the startup configuration that is parsed through the include filter, showing only the lines that include the string "hostname".

When you make changes to the running configuration, it does not affect the startup configuration. The startup configuration still has SW2 configured as the hostname.

```
Temp# show startup-config | include hostname
hostname SW2
```

Step 4 Use the **reload** command which will reboot the switch. This action will cause the switch to throw away the running configuration and read the startup configuration from scratch.

Answer **no** to the query about saving the modified configuration. The goal is to demonstrate how to return to the old configuration. If you save the modified configuration, the system will overwrite the old configuration.

After the reload, as indicated by the system prompt, the hostname has returned to SW2.

```
Temp# reload

System configuration has been modified. Save? [yes/no]: no
Proceed with reload? [confirm] <Enter>
<... output omitted ...>
Press RETURN to get started! <Enter>
<... output omitted ...>
SW2>
```

To make changes to the running configuration permanent, you have to save the running configuration over the startup configuration.

Step 5 Change the hostname one more time.

This time, set the hostname to "ThisWillStick."

```
SW2> enable
SW2# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
SW2(config)# hostname ThisWillStick
ThisWillStick(config)# end
ThisWillStick#
```

Step 6 Copy the running configuration over the startup configuration.

```
ThisWillStick# copy running-config startup-config
Destination filename [startup-config]? <Enter>
Building configuration...
Compressed configuration from 936 bytes to 641 bytes[OK]
ThisWillStick#
```

After this copy operation, the change to the hostname is reflected in the startup configuration and will now be able to survive a reload event.

Note Optionally, you can use the **show startup-configuration** command to verify that the change is reflected there.

Step 7 Use the **reload** command again, and verify that the new hostname setting is still in place after the reboot event.

```
ThisWillStick# reload
Proceed with reload? [confirm] <Enter>
<... output omitted ...>
Press RETURN to get started! <Enter>
<... output omitted ...>
ThisWillStick>
```

The hostname does, indeed, remain as ThisWillStick.

Step 8 Now erase the startup configuration with the **erase startup-config** command.

```
ThisWillStick> enable
ThisWillStick# erase startup-config
Erasing the nvram filesystem will remove all configuration files! Continue?
[confirm]  <Enter>
[OK]
Erase of nvram: complete
*Jul  6 08:40:12.990: %SYS-7-NV_BLOCK_INIT: Initialized the geometry of nvram
ThisWillStick#
```

Note Even though the system erased the startup configuration, this action does not have an effect on the running configuration. In fact, (do not do this now) you could use the **copy running-config startup-config** command now to put the startup configuration back to the way it was.

Step 9 Verify that the system actually erased the startup configuration using **show startup-config**.

```
ThisWillStick# show startup-config
startup-config is not present
```

Step 10 Reload the switch.

Understand that the switch will attempt to read the startup configuration and find it missing. This situation will essentially set the switch back to the factory default state. Do not worry, when the lab is reinitialized, the lab system will set all device configurations appropriately.

```
ThisWillStick# reload
Proceed with reload? [confirm] <Enter>
<... output omitted ...>
Press RETURN to get started!  <Enter>
Switch>
```

Step 11 Verify that the hostname setting in the running configuration is the default value (Switch).

```
Switch> enable
Switch# show running-config | include hostname
hostname Switch
Switch#
```

Task 4: Improve User Experience in CLI

In this session, you will be able to practice using terminal history. Recalling previous commands is useful simply to reduce typing. When you recall a command, you can simply press Enter to use the exact same command, or you can edit it to suit your new purpose. The lab is prepared with the devices that are represented in the topology, but for this session you will only be using R1.

The prompt displays the hostname that is configured on the device. You will modify this component of the router configuration as you experiment with the startup and running configurations.

Activity

Step 1 On R1, use the **enable** command to access the privileged EXEC.

```
R1> enable
R1#
```

Step 2 Enter the sequence of commands that are shown below.

The sequence is rather arbitrary. The selection criteria were to include three EXEC commands and two configuration mode commands. Do not be concerned if the commands are new to you. This part will simply give you a little bit of data in the terminal history.

- **show ip route** (in privileged EXEC mode)
- **show clock** (in privileged EXEC mode)
- **show ip interface brief** (in privileged EXEC mode)
- **configure terminal** (to go in global configuration mode)
- **clock timezone EST 0** (in global configuration mode)
- **no ip domain-lookup** (in global configuration mode)

```
R1# show ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       + - replicated route, % - next hop override

Gateway of last resort is not set

      10.0.0.0/8 is variably subnetted, 4 subnets, 2 masks
C        10.10.1.0/24 is directly connected, Ethernet0/0
L        10.10.1.1/32 is directly connected, Ethernet0/0
C        10.10.3.0/24 is directly connected, Loopback0
L        10.10.3.1/32 is directly connected, Loopback0


R1# show clock
*00:47:02.857 PST Mon Jul 6 2015


R1# show ip interface brief
Interface                  IP-Address      OK? Method Status
Protocol
Ethernet0/0                10.10.1.1       YES NVRAM  up                    up
Ethernet0/1                unassigned      YES NVRAM  administratively down
down
Ethernet0/2                unassigned      YES NVRAM  administratively down
down
Ethernet0/3                unassigned      YES NVRAM  administratively down
down
Serial1/0                  unassigned      YES NVRAM  administratively down
down
Serial1/1                  unassigned      YES NVRAM  administratively down
down
Serial1/2                  unassigned      YES NVRAM  administratively down
down
Serial1/3                  unassigned      YES NVRAM  administratively down
down
Loopback0                  10.10.3.1       YES NVRAM  up                    up


R1# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)# clock timezone EST 0
*Jul  6 08:48:41.931: %SYS-6-CLOCKUPDATE: System clock has been updated from
00:48:41 PST Mon Jul 6 2015 to 08:48:41 EST Mon Jul 6 2015, configured from
console by console.
R1(config)# no ip domain-lookup
```

Improving User Experience in CLI

The Cisco IOS CLI includes many features that make the configuration process easier and faster. These features include command-line editing keys, command history, and filtering parameters.

Command-Line Editing Keys

Command-line editing keys are shortcuts and hot keys that the CLI provides. Use these shortcuts and hot keys to move the cursor around on the command line for corrections or changes. Use them also to make configuring, monitoring, and troubleshooting easier. The table describes each of the shortcuts for command-line editing and controlling command entry.

Command-Line Editing Key Sequence	Description
Ctrl-A	Moves the cursor to the beginning of the command line
Ctrl-C	Aborts the current command and exits the configuration mode
Ctrl-E	Moves the cursor to the end of the command line
Esc-B	Moves the cursor back one word
Esc-F	Moves the cursor forward one word
Ctrl-B	Moves the cursor back one character
Ctrl-F	Moves the cursor forward one character
Ctrl-D	Deletes a single character at the cursor
Backspace	Removes one character to the left of the cursor
Ctrl-P	Redisplays the current command line
Ctrl-U	Erases a line
Ctrl-W	Erases a word to the left of the cursor
Ctrl-Z	Ends configuration mode and returns to the EXEC prompt
Tab	Completes a partially entered command if enough characters have been entered to make it unambiguous
Ctrl-Shift-6	Allows the user to interrupt a Cisco IOS process such as ping or traceroute
Ctrl-P or Up Arrow	Recalls last (previous) commands
Ctrl-N or Down Arrow	Recalls more recent commands

Note The Esc key is not functional on all terminals.

Command History

The Cisco CLI provides a history or record of commands that users have entered. You will find this feature, which is called the command history, particularly useful in helping you to recall long or complex commands or entries.

With the command history feature, you can complete the following tasks:

- **Display the contents of the command buffer:** By default, command history is enabled, and the system records the last 10 command lines in its history buffer.
- **Set the command history buffer size:** To change the number of command lines that the system will record during the current terminal session only, use the **terminal history** command in user EXEC mode.
- **Recall previously entered commands that are stored in the history buffer:** There is a buffer for EXEC mode and another buffer for the configuration mode. To recall commands in the history buffer, press **Ctrl-P** or the **Up Arrow** key. The command output begins with the most recent command. Repeat the key sequence to recall successively older commands.

  To return to more recent commands in the history buffer (after recalling older commands with Ctrl-P or the Up Arrow key), press **Ctrl-N**, or the **Down Arrow** key. Repeat the key sequence to recall successively more recent commands.

  On most computers, there are additional select and copy functions available. Copy a previous command string, then paste or insert it as the current command entry, and press **Enter**.

  When you use **show** commands such as **show running-config**, Cisco IOS Software automatically pauses when displaying the output after a specified number of lines. The process of displaying the output pauses, and Cisco IOS Software displays "--More--." It then waits for user input to continue with the display process. You can press the Spacebar key to display another set of subsequent lines or press Enter to display a single line.
- **Set the number of lines on the current terminal screen:** You can use the **terminal length** command, followed by a number, to control the number of lines that the CLI displays without pausing during the output. A value of zero prevents the router from pausing between screens of output. By default, the value is set to 24.

Step 3 Now, while remaining in the configuration mode, use the **Up Arrow** and **Down Arrow** keys to scroll through the terminal history buffer.

Note that you do not see the EXEC commands. There is a separate terminal history buffer for configuration and EXEC modes.

Step 4 Leave the configuration mode (use **end, exit,** or press **Ctrl-Z**) to return to privileged EXEC.

Step 5 Again use the **Up Arrow** and **Down Arrow** keys to show that you can recall previous commands.

Step 6 Recall the **show ip route** command and then press the **Enter** key to resubmit it without any edits.

It is a common exercise to revisit show commands that display operational status as you make changes to the configurations on IOS devices and their neighbors.

Step 7 Now, type the following command, purposely mistyping "show" as "snow."

```
R1# snow ip interface brief
      ^
% Invalid input detected at '^' marker.
```

Everyone makes typographical errors. Dealing with them is one of the best uses of the terminal history and the command line editing tools.

Step 8 Follow this sequence to quickly and easily correct the typographical error and resubmit the corrected command:

a. Press the **Up Arrow** key once to retrieve the previous command.

b. Press **Ctrl-A** to move the cursor to the beginning of the line.

c. Press the **Right Arrow** twice to move the cursor to the right of the incorrect letter "n."

d. Press **Backspace** to erase the letter "n."

e. Press **h** to insert the correct letter "h."

f. Press **Enter** to resubmit the corrected command.

Step 9 Return to the global configuration mode.

```
R1# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#
```

Note As before, you will be using commands that you are not familiar with to facilitate the demonstration of the power of the terminal history buffer. Do not concern yourself with commands themselves. Instead, focus on how beneficial the terminal history buffer can be.

Step 10 Configure the description of interface Serial 1/0 and enable the interface by overriding the default **shutdown** command.

```
R1(config)# interface Serial 1/0
R1(config-if)# description Link to SP1
R1(config-if)# no shutdown
*Jul  6 08:51:13.776: %LINK-3-UPDOWN: Interface Serial1/0, changed state to up
*Jul  6 08:51:14.780: %LINEPROTO-5-UPDOWN: Line protocol on Interface
Serial1/0, changed state to up
R1(config-if)#
```

Step 11 Repeat a very similar configuration for interface Serial 1/1.

The following process can make this task relatively easy:

a. Press the **Up Arrow** key three times to recall the interface command. Edit the 1/0 to be 1/1 and press the **Enter** key to resubmit the edited command.

b. Press the **Up Arrow** key three times, to recall the description command, edit the SP1 to be SP2 and press the **Enter** key to resubmit the edited command.

c. Press the **Up Arrow** key three times, to recall the **no shutdown** command, and press the **Enter** key to resubmit the command without any editing.

The resulting sequence should look like the following example:

```
R1(config)# interface Serial 1/1
R1(config-if)# description Link to SP2
R1(config-if)# no shutdown
*Jul  6 09:02:22.638: %LINK-3-UPDOWN: Interface Serial1/1, changed state to up
*Jul  6 09:02:23.642: %LINEPROTO-5-UPDOWN: Line protocol on Interface
Serial1/1, changed state to up
R1(config-if)#
```

Step 12 Leave the configuration mode by using **end**, **exit** (2 times), or pressing **Ctrl-Z** to return to privileged EXEC.

Optionally, you can save the running configuration to the startup configuration, but it is not necessary in the automated lab environment.

Filtering Parameters

Another useful feature that improves the user experience in the CLI is the filtering of **show** outputs. Using filtering, you can display only the parts of **show** outputs that you are interested in. You can filter outputs by typing the pipe (|) character after a **show** command, followed by a filtering parameter and a filtering expression. The table describes filtering parameters that are available for output filtering.

Parameter	Description
begin	Shows all output lines, starting with the line that matches the filtering expression
exclude	Excludes all output lines that match the filtering expression
include	Includes all output lines that match the filtering expression
section	Shows the entire section that starts with the filtering expression

Step 13 On the R1 router use **begin** and **include** options with **show running-config** command and filtering expression *interface*.

You should see following output when using **begin** option:

```
R1# show running-config | begin interface
interface Loopback0
 ip address 10.10.3.1 255.255.255.0
!
interface Ethernet0/0
 description Link to SW2
 ip address 10.10.1.1 255.255.255.0
!
interface Ethernet0/1
 no ip address
 shutdown
!
interface Ethernet0/2
 no ip address
 shutdown
!
interface Ethernet0/3
 no ip address
 shutdown
!
interface Serial1/0
 no ip address
 shutdown
 serial restart-delay 0
!
interface Serial1/1
 no ip address
 shutdown
 serial restart-delay 0
!
interface Serial1/2
 no ip address
 shutdown
 serial restart-delay 0
!
interface Serial1/3
 no ip address
 shutdown
 serial restart-delay 0
!
ip forward-protocol nd
!
!
no ip http server
no ip http secure-server
!
!
!
!
control-plane
!
!
!
!
!
!
!
line con 0
 logging synchronous
line aux 0
line vty 0 4
 login
 transport input all
!
!
end
```

You should see following output when using **include** option:

```
R1# show running-config | include interface
interface Loopback0
interface Ethernet0/0
interface Ethernet0/1
interface Ethernet0/2
interface Ethernet0/3
interface Serial1/0
interface Serial1/1
interface Serial1/2
interface Serial1/3
```

Step 14 On the R1 router use **section** option with **show running-config** command and filtering expression *interface*.

You should see following output when using **section** option:

```
R1# show running-config | section interface
interface Loopback0
 ip address 10.10.3.1 255.255.255.0
interface Ethernet0/0
 description Link to SW2
 ip address 10.10.1.1 255.255.255.0
interface Ethernet0/1
 no ip address
 shutdown
interface Ethernet0/2
 no ip address
 shutdown
interface Ethernet0/3
 no ip address
 shutdown
interface Serial1/0
 no ip address
 shutdown
 serial restart-delay 0
interface Serial1/1
 no ip address
 shutdown
 serial restart-delay 0
interface Serial1/2
 no ip address
 shutdown
 serial restart-delay 0
interface Serial1/3
 no ip address
 shutdown
 serial restart-delay 0
```

Step 15 On the R1 router use **exclude** option with **show running-config** command and filtering expression *!*.

You should see following output when using **exclude** option:

```
R1# show running-config | exclude !
Building configuration...

Current configuration : 1223 bytes
version 15.2
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
hostname R1
boot-start-marker
boot-end-marker
no aaa new-model
clock timezone PST -8 0
mmi polling-interval 60
no mmi auto-configure
no mmi pvc
mmi snmp-timeout 180


no ip domain lookup
ip cef
no ipv6 cef
multilink bundle-name authenticated
redundancy
interface Loopback0
 ip address 10.10.3.1 255.255.255.0
interface Ethernet0/0
 description Link to SW2
 ip address 10.10.1.1 255.255.255.0
interface Ethernet0/1
 no ip address
 shutdown
interface Ethernet0/2
 no ip address
 shutdown
interface Ethernet0/3
 no ip address
 shutdown
interface Serial1/0
 no ip address
 shutdown
 serial restart-delay 0
interface Serial1/1
 no ip address
 shutdown
 serial restart-delay 0
interface Serial1/2
 no ip address
 shutdown
 serial restart-delay 0
interface Serial1/3
 no ip address
 shutdown
 serial restart-delay 0
ip forward-protocol nd
no ip http server
no ip http secure-server
control-plane
line con 0
 logging synchronous
line aux 0
line vty 0 4
 login
 transport input all
end
```

This is the end of the discovery lab.

Challenge

1. Which network service is not delivered by Cisco IOS Software?

   A. features to carry the chosen network protocols and functions
   B. connectivity for high-speed traffic between devices
   C. security to control access and prohibit unauthorized network use
   D. scalability to add interfaces and capability as needed for network growth
   E. Microsoft operating system that is called MS-DOS

2. How can you generally access Cisco IOS Software services?

   A. using a CLI
   B. using an MS-DOS
   C. using an EXEC session
   D. using a GNOME

3. In the Cisco IOS Software what are the two EXEC access levels? (Choose two.)

   A. admin EXEC
   B. user EXEC
   C. privileged EXEC
   D. basic EXEC
   E. advanced EXEC

4. Which Cisco IOS Software EXEC levels allow a person to access only a limited number of basic monitoring commands?

   A. admin EXEC
   B. user EXEC
   C. privileged EXEC
   D. basic EXEC
   E. advanced EXEC

5. Which Cisco IOS command do you use to change from user EXEC level into privileged EXEC level?

   A. **enable**
   B. **disable**
   C. **admin**
   D. **configure**
   E. **configure terminal**

6. Which Cisco IOS command do you use to change from privileged EXEC level into user EXEC level?

   A. **enable**
   B. **disable**
   C. question mark (**?**)
   D. **exit**

7. Which Cisco IOS command do you use to display configuration in RAM?

   A. **show startup-config**
   B. **show config**
   C. **show ram-config**
   D. **show running-config**

Answer Key

Challenge

1. E
2. A
3. B, C
4. B
5. A
6. B
7. D

Lesson 5: Starting a Switch

Introduction

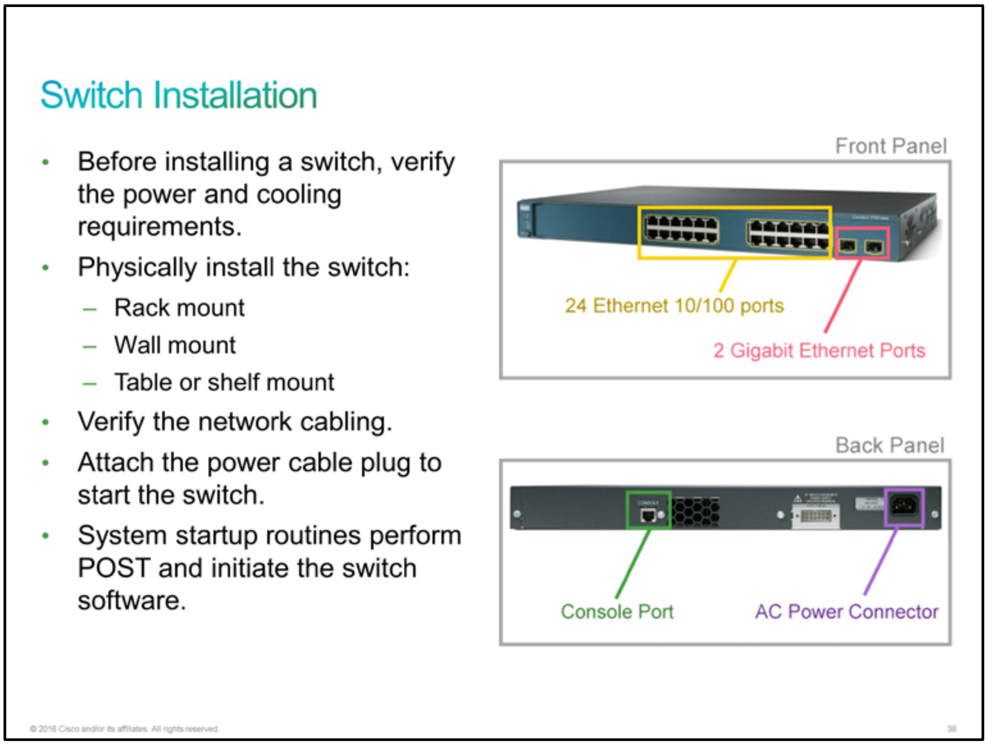
Congratulations, you got the job. You now have the opportunity to perform your first installation. A law firm contracted CCS to install a small network to share printers and other resources, using a switch. Bob tells you that, in addition to performing the physical installation, you will need to be able to specify the hostname, enable the interface, assign a host IP address, and configure the default gateway and interface descriptions.

Bob asks if you would like to go on site to set up the switch or review one or more topics before performing the tasks.

You might want to review general requirements for physical switch installation. You should also know how to read switch LED indicators to recognize the status of a switch and how to access a switch CLI. Get familiar with accessing the switch CLI and configuration commands. You should also not forget to review the show commands, which enable you to verify the status of the switch.

Switch Installation

Before you physically install a Catalyst switch, you must have the correct power and operating environment. When you have correctly connected cable, you can power up the switch.



Switch Installation

- Before installing a switch, verify the power and cooling requirements.
- Physically install the switch:
  - Rack mount
  - Wall mount
  - Table or shelf mount
- Verify the network cabling.
- Attach the power cable plug to start the switch.
- System startup routines perform POST and initiate the switch software.

38

Physical installation and startup of a Catalyst switch requires completion of these steps:

1. Before performing physical installation, verify the following:
   - Switch power requirements
   - Switch operating environment requirements (operational temperature and humidity)
2. Use the appropriate installation procedures for rack mounting, wall mounting, or table or shelf mounting.
3. Before starting the switch, verify that network cable connections are secure.
4. Attach the power cable plug to the power supply socket of the switch. The switch will start. Some Catalyst switches do not have power buttons.
5. Observe the boot sequence:
   - When the switch is on, POST begins. During POST, the switch LED indicators blink while a series of tests determine that the switch is functioning properly.
   - The Cisco IOS Software output text is displayed on the console.

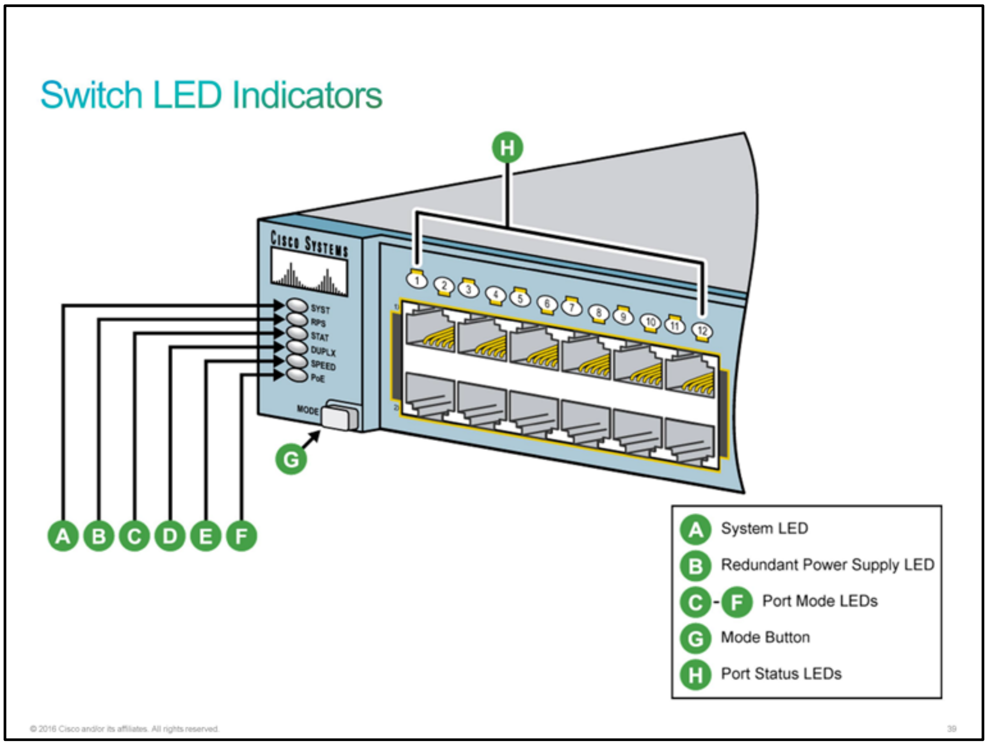
When all startup procedures are finished, the switch is ready to configure.

Switch LED Indicators

Typically, before turning on a device, you need to plug it in. However, some Cisco switches do not have power switches, so when you plug them in, they power up automatically. Because of this fact, you should make sure that the console cable is connected and the terminal program running before you plug in the switch the first time. This preparation will allow you to monitor the boot process of the switch. As the switch powers on, it begins POST, a series of tests that run automatically to ensure that the switch functions properly. Ensuring the switch passes POST is the first step of deploying a switch.

When you need to examine how a switch is working, or to verify its status, and to troubleshoot any problems, you usually mostly use commands from the Cisco IOS CLI. However, the switch hardware does include several LEDs that provide some status and troubleshooting information. Generally, when the Cisco switch is functioning normally, the LEDs are lit in green, and if there is a malfunction, the LEDs are lit in amber.

The figure here shows the front of a Cisco Catalyst 3750 switch, with six LEDs on the left, one LED over each port, and a mode button.



Switch LED Indicators

A System LED

B Redundant Power Supply LED

C - F Port Mode LEDs

G Mode Button

H Port Status LEDs

© 2016 Cisco and/or its affiliates. All rights reserved.

39

LED Status

Letter in Figure	Name	Description
A	SYST	Implies the overall system status.
B	RPS	Suggests the status of the extra (redundant) power supply.
C	STAT	If on (green), each port LED implies the status of that port.

Letter in Figure	Name	Description
D	DUPLX	If on (green), each port LED implies the duplex of that port (on is full duplex; off means half duplex).
E	SPEED	If on (green), each port LED implies the speed of that port, as follows: off means 10 Mbps, solid green means 100 Mbps, and flashing green means 1 Gbps.
F	PoE	Some switches have a PoE LED in the system status group of LEDs. This LED indicates the per-port and system PoE status.
G	MODE	A button that cycles the meaning of the LEDs through three states (STAT, DUPLX, SPEED).
H	Port	Has different meanings, depending on the port mode as toggled using the mode button.

To help make sense of the LEDs, consider the example of the SYST LED for a moment. This LED provides a quick overall status of the switch, with three simple states on most Cisco Catalyst 2960 switch models:

- Off: The switch is not powered on.
- On (green): The switch is powered on and operational. Cisco IOS Software has been loaded.
- On (amber): The switch POST process failed and the Cisco IOS Software did not load.

So, just looking at the SYST LED on the switch tells you whether the switch is working and, if it is not, whether this issue is due to the loss of power (the SYST LED is off) or some kind of POST problem (the LED is amber).

Click the **Play** Button to watch a short video about Cisco catalyst switch LED indicators.

Connecting to a Console Port

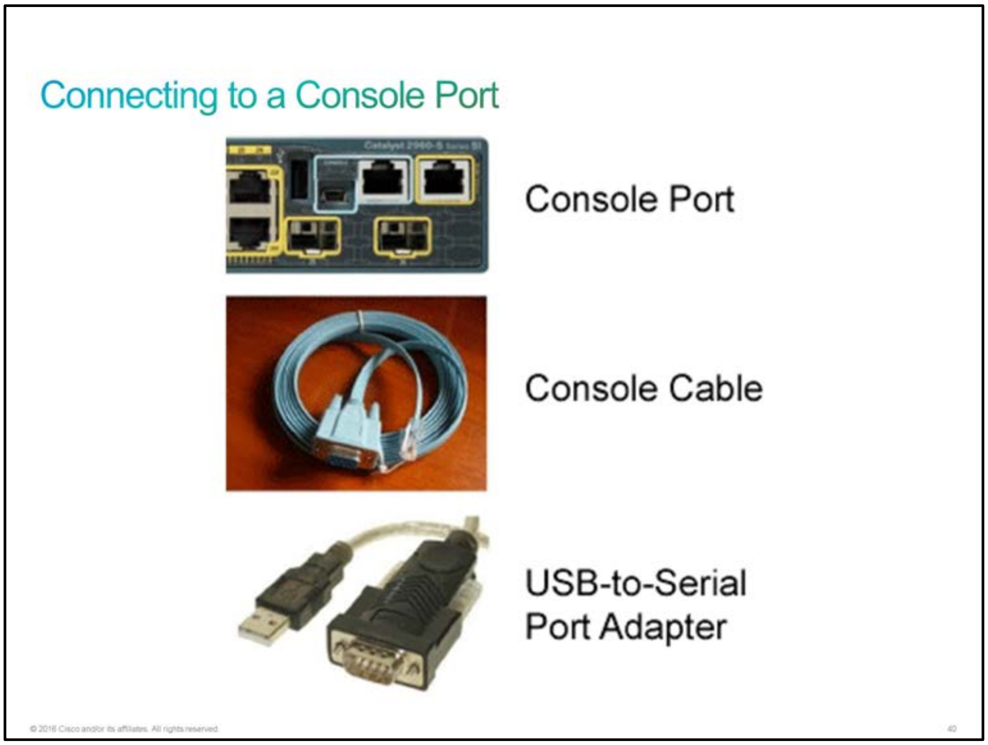
Unlike a computer host, Cisco switches do not have a keyboard, monitor, or mouse device to allow direct user interaction. Upon initial installation, you can configure the switch from a PC that is connected directly through the console port on the switch.

You need the following equipment to access a Cisco device through the console port:

- RJ-45-to-DB-9 console cable
- PC or equivalent with serial port and communications software, such as HyperTerminal, configured with these settings:
  - Speed: 9600 bps
  - Data bits: 8
  - Parity: None
  - Stop bit: 1
  - Flow control: None

Modern computers and notebooks rarely include built-in serial ports. You often use a USB-to-RS-232-compatible serial port adapter instead.

On newer Cisco network devices, a USB serial console connection is also supported. You need a suitable USB cable (USB Type A-to-5-pin mini Type B) and operating system device driver to establish connectivity.



Connecting to a Console Port

Console Port

Console Cable

USB-to-Serial Port Adapter

© 2016 Cisco and/or its affiliates. All rights reserved.

40

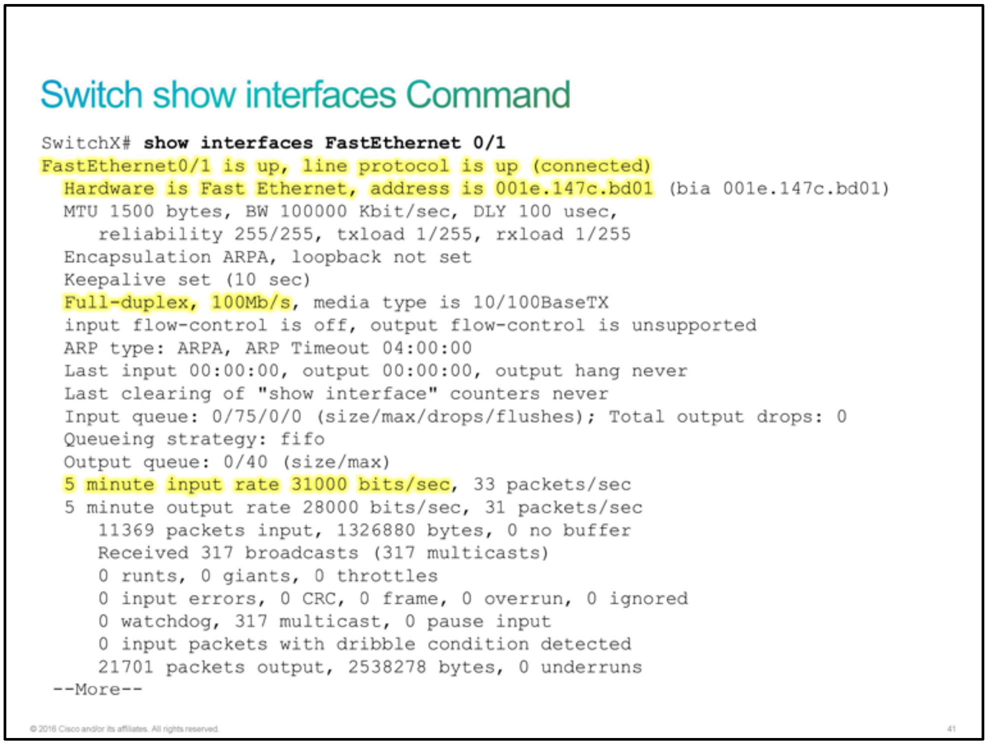
Note	Only one console port can be active at a time. When a cable is plugged into the USB console port, the RJ-45 port becomes inactive. When the USB cable is removed from the USB port, the RJ-45 port becomes active.

When a console connection is established, you gain access to user EXEC mode, by default. To start configuration, you must enter privileged EXEC mode by using the **enable** command.

Basic Show Commands and Information

Switch show version Command

- **show interfaces**: The **show interfaces** command displays the status and statistical information for the network interfaces of the switch. The resulting output varies, depending on the network for which a particular interface has been configured. You usually enter this command with the options *type* and *slot/number*. The *type* option allows values such as **FastEthernet** and **GigabitEthernet.** The *slot/number* option indicates slot 0 and the port number on the selected interface (for example, **fa0/1**).



Switch show interfaces Command

```
SwitchX# show interfaces FastEthernet 0/1
FastEthernet0/1 is up, line protocol is up (connected)
  Hardware is Fast Ethernet, address is 001e.147c.bd01 (bia 001e.147c.bd01)
  MTU 1500 bytes, BW 100000 Kbit/sec, DLY 100 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set
  Keepalive set (10 sec)
  Full-duplex, 100Mb/s, media type is 10/100BaseTX
  input flow-control is off, output flow-control is unsupported
  ARP type: ARPA, ARP Timeout 04:00:00
  Last input 00:00:00, output 00:00:00, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  5 minute input rate 31000 bits/sec, 33 packets/sec
  5 minute output rate 28000 bits/sec, 31 packets/sec
     11369 packets input, 1326880 bytes, 0 no buffer
     Received 317 broadcasts (317 multicasts)
     0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
     0 watchdog, 317 multicast, 0 pause input
     0 input packets with dribble condition detected
     21701 packets output, 2538278 bytes, 0 underruns
 --More--
```

41

The table shows some of the fields in the display that you will find useful for verifying fundamental switch details:

Fundamental Switch Details

Output	Description
FastEthernet0/1 is up, line protocol is up (connected)	Indicates the status of the interface hardware. In this example, it is functioning correctly. The hardware status is followed by the status of the line protocol, which in this example is also operational and active.
Hardware is Fast Ethernet, address is 001e.147c.bd01	Indicates the MAC address of the interface
Full-duplex, 100 Mb/s	Shows the type and mode of connection. Other possibilities include half-duplex, 10 Mb/s.

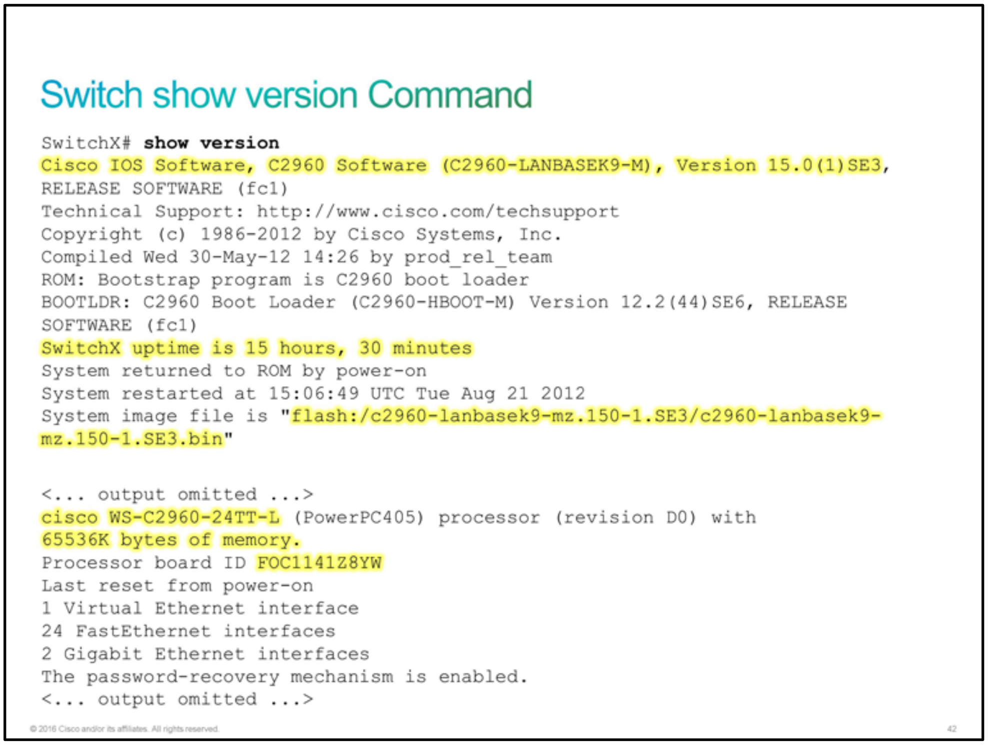
Output	Description
5 minute input rate 31000 bits/sec	Reports interface traffic statistics for average input rate

Note	The **show interfaces** command is used frequently when you are configuring and monitoring network devices.

Switch show interfaces Command

After you log into a Cisco switch, you can verify the switch software and hardware status by using several commands that you execute from privileged EXEC mode. These commands include the **show version**, **show interfaces**, and **show running-config** commands. Here is a look at each of these commands in more detail.

- **show version**: You can use the **show version** IOS command in privileged EXEC mode to verify the IOS version and release numbers of the IOS software that is running on a Cisco switch.



Switch show version Command

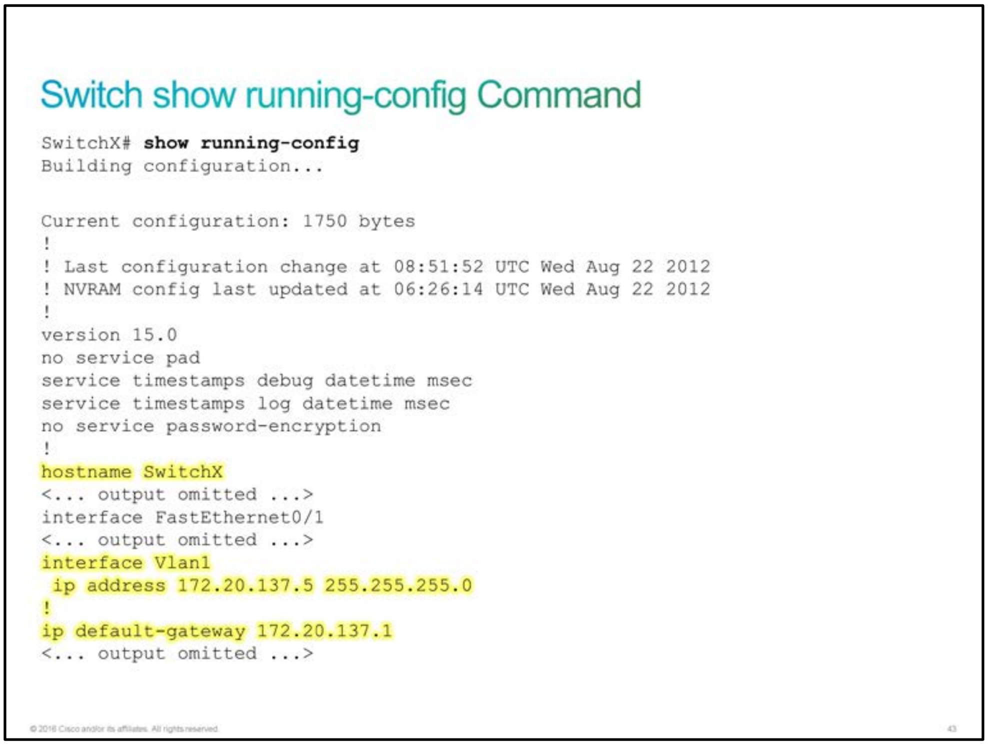
```
SwitchX# show version
Cisco IOS Software, C2960 Software (C2960-LANBASEK9-M), Version 15.0(1)SE3,
RELEASE SOFTWARE (fc1)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2012 by Cisco Systems, Inc.
Compiled Wed 30-May-12 14:26 by prod_rel_team
ROM: Bootstrap program is C2960 boot loader
BOOTLDR: C2960 Boot Loader (C2960-HBOOT-M) Version 12.2(44)SE6, RELEASE
SOFTWARE (fc1)
SwitchX uptime is 15 hours, 30 minutes
System returned to ROM by power-on
System restarted at 15:06:49 UTC Tue Aug 21 2012
System image file is "flash:/c2960-lanbasek9-mz.150-1.SE3/c2960-lanbasek9-
mz.150-1.SE3.bin"

<... output omitted ...>
cisco WS-C2960-24TT-L (PowerPC405) processor (revision D0) with
65536K bytes of memory.
Processor board ID FOC1141Z8YW
Last reset from power-on
1 Virtual Ethernet interface
24 FastEthernet interfaces
2 Gigabit Ethernet interfaces
The password-recovery mechanism is enabled.
<... output omitted ...>
```

The following table describes some of the output fields of the **show version** command:

Output Fields from the show version Command

Output	Description
Cisco IOS Software version	Identification of the software by name and version number Always specify the complete version number when reporting a possible software problem. In this example, the switch is running Cisco IOS Release 15.0(1)SE3.
Switch uptime	Current days and time since the system was last booted In this example, the switch uptime is 15 hours and 30 minutes.
Switch platform	Hardware platform information, including revision and amount of RAM
Processor board ID	Device serial number

Switch show running-config Command



Switch show running-config Command

```
SwitchX# show running-config
Building configuration...

Current configuration: 1750 bytes
!
! Last configuration change at 08:51:52 UTC Wed Aug 22 2012
! NVRAM config last updated at 06:26:14 UTC Wed Aug 22 2012
!
version 15.0
no service pad
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname SwitchX
<... output omitted ...>
interface FastEthernet0/1
<... output omitted ...>
interface Vlan1
 ip address 172.20.137.5 255.255.255.0
!
ip default-gateway 172.20.137.1
<... output omitted ...>
```

© 2016 Cisco and/or its affiliates. All rights reserved. 43

- **show running-config**: The **show running-config** command displays the current running (active) configuration file of the switch. This command requires privileged EXEC mode access. This command displays the IP address, subnet mask, and default gateway settings.

Discovery 2: Perform Basic Switch Configuration

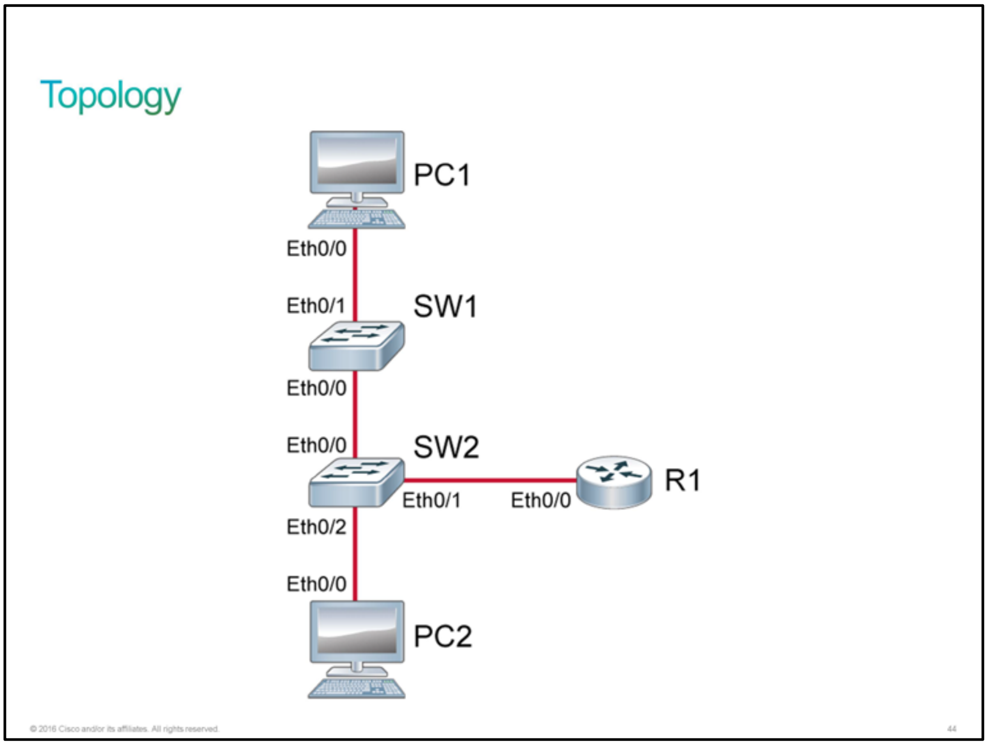
Introduction

This discovery session guides you through the initial configuration of a switch with Cisco IOS Software. The lab is prepared with the devices that are represented in the topology diagram, with the IP addresses as depicted in the table. Note that PC1, PC2, SW2, and R1 are fully configured. In this discovery session, your task will be to provide an initial configuration for SW1. During the session, you will configure and verify each of the following settings on SW1:

- Hostname
- IP address
- Default gateway
- Interface descriptions on the interfaces connecting to PC1 and SW2

You will also verify switch settings by using different **show** commands.

Topology



Job Aid

Device Information

Device Information Table

Device	Characteristic	Value
PC1	Hostname	PC1
PC1	IP address	10.10.1.10/24
PC1	Default gateway	10.10.1.1
PC2	Hostname	PC2
PC2	IP address	10.10.1.20/24
PC2	Default gateway	10.10.1.1
SW1	Hostname	SW1
SW1	VLAN 1 IP address	10.10.1.2/24
SW1	Default gateway	10.10.1.1
SW1	Ethernet0/0 description	Link to SW2
SW1	Ethernet0/1 description	Link to PC1
SW2	Hostname	SW2
SW2	VLAN 1 IP address	10.10.1.3/24
SW2	Default gateway	10.10.1.1
SW2	Ethernet0/0 description	Link to SW1
SW2	Ethernet0/1 description	Link to R1
SW2	Ethernet0/2 description	Link to PC2
R1	Hostname	R1
R1	Ethernet0/0 description	Link to SW2
R1	Ethernet0/0 IP address	10.10.1.1/24
R1	Loopback 0 IP	10.10.3.1/24

PCs in the virtual lab environment are simulated as routers, so you should use Cisco IOS commands to configure them or make verifications.

Task 1: Configure a Switch from the Command Line

Activity

Step 1 Access the console of SW1 and use the **enable** command to access the privileged EXEC.

On SW1, enter the following command:

```
Switch> enable
Switch#
```

You can, of course, use unambiguous abbreviations for commands, such as **en**. You can also take advantage of tab completion using something like **en<tab>**.

The change of the last character in the prompt from > to # is an indication that you have successfully accessed privileged mode.

Step 2 Enter the global configuration mode using the configure terminal command.

On SW1, enter the following command:

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)#
```

The change in prompt to include (config) indicates that you are now in the global configuration mode

Step 3 Set the hostname of the switch to SW1 by using the **hostname** command.

On SW1, enter the following command:

```
Switch(config)# hostname SW1
SW1(config)#
```

The prompt reflects the hostname. It was the default (Switch) and is now set to SW1.

Step 4 In this topology, only VLAN 1 is in use. Set the IP address that SW1 uses on VLAN 1 to 10.10.1.2 with a 24-bit subnet mask. To do so, you will have to enter the interface configuration mode and use the **ip address** command. You will also need to enable the interface with the **no shutdown** command.

On SW1, enter the following command:

```
SW1(config)# interface vlan 1
SW1(config-if)# ip address 10.10.1.2 255.255.255.0
SW1(config-if)# no shutdown
```

Again, the prompt changes as you move through the hierarchy of CLI modes. The prompt includes *config*, indicating that you are in the global configuration mode.

Step 5 Next, set the SW1 default gateway to 10.10.1.1. You do this action from the global configuration mode. Use the **exit** command to return to the global configuration mode, then use the **ip default-gateway** command appropriately.

On SW1, enter the following command:

```
SW1(config-if)# exit
SW1(config)# ip default-gateway 10.10.1.1
```

Again, the prompt changes as you move through the hierarchy of CLI modes. The prompt includes *config*, indicating that you are in the global configuration mode.

Step 6 Finish the configuration requirements by setting the descriptions on interfaces Ethernet 0/0 and Ethernet 0/1, which are links to SW2 and PC1 respectively. The **description** command is available in the interface configuration mode.

On SW1, enter the following command:

```
SW1(config)# interface ethernet 0/0
SW1(config-if)# description Link to SW2
SW1(config-if)# interface ethernet 0/1
SW1(config-if)# description Link to PC1
```

Step 7 SW1 is now properly configured. Leave configuration mode and return to privileged EXEC mode.

On SW1, enter the following command:

```
SW1(config-if)# end
SW1#
```

You can accomplish the same thing in several ways within Cisco IOS. Instead of using the **end** command to go from interface configuration mode all the way back to privileged EXEC, you could also have used the **exit** command twice, or simply pressed **Ctrl-Z**.

Task 2: Verify the Switch Initial Startup Status

Activity

This discovery session assumes that you have just finished configuration of the following settings on SW1:

- Hostname
- IP address
- Default gateway
- Interface descriptions on the interfaces connecting to PC1 and SW2

You will now verify these settings on SW1. Consult the topology diagram and configuration specifications table for the complete connectivity and configuration details. Note that PC1, PC2, SW2, and R1 were already fully configured. In this discovery session, you will focus solely on SW1.

Step 1 On SW1, verify the correct IP address configuration on interface VLAN1. To verify proper IP configuration, you have several options. Normally, you have several ways to verify configuration elements. Often, these include directly viewing the configuration, showing operational status, and verifying behavior. You will utilize all three methodologies here.

On SW1, enter the following command:

One way that you can verify the configuration is by simply viewing it with the **show running-config** command. You can pare down the output of this command by piping it to the include or the section filter. But, since viewing the configuration of a particular interface is a common exercise, you can specify an interface directly to the **show running-config** command. Give the following a try:

```
SW1# show running-config interface vlan 1
Building configuration...

Current configuration : 59 bytes
!
interface Vlan1
 ip address 10.10.1.2 255.255.255.0
end
```

Another option that you have for verifying the IP address configuration is by viewing the status of interfaces. Use the **show ip interface brief** command to see the status of interface VLAN1. Verify that it is up and that the IP address is correct.

```
SW1# show ip interface brief
Interface              IP-Address      OK? Method Status
Protocol
<... output omitted ...>
Vlan1                  10.10.1.2       YES manual up                    up
```

You might find looking at the configuration or at system status useful, but often the most satisfying method is verifying system behavior. If the interface has been properly configured, you should be able to ping other IP addresses on the local subnet. Try to ping PC1, PC2, and R1.

```
SW1# ping 10.10.1.10
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.1.10, timeout is 2 seconds:
.!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 1/2/5 ms
SW1# ping 10.10.1.20
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.1.20, timeout is 2 seconds:
.!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 1/1/1 ms
SW1# ping 10.10.1.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.1.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/202/1009 ms
```

The **ping** command will send five ICMP echo requests and wait for a reply after each request. A period (**.**) indicates a timeout on the reply. An exclamation point (**!**) indicates that the reply was received.

It is common for a timeout to occur on the first echo request, but you may not see it. It happens when the local system does not have an entry in its ARP table for the remote system.

A ping can provide rudimentary performance indications. Timeouts are obviously bad, but the command also displays response time statistics for the replies that were received.

Step 2 Now, verify that the default gateway is configured appropriately. Again, you have multiple options.

On SW1, enter the following command:

View the running configuration, including only lines which include the string "default".

```
SW1# show running-config | include default
ip default-gateway 10.10.1.1
```

Besides looking at the configuration, you verify the status. Use the **show ip route** command to view the IP routing table of SW1.

```
SW1# show ip route
Default gateway is 10.10.1.1
Host               Gateway           Last Use    Total Uses  Interface
ICMP redirect cache is empty
```

Again, you can also verify system behavior. If your default gateway is properly set, you should be able to ping IP addresses on remote subnets. Try to ping the address 10.10.3.1 which is on the other side of R1.

```
SW1# ping 10.10.3.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.3.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/203/1012 ms
```

Step 3 The last thing to verify is the description on the appropriate interfaces. You should have configured descriptions on both Ethernet 0/0 and Ethernet 0/1. As usual, there are multiple strategies that you can use for verification.

On SW1, enter the following command:

View the running configuration, but use the section filter to view sections that include the "0/0" string. Repeat this action for the "0/1" string.

```
SW1# show running-config | section 0/0
interface Ethernet0/0
 description Link to SW2
 duplex auto
SW1# show running-config | section 0/1
interface Ethernet0/1
 description Link to PC1
 duplex auto
```

Verify the system status using the **show interface status** command.

```
SW1# show interface status
Port        Name          Status       Vlan       Duplex  Speed Type
Et0/0       Link to SW2   connected    trunk        auto   auto unknown
Et0/1       Link to PC1   connected    1            auto   auto unknown
<... output omitted ...>
```

This is the end of the discovery lab.

Challenge

1. Which of the following happens first when a switch is powered on?

   A. IOS is booted up
   B. Linux is booted up
   C. POST begins
   D. Command Line appears

2. Some Catalyst switches do not have power ON buttons. True or False?

   A. True
   B. False

3. On some simple Cisco Catalyst Switches, the SYST LED will blink which color when the switch POST process fails and the Cisco IOS Software does not load?

   A. RED
   B. AMBER
   C. GREEN
   D. WHITE

4. Which of the following will help connect you to a Cisco device through the console port?

   A. Console Cable
   B. Serial Cable
   C. USB Cable
   D. RJ45 Cable

5. Which of the following would you use on the PC to connect through the console cable to the Cisco Device?

   A. Command Line
   B. Terminal
   C. HyperTerminal
   D. Web based Application

6. Each port on a Cisco Catalyst Switch has two LEDs to show its status details. True or False?

   A. True
   B. False

7. The console port on a Cisco Device looks like which of the following:

   A. RJ45 port
   B. Serial DB-9 Port
   C. USB port

Answer Key

Challenge

1. C
2. A
3. B
4. A
5. C
6. B
7. A

Lesson 6: Understanding Ethernet and Switch Operation

Introduction

CCS has received an email from a lawyer at the XYZ law firm, for which you recently implemented a small network. The lawyer, Li, explains that XYZ has decided to contract with an IT services company for all its networking needs. As the most computer-savvy employee of the firm, Li has taken on the responsibility of selecting the IT services company and then serving as a liaison between it and the law firm. Because he enjoyed working with you during the network implementation, he wants to give CCS the opportunity to win the contract.

Li is a highly curious individual who takes every opportunity to learn about networking, so in addition to requesting that CCS contact him about providing IT services, he lists several topics that he would like to discuss:

- Ethernet LAN connection media
- Ethernet frame structure
- Ethernet addresses
- Switch operation
- Duplex communication

Hoping to get the contract, Bob asks you to go to the law firm and speak to Li in person.

Before talking to Li, you should feel confident discussing different Ethernet media options, including the most common connectors and cable types. You might want to refresh your knowledge of the Ethernet frame structure and also MAC addresses and their function.

Make sure that you are familiar with switch operation, duplex options, and collision domains.

Ethernet LAN Connection Media

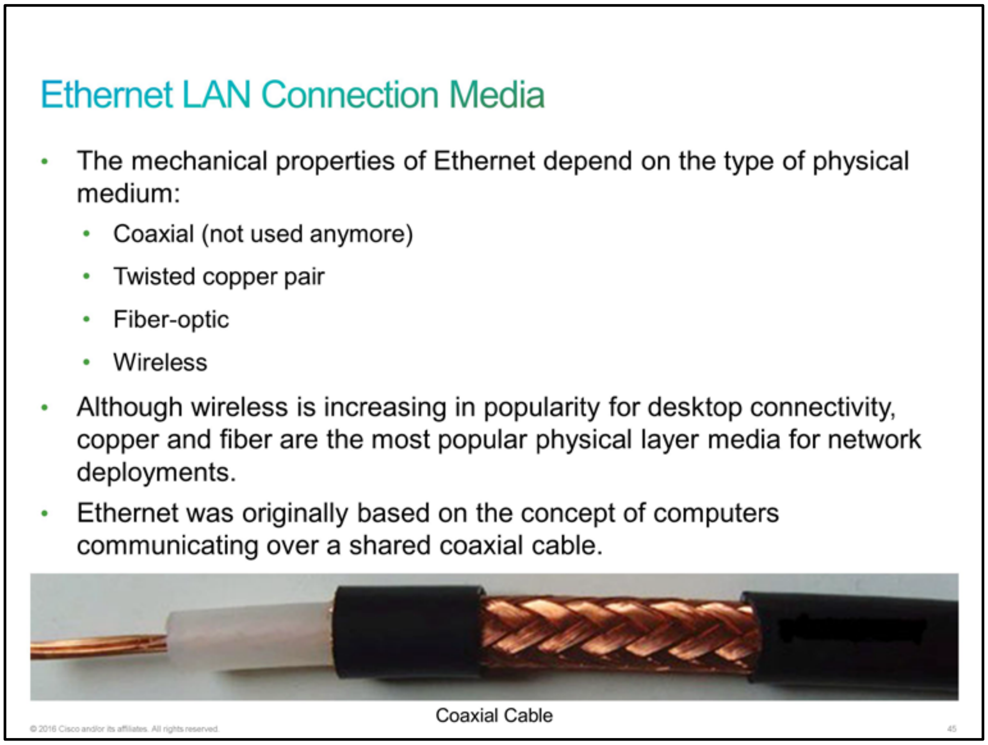
To connect a switch to a LAN, you must use some sort of media. The most common LAN media is Ethernet. Ethernet is not just a type of cable or protocol. It is a network standard that the IEEE published. So you can hear various Ethernet terms, such as Ethernet protocols, Ethernet cables, Ethernet ports, and Ethernet switches. Ethernet is basically a set of guidelines that enable various network components to work together. These guidelines specify cabling and signaling at the physical and data link layers of the OSI model. For example, Ethernet standards recommend different types of cable and specify maximum segment lengths for each type.

The names of the standards (shown in the top row of the table) specify the transmission speed, the type of signaling, and the type of cabling. For example, in the standard name 10BASE-T, the "10" specifies a transmission speed of 10 Mbps, the word "base" refers to baseband signaling (which means that only Ethernet signals are carried on the medium), and the letter "T" represents twisted-pair cabling. Twisted-pair cabling is a type of wiring in which two conductors are twisted together for the purposes of canceling EMI from external sources.

Ethernet Media Standards

Requirement	100BASE-TX	100BASE-FX	1000BASE-T	1000BASE-SX	1000BASE-LX
Media	TIA Category 5 UTP two-pair	62.5/125 micron multimode fiber	TIA Category 5, 5e UTP four-pair	62.5/50 micron multimode fiber	9 micron single-mode fiber
Maximum Segment Length	100 m (328 ft)	400 m (1312.3 ft)	100 m (328 ft)	275 m (62.5 micron) 550 m (50 micron)	5–10 km (1.86–6.2 miles)

Requirement	100BASE-TX	100BASE-FX	1000BASE-T	1000BASE-SX	1000BASE-LX
Connector	ISO 8877 (RJ-45)	Duplex MIC ST	ISO 8877 (RJ-45)	—	—



Ethernet LAN Connection Media

- The mechanical properties of Ethernet depend on the type of physical medium:
  - Coaxial (not used anymore)
  - Twisted copper pair
  - Fiber-optic
  - Wireless
- Although wireless is increasing in popularity for desktop connectivity, copper and fiber are the most popular physical layer media for network deployments.
- Ethernet was originally based on the concept of computers communicating over a shared coaxial cable.

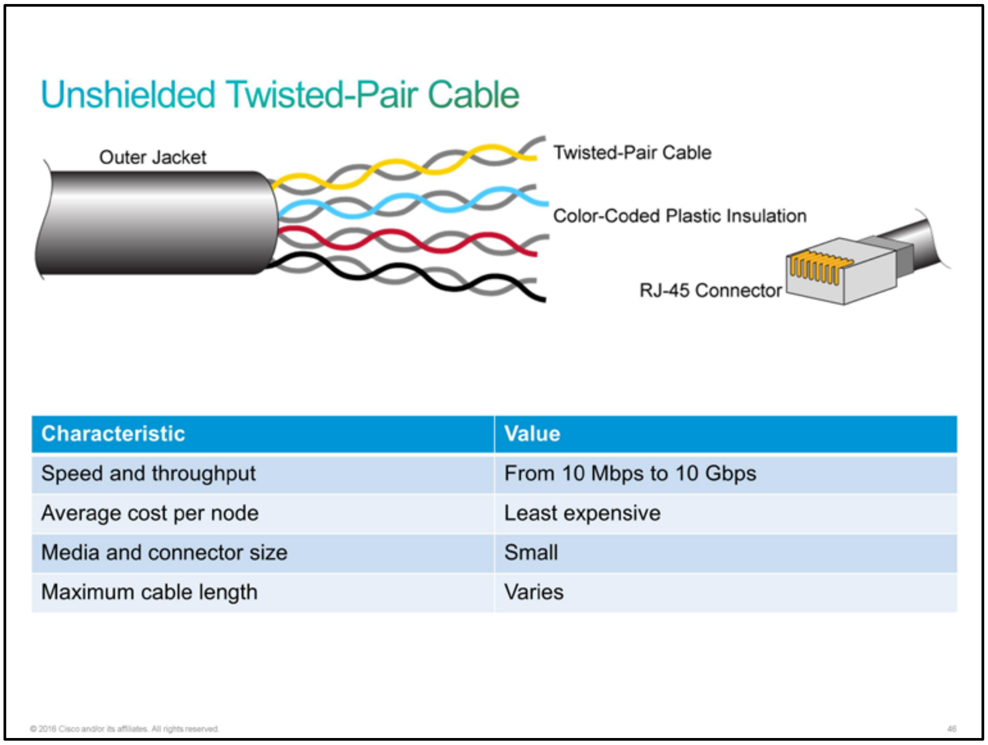
Coaxial Cable

© 2016 Cisco and/or its affiliates. All rights reserved. 45

Copper Media

Take a look at copper media. Most Ethernet networks use UTP copper cabling for short and medium-length distances because of its low cost, when compared to fiber-optic or coaxial cable.

Unshielded Twisted-Pair Cable



Unshielded Twisted-Pair Cable

Characteristic	Value
Speed and throughput	From 10 Mbps to 10 Gbps
Average cost per node	Least expensive
Media and connector size	Small
Maximum cable length	Varies

Ethernet over twisted-pair technologies use twisted-pair cables for the physical layer of an Ethernet computer network. Twisted-pair cabling is a type of wiring in which two conductors (the forward and return conductors of a single circuit) are twisted together for the purposes of canceling EMI from external sources (for example, electromagnetic radiation from UTP cables and crosstalk between neighboring pairs).

A UTP cable is a four-pair wire. Each of the eight individual copper wires in a UTP cable is covered by an insulating material. In addition, the wires in each pair are twisted around each other. The advantage of a UTP cable is its ability to cancel interference, because the twisted-wire pairs limit signal degradation from EMI and RFI. To further reduce crosstalk between the pairs in a UTP cable, the number of twists in the wire pairs varies. Cables must follow precise specifications regarding how many twists or braids are permitted per meter.

A UTP cable is used in various types of networks. When used as a networking medium, a UTP cable has four pairs of either 22- or 24-gauge copper wire. A UTP that is used as a networking medium has an impedance of 100 ohms, differentiating it from other types of twisted-pair wiring such as that used for telephone wiring. A UTP cable has an external diameter of approximately 0.43 cm (0.17 inches), and its small size can be advantageous during installation.

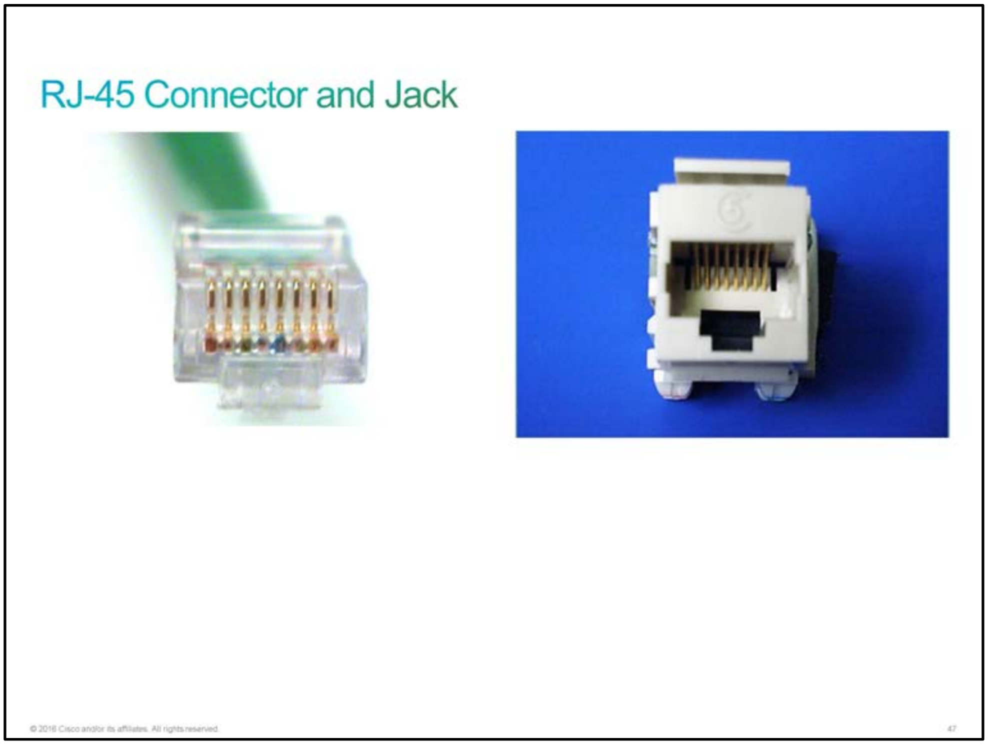
Several categories of UTP cable exist:

- **Category 1:** Used for telephone communications, not suitable for transmitting data
- **Category 2:** Capable of transmitting data at speeds of up to 4 Mbps
- **Category 3:** Used in 10BASE-T networks—can transmit data at speeds of up to 10 Mbps
- **Category 4:** Used in Token Ring networks—can transmit data at speeds of up to 16 Mbps

- **Category 5:** Capable of transmitting data at speeds of up to 100 Mbps
- **Category 5e:** Used in networks running at speeds of up to 1000 Mbps (1 Gbps)
- **Category 6:** Consists of four pairs of 24-gauge copper wires, which can transmit data at speeds of up to 10 Gbps
- **Category 6a:** Used in networks running at speeds of up to 10 Gbps
- **Category 7:** Used in networks running at speeds of up to 10 Gbps

RJ-45 Connector and Jack

UTP cables are used with RJ-45 connectors. The figure shows a UTP cable with an RJ-45 connector and a jack.



RJ-45 Connector and Jack

© 2016 Cisco and/or its affiliates. All rights reserved. 47

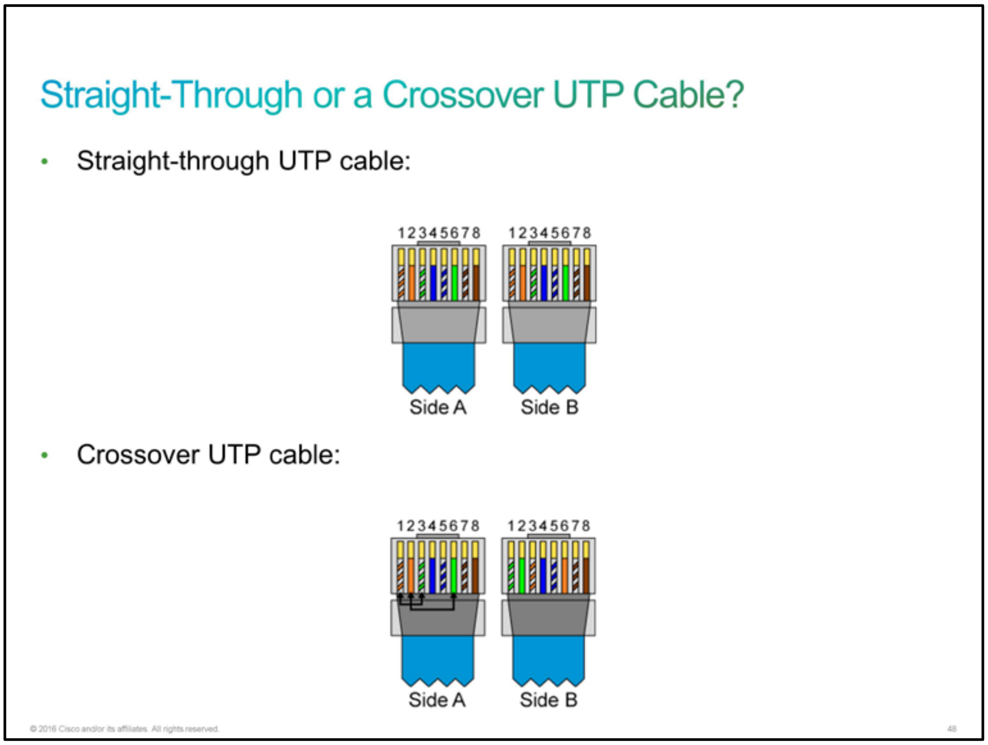
The RJ-45 plug is the male component, which is crimped at the end of the cable. As you look at the male connector from the front, as shown in the figure, the pin locations are numbered from 8 on the left to 1 on the right.

The jack is the female component in a network device, wall, cubicle partition outlet, or patch panel. As you look at the female connector from the front, as shown in the figure, the pin locations are numbered from 1 on the left to 8 on the right.

Straight-Through or a Crossover UTP Cable?

When choosing a UTP cable, you must also determine whether you need a straight-through UTP cable or a crossover UTP cable. Straight-through cables are primarily used for connecting unlike devices, while crossover cables are used for connecting like devices. To tell the difference in the two types of cabling, hold the ends of the cable next to each other with the connector side of each end facing you.

The cable is a straight-though cable if each of the eight pins corresponds to the same pin on the opposite side, as shown in the figure. The cable is a crossover cable if some of the wires on one end of the cable are crossed to a different pin on the other side of the cable, as shown in the figure.

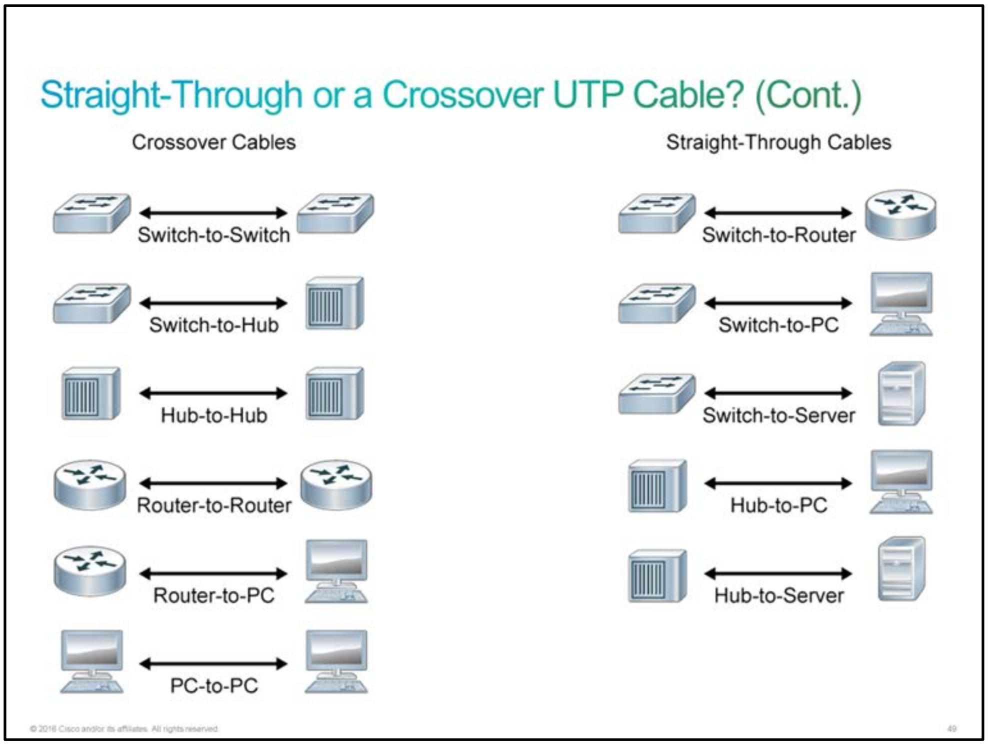


Straight-Through or a Crossover UTP Cable?

- Straight-through UTP cable:

- Crossover UTP cable:

The following figure shows when to use straight-through and crossover cables.



Straight-Through or a Crossover UTP Cable? (Cont.)

Crossover Cables

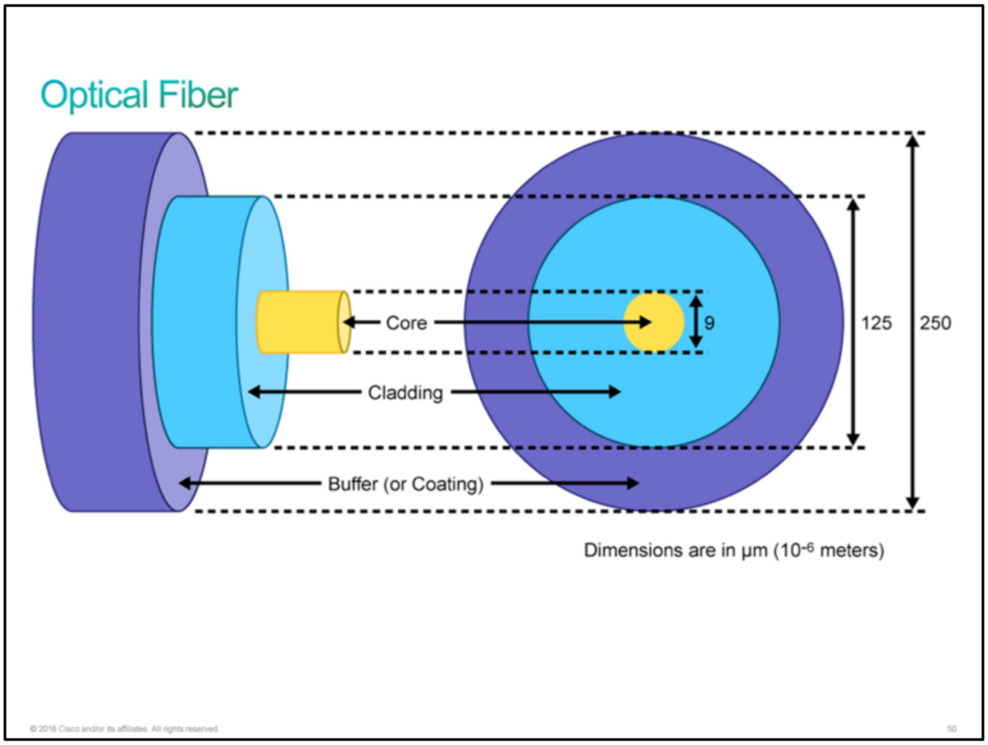
- Switch-to-Switch
- Switch-to-Hub
- Hub-to-Hub
- Router-to-Router
- Router-to-PC
- PC-to-PC

Straight-Through Cables

- Switch-to-Router
- Switch-to-PC
- Switch-to-Server
- Hub-to-PC
- Hub-to-Server

49

Optical Fiber



Optical Fiber

Dimensions are in µm (10^{-6} meters)

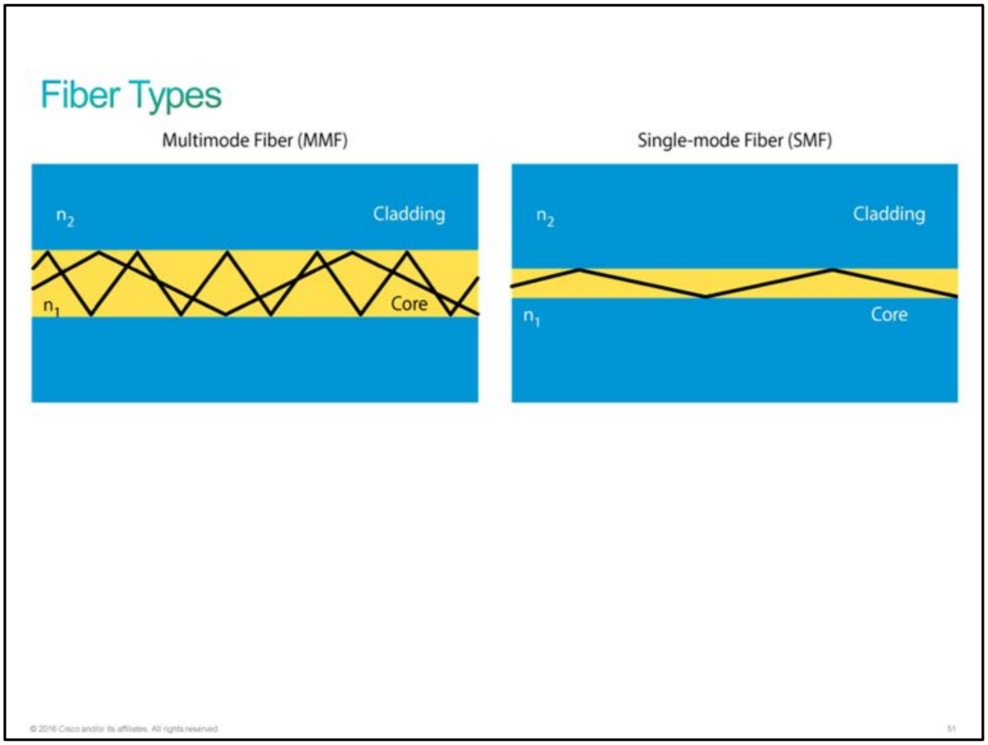
50

An optical fiber is a flexible, transparent fiber that is made of very pure glass (silica) and is not much larger than a human hair. It acts as a waveguide, or "light pipe," to transmit light between the two ends of the fiber. Optical fibers are widely used in fiber-optic communication, which permits transmission over longer distances and at higher bandwidths (data rates) than other forms of communication. Fibers are used instead of metal wires because signals travel along them with less loss and with immunity to electromagnetic interference.

The two fundamental components that allow a fiber to confine light are the core and the cladding. Most of the light travels from the beginning to the end inside the core. The cladding around the core provides confinement. The diameters of the core and cladding are shown in the figure, but the core diameter may vary for various fiber types. In this case, the core diameter of 9 micrometers is very small—the diameter of a human hair is about 50 micrometers. The outer diameter of the cladding is a standard size of 125 micrometers. Standardizing the size means that component manufacturers can make connectors for all fiber-optic cables.

The third element in this picture is the buffer (coating), which has nothing to do with the confinement of the light in the fiber. Its purpose is to protect the glass from scratches and moisture. The fiber-optic cable can be easily scratched and broken, like a glass pane. If the fiber is scratched, the scratch could propagate and break the fiber. Another important aspect is the need to keep the fiber dry.

Fiber Types



Fiber Types

The most significant difference between SMF and MMF is in the ability of the fiber to send light over a long distance at high bit rates. In general, MMF is used for shorter distances at a lower bit rate than SMF. For long-distance communications, SMF is preferred. There are many variations of fiber for both MMF and SMF.

The most significant physical difference is in the size of the core. The glass in the two fibers is the same, and the index of refraction change is similar. The core diameter can make a major difference. The diameter of the fiber cladding is universal for matching fiber ends.

The effect of having different-size cores in the fiber is that the two fiber types will support various ways for the light to get through the fiber. MMF supports multiple ways for the light from one source to travel through the fiber (the source of the designation "multimode"). Each path can be thought of as a mode.

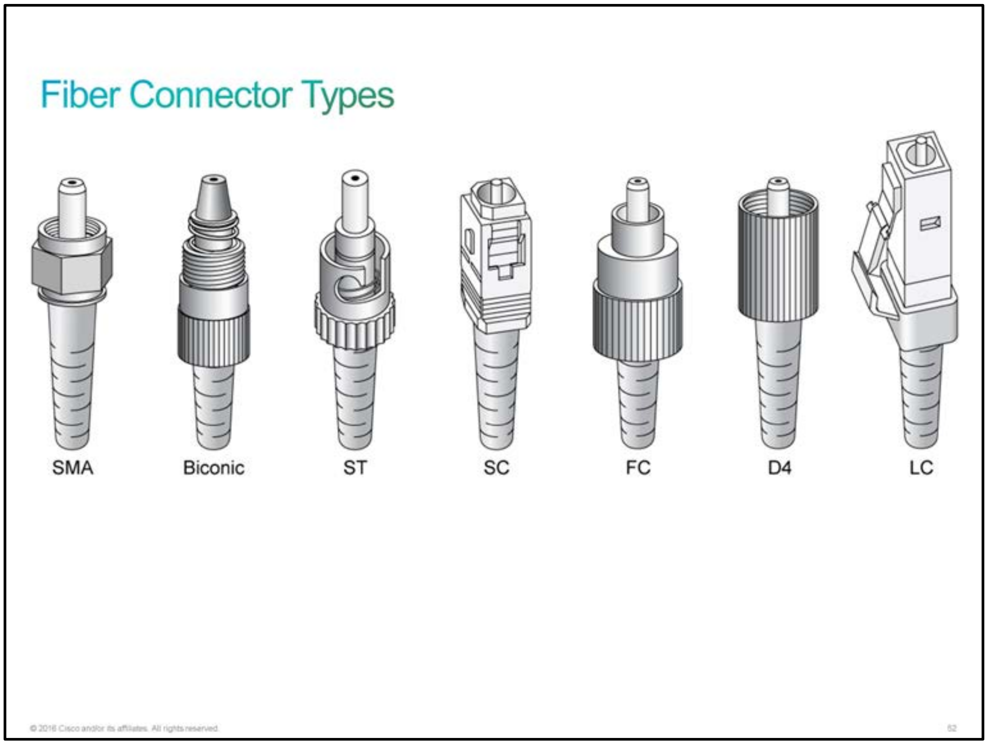
For SMF, the possible ways for light to get through the fiber have been reduced to one, a "single mode." It is not exactly one, but it is a useful approximation.

The table summarizes MMF and SMF characteristics.

MMF and SMF Characteristics

MMF Characteristics	SMF Characteristics
LED transmitter is usually used	Laser transmitter is usually used
Lower bandwidth and speed	Higher bandwidth and speed
Shorter distances	Longer distances
Less expensive	More expensive

Fiber Connector Types



Fiber Connector Types

An optical fiber connector terminates the end of an optical fiber. Various optical fiber connectors are available. The main differences among the types of connectors are dimensions and methods of mechanical coupling. Generally, organizations standardize on one type of connector, depending on the equipment that they commonly use, or they standardize per type of fiber (one for MMF, one for SMF). There are about 70 connector types in use today.

There are three types of connectors:

- Threaded
- Bayonet
- Push-pull

These materials are used for connectors:

- Metal
- Plastic sleeve

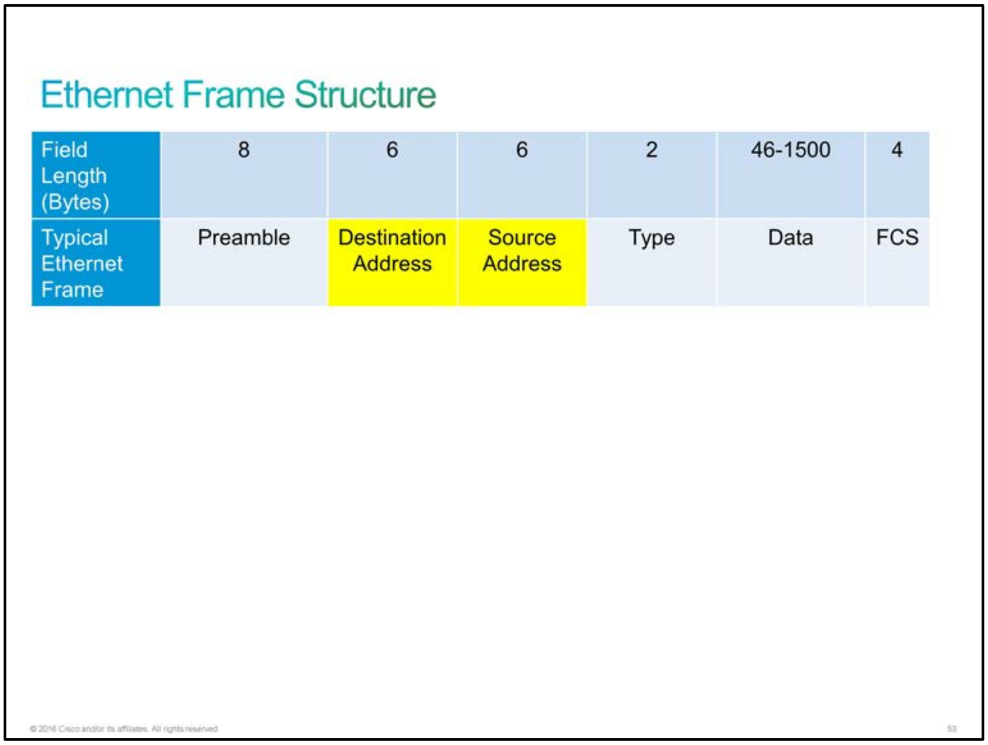
Here you can see the most common types of connectors and their typical uses:

- **ST:** For patch panels (for their durability)
- **FC:** For patch panels; used by service providers
- **SC:** For enterprise equipment
- **LC**: For enterprise equipment, commonly used on SFP modules

In data communications and telecommunications applications today, small-form-factor connectors (for example, LCs) are replacing the traditional connectors (for example, SCs), mainly to pack more connectors on the faceplate and as a result reduce system footprints.

Ethernet Frame Structure

Bits that are transmitted over an Ethernet LAN are organized into frames.



Ethernet Frame Structure

Field Length (Bytes)	8	6	6	2	46-1500	4
Typical Ethernet Frame	Preamble	Destination Address	Source Address	Type	Data	FCS

53

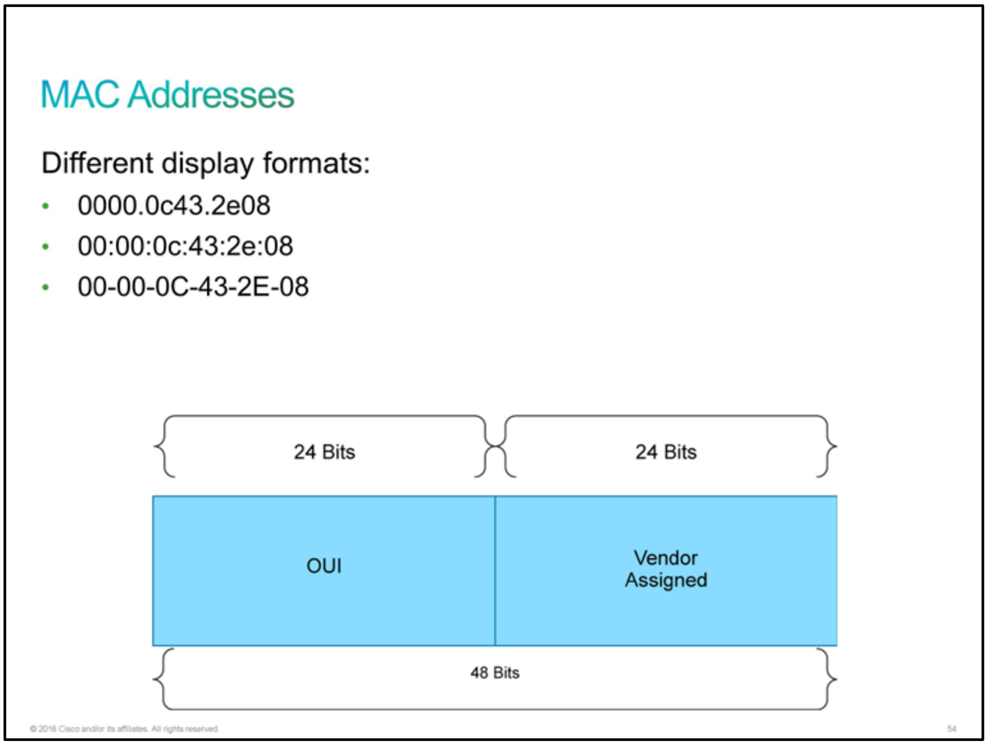
In Ethernet terminology, the container into which data is placed for transmission is called a *frame*. The frame contains header information, trailer information, and the actual data that is being transmitted.

The table shows the most important fields of a MAC layer of the Ethernet frame:

- **Preamble:** This field consists of 8 bytes of alternating 1s and 0s that are used to synchronize the signals of the communicating computers.
- **Destination address:** This field contains the address of the NIC on the local network to which the packet is being sent.
- **Source address:** This field contains the address of the NIC of the sending computer.
- **Type:** This field contains a code that identifies the network layer protocol.
- **Data and pad:** This field contains the data that is received from the network layer on the transmitting computer. This data is then sent to the same protocol on the destination computer. If the data is shorter than the minimum length of 46 bytes, a string of extraneous bits is used to pad the field.
- **FCS:** The FCS field includes a checking mechanism to ensure that the packet of data has been transmitted without corruption.

MAC Addresses

All network devices on the same network must have a unique MAC address. The MAC address is the means by which data is directed to the proper destination device. The MAC address of a device is an address that is burned into the NIC. Therefore, it is also referred to as the physical address or BIA. The MAC address is expressed as groups of hexadecimal digits that are organized in pairs or quads.



MAC Addresses

Different display formats:

- 0000.0c43.2e08
- 00:00:0c:43:2e:08
- 00-00-0C-43-2E-08

Note What is hexadecimal?

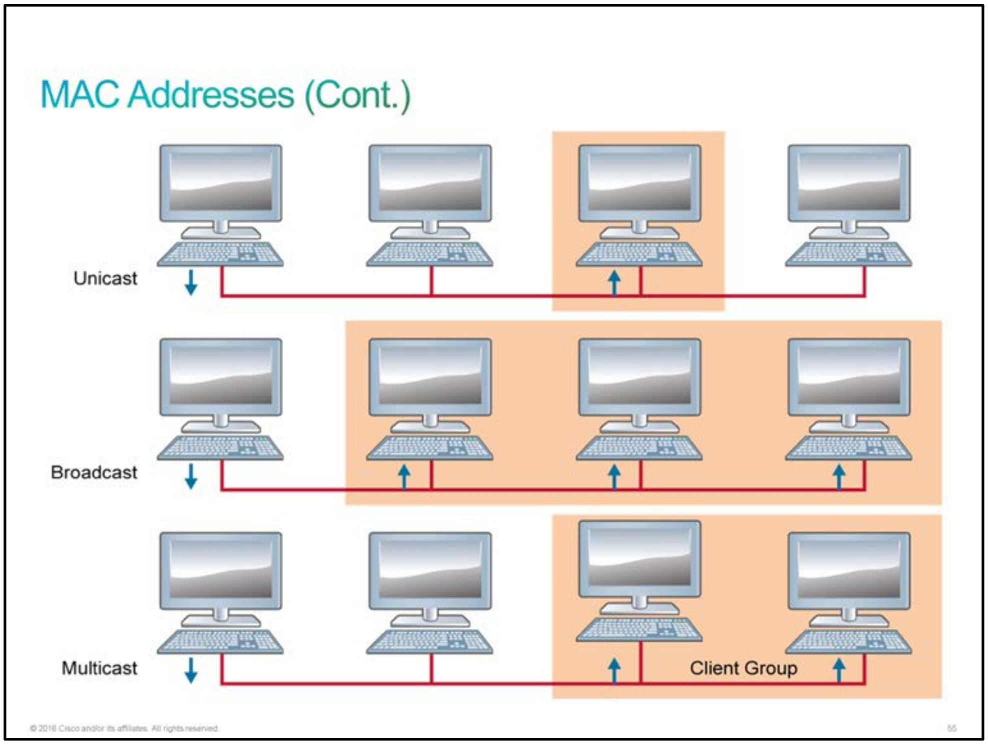
Note Hexadecimal (often referred to as simply *hex*) is a numbering system with a base of 16. This means that it uses 16 unique symbols as digits. The decimal system that you use on a daily basis has a base of 10, which means that it is made up of 10 unique symbols, 0 through 9. The valid symbols in hexadecimal are 0,1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, and F. In decimal, A, B, C, D, E, and F equal 10, 11, 12, 13, 14, and 15. Each hexadecimal digit is 4 bits long, because it requires 4 bits in binary to count to 15. Because a MAC address is made up of 12 hexadecimal digits, it is 48 bits long.

A MAC address is made up of 12 hexadecimal numbers, which means it has 48 bits. There are two main components of a MAC. The first 24 bits constitute the OUI. The last 24 bits constitute the vendor-assigned end-station address.

- **24-bit OUI:** The OUI identifies the manufacturer of the NIC. The IEEE regulates the assignment of OUI numbers. Within the OUI, there are two bits that have meaning only when used in the destination address:
  - **Broadcast or multicast bit:** This bit indicates to the receiving interface that the frame is destined for all or a group of end stations on the LAN segment.

  - **Locally administered address bit:** Normally, the combination of the OUI and a 24-bit station address is universally unique. However, if the address is modified locally, this bit should be set.
- **24-bit vendor-assigned end-station address:** This portion uniquely identifies the Ethernet hardware.

The MAC address identifies the location of a specific computer on a LAN. Unlike other kinds of addresses that are used in networks, the MAC address should not be changed unless there is some specific need to do so.



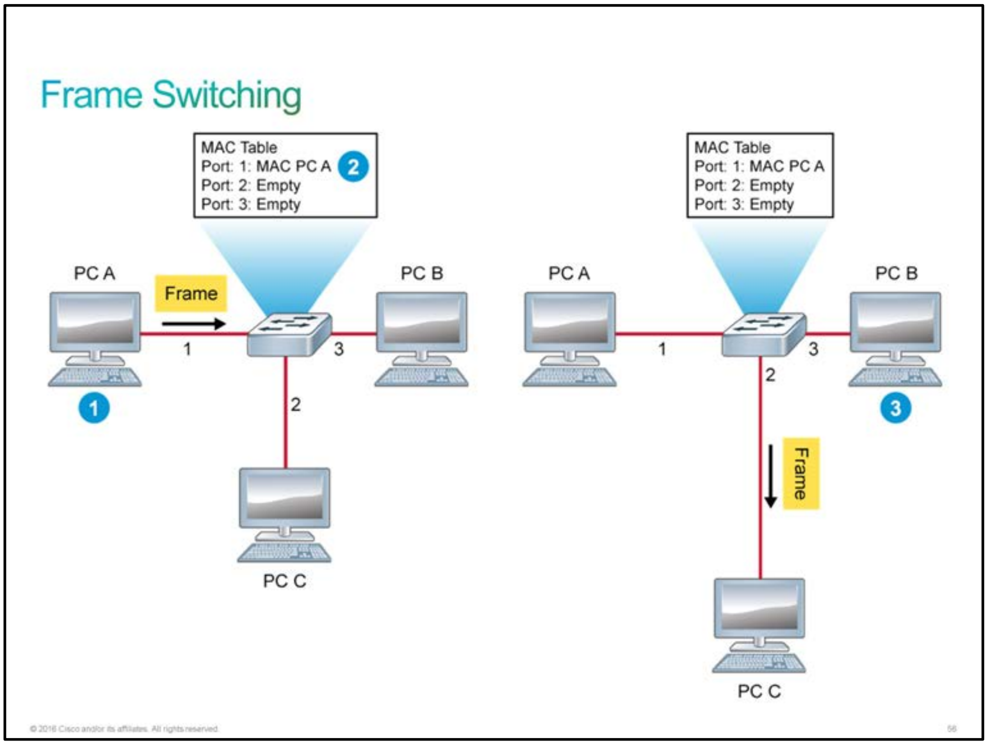
MAC Addresses (Cont.)

There are three major types of network communications:

- **Unicast:** Communication in which a frame is sent from one host and is addressed to one specific destination. In a unicast transmission, there is only one sender and one receiver. Unicast transmission is the predominant form of transmission on LANs and within the Internet.
- **Broadcast:** Communication in which a frame is sent from one address to all other addresses. In this case, there is only one sender, but the information is sent to all the connected receivers. Broadcast transmission is essential for sending the same message to all devices on the LAN.
- **Multicast:** Communication in which information is sent to a specific group of devices or clients. Unlike broadcast transmission, in multicast transmission, clients must be members of a multicast group to receive the information.

Frame Switching

The switch builds and maintains a table, which is called the MAC table, that matches the destination MAC address with the port that is used to connect to a node. The MAC table is stored in the CAM, which enables very fast lookups.

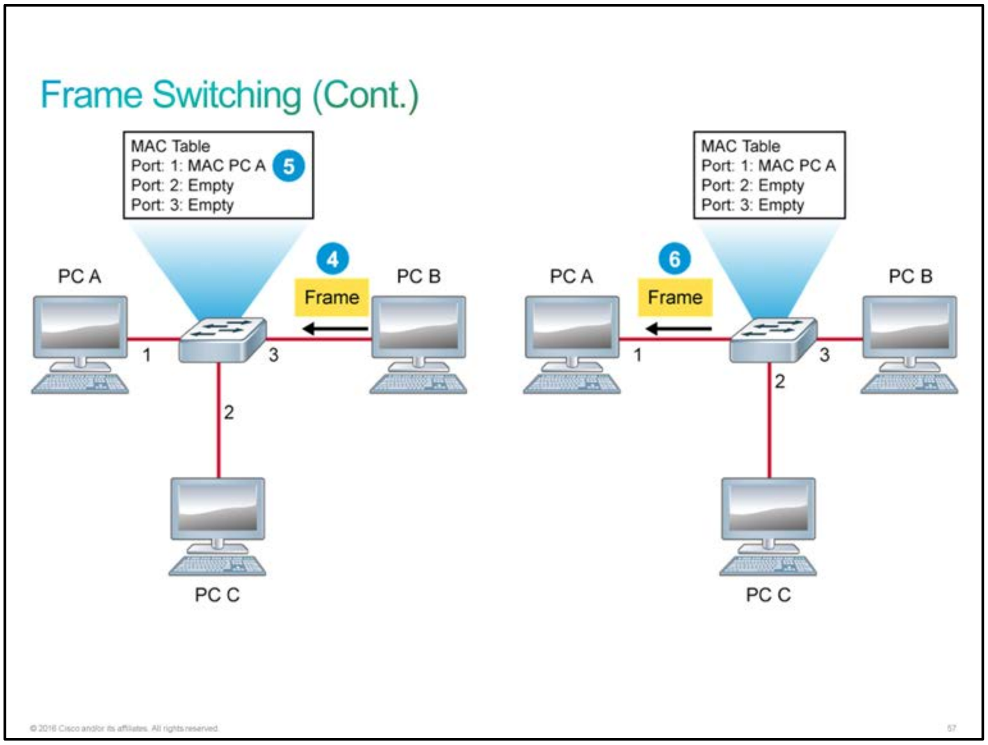


Frame Switching

MAC Table
Port: 1: MAC PC A
Port: 2: Empty
Port: 3: Empty

MAC Table
Port: 1: MAC PC A
Port: 2: Empty
Port: 3: Empty

56



Frame Switching (Cont.)

MAC Table
Port: 1: MAC PC A
Port: 2: Empty
Port: 3: Empty

MAC Table
Port: 1: MAC PC A
Port: 2: Empty
Port: 3: Empty

57

For each incoming frame, the destination MAC address in the frame header is compared to the list of addresses in the MAC table. Switches then use MAC addresses as they decide whether to filter, forward, or flood frames.

The switch creates and maintains a table using the source MAC addresses of incoming frames and the port number through which the frame entered the switch. When an address is not known, a switch learns the network topology by analyzing the source address of incoming frames from all the attached networks. The table describes the switching process:

Switching Frames Procedure

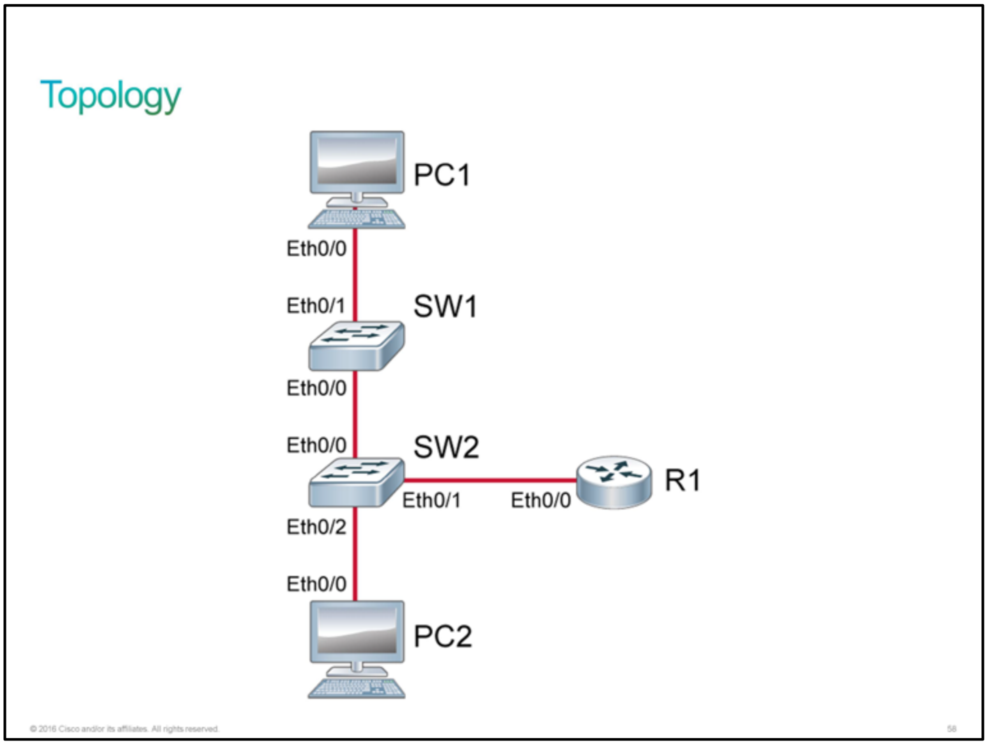
Step	Action
1	The switch receives a frame from PC A on port 1.
2	The switch enters the source MAC address and the switch port that received the frame into the MAC table.
3	The switch checks the table for the destination MAC address. Because the destination address is not known, the switch floods the frame to all the ports except the port on which it received the frame.
4	The destination device with the matching MAC address replies to the unicast with a unicast frame addressed to PC A.
5	The switch enters the source MAC address of PC B and the port number of the switch port that received the frame into the MAC table. The destination address of the frame and its associated port is found in the MAC table.
6	The switch can now forward frames between the source and destination devices without flooding because it has entries in the MAC table that identify the associated ports.

Discovery 3: Observe How a Switch Operates

Introduction

This discovery session will let you observe how a switch maintains its MAC address table, which it uses to control the forwarding of frames. The lab is prepared with the devices that are represented in the topology diagram with the IP addresses as depicted in the table. All devices are fully configured.

Topology



Job Aid

Device Information

Device Information Table

Device	Characteristic	Value
PC1	Hostname	PC1
PC1	IP address	10.10.1.10/24
PC1	Default gateway	10.10.1.1
PC2	Hostname	PC2

Device	Characteristic	Value
PC2	IP address	10.10.1.20/24
PC2	Default gateway	10.10.1.1
SW1	Hostname	SW1
SW1	VLAN 1 IP address	10.10.1.2/24
SW1	Default gateway	10.10.1.1
SW1	Ethernet0/0 description	Link to SW2
SW1	Ethernet0/1 description	Link to PC1
SW2	Hostname	SW2
SW2	VLAN 1 IP address	10.10.1.3/24
SW2	Default gateway	10.10.1.1
SW2	Ethernet0/0 description	Link to SW1
SW2	Ethernet0/1 description	Link to R1
SW2	Ethernet0/2 description	Link to PC2
R1	Hostname	R1
R1	Ethernet0/0 description	Link to SW2
R1	Ethernet0/0 IP address	10.10.1.1/24
R1	Loopback 0 IP	10.10.3.1/24

PCs in the virtual lab environment are simulated as routers, so you should use Cisco IOS commands to configure them or make verifications.

Task 1: Observe How a Switch Operates

Activity

Step 1 First, determine the MAC addresses of the Ethernet0/0 interface on PC1, PC2, and R1.

The **show interface** command displays the MAC address of the interface along with a lot of other information. To reduce the amount of output, allowing you to focus on the line that contains the MAC address, you can pipe the **show interface** output to include filter as shown here.

One at a time, access the console connection to PC1, PC2, and R1 and execute the **show interface** command.

```
PC1# sh int e0/0 | include address
  Hardware is AmdP2, address is aabb.cc00.7600 (bia aabb.cc00.7600)
  Internet address is 10.10.1.10/24


PC2# sh int e0/0 | include address
  Hardware is AmdP2, address is aabb.cc00.7700 (bia aabb.cc00.7700)
  Internet address is 10.10.1.20/24


R1# sh int e0/0 | include address
  Hardware is AmdP2, address is aabb.cc00.7500 (bia aabb.cc00.7500)
  Internet address is 10.10.1.1/24
```

In the emulated environment of the lab, the MAC addresses are similar to each other. This similarity will make it easy to distinguish them as the steps of this discovery progress.

MAC addresses in your output may be different.

Step 2 Access the console of SW2, and enter the **show mac address-table** command.

On SW2, enter the following command:

```
SW2# show mac address-table
          Mac Address Table
-------------------------------------------

Vlan    Mac Address       Type        Ports
----    -----------       --------    -----
   1    aabb.cc00.7500    DYNAMIC     Et0/1
   1    aabb.cc00.7700    DYNAMIC     Et0/2
Total Mac Addresses for this criterion: 2
```

This output is consistent with the information from the Job Aid table. The MAC address that is associated with PC2 is seen on interface Ethernet0/2 and the MAC address that is associated with R1 is associated with interface Ethernet0/1.

PC2 and R1 are both directly connected to SW2 and forward frames very regularly. It is expected that their addresses will remain in the MAC address table almost constantly. You may also see the MAC address of PC1 in the table as well.

Step 3 In this step, be prepared to press **Up Arrow** to use the IOS command recall feature to quickly repeat the **show mac address-table** command after clearing the MAC address table. After clearing the MAC address, table you should find that the MAC address for PC2 and R1 (which are directly connected to SW2) will repopulate themselves quickly.

Clear the MAC address table and use command recall to repeatedly execute the **show mac address-table** command until both addresses are populated. On SW2, enter the following command:

```
SW2# clear mac address-table dynamic
SW2# show mac address-table
          Mac Address Table
-------------------------------------------

Vlan    Mac Address       Type        Ports
----    -----------       --------    -----
   1    aabb.cc00.7500    DYNAMIC     Et0/1
   1    aabb.cc00.7700    DYNAMIC     Et0/2
Total Mac Addresses for this criterion: 2
```

The directly connected systems will populate quickly as they send Ethernet frames to the switch.

The MAC address of PC1, which is one hop away, is not in the table yet.

Step 4 Repeat a similar process on SW1. Clear the MAC address table, and then observe the population of the table.

The MAC address of PC1 should populate in just a few seconds. On SW1, enter the following commands.

```
SW1# clear mac address-table dynamic
SW1# show mac address-table
          Mac Address Table
-------------------------------------------

Vlan    Mac Address       Type        Ports
----    -----------       --------    -----
SW1# show mac address-table
          Mac Address Table
-------------------------------------------

Vlan    Mac Address       Type        Ports
----    -----------       --------    -----
   1    aabb.cc00.7600    DYNAMIC     Et0/1
Total Mac Addresses for this criterion: 1
```

Step 5 Generate some traffic from PC1 to R1 and PC2. This traffic will have to travel across both SW1 and SW2. Because the MAC address of PC1 is not known to SW2 and the MAC addresses of R1 or PC2 are not known to SW1, there will be flooding of initial Ethernet frames. This flooding will happen much too fast to recognize it in the lab. But the final result should be that all three endpoints (PC1, PC2, and R1) appear in the MAC address tables of both switches.

Access the console of PC1 and ping R1.

```
PC1# ping 10.10.1.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.1.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
```

Access the console of PC1 and ping PC2.

```
PC1# ping 10.10.1.20
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.1.20, timeout is 2 seconds:
.!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 1/2/5 ms
```

Step 6 Access the console of SW1 and execute the **show mac address-table** command. Repeat the same on SW2.

On SW1, enter the following command:

```
SW1# show mac address-table
          Mac Address Table
-------------------------------------------

Vlan    Mac Address       Type        Ports
----    -----------       --------    -----
   1    aabb.cc00.7500    DYNAMIC     Et0/0
   1    aabb.cc00.7600    DYNAMIC     Et0/1
   1    aabb.cc00.7700    DYNAMIC     Et0/0
Total Mac Addresses for this criterion: 3
```

On SW2, enter the following command.

```
SW2# show mac address-table
          Mac Address Table
-------------------------------------------

Vlan    Mac Address       Type        Ports
----    -----------       --------    -----
   1    aabb.cc00.7500    DYNAMIC     Et0/1
   1    aabb.cc00.7600    DYNAMIC     Et0/0
   1    aabb.cc00.7700    DYNAMIC     Et0/2
Total Mac Addresses for this criterion: 3
```

In the MAC address table of SW1, the MAC addresses of PC2 and R1 are both associated with the interface Ethernet0/0. Interface Ethernet0/0 is the link to SW2. Any Ethernet frames that are destined for either of these MAC addresses must be forwarded to SW2 for delivery.

In the MAC address table of SW2, the MAC address of PC1 is associated with the interface Ethernet0/0. Interface Ethernet0/0 is the link to SW1. Any Ethernet frames that are destined for this MAC address must be forwarded to SW1 for delivery.

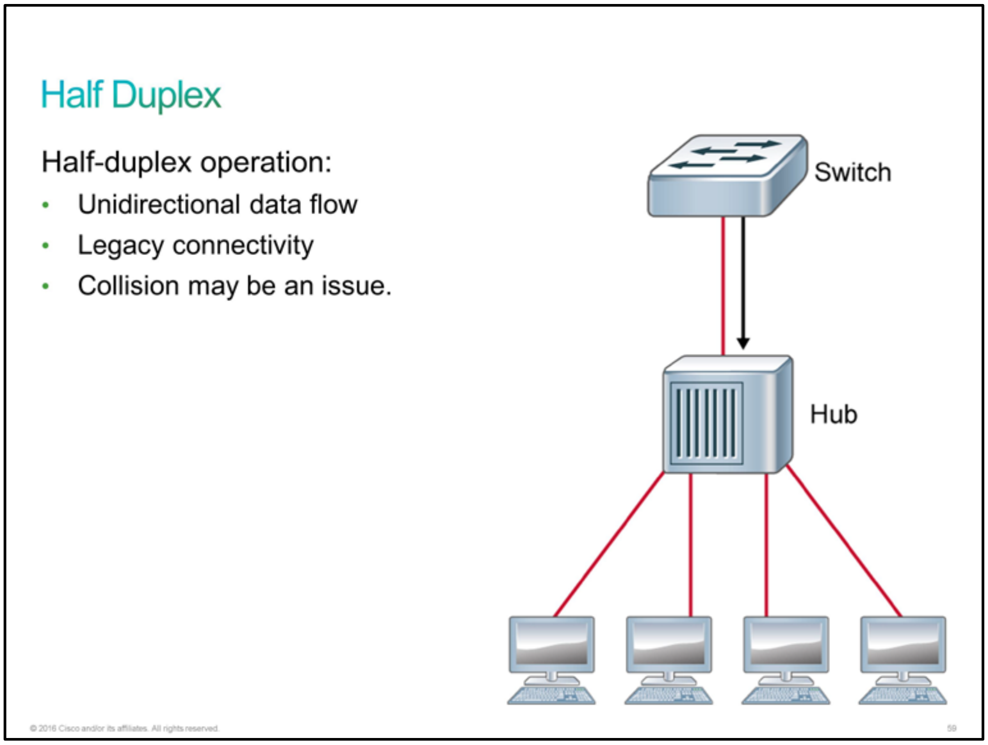
This is the end of the discovery lab.

Duplex Communication

The term *duplex communication* is used to describe a communications channel that can carry signals in both directions, as opposed to a simplex channel, which carries a signal in only one direction. There are two types of duplex settings that are used for communications on an Ethernet network, full duplex and half duplex.

Half Duplex

Half-duplex communication relies on a unidirectional data flow, which means that data can go only in one direction at a time. Sending and receiving data are not performed at the same time. Half-duplex communication is similar to communication with walkie-talkies or two-way radios, in which only one person can talk at a time. Because data can flow in only one direction at a time, each device in a half-duplex system must constantly wait its turn to transmit data. This constant waiting results in performance issues. As a result, full-duplex communication has replaced half duplex in more current hardware. Half-duplex connections are typically seen in older hardware, such as hubs.



Half Duplex

Half-duplex operation:

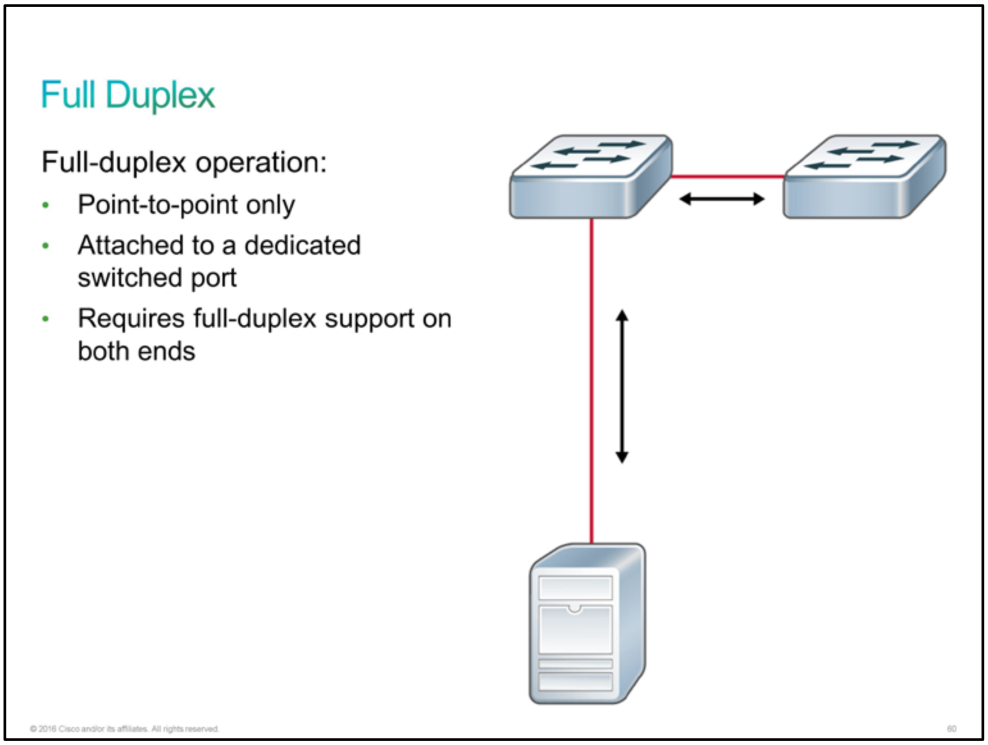
- Unidirectional data flow
- Legacy connectivity
- Collision may be an issue.

59

If a device transmits while another is also transmitting, a collision occurs. Therefore, half-duplex communication implements Ethernet CSMA/CD to help reduce the potential for collisions and to detect them when they do happen. CSMA/CD allows a collision to be detected, which causes the offending devices to stop transmitting. Each device retransmits after a random amount of time has passed. Because the time at which each device retransmits is random, the possibility that they again collide during retransmission is very small.

Full Duplex

Full-duplex communication is like telephone communication, in which each person can talk and hear what the other person says simultaneously. In a full-duplex communication, the data flow is bidirectional, so data can be sent and received at the same time. The bidirectional support enhances performance by reducing the wait time between transmissions. Most Ethernet, Fast Ethernet, and Gigabit Ethernet NICs sold today offer full-duplex capability. In full-duplex mode, the collision-detection circuit is disabled. Frames that the two connected end nodes send cannot collide because the end nodes use two separate circuits in the network cable.



Full Duplex

Full-duplex operation:

- Point-to-point only
- Attached to a dedicated switched port
- Requires full-duplex support on both ends

60

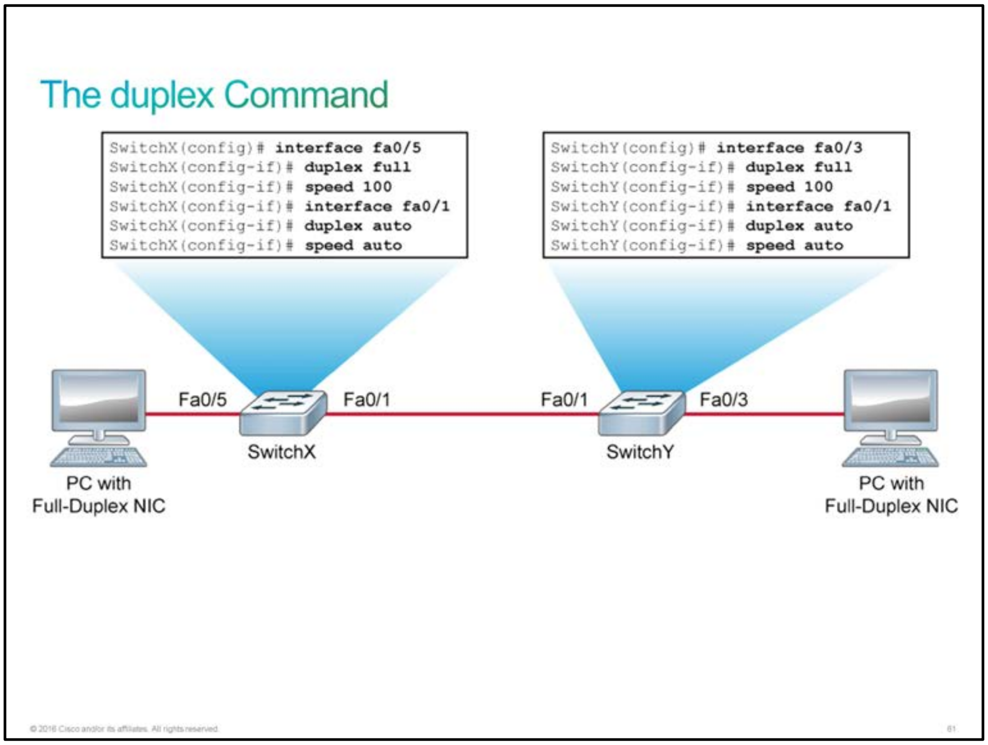
Each full-duplex connection uses only one port. Full-duplex communications require a direct connection between two nodes that both support full duplex. If one of the nodes is a switch, the switch port to which the other node is connected must be configured to operate in the full-duplex mode. The primary cause of duplex issues is mismatched settings on two directly connected devices. For example, the switch is configured for full duplex and the attached PC is configured for half duplex.

The duplex Command

The **duplex** command is used to specify the duplex mode of operation for switch ports. The **duplex** command supports the following options:

- The **full** option sets the full-duplex mode.
- The **half** option sets the half-duplex mode.
- The **auto** option sets autonegotiation of the duplex mode. With autonegotiation enabled, the two ports communicate to decide the best mode of operation.

The figure shows an example of duplex and speed configurations on the Fast Ethernet interfaces of two switches. To prevent mismatch issues, the settings on each interface are configured to match the settings of the directly connected interfaces. For example, interface Fa0/5 on SwitchX is configured for full duplex because it connects to a PC with a full-duplex NIC. The interface Fa0/1 on SwitchX is configured to autonegotiate speed and duplex settings with its neighbor, SwitchY.



The duplex Command

```
SwitchX(config)# interface fa0/5
SwitchX(config-if)# duplex full
SwitchX(config-if)# speed 100
SwitchX(config-if)# interface fa0/1
SwitchX(config-if)# duplex auto
SwitchX(config-if)# speed auto
```

```
SwitchY(config)# interface fa0/3
SwitchY(config-if)# duplex full
SwitchY(config-if)# speed 100
SwitchY(config-if)# interface fa0/1
SwitchY(config-if)# duplex auto
SwitchY(config-if)# speed auto
```

For 100BASE-FX ports, the default option is **full**, and they cannot autonegotiate. 100BASE-FX ports operate only at 100 Mbps in full-duplex mode. For Fast Ethernet and 10/100/1000 ports, the default option is **auto**. The 10/100/1000 ports operate in either half- or full-duplex mode when their speed is set to 10 or 100 Mbps, but when their speed is set to 1000 Mbps, they operate only in the full-duplex mode.

Autonegotiation can at times produce unpredictable results. By default, when autonegotiation fails, a Cisco Catalyst switch sets the corresponding switch port to half-duplex mode. Autonegotiation failure happens when an attached device does not support autonegotiation. If the device is manually configured to also operate in the half-duplex mode, there is no problem. However, if the device is manually configured to operate in the full-duplex mode, there is a duplex mismatch. A duplex mismatch causes late collision errors at the end of the connection. To avoid this situation, manually set the duplex parameters of the switch to match the attached device.

You can use the **show interfaces** command in the privileged EXEC mode to verify the duplex settings on a switch. This command displays statistics and statuses for all interfaces or for the interface that you specify. The following example shows the duplex and speed settings of a Fast Ethernet interface.

```
SwitchX# show interfaces FastEthernet0/5
FastEthernet0/5 is up, line protocol is up (connected)
  Hardware is Fast Ethernet, address is 0022.91c4.0e01 (bia 0022.91c4.0e01)
  MTU 1500 bytes, BW 100000 Kbit, DLY 100 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set
  Keepalive set (10 sec)
  Full-duplex, 100Mb/s, media type is 10/100BaseTX
  input flow-control is off, output flow-control is unsupported
  <... output omitted ...>
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     7289 packets input, 927927 bytes, 0 no buffer
     Received 184 broadcasts (1380 multicasts
     0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
     0 watchdog, 1380 multicast, 0 pause input
     0 input packets with dribble condition detected
     39965 packets output, 7985339 bytes, 0 underruns
     0 output errors, 0 collisions, 1 interface resets
     0 babbles, 0 late collision, 0 deferred
     0 lost carrier, 0 no carrier, 0 PAUSE output
     0 output buffer failures, 0 output buffers swapped out
```

Challenge

1. Which type of physical medium is no longer use for Ethernet?

   A. coaxial
   B. twisted copper pair
   C. fiber-optic
   D. wireless

2. Match the UTP cable category with its description.

category 5	used in 10BASE-T networks—can transmit data at speeds of up to 10 Mbps
category 5e	capable of transmitting data at speeds of up to 100 Mbps
category 6a	used in networks running at speeds of up to 1000 Mbps (1 Gbps)
category 3	used in networks running at speeds up to 10 Gbps

3. Which is not an optical fiber connector type?

   A. threaded
   B. bayonet
   C. push-pull
   D. metal
   E. RJ-45

4. Place the Ethernet frame fields into the correct order.

4.	FCS
3.	preamble
6.	type
5.	source address
1.	destination address
2.	data and pad

5. Which three formats are correct MAC address formats? (Choose three.)

   A. 0000.0c43.2e08
   B. 00:00:0c:43:2e:08
   C. 00-00-0C-43-2E-08
   D. 00000C432E08
   E. 0000-0C43-2E08
   F. 00:00-0c:43-2e:08

6. Match the network communication type with its description.

broadcast	communication in which a frame is sent from one host and addressed to one specific destination
multicast	communication in which a frame is sent from one address to all other addresses
unicast	communication in which information is sent to a specific group of devices or clients

7. Which three characteristics are full-duplex operation characteristics? (Choose three.)

   A. unidirectional data flow
   B. point-to-point only
   C. legacy connectivity
   D. attached to a dedicated switched port
   E. collision may be an issue
   F. requires full-duplex support on both ends

Answer Key

Challenge

1. A
2.

category 3	used in 10BASE-T networks—can transmit data at speeds of up to 10 Mbps
category 5	capable of transmitting data at speeds of up to 100 Mbps
category 5e	used in networks running at speeds of up to 1000 Mbps (1 Gbps)
category 6a	used in networks running at speeds up to 10 Gbps

3. E
4.

6.	FCS
1.	preamble
4.	type
3.	source address
2.	destination address
5.	data and pad

5. A, B, C
6.

unicast	communication in which a frame is sent from one host and addressed to one specific destination
broadcast	communication in which a frame is sent from one address to all other addresses
multicast	communication in which information is sent to a specific group of devices or clients

7. B, D, F

Lesson 7: Troubleshooting Common Switch Media Issues

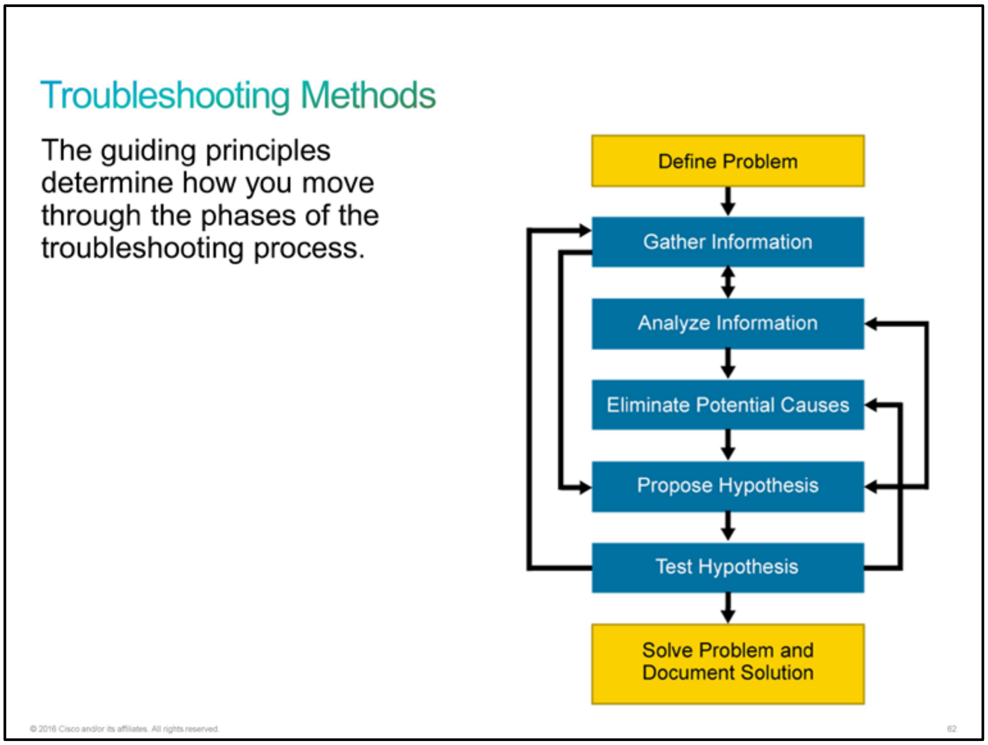
Introduction

The new job is going well and you are taking on more responsibility. You are now doing phone technical support and switch troubleshooting. Lawyer Li calls CCS, complaining of intermittent connectivity since you installed the switch at the law firm. Bob tells you to be prepared to answer Li's questions about the common switched network media issues and port issues, and troubleshoot them if required.

Remember that most issues affecting a switched network are encountered during the original implementation. After a network is installed, it should continue to operate without problems. However, that is only true in theory. Cabling becomes damaged, configurations change, and new devices are connected to the switch, which requires switch configuration changes. Ongoing maintenance is necessary. Before going on site, you might want to review the most common media and port issues and how to troubleshoot them.

Troubleshooting Methods

A troubleshooting method is a guiding principle that determines how you move through the phases of the troubleshooting process



Troubleshooting Methods

The guiding principles determine how you move through the phases of the troubleshooting process.

Define Problem

Gather Information

Analyze Information

Eliminate Potential Causes

Propose Hypothesis

Test Hypothesis

Solve Problem and Document Solution

62

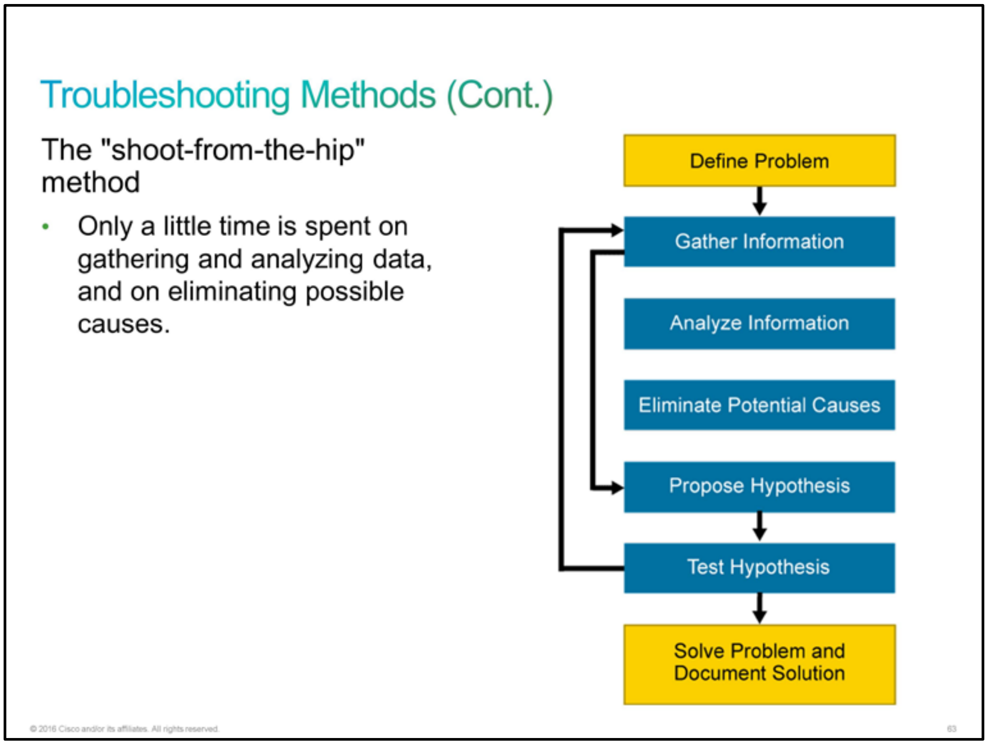
All troubleshooting processes include the elements of gathering and analyzing information, eliminating possible causes, and formulating and testing hypotheses. However, the time one spends on each of those phases, and how one moves from phase to phase, can be significantly different from person to person. It is a key differentiator between effective and less-effective troubleshooters.

In a typical troubleshooting process, for a complex problem, you would continually move between the different processes: gather some information, analyze it, eliminate some possibilities, gather more information, analyze again, formulate a hypothesis, test it, reject it, eliminate some more possibilities, gather more information, and so on.

If you do not use a structured approach but move between the phases randomly, you might eventually find the solution but the process will be very inefficient. In addition, if your approach has no structure, it is practically impossible to hand it over to someone else without losing all the progress that was made up to that point. You also may need to stop and resume your own troubleshooting process.

A structured approach to troubleshooting (no matter what the exact method is) will yield more predictable results in the end and will make it easier to pick up the process where you left off in a later stage or to hand it over to someone else.

Quickly formulating a first hypothesis that is based on common problem causes and corresponding solutions can be very effective in the short run.



Troubleshooting Methods (Cont.)

The "shoot-from-the-hip" method

- Only a little time is spent on gathering and analyzing data, and on eliminating possible causes.

Define Problem

Gather Information

Analyze Information

Eliminate Potential Causes

Propose Hypothesis

Test Hypothesis

Solve Problem and Document Solution

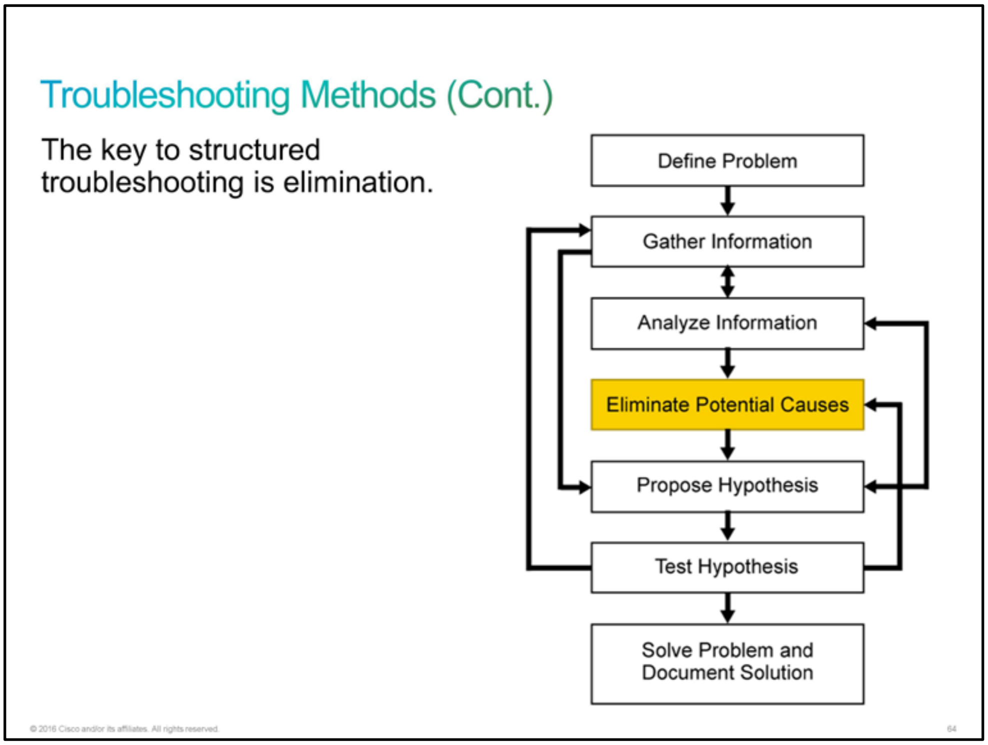
© 2016 Cisco and/or its affiliates. All rights reserved. 63

A troubleshooting method that is commonly deployed by both inexperienced and experienced troubleshooters is the "shoot-from-the-hip" method, where, after a very short period of gathering information, the troubleshooter quickly makes a change to see if it solves the problem. This action might seem like random troubleshooting, but usually, the guiding principle for this method is knowing common symptoms and their corresponding causes.

Look at the following example: A user reports a LAN performance problem to you. In 90 percent of similar problems in the past in this environment, the problem was caused by a duplex mismatch, and the solution was to configure the switch port for 100 Mbps full duplex. An obvious thing to do is to quickly verify the duplex setting of the switch port to which the user connects and to change it to 100 Mbps full duplex to see if that fixes the problem.

When it works, this method can be very effective, because very little time is spent on gathering data, analyzing, and eliminating possible causes. However, the downside is that if it does not work, you have not come any closer to a possible solution.

Experienced troubleshooters can use this method effectively, and it can also be a useful tool for an inexperienced troubleshooter. However, the main factor in using this method effectively is knowing when to stop and switch to a more methodical approach.



Troubleshooting Methods (Cont.)

The key to structured troubleshooting is elimination.

Define Problem

Gather Information

Analyze Information

Eliminate Potential Causes

Propose Hypothesis

Test Hypothesis

Solve Problem and Document Solution

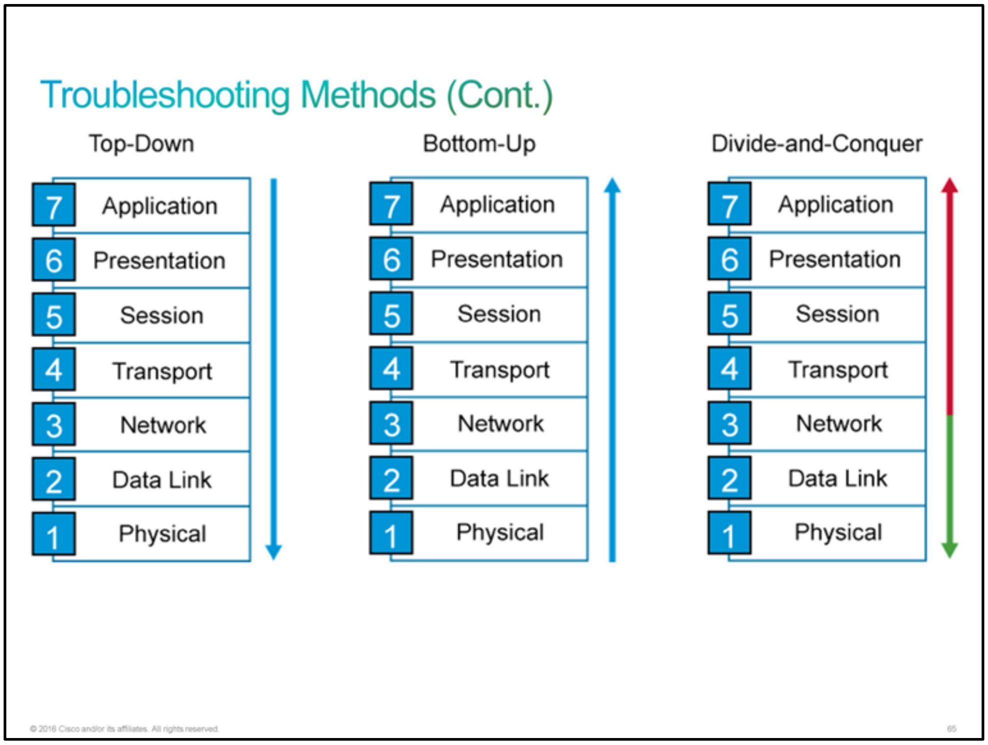
© 2016 Cisco and/or its affiliates. All rights reserved.

64

A structured troubleshooting method is a guideline that helps you move through the different phases of the troubleshooting process. The key to all structured troubleshooting methods is the elimination of the causes.

By systematically eliminating possible problem causes, you can reduce the scope of the problem until you manage to isolate and solve the problem. If it turns out that you lack the knowledge or experience to solve the problem yourself, you can hand it over as a better-defined problem. So, even if you do not manage to solve the problem, you will increase the chances that someone else can find the cause of the problem and resolve it quickly and efficiently.

Several different structured troubleshooting approaches exist and the approach to use may be chosen depending on the problem.

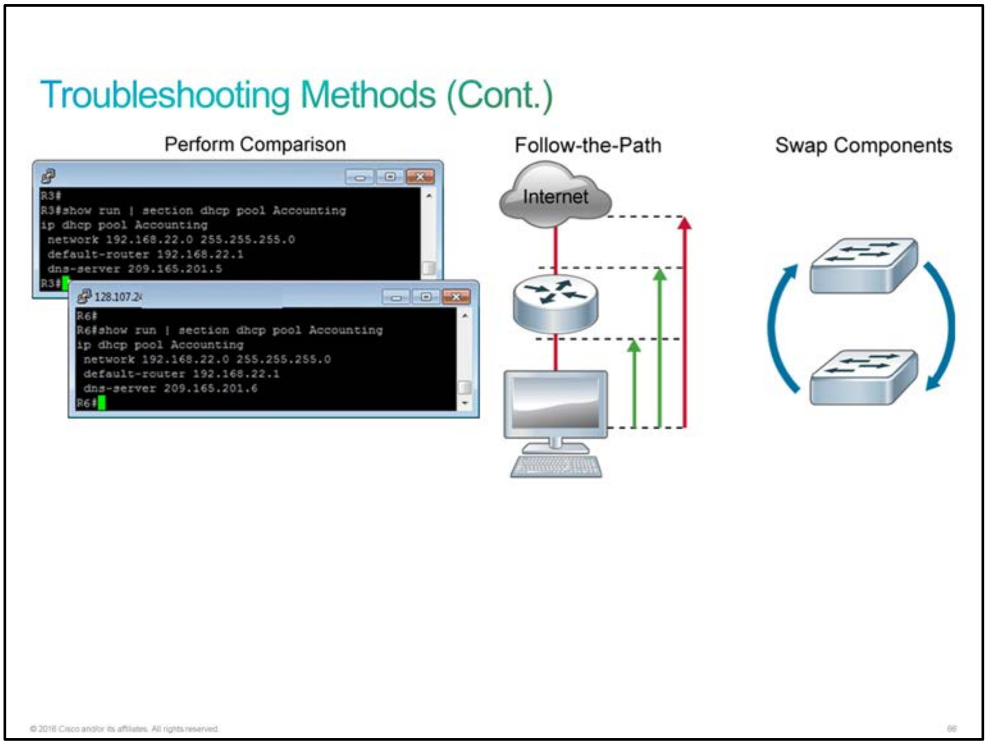


Troubleshooting Methods (Cont.)

Top-Down	Bottom-Up	Divide-and-Conquer
7 Application	7 Application	7 Application
6 Presentation	6 Presentation	6 Presentation
5 Session	5 Session	5 Session
4 Transport	4 Transport	4 Transport
3 Network	3 Network	3 Network
2 Data Link	2 Data Link	2 Data Link
1 Physical	1 Physical	1 Physical

65

- **Top-down method:** Work from the application layer in the OSI model down to the physical layer.
- **Bottom-up method:** Work from the physical layer in the OSI model up to the application layer.
- **Divide-and-conquer method:** Start in the middle of the OSI layers (usually the network layer) and then go up or down, depending on the results.



Troubleshooting Methods (Cont.)

```
R3#
R3#show run | section dhcp pool Accounting
ip dhcp pool Accounting
 network 192.168.22.0 255.255.255.0
 default-router 192.168.22.1
 dns-server 209.165.201.5
R3#
```

```
R6#
R6#show run | section dhcp pool Accounting
ip dhcp pool Accounting
 network 192.168.22.0 255.255.255.0
 default-router 192.168.22.1
 dns-server 209.165.201.6
R6#
```

- **Perform comparison method:** Compare devices or processes of the network that are operating correctly to devices or processes that are not operating as expected. Gather clues by spotting significant differences.
- **Follow-the-path method:** Determine the path that packets follow through the network from the source to the destination and track the packets along the path.
- **Swap components method:** Physically move components and observe if the problem moves with the components or not.

Troubleshooting Tools

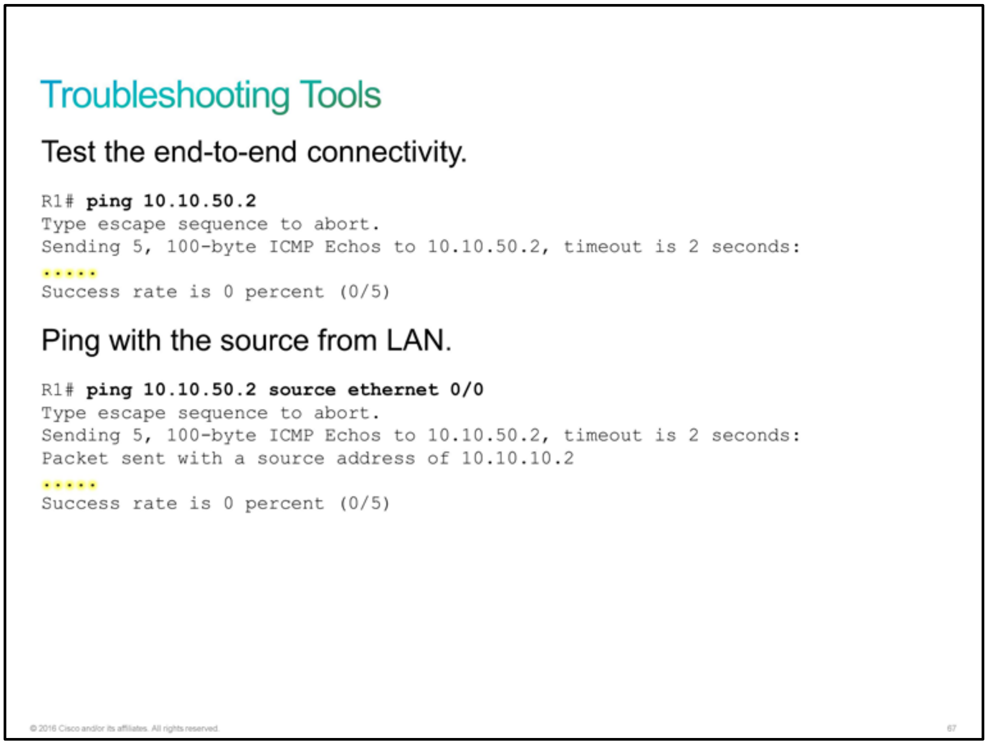
The basic purpose of ping is to check the following aspects:

- Reachability
- RTT
- Packet loss

After sending ICMP echo requests, if an ICMP echo reply packet is received within the default 2-second (configurable) timeout, an "!" is printed, meaning that the reply was received before the timeout expired, or a "." is printed, meaning that the reply was not received before the timeout expired.

The router, as a result, prints the min/avg/max RTT in milliseconds.

Note	When pinging, processing delays can be significant, because the router considers that responding to a ping is a low priority task.



Troubleshooting Tools

Test the end-to-end connectivity.

```
R1# ping 10.10.50.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.50.2, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)
```

Ping with the source from LAN.

```
R1# ping 10.10.50.2 source ethernet 0/0
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.50.2, timeout is 2 seconds:
Packet sent with a source address of 10.10.10.2
.....
Success rate is 0 percent (0/5)
```

© 2016 Cisco and/or its affiliates. All rights reserved. 67

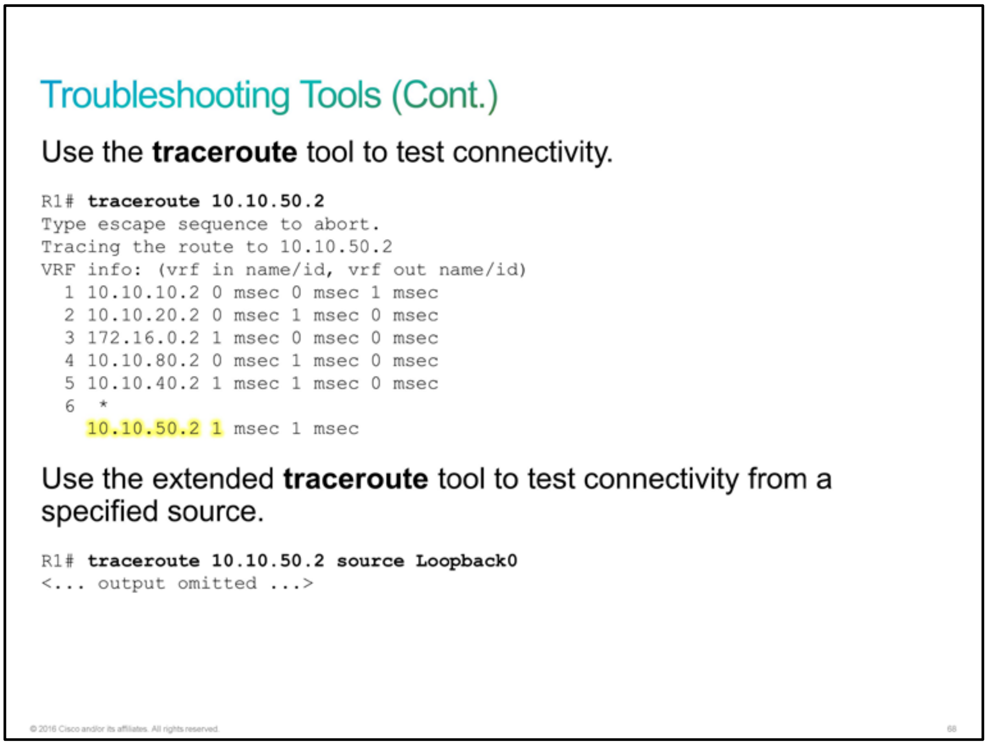
Traceroute works by sending the remote host a sequence of three UDP datagrams with a TTL of 1 in the IP header and the destination ports 33434 (first packet), 33435 (second packet), and 33436 (third packet). The TTL of 1 causes the datagram to "timeout" when it hits the first router in the path. The router responds with an ICMP "time exceeded" message, meaning that the datagram has expired.

The next three UDP datagrams are sent with TTL of 2 to destination ports 33437, 33438, and 33439.

After passing the first router, the datagram arrives at the ingress interface of the second router. The router responds with an ICMP "time exceeded" message.

This process continues until the packet reaches the final destination and all the routers along the path send ICMP "time exceeded," messages.

When the packet reaches the final destination, the device responds with an ICMP "port unreachable."



Troubleshooting Tools (Cont.)

Use the **traceroute** tool to test connectivity.

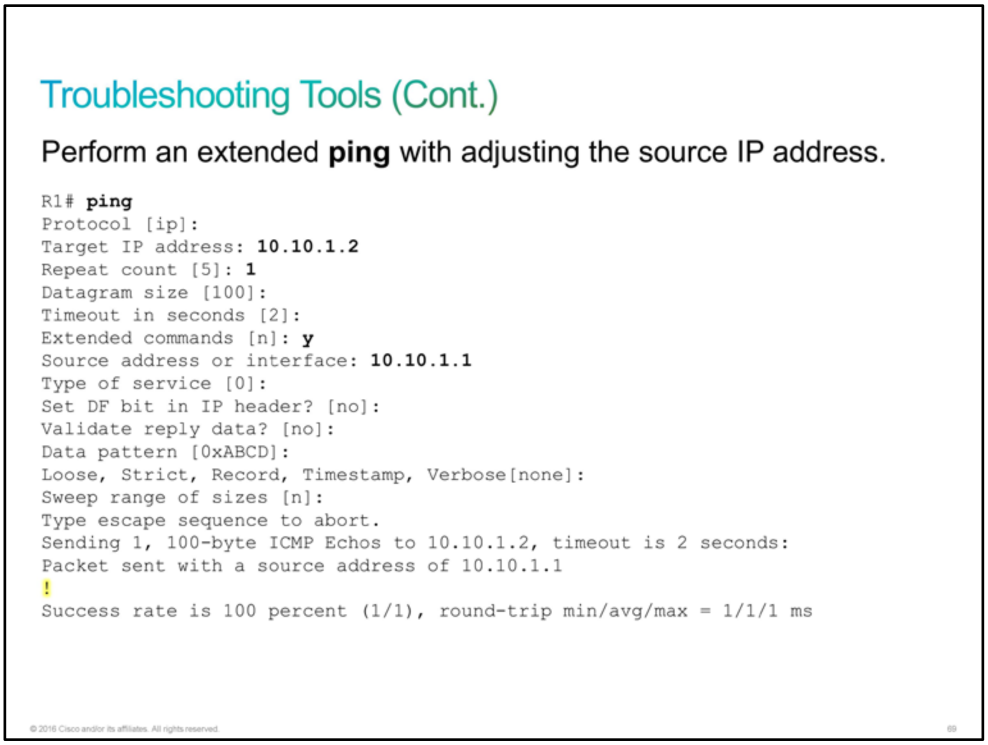
```
R1# traceroute 10.10.50.2
Type escape sequence to abort.
Tracing the route to 10.10.50.2
VRF info: (vrf in name/id, vrf out name/id)
  1 10.10.10.2 0 msec 0 msec 1 msec
  2 10.10.20.2 0 msec 1 msec 0 msec
  3 172.16.0.2 1 msec 0 msec 0 msec
  4 10.10.80.2 0 msec 1 msec 0 msec
  5 10.10.40.2 1 msec 1 msec 0 msec
  6  *
    10.10.50.2 1 msec 1 msec
```

Use the extended **traceroute** tool to test connectivity from a specified source.

```
R1# traceroute 10.10.50.2 source Loopback0
<... output omitted ...>
```

© 2016 Cisco and/or its affiliates. All rights reserved. 68

When a normal **ping** command is sent from a router, the source address of the ping is the IP address of the interface that the packet uses to exit the router. If an extended **ping** command is used, the source IP address can be changed to any IP address on the router.



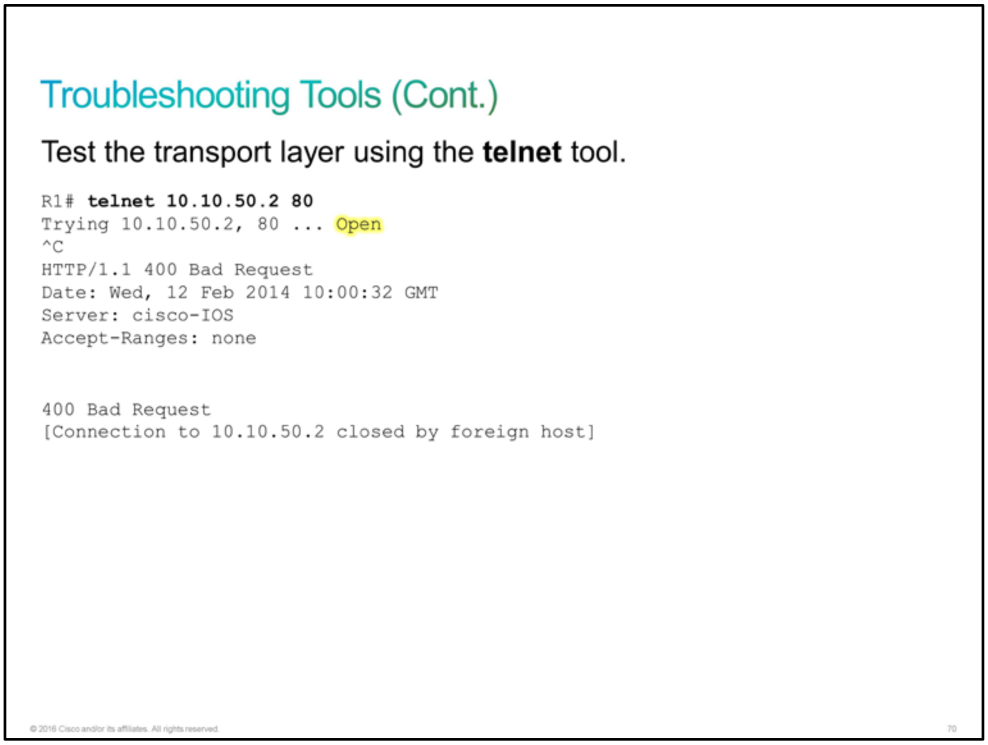
Troubleshooting Tools (Cont.)

Perform an extended **ping** with adjusting the source IP address.

```
R1# ping
Protocol [ip]:
Target IP address: 10.10.1.2
Repeat count [5]: 1
Datagram size [100]:
Timeout in seconds [2]:
Extended commands [n]: y
Source address or interface: 10.10.1.1
Type of service [0]:
Set DF bit in IP header? [no]:
Validate reply data? [no]:
Data pattern [0xABCD]:
Loose, Strict, Record, Timestamp, Verbose[none]:
Sweep range of sizes [n]:
Type escape sequence to abort.
Sending 1, 100-byte ICMP Echos to 10.10.1.2, timeout is 2 seconds:
Packet sent with a source address of 10.10.1.1
!
Success rate is 100 percent (1/1), round-trip min/avg/max = 1/1/1 ms
```

© 2016 Cisco and/or its affiliates. All rights reserved. 69

When you telnet to a remote device, the default port number is used. The default port for telnet is 23. You can use a different port number, from the range 1 to 65535, to test if a remote device is listening to the port.



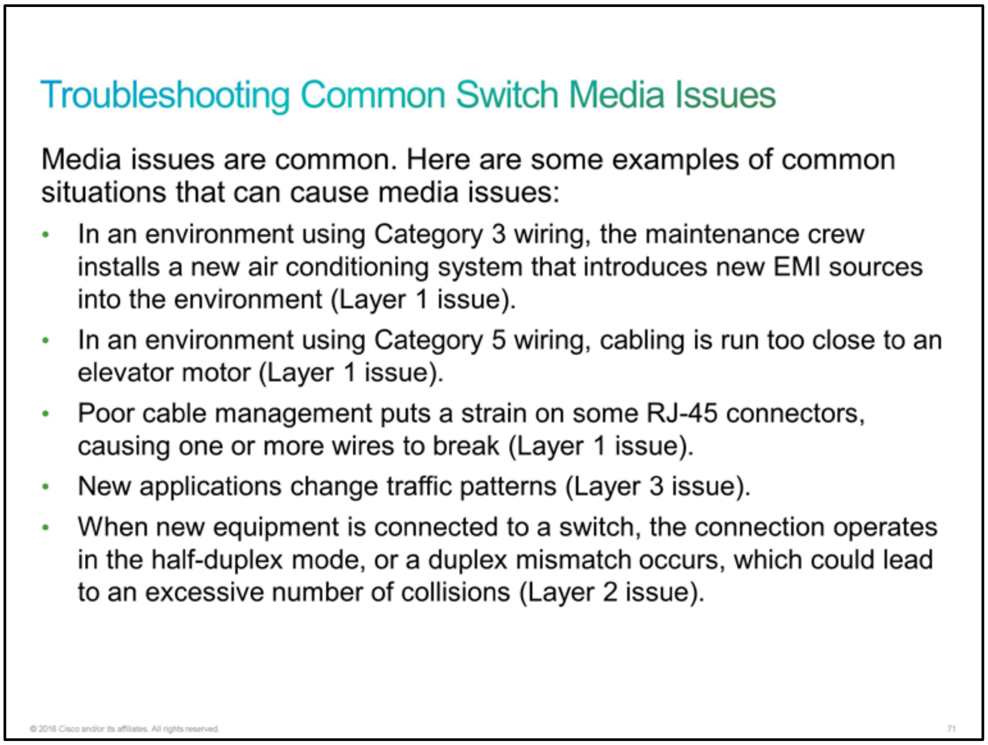
Troubleshooting Tools (Cont.)

Test the transport layer using the **telnet** tool.

```
R1# telnet 10.10.50.2 80
Trying 10.10.50.2, 80 ... Open
^C
HTTP/1.1 400 Bad Request
Date: Wed, 12 Feb 2014 10:00:32 GMT
Server: cisco-IOS
Accept-Ranges: none


400 Bad Request
[Connection to 10.10.50.2 closed by foreign host]
```

Troubleshooting Common Switch Media Issues

Switches operate at multiple layers of the OSI model. At Layer 1 of the OSI model, switches provide an interface to the physical media. At Layer 2 of the OSI model, they provide switching of frames based on MAC addresses. Therefore, switch problems are generally seen as Layer 1 and Layer 2 issues. Layer 3 issues, concerning IP connectivity to the switch for management purposes, could also occur.

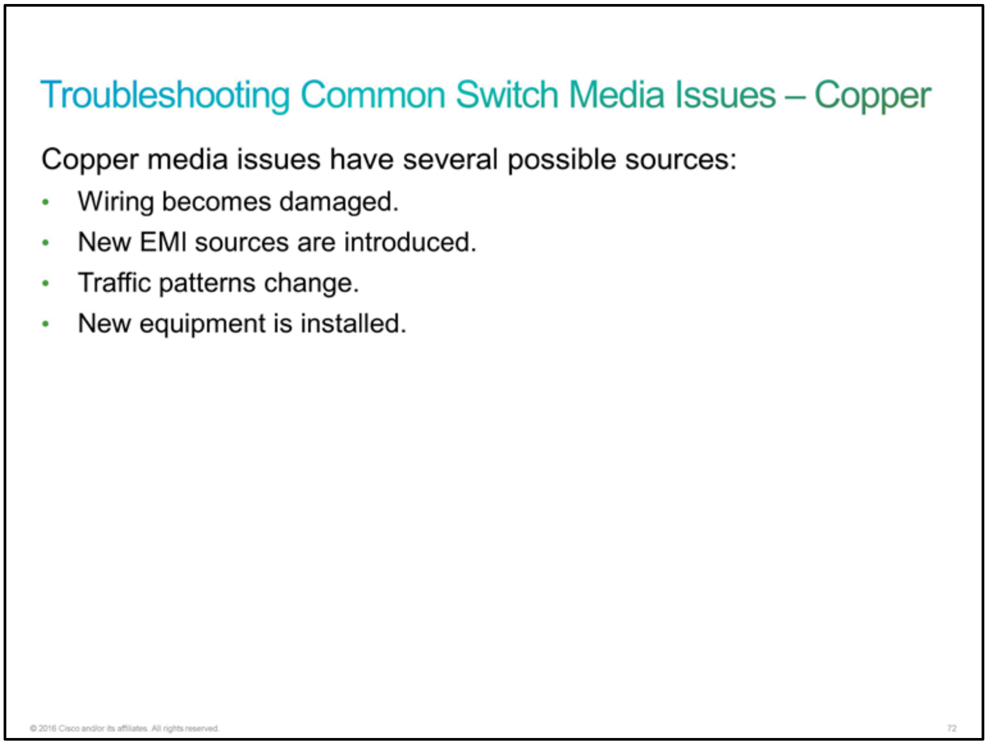


Troubleshooting Common Switch Media Issues

Media issues are common. Here are some examples of common situations that can cause media issues:

- In an environment using Category 3 wiring, the maintenance crew installs a new air conditioning system that introduces new EMI sources into the environment (Layer 1 issue).
- In an environment using Category 5 wiring, cabling is run too close to an elevator motor (Layer 1 issue).
- Poor cable management puts a strain on some RJ-45 connectors, causing one or more wires to break (Layer 1 issue).
- New applications change traffic patterns (Layer 3 issue).
- When new equipment is connected to a switch, the connection operates in the half-duplex mode, or a duplex mismatch occurs, which could lead to an excessive number of collisions (Layer 2 issue).

Troubleshooting Common Switch Media Issues – Copper



Troubleshooting Common Switch Media Issues – Copper

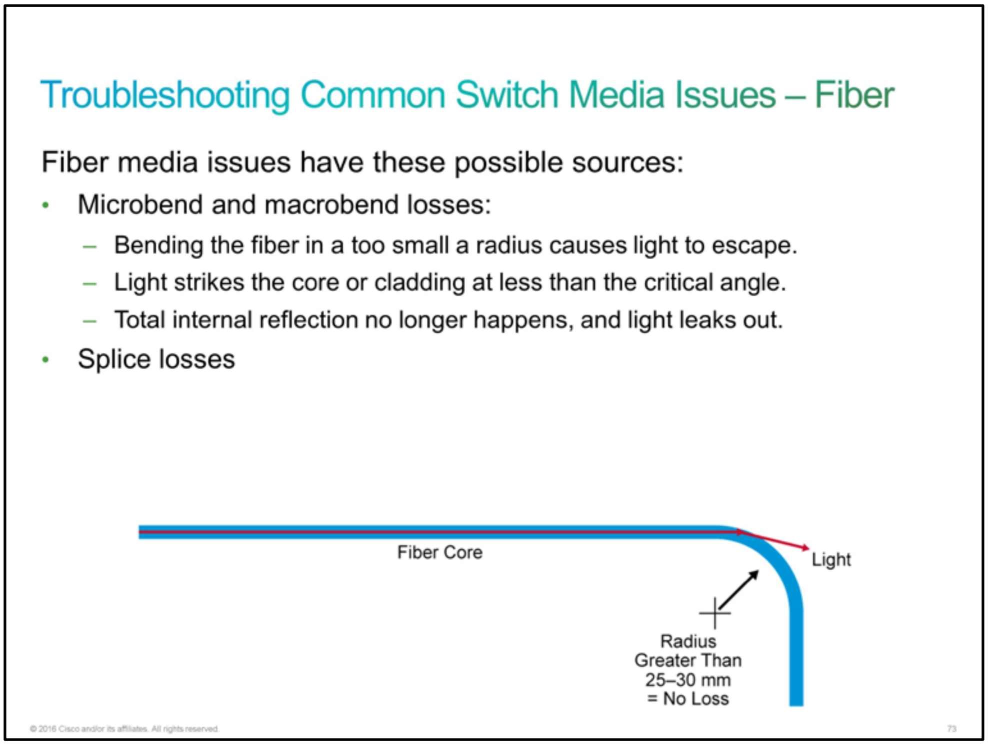
Copper media issues have several possible sources:

- Wiring becomes damaged.
- New EMI sources are introduced.
- Traffic patterns change.
- New equipment is installed.

72

Damaged wiring and EMI commonly show up as excessive collisions and noise. Changes in traffic patterns and the installation of a hub will show up as collisions and runt frames.

Troubleshooting Common Switch Media Issues – Fiber



Troubleshooting Common Switch Media Issues – Fiber

Fiber media issues have these possible sources:

- Microbend and macrobend losses:
  - Bending the fiber in a too small a radius causes light to escape.
  - Light strikes the core or cladding at less than the critical angle.
  - Total internal reflection no longer happens, and light leaks out.
- Splice losses

Fiber Core
Light
Radius
Greater Than
25–30 mm
= No Loss

73

There are several ways in which light can be lost from the fiber. Some are due to manufacturing problems (for example, microbends, macrobends, and splicing fibers that do not have their cores centered), while others are physics problems (back reflections), because light reflects whenever it encounters a change in the index of refraction.

Macrobends are typically applied to the fiber during fiber installation.

There is another explanation for why light leaks out at a macrobend. Part of the traveling wave, which is called the evanescent wave, travels inside the cladding. Around the bend, part of the evanescent wave must travel faster than the speed of light in the material. This occurrence is not possible, so that part radiates out of the fiber.

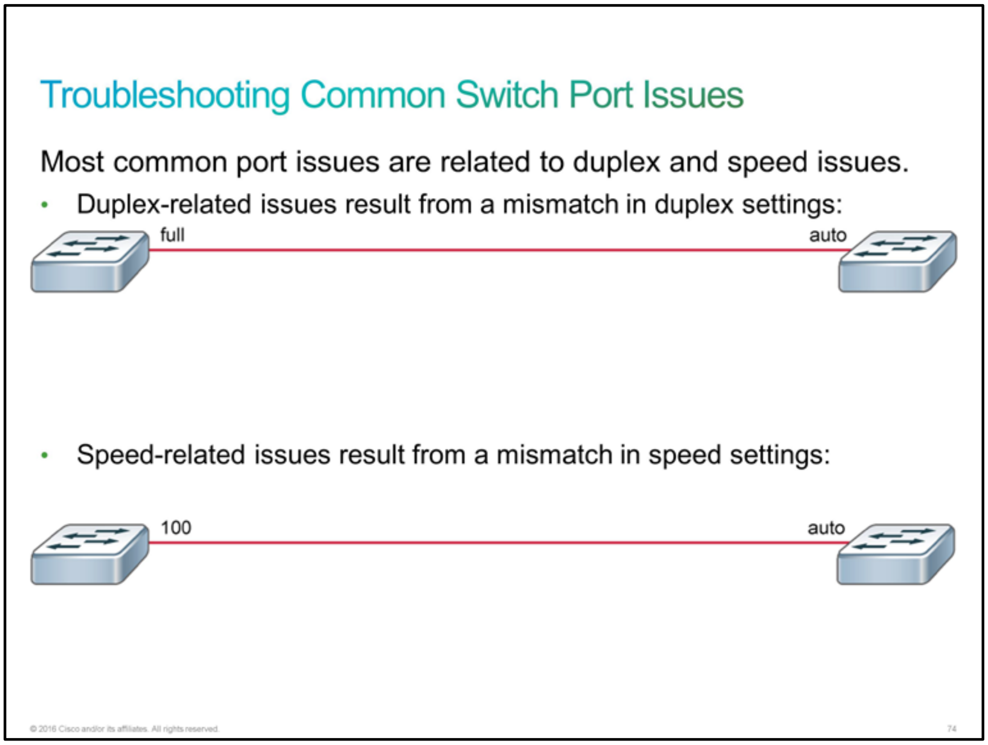
Bend losses can be minimized by designing a larger index difference between the core and the cladding. Another approach is to operate at the shortest possible wavelength and perform good installations.

Splices are a way to connect two fibers by fusing their ends. The best way to align the fiber core is by using the outside diameter of the fiber as a guide. If the core is at the center of the fiber, a good splice can be achieved. If the core is off center, then it is impossible to create a good splice. You would have to cut the fiber further upstream and test again.

Another possibility is that the fibers to be spliced could have dirt on their ends. Dirt can cause many problems, particularly if the dirt intercepts some or all the light from the core. Recall that the core for SMF is only 9 micrometers.

Troubleshooting Common Switch Port Issues

Port access issues will most likely have visible symptoms, such as users being unable to connect to the network. These problems are sometimes related to faulty media and equipment, such as NICs, but more often port issues are related to duplex and speed settings.



Troubleshooting Common Switch Port Issues

Most common port issues are related to duplex and speed issues.

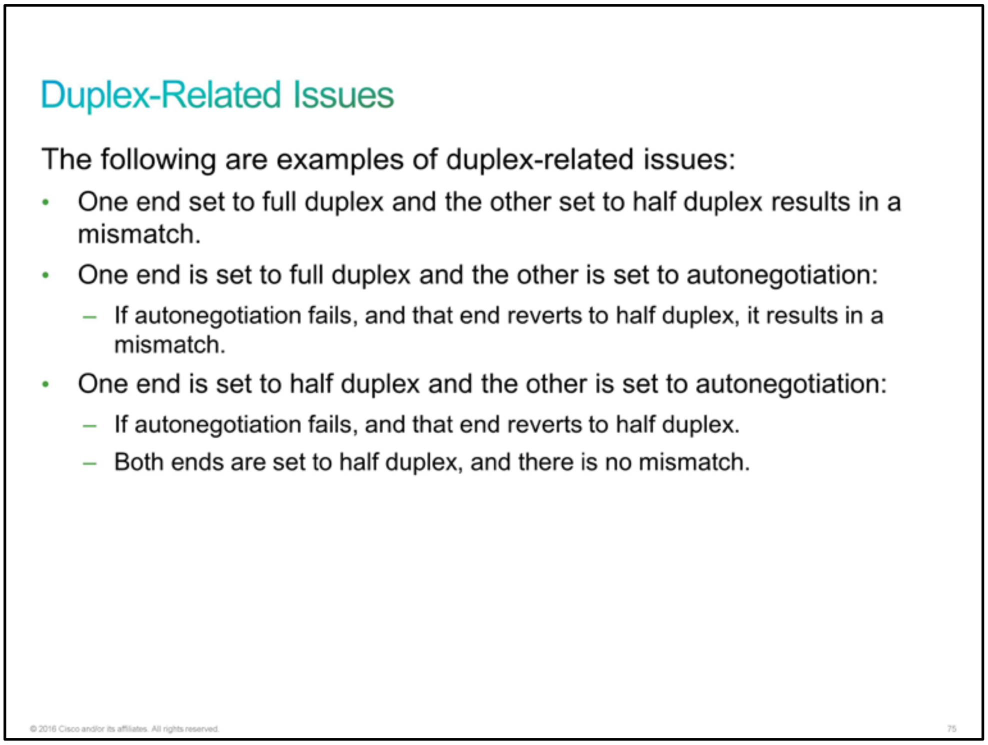
- Duplex-related issues result from a mismatch in duplex settings:

- Speed-related issues result from a mismatch in speed settings:

74

A common issue with speed and duplex occurs when the duplex settings are mismatched between two switches, between a switch and a router, or between a switch and a workstation or server. This mismatch can occur when you manually hard-code the speed and duplex, or from autonegotiation issues between the two devices.

Duplex-Related Issues

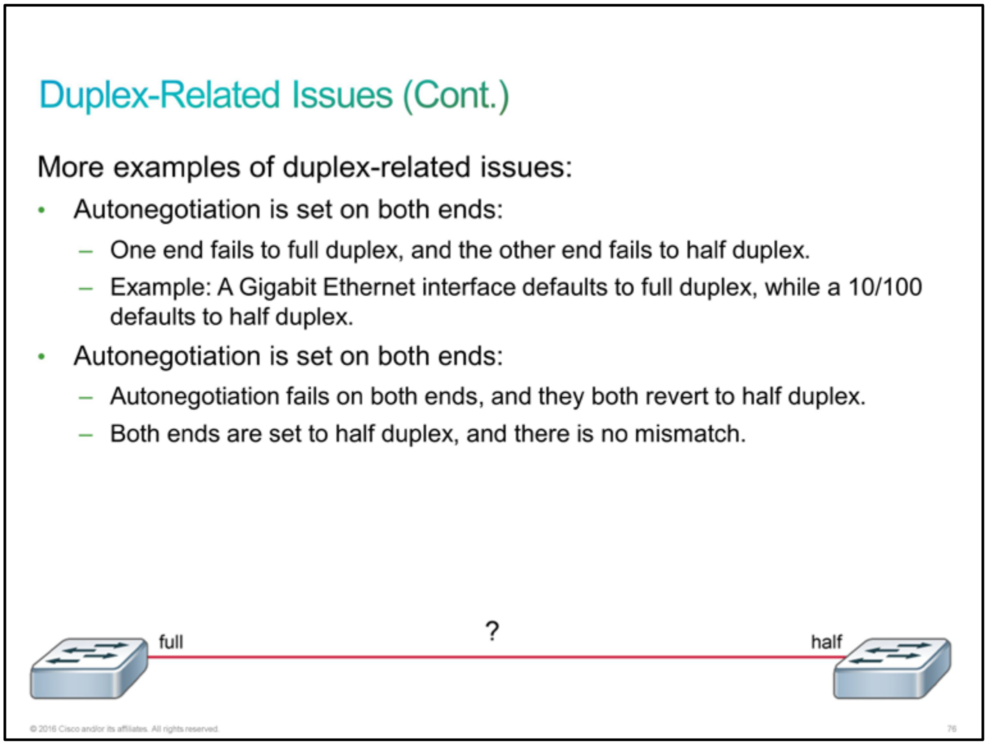


Duplex-Related Issues

The following are examples of duplex-related issues:

- One end set to full duplex and the other set to half duplex results in a mismatch.
- One end is set to full duplex and the other is set to autonegotiation:
  - If autonegotiation fails, and that end reverts to half duplex, it results in a mismatch.
- One end is set to half duplex and the other is set to autonegotiation:
  - If autonegotiation fails, and that end reverts to half duplex.
  - Both ends are set to half duplex, and there is no mismatch.

75



Duplex-Related Issues (Cont.)

More examples of duplex-related issues:

- Autonegotiation is set on both ends:
  - One end fails to full duplex, and the other end fails to half duplex.
  - Example: A Gigabit Ethernet interface defaults to full duplex, while a 10/100 defaults to half duplex.
- Autonegotiation is set on both ends:
  - Autonegotiation fails on both ends, and they both revert to half duplex.
  - Both ends are set to half duplex, and there is no mismatch.

76

A duplex mismatch is a situation in which the switch operates at full duplex and the connected device operates at half duplex, or vice versa. The result of a duplex mismatch is extremely slow performance, intermittent connectivity, and loss of connection. Other possible causes of data-link errors at full duplex are bad cables, a faulty switch port, or NIC software or hardware issues.

Use the **show interface** command to verify the duplex settings.

If the mismatch occurs between two Cisco devices with Cisco Discovery Protocol enabled, you will see Cisco Discovery Protocol error messages on the console or in the logging buffer of both devices. Cisco Discovery Protocol is useful for detecting errors and for gathering port and system statistics on nearby Cisco devices. Whenever there is a duplex mismatch (in this example, on the FastEthernet0/1 interface), the consoles of Cisco Catalyst switches that run Cisco IOS Software display these error messages:

```
%CDP-4-DUPLEX_MISMATCH: duplex mismatch discovered on FastEthernet0/1 (not half
duplex)
```

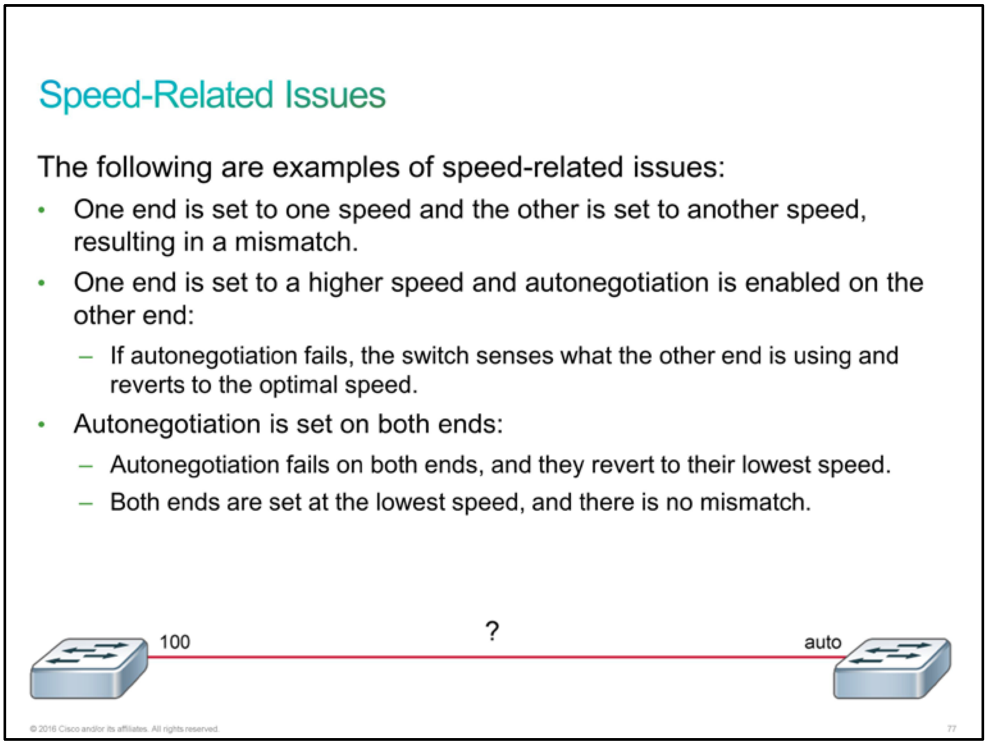
Also, for switches with Cisco IOS Software, these messages appear for link up or down situations (in this example, on the FastEthernet0/1 interface):

```
%LINK-3-UPDOWN: Interface FastEthernet0/1, changed state to up
%LINK-3-UPDOWN: Interface FastEthernet0/1, changed state to down
```

Use the **duplex** *mode* command to configure duplex operation on an interface. The following are available duplex modes:

- Mode **full**: Specifies full-duplex operation.
- Mode **half**: Specifies half-duplex operation.
- Mode **auto**: Specifies the autonegotiation capability. The interface automatically operates at half or full duplex, depending on environmental factors, such as the type of media and the transmission speeds for the peer routers, hubs, and switches that are used in the network configuration.

Speed-Related Issues



Speed-Related Issues

The following are examples of speed-related issues:

- One end is set to one speed and the other is set to another speed, resulting in a mismatch.
- One end is set to a higher speed and autonegotiation is enabled on the other end:
  - If autonegotiation fails, the switch senses what the other end is using and reverts to the optimal speed.
- Autonegotiation is set on both ends:
  - Autonegotiation fails on both ends, and they revert to their lowest speed.
  - Both ends are set at the lowest speed, and there is no mismatch.

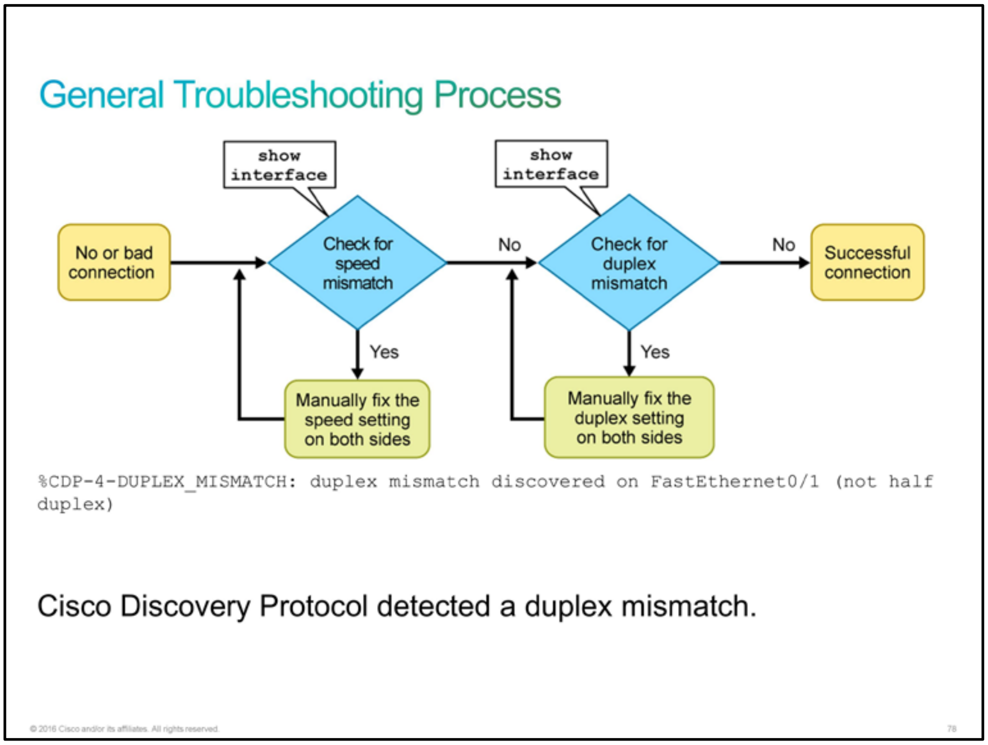
77

Use the **show interface** command to verify the speed settings.

General Troubleshooting Process

To troubleshoot switch port issues when you have no connection or a bad connection between a switch and another device, use this general process:

1. Use the **show interface** command to check whether there is a speed mismatch between the switch and a device on the other side (switch, router, server, and so on). If there is a speed mismatch, set the speed on both sides to the same value.
2. Use the **show interface** command to check whether there is a duplex mismatch between the switch and a device on the other side. It is recommended that you use full duplex if both sides support it.

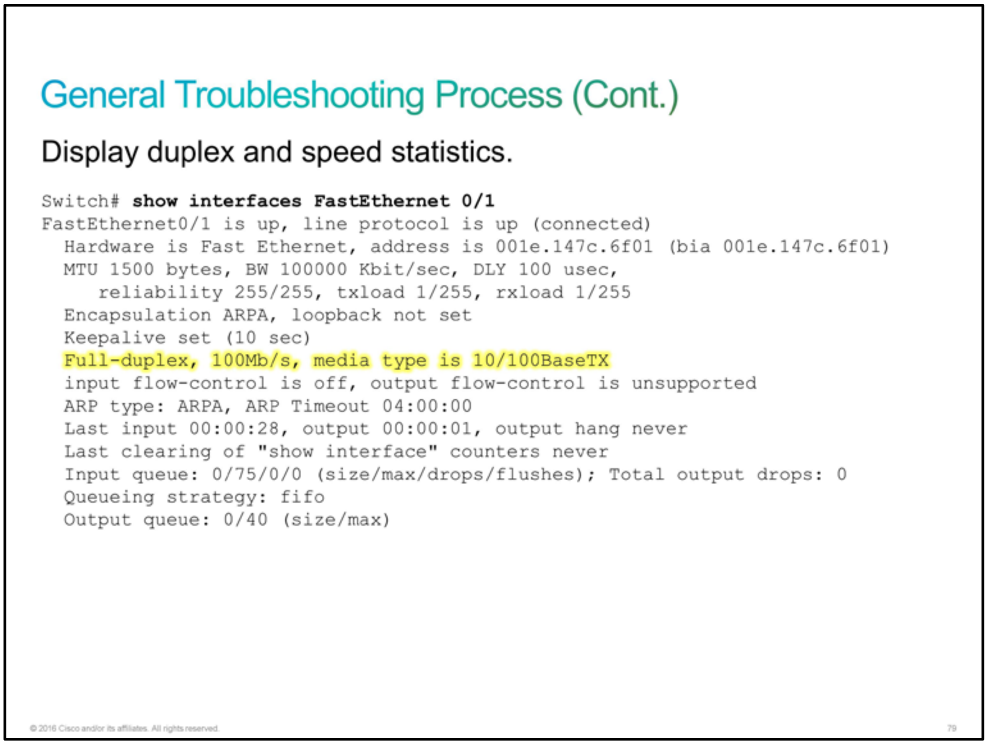


General Troubleshooting Process

show interface
show interface
No or bad connection
Check for speed mismatch
No
Check for duplex mismatch
No
Successful connection
Yes
Yes
Manually fix the speed setting on both sides
Manually fix the duplex setting on both sides

```
%CDP-4-DUPLEX_MISMATCH: duplex mismatch discovered on FastEthernet0/1 (not half
duplex)
```

Cisco Discovery Protocol detected a duplex mismatch.



General Troubleshooting Process (Cont.)

Display duplex and speed statistics.

```
Switch# show interfaces FastEthernet 0/1
FastEthernet0/1 is up, line protocol is up (connected)
  Hardware is Fast Ethernet, address is 001e.147c.6f01 (bia 001e.147c.6f01)
  MTU 1500 bytes, BW 100000 Kbit/sec, DLY 100 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set
  Keepalive set (10 sec)
  Full-duplex, 100Mb/s, media type is 10/100BaseTX
  input flow-control is off, output flow-control is unsupported
  ARP type: ARPA, ARP Timeout 04:00:00
  Last input 00:00:28, output 00:00:01, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
```

The figure shows an example of the **show interface** command output. The example highlights duplex and speed settings for the interface FastEthernet0/1.

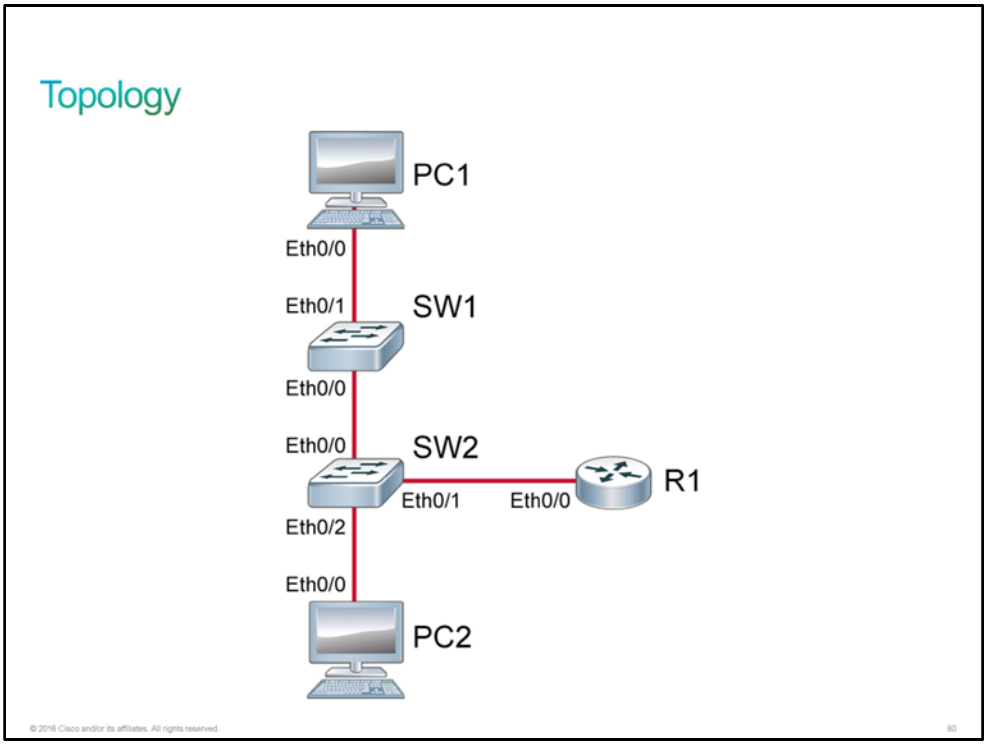
Based on the output of the **show interface** command, you can find, diagnose, and correct the duplex or speed mismatch between the switch and the device on the other side.

Discovery 4: Troubleshoot Switch Media and Port Issues

Introduction

In this activity, you will use troubleshooting guidelines to isolate and correct switch media issues. You will follow troubleshooting guidelines to determine the source of connectivity problems between a computer and a switch, and between a router and a switch, and fix them.

Topology



Job Aid

Device Information

Device Information Table

Device	Characteristic	Value
PC1	Hostname	PC1
PC1	IP address	10.10.1.10/24
PC1	Default gateway	10.10.1.1
PC2	Hostname	PC2

Device	Characteristic	Value
PC2	IP address	10.10.1.20/24
PC2	Default gateway	10.10.1.1
SW1	Hostname	SW1
SW1	VLAN 1 IP address	10.10.1.2/24
SW1	Default gateway	10.10.1.1
SW1	Ethernet0/0 description	Link to SW2
SW1	Ethernet0/1 description	Link to PC1
SW2	Hostname	SW2
SW2	VLAN 1 IP address	10.10.1.3/24
SW2	Default gateway	10.10.1.1
SW2	Ethernet0/0 description	Link to SW1
SW2	Ethernet0/1 description	Link to R1
SW2	Ethernet0/2 description	Link to PC2
R1	Hostname	R1
R1	Ethernet0/0 description	Link to SW2
R1	Ethernet0/0 IP address	10.10.1.1/24
R1	Loopback 0 IP	10.10.3.1/24

PCs in the virtual lab environment are simulated as routers, so you should use Cisco IOS commands to configure them or make verifications.

Task 1: Troubleshoot Port Issues

Activity

Step 1 John calls you about an issue that he is experiencing while using PC2. He says that PC2 has no network connectivity, and he insists that somebody unplugged his computer from the switch. The senior engineers are out. You are the only one who can solve this problem right now. You have access only to switch SW2. Troubleshoot connectivity between the computer PC2 and the switch SW2.

From SW2, determine if you can ping PC2.

```
SW2# ping 10.10.1.20
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.1.20, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)
```

When you issue a ping from SW2 to PC2, your success rate is 0 percent, so there is no Layer 3 connectivity between the two devices.

Step 2 Verify the status of interface Ethernet0/2 on SW2. Interface Ethernet0/2 connects to PC2.

The output of the **show interfaces Ethernet0/2** command tells you that the interface toward PC2 is administratively down, which means that the administrator disabled the interface.

```
SW2# show interfaces Ethernet0/2
Ethernet0/2 is administratively down, line protocol is down (disabled)
  Hardware is AmdP2, address is aabb.cc00.0520 (bia aabb.cc00.0520)
  Description: Link to PC2
  MTU 1500 bytes, BW 10000 Kbit/sec, DLY 1000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set
  Keepalive set (10 sec)
  Auto-duplex, Auto-speed, media type is unknown
<... output omitted ...>
```

Step 3 Correct the issue so that John can continue his work. Do not forget to verify Layer 3 connectivity between PC2 and SW2.

Enter the interface configuration mode for Ethernet0/2 and enable the interface with the **no shutdown** command.

```
SW2(config)# interface Ethernet 0/2
SW2(config-if)# no shutdown
```

Wait for 30 seconds and verify Layer 3 connectivity between PC2 and SW2 by issuing the **ping** command. It should be successful.

```
SW2# ping 10.10.1.20
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.1.20, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
```

Step 4 Save the configuration of SW2.

It is important to save the configuration of SW2 because the **no shutdown** command would disappear if the switch is restarted. John would again be cut off from the network.

```
SW2# copy running-config startup-config
Destination filename [startup-config]? <Enter>
Building configuration...
Compressed configuration from 985 bytes to 665 bytes[OK]
```

This is the end of the discovery lab.

Challenge

1. Identify three subprocesses of the troubleshooting process. (Choose three.)

   A. elimination
   B. termination
   C. testing
   D. calculation
   E. problem definition
   F. compilation

2. Which verification method would you typically use to verify that you have a bad cable?

   A. top-down
   B. bottom-up
   C. follow-the-path
   D. divide-and-conquer
   E. swap-components

3. What are the three basic purposes of a ping test? (Choose three.)

   A. reachability
   B. RTT
   C. packet loss
   D. MTU on the path
   E. hops in the path
   F. operation system running on each hop

4. What is not a copper media issues source?

   A. Wiring becomes damaged
   B. Bending the cooper in too small a radius
   C. New EMI sources are introduced
   D. Traffic patterns change
   E. New equipment is installed

5. What are two possible sources of fiber media issues? (Choose two.)

   A. wiring becomes damaged
   B. macrobend losses
   C. new EMI sources are introduced
   D. traffic patterns change
   E. splice losses

6. Which command should be used to verify speed setting?

   A. **show speed**
   B. **show interface speed**
   C. **show interface**
   D. **debug speed**

7. Which command should be used to verify duplex setting?

   A. **show interface**
   B. **show duplex**
   C. **show interface duplex**
   D. **debug duplex**

Answer Key

Challenge

1. A, C, E
2. E
3. A, B, C
4. B
5. B, E
6. C
7. A

Module 2: Establishing Internet Connectivity

Introduction

This module explains all necessary technologies to successfully establish an Internet connection and provide Internet access to local area users. Internet protocol and IP addressing are explained. Subnets and steps needed to perform subnetting are introduced. Transport layer protocols, TCP and UDP, are briefly described, but details are avoided; the focus is to provide enough knowledge to later understand how NAT operates. The router, its role and operation, is explained. Basic configuration steps are shown with configuration examples. The packet delivery process is illustrated, and the ARP protocol is introduced. Access control lists are introduced at an overview level, but only a standard ACL is explained at this point due to its use in NAT. Other use cases and configuration options are discussed in the next module. Finally, NAT is explained. Configuration examples are used, with the focus on configuring PAT.

Lesson 1: Understanding the TCP/IP Internet Layer

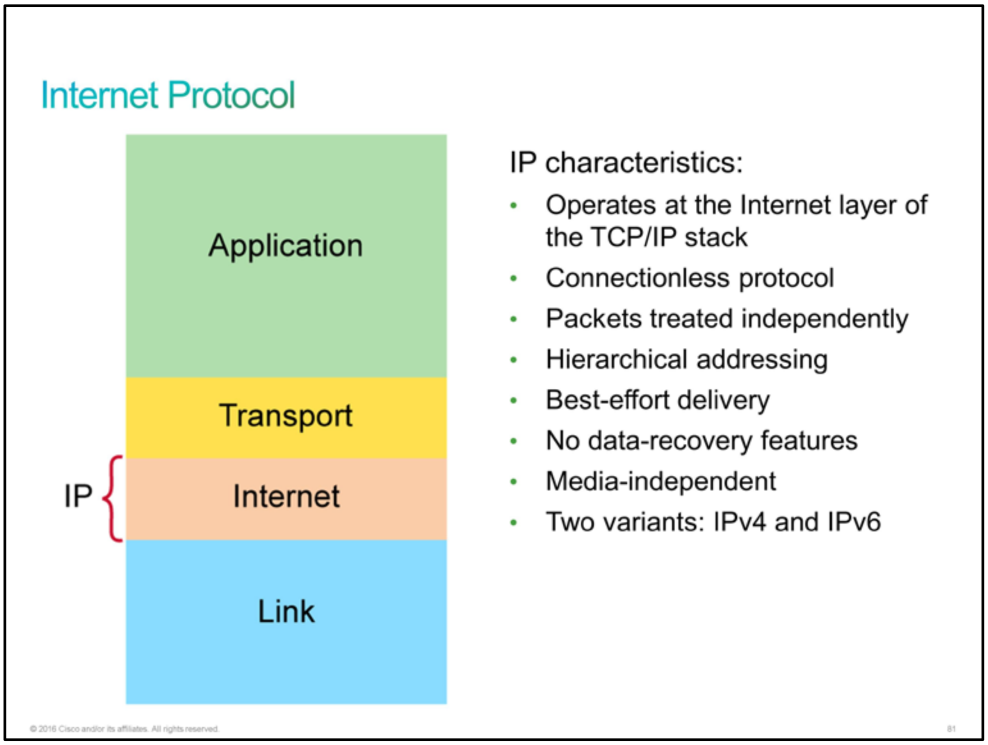
Introduction

CCS is happy with your work involving switches and wants to start assigning you to deployments involving routers and Internet connectivity. Before they do, you will need to demonstrate your knowledge of IPv4 and IPv4 addressing.

You will need to understand the Internet Protocol and its general characteristics, IP addressing, its structure (network and the host portion of addresses), and the IPv4 address fields. You will go through Internet address classes and the types of reserved IP addresses, where you will focus on relevant types (network address, broadcast address). You will also mention the problem of memorizing IP addresses and how DNS is introduced as a solution. At the end, you will explain how to verify of IP settings on end host devices.

Internet Protocol

The IP component of TCP/IP determines where packets of data are routed, based on their destination addresses. IP has certain characteristics that are related to how it manages this function.



Internet Protocol

Application

Transport

IP

Internet

Link

IP characteristics:

- Operates at the Internet layer of the TCP/IP stack
- Connectionless protocol
- Packets treated independently
- Hierarchical addressing
- Best-effort delivery
- No data-recovery features
- Media-independent
- Two variants: IPv4 and IPv6

81

IP uses packets to carry information through the network. A packet is a self-contained, independent entity that contains data and sufficient information to be routed from the source to the destination without reliance on earlier exchanges.

IP has these characteristics:

- IP operates at Layer 3 of the OSI model (network layer) and at the Internet layer of the TCP/IP stack.
- IP is a connectionless protocol, in which a one-way datagram is sent to the destination without advance notification to the destination device. The destination device receives the data and does not return any status information to the sending device.
- Each packet is treated independently, which means that each packet can travel a different way to the destination.
- IP uses hierarchical addressing, in which the network ID is the equivalent of a street, and the host ID is the equivalent of a house or an office building on that street.
- IP provides service on a best-effort basis and does not guarantee packet delivery. A packet can be misdirected, duplicated, or lost on the way to its destination.
- IP does not provide any special features that recover corrupted packets. Instead, the end systems of the network provide these services.
- IP operates independently of the medium that is carrying the data.
- There are two types of IP addresses: IPv4 and IPv6.

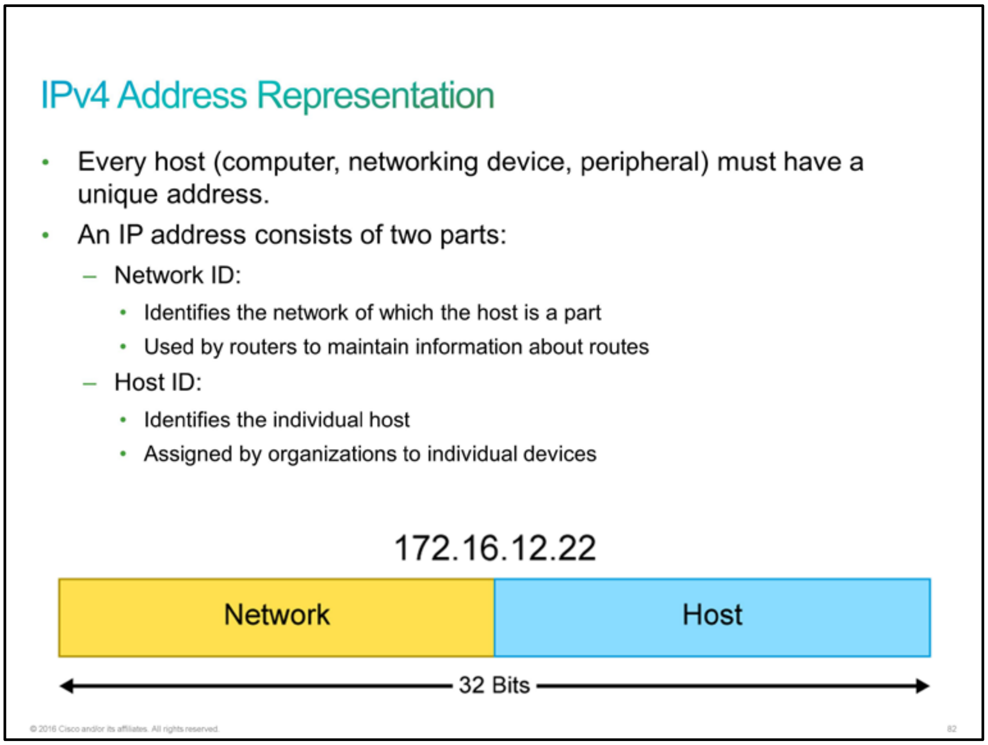
Example: Delivering a Letter Through a Postal Service

An analogy for IP services would be mail delivery by a postal service. For example, you live in San Francisco and your mother lives in New York. You write three letters to your mother. You seal each letter in a separate envelope, address each letter, and write your return address in the upper left-hand corner of each envelope.

You deposit the three letters in the outgoing mail slot at your local post office. The postal service makes its best attempt to deliver all three letters to your mother in New York. However, the postal service will not guarantee that the letters will arrive at their destination. It will not guarantee that all three letters will be processed by the same carrier or take the same route. And it will not guarantee that the letters will arrive in the order in which you mailed them.

IPv4 Address Representation

Every host must be assigned a unique address to communicate on an IP network. Common IP hosts include PCs, laptops, printers, web servers, smart phones, and tablets.



IPv4 Address Representation

- Every host (computer, networking device, peripheral) must have a unique address.
- An IP address consists of two parts:
  - Network ID:
    - Identifies the network of which the host is a part
    - Used by routers to maintain information about routes
  - Host ID:
    - Identifies the individual host
    - Assigned by organizations to individual devices

© 2016 Cisco and/or its affiliates. All rights reserved. 82

Physical street addresses are necessary to identify the locations of specific homes and businesses so that mail can reach them efficiently. In the same way, logical IP addresses are used to identify the location of specific devices on an IP network so that data can reach those network locations. Every host that is connected to the Internet has a unique 32-bit IP address that identifies it. Without a structure for allocating all those IP addresses, it would be impossible to route packets efficiently. Learning how IP addresses are structured and how they function in the operation of a network provides an understanding of how IP packets are forwarded over networks using TCP/IP.

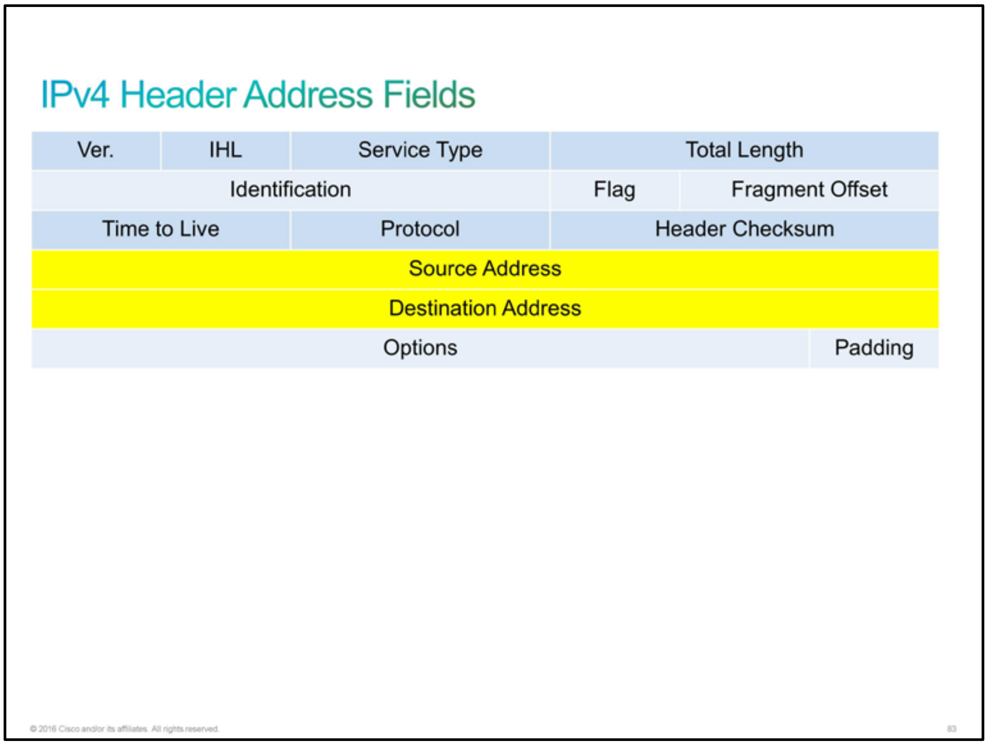
Two versions of the Internet Protocol are in use, IPv4 and IPv6. IPv4 is the most common type of address that is currently used on the Internet. It has been the mainstay protocol since the 1980s. The IPv6 address was designed to solve the problem of global IP address exhaustion. Adoption of IPv6 was initially very slow, but is now reaching wider deployment.

An IP address is hierarchical and consists of two parts:

- **The network address portion (network ID):** Describes the network of which this IP address is a part.
- **The host address portion (host ID):** Identifies a specific endpoint. These endpoints are the servers, computers, and other devices that are connected to the network. Host IDs are assigned to individual devices (end-user devices, printers, network devices, and so on).

IPv4 Header Address Fields

Before you can send an IP packet, there needs to be a format that all IP devices agree upon to route a packet from the source to the destination. All that information is contained in the IP header. The IPv4 header is basically a container for values that are required to achieve host-to-host IP communications. Some fields (such as the IP version) are static, and others, such as TTL, are modified continually in transit.



IPv4 Header Address Fields

<table>
<tr><td>Ver.</td><td>IHL</td><td>Service Type</td><td colspan="3">Total Length</td></tr>
<tr><td colspan="3">Identification</td><td>Flag</td><td colspan="2">Fragment Offset</td></tr>
<tr><td colspan="2">Time to Live</td><td>Protocol</td><td colspan="3">Header Checksum</td></tr>
<tr><td colspan="6">Source Address</td></tr>
<tr><td colspan="6">Destination Address</td></tr>
<tr><td colspan="5">Options</td><td>Padding</td></tr>
</table>

83

The IPv4 header has several fields. At this point, these two fields are most important for you:

- **Source Address:** Specifies the 32-bit binary value that represents the IP address of the sending endpoint
- **Destination Address:** Specifies the 32-bit binary value that represents the IP address of the receiving endpoint

Other fields in the header are the following:

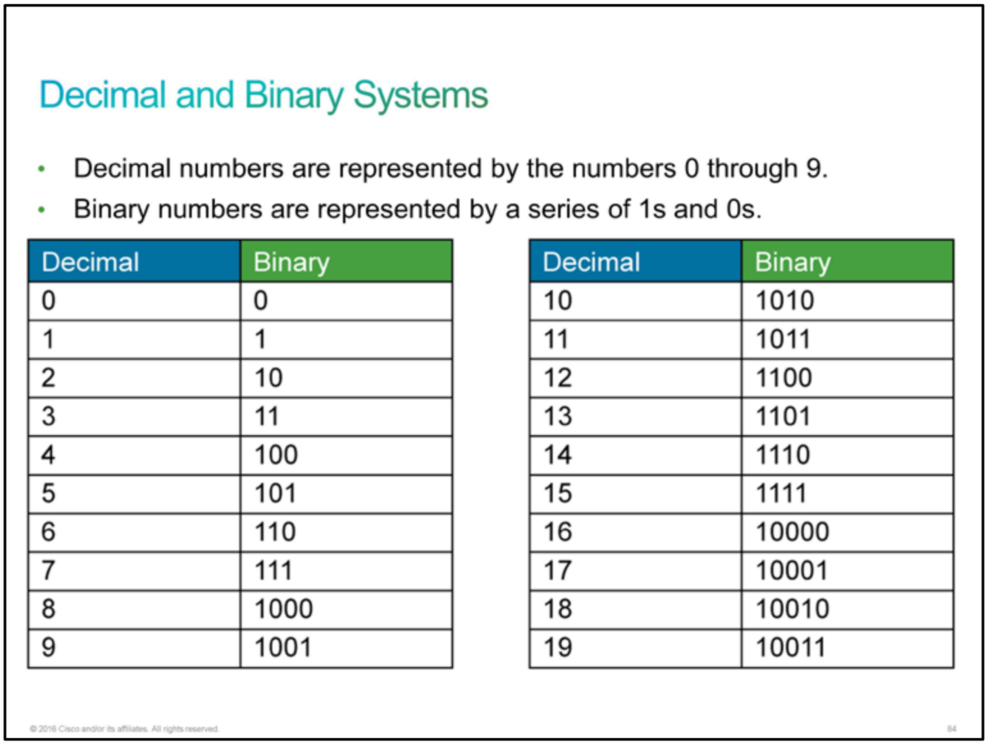
- **Version:** Describes the version of the Internet Protocol
- **IHL:** Internet Header Length; describes the length of the header
- **Service Type:** Provides information on the desired quality of service
- **Total Length:** Describes the length of a packet, including header and data
- **Identification:** Used for unique fragment identification
- **Flag:** Sets various control flags regarding fragmentation
- **Fragment Offset:** Indicates where specific fragment belongs
- **Time to Live:** Limits the lifetime of a packet
- **Protocol:** Indicates the protocol that is used in the data portion of an IP packet

- **Header Checksum:** Used for header error detection
- **Options:** Includes optional parameters
- **Padding:** Used to ensure that the header ends on a 32-bit boundary

If you would like to learn more about the IPv4 header fields, go to http://tools.ietf.org/html/rfc791.

Decimal and Binary Systems

The decimal (base 10) system is the numbering system that is used in everyday mathematics, and the binary (base 2) system is the foundation of computer operations. Network device addresses use the binary system to define their location on the network. The IP address is based on a dotted-decimal notation of a binary number. Having a basic understanding of the mathematical properties of a binary system helps you to understand networking.



Decimal and Binary Systems

- Decimal numbers are represented by the numbers 0 through 9.
- Binary numbers are represented by a series of 1s and 0s.

Decimal	Binary
0	0
1	1
2	10
3	11
4	100
5	101
6	110
7	111
8	1000
9	1001

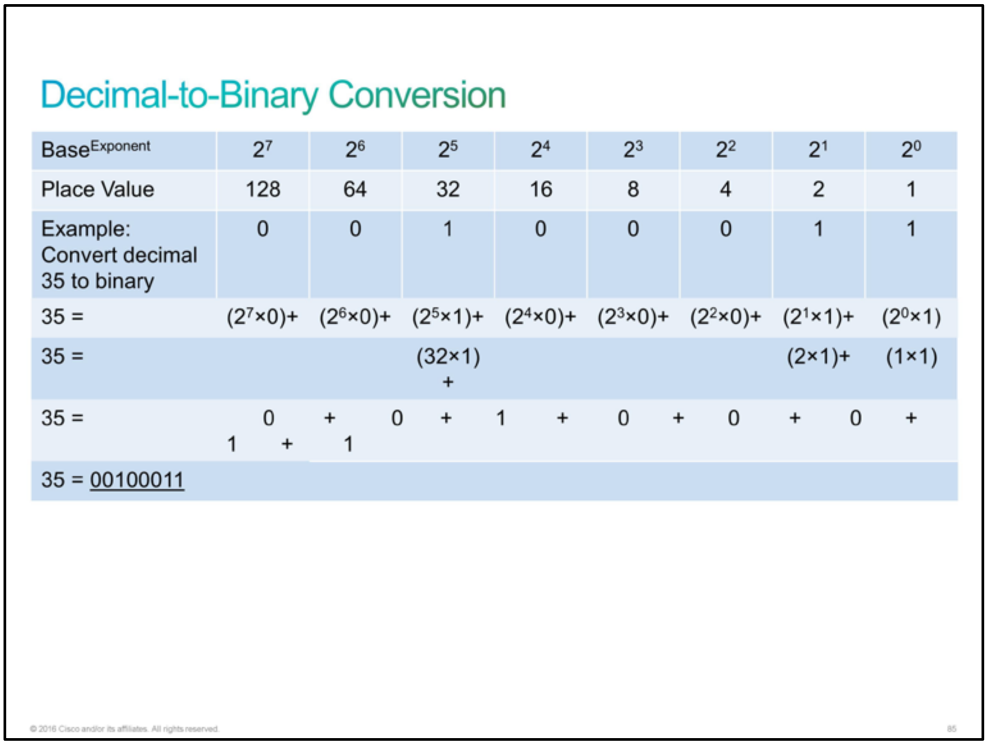
Decimal	Binary
10	1010
11	1011
12	1100
13	1101
14	1110
15	1111
16	10000
17	10001
18	10010
19	10011

In the decimal system, the digits are 0, 1, 2, 3, 4, 5, 6, 7, 8, and 9. When quantities higher than 9 are required, the decimal system begins with 10 and continues all the way to 99. Then the decimal system begins again with 100, and so on, with each column to the left raising the exponent by 1.

The binary system uses only the digits 0 and 1. Therefore, the first digit is 0, followed by 1. If a quantity higher than 1 is required, the binary system goes to 10, followed by 11. The binary system continues with 100, 101, 110, 111, then 1000, and so on. This figure shows the binary equivalent of the decimal numbers 0 through 19.

Decimal-to-Binary Conversion

You can convert decimal numbers to binary numbers through a specific process.



Decimal-to-Binary Conversion

Base^{Exponent}	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
Place Value	128	64	32	16	8	4	2	1
Example: Convert decimal 35 to binary	0	0	1	0	0	0	1	1
35 =	$(2^7\times0)+$	$(2^6\times0)+$	$(2^5\times1)+$	$(2^4\times0)+$	$(2^3\times0)+$	$(2^2\times0)+$	$(2^1\times1)+$	$(2^0\times1)$
35 =			(32×1) +				$(2\times1)+$	(1×1)
35 =	0 + 0 + 1 + 0 + 0 + 0 + 1 + 1							
35 = 00100011								

85

This figure shows a simple binary conversion of the decimal number 35. The base exponent line shows base-2 numbers and their exponents (2 × 2 = 4 × 2 = 8, and so on). The decimal value of the base exponent number is listed in the second row, and the binary number is displayed in the third row. The table describes the steps to determine the binary number. Note that the first 2 bits of the binary number are 0s. These zeros are known as leading 0s. In reality, the decimal number 35 would only be a 6-bit binary number. Because IP addresses are laid out as four sets of octets, the binary number is made into an octet by placing 0s to the left of the 6-bit number.

The table shows the steps for converting the number 35 to a binary number.

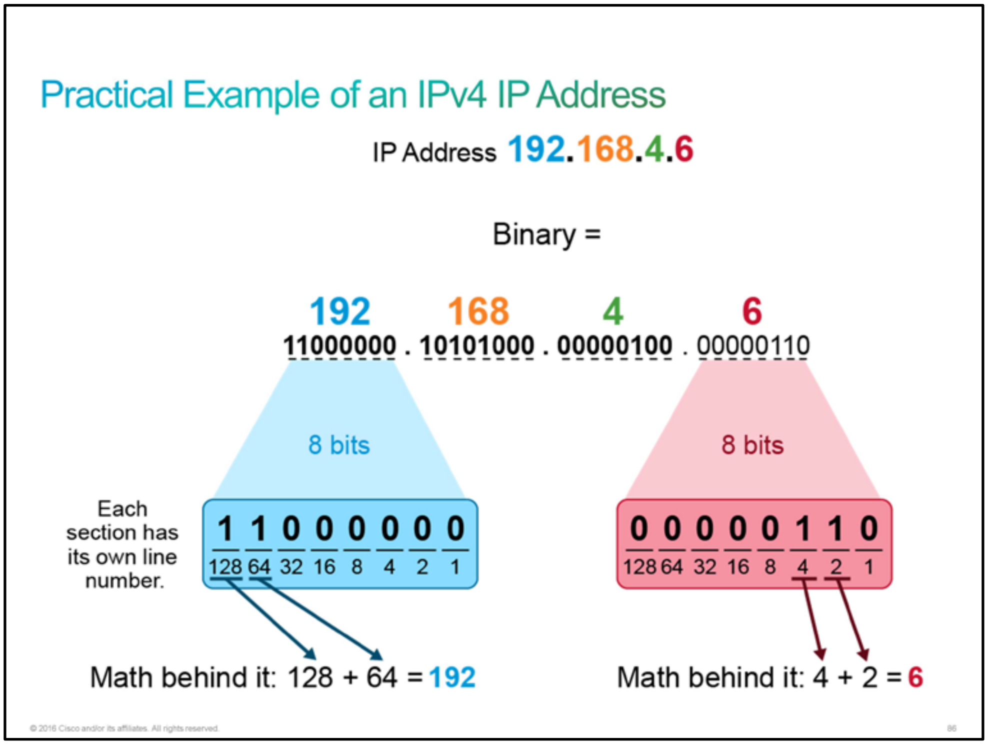
Procedure for Converting a Decimal Number to a Binary Number

Step	Action
1.	Looking at the table, what is the greatest power of 2 that is less than or equal to 35? 128 does not go into 35, so place a 0 in that column.
2.	64 does not go into 35, so place a 0 in that column.
3.	2^5 (32) is smaller than 35. 32 goes into 35 one time. Place a 1 in that column.
4.	Calculate how much is left over by subtracting 32 from 35. The result is 3.
5.	Check to see if 16 (the next lower power of 2) fits into 3. Because it does not, a 0 is placed in that column.

Step	Action
6.	The value of the next number is 8, which is larger than 3, so a 0 is placed in that column also.
7.	The next value is 4, which is still larger than 3, so it, too, receives a 0.
8.	The next value is 2, which is smaller than 3. Because 2 fits into 3 one time, place a 1 in that column.
9.	Subtract 2 from 3, and the result is 1.
10.	The decimal value of the last bit is 1, which fits into the remaining number. Therefore, place a 1 in the last column. The binary equivalent of the decimal number 35 is 00100011.

Practical Example of an IPv4 IP Address

An IP address is in the form of four sets of decimal numbers that are separated by dots. The decimal number in each and every set is in the range from 0 to 255. Each set is called an octet. There are four octets in an IP address. All computer systems understand IP addresses only in the binary form. The following example shows how an IP address is translated into the binary form.



Practical Example of an IPv4 IP Address

IP Address 192.168.4.6

Binary =

192
168
4
6
11000000 . 10101000 . 00000100 . 00000110
8 bits
8 bits
Each section has its own line number.
1 1 0 0 0 0 0 0
128 64 32 16 8 4 2 1
0 0 0 0 0 1 1 0
128 64 32 16 8 4 2 1
Math behind it: 128 + 64 = 192
Math behind it: 4 + 2 = 6
© 2016 Cisco and/or its affiliates. All rights reserved.
86

An easy way how to begin with the translation is to create basic placeholders for your binary conversion. Place 32 underscores (_) under the IP address and separate them with dots. Why 32? Because you will translate each set into an 8-bit binary number; 4 × 8 = 32. Each underscore has a set value. Values from right to left are: 1, 2, 4, 8, 16, 32, 64, 128. Can you see a pattern? Each value increases by 2^n. Now you start adding these numbers so that the total equals your set. The figure shows the translation for the first and last set.

The value of the first set is 192. If you add 128 and 64 (first two underscore values) that equals 192. Every time that you add a number, write 1 on the underscore (each bit is represented by a 1 or a 0). When you finish, write 0 on the underscores that you did not use (those bits are set to off). The first octet is translated to 11000000.

The value of the last set is 6. If you use the second and third bit (from the right), you match that number, so the last set translates to 00000110. Try to translate the remaining sets on your own.

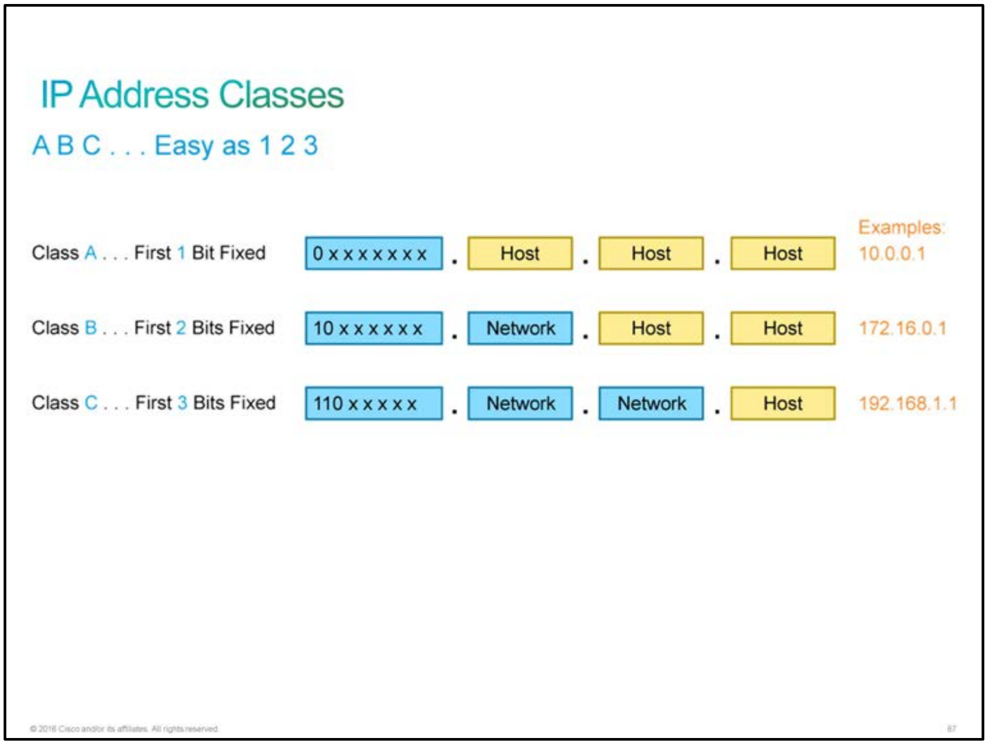
IP Address Classes

To accommodate networks of different sizes and help in classifying them, IP addresses are divided into categories that are called *classes*.

Assigning IP addresses to classes is known as *classful addressing*. During the early days of the Internet, the IANA determined the classes.

Each IP address is broken down into a network ID and a host ID. In addition, a bit or bit sequence at the start of each address determines the class of the address. The figure shows three of the five IP classes.

Note	IP hosts use only Class A, B, and C IP addresses for unicast (host-to-host) communications. Class D and Class E are included for completeness, but they are outside the scope of this discussion.



IP Address Classes

A B C . . . Easy as 1 2 3

					Examples:
Class A . . . First 1 Bit Fixed	0 x x x x x x x	Host	Host	Host	10.0.0.1
Class B . . . First 2 Bits Fixed	10 x x x x x x	Network	Host	Host	172.16.0.1
Class C . . . First 3 Bits Fixed	110 x x x x x	Network	Network	Host	192.168.1.1

© 2016 Cisco and/or its affiliates. All rights reserved.

87

Class A

A Class A address block is designed to support extremely large networks with more than 16 million host addresses. The Class A address uses only the first octet (8 bits) of the 32-bit number to indicate the network address. The remaining 3 octets of the 32-bit number are used for host addresses. The first bit of a Class A address is always a 0. Because the first bit is a 0, the lowest number that can be represented is 00000000 (decimal 0), and the highest number that can be represented is 01111111 (decimal 127). However, these two network numbers, 0 and 127, are reserved and cannot be used as network addresses. Any address that has a value between 1 and 126 in the first octet of the 32-bit number is a Class A address.

Class B

The Class B address space is designed to support the needs of moderate to large networks with more than 65,000 hosts. The Class B address uses 2 of the 4 octets (16 bits) to indicate the network address. The remaining two octets specify host addresses. The first 2 bits of the first octet of a Class B address are always binary 10. Starting the first octet with binary 10 ensures that the Class B space is separated from the upper levels of the Class A space. The remaining 6 bits in the first octet may be populated with either 1s or 0s. Therefore, the lowest number that can be represented with a Class B address is 10000000 (decimal 128), and the highest number that can be represented is 10111111 (decimal 191). Any address that has a value in the range of 128 to 191 in the first octet is a Class B address.

Class C

The Class C address space is the most commonly available address class. This address space is intended to provide addresses for small networks with a maximum of 254 hosts. In a Class C address, the first three octets (24 bits) of the IP address identify the network portion, with the remaining octet reserved for the host portion. A Class C address begins with binary 110. Therefore, the lowest number that can be represented is 11000000 (decimal 192), and the highest number that can be represented is 11011111 (decimal 223). If an address contains a number in the range of 192 to 223 in the first octet, it is a Class C address.

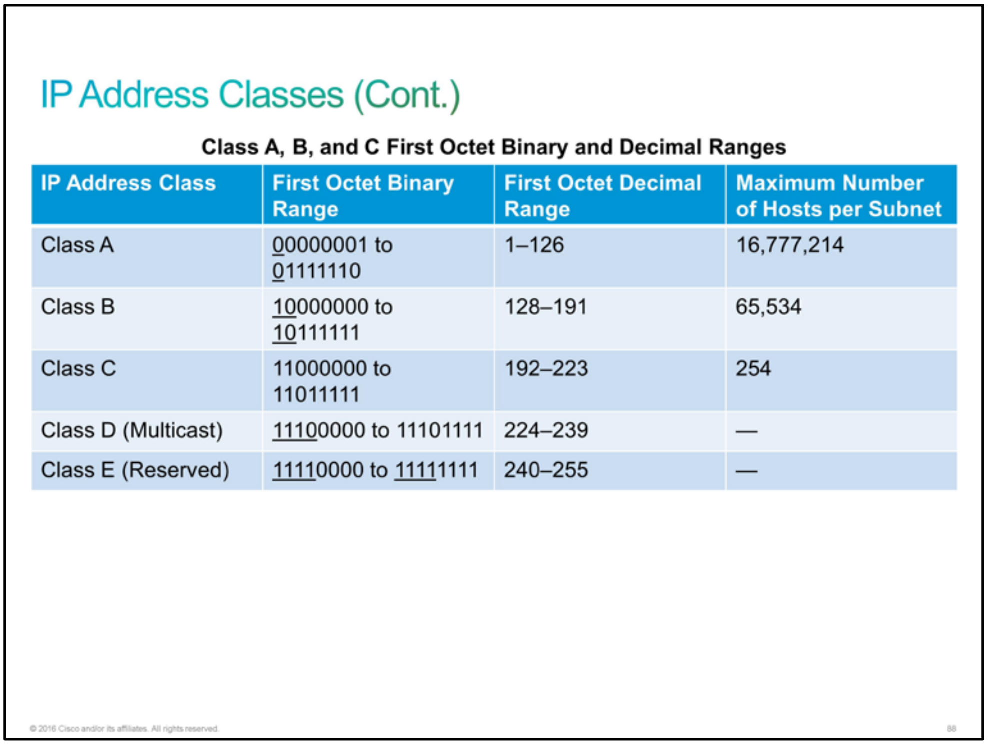
Class D

Class D (multicast) IP addresses are dedicated to multicast applications such as streaming media. Multicasts are a special type of broadcast, in that only hosts that request to participate in the multicast group will buffer the traffic to the IP address of that group. Unlike IP addresses in Classes A, B, and C, multicast addresses are always the destination address and never the source. A Class D address begins with binary 1110. Therefore, the lowest number that can be represented is 11100000 (decimal 224), and the highest number that can be represented is 11101111 (decimal 239). If an address contains a number in the range of 224 to 239 in the first octet, it is a Class D address.

Class E

Class E (reserved) IP addresses are reserved by the IANA as a block of experimental addresses. Class E IP addresses should never be assigned to IP hosts. A Class E address begins with binary 1111. Therefore, the lowest number that can be represented is 11110000 (decimal 240), and the highest number that can be represented is 11111111 (decimal 255). If an address contains a number in the range of 240 to 255 in the first octet, it is a Class E address.

This table shows the IP address range of the first octet (in decimal and binary) for Class A, B, and C IP addresses, and also the number of host addresses that are available for each class of addresses.



IP Address Classes (Cont.)

Class A, B, and C First Octet Binary and Decimal Ranges

IP Address Class	First Octet Binary Range	First Octet Decimal Range	Maximum Number of Hosts per Subnet
Class A	00000001 to 01111110	1–126	16,777,214
Class B	10000000 to 10111111	128–191	65,534
Class C	11000000 to 11011111	192–223	254
Class D (Multicast)	11100000 to 11101111	224–239	—
Class E (Reserved)	11110000 to 11111111	240–255	—

Note Class A addresses 127.0.0.0 to 127.255.255.255 cannot be used. This range is reserved for loopback and diagnostic functions.

Each class of network allows a fixed number of hosts. In a Class A network, the first octet is assigned to the network, leaving the last three octets to be assigned to hosts. The first host address in each network (all 0s) is reserved for the actual network address, and the final host address in each network (all 1s) is reserved for broadcasts.

In a Class A network, the last three octets are used as host addresses. An octet is 8 bits, so 3 octets is 24 bits. So the maximum number of hosts in a Class A network is $2^{24} – 2$ (subtracting the network and broadcast reserved addresses), or 16,777,214.

In a Class B network, the first 2 octets are assigned to the network. The final 2 octets (16 bits) are assigned to hosts. The maximum number of hosts in a Class B network is $2^{16} – 2$, or 65,534.

In a Class C network, the first 3 octets are assigned to the network. The final octet (8 bits) can be assigned to hosts, so the maximum number of hosts is $2^{8} – 2$, or 254.

Class D and Class E IP addresses are special cases. They are never assigned to hosts as source IP addresses.

Reserved IPv4 Addresses

Certain IP addresses are reserved and cannot be assigned to individual devices on a network. Reserved IP addresses include a network address, which is used to identify the network itself, and a broadcast address, which is used for broadcasting packets to all the devices on a network.

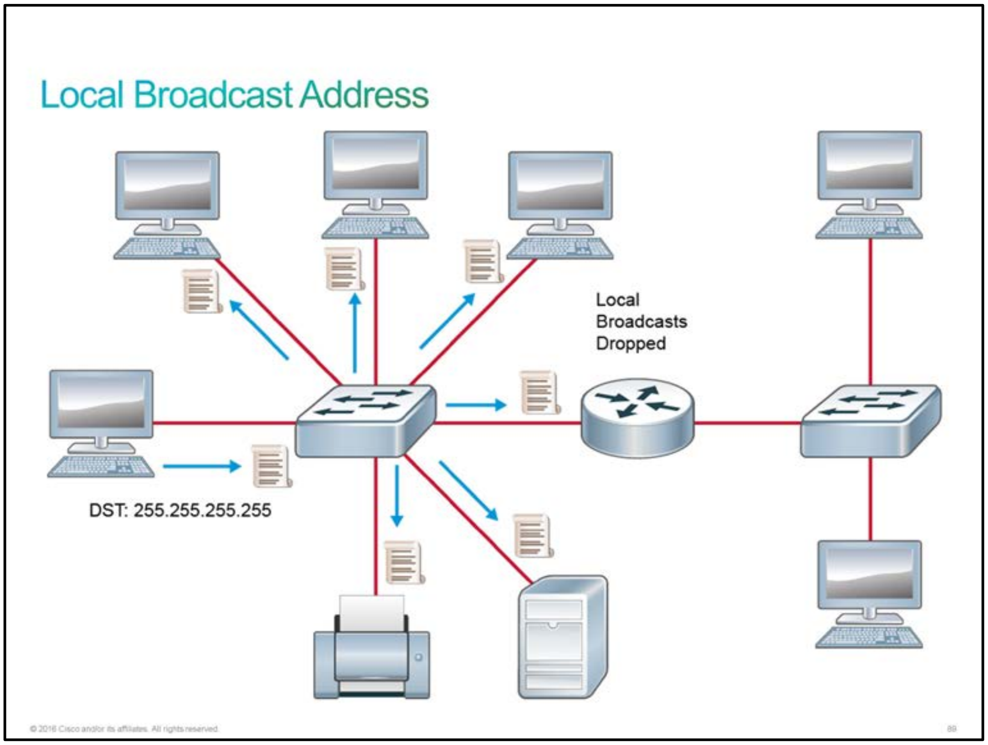
Network Address

The network address is a standard way to refer to a network. An IP address that has binary 0s in all the host bit positions is reserved for the network address. For example, in a Class A network, 10.0.0.0 is the IP address of the network containing the host 10.1.2.3. All hosts in 10.0.0.0 will have the same network bits. The IP address 172.16.0.0 is a Class B network address, and 192.16.1.0 is a Class C network address. A router uses the network IP address when it searches its IP routing table for the destination network location.

In a Class B network address, the first two octets are the network portion. The last two octets contain 0s because those 16 bits are for host numbers and are used for devices that are attached to the network. In the IP address 172.16.0.0, the first two octets are reserved for the network address and are never used as an address for any device that is attached to it. An example of an IP address for a device on the network is 172.16.16.1. In this example, 172.16 is the network address portion and 16.1 is the host address portion.

Local Broadcast Address

If an IP device wants to communicate with all the devices on the local network, it sets the destination address to all 1s (255.255.255.255) and transmits the packet. For example, hosts that do not know their network number and are asking a server for it may use this address. The local broadcast is never routed beyond the local network (subnet).

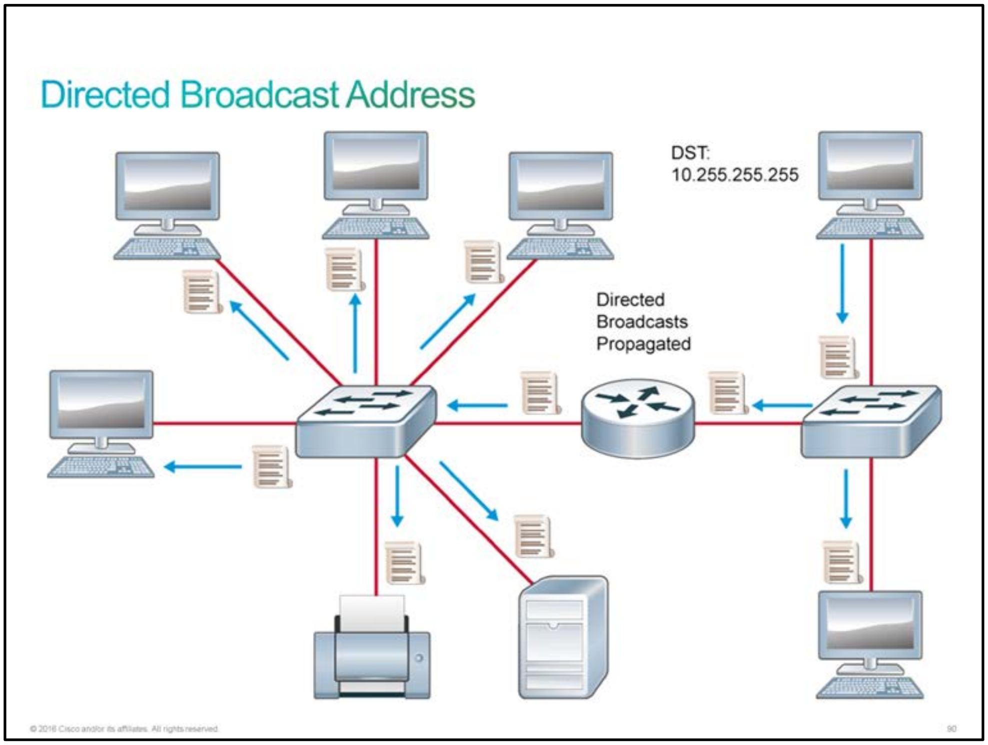


Local Broadcast Address

Directed Broadcast Address

The broadcast IP address of a network is a special address for each network that allows communication to all the hosts in that network. To send data to all the hosts in a network, a host can send a single packet that is addressed to the broadcast address of the network. The broadcast address uses the highest address in the network range, which is the address in which the bits in the host portion are all 1s. For the network 10.0.0.0, with 8 network bits, the broadcast address would be 10.255.255.255. This address is also referred to as the *directed broadcast*. Assuming a hypothetical network where every Class A IP host address was in use, a ping to 10.255.255.255 would receive a response from all 16,777,214 hosts.

For the network address 172.16.0.0, the last 16 bits make up the host field (or host part of the address). The broadcast that would be sent out to all the devices on that network would include a destination address of 172.16.255.255.



Directed Broadcast Address

Note	The directed broadcast address can be routed over your company's intranet and over the Internet. In the 1990s, a popular DoS attack referred to as a Smurf used directed broadcasts to send so much traffic to an intended victim that they could not send or receive any legitimate traffic. For this reason, Cisco IOS defaults to disallowing directed broadcasts. This capability can be restored with the **ip directed-broadcast** command in the global configuration mode. It is a best practice to leave directed broadcasts disabled unless you have a specific use case. Routers began using the **no ip directed-broadcast** command as a platform default starting with Cisco IOS Release 12.0.

Local Loopback Address

A local loopback address is used to let the system send a message to itself for testing. The loopback address creates a shortcut method for TCP/IP applications and services that run on the same device to communicate with one another. A typical local loopback IP address is 127.0.0.1. On a Microsoft Windows 7 host, you can ping any IP address in the 127.0.0.0/8 range.

Network ID

The network portion of an IP address is also referred to as the *network ID*. A network ID is important because most hosts on a network can directly communicate only with devices in the same network. If the hosts need to communicate with devices that have interfaces that are assigned to another network ID, they must go through a network device that can route data between the networks. This holds true even when the devices share the same physical media segment. The network ID cannot be assigned to a host. For example, 10.0.0.0 cannot be assigned because it is the network ID for that Class A network.

A network ID enables a router to transmit an IP packet onto the appropriate network segment.

All Zeros Address

The address 0.0.0.0 indicates the host in "this" network and is used only as a source address. An example use case is the DHCP assignment process before the host has a valid IP address.

For more information about reserved IPv4 addresses, refer to RFC 5735, at http://tools.ietf.org/html/rfc5735.

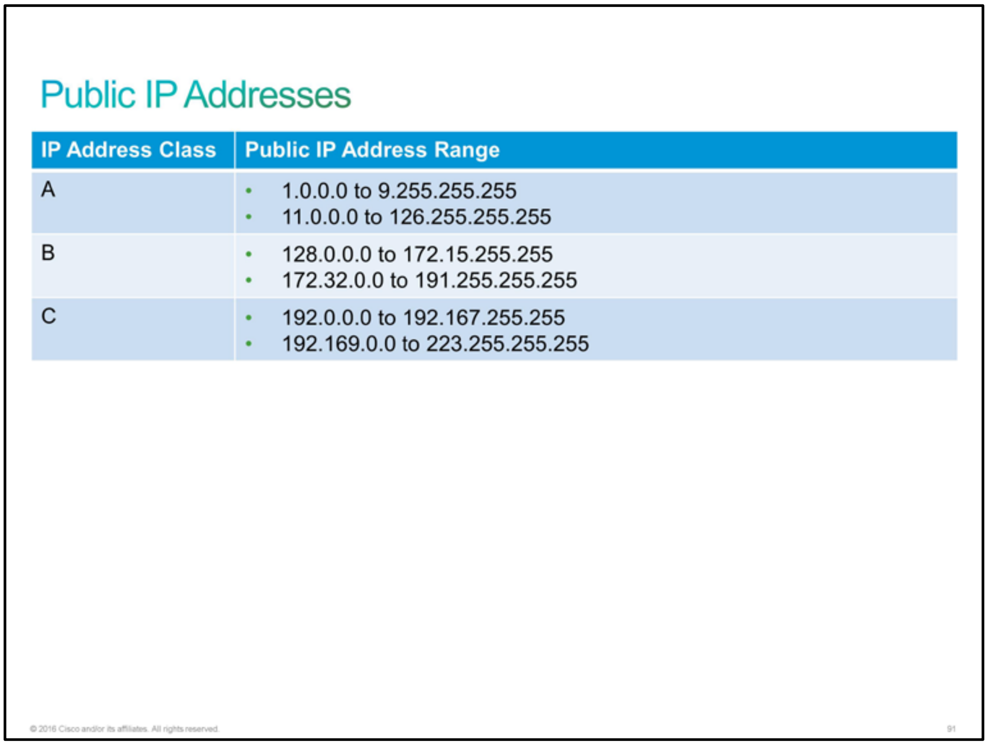
Private vs. Public IP Addresses

As the Internet began to grow exponentially in the 1990s, it became clear that if the current growth trajectory continued, eventually there would not be enough IP addresses for everyone that wanted one. Work began on a permanent solution, which would become IPv6, but in the interim, several other solutions were developed. These included NAT, CIDR, private IP addressing, and VLSM.

Public IP Addresses

Hosts that are publicly accessible over the Internet require public IP addresses. Internet stability depends directly on the uniqueness of publicly used network addresses. Therefore, a mechanism is needed to ensure that addresses are, in fact, unique. The allocation of IP addresses is managed by the IANA.

With few exceptions, businesses and home Internet users receive their IP address assignment from their LIR, which is typically their ISP. These IP addresses are called "provider-dependent" because they are linked to the ISP. If you change ISPs, you will have to readdress your Internet-facing hosts.



Public IP Addresses

IP Address Class	Public IP Address Range
A	• 1.0.0.0 to 9.255.255.255 • 11.0.0.0 to 126.255.255.255
B	• 128.0.0.0 to 172.15.255.255 • 172.32.0.0 to 191.255.255.255
C	• 192.0.0.0 to 192.167.255.255 • 192.169.0.0 to 223.255.255.255

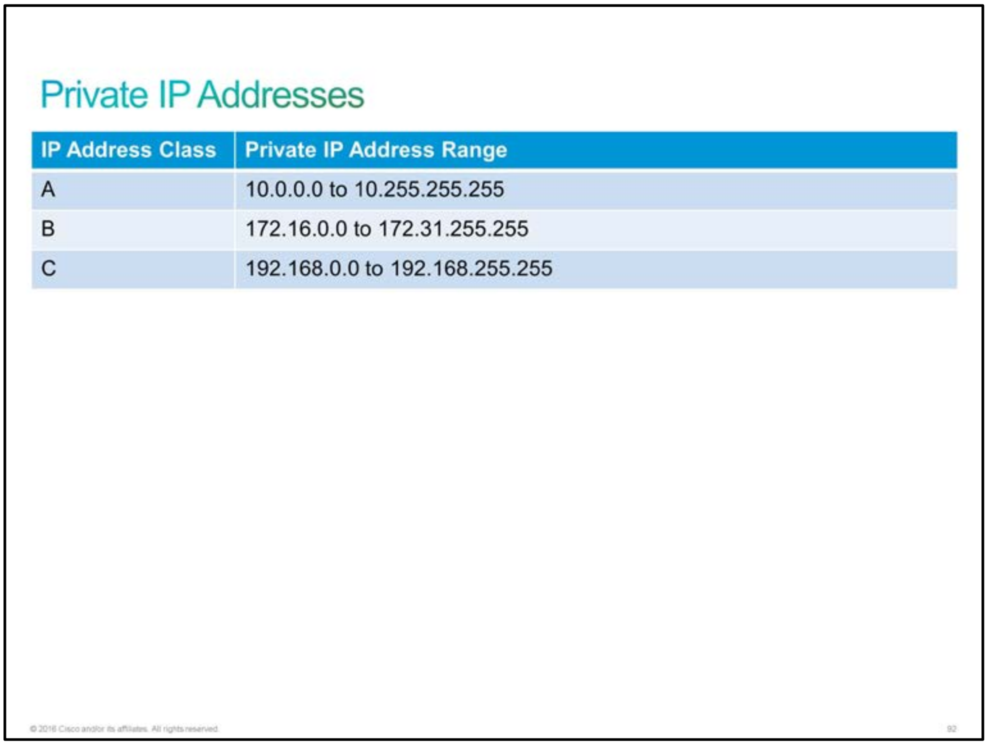
© 2016 Cisco and/or its affiliates. All rights reserved. 91

Private IP Addresses

Internet hosts require a globally unique IP address, but private hosts that are not connected to the Internet can use any valid address, as long as it is unique within the private network. However, because many private networks exist alongside public networks, deploying arbitrary IP addresses is strongly discouraged.

In February of 1996, the IETF published RFC 1918, "Address Allocation for Private Internets," to both ease the accelerating depletion of globally routable IP addresses and provide companies an alternative to using arbitrary IP addresses. Three blocks of IP addresses (one Class A network, 16 Class B networks, and 256 Class C networks) are designated for private, internal use.

Addresses in these ranges are not routed on the Internet backbone. Internet routers are configured to discard private addresses. In a private intranet, these private addresses can be used instead of globally unique addresses. When a network that is using private addresses requires Internet connectivity, it is necessary to translate the private addresses to public addresses. This translation process is called NAT. A router or firewall is often the network device that performs NAT.



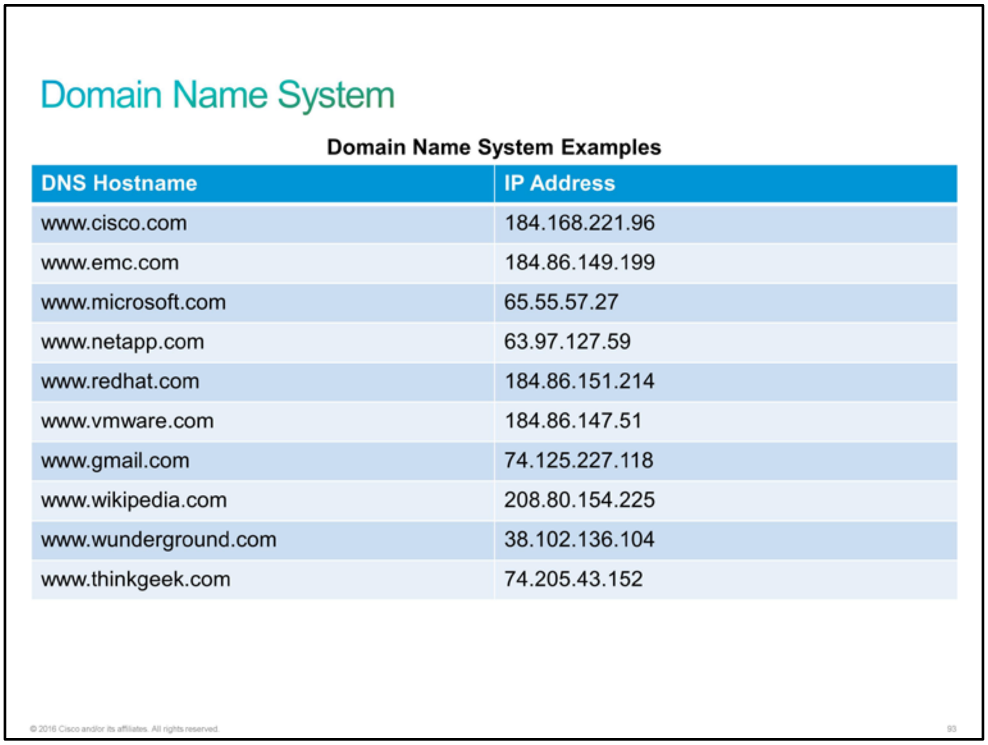
Private IP Addresses

IP Address Class	Private IP Address Range
A	10.0.0.0 to 10.255.255.255
B	172.16.0.0 to 172.31.255.255
C	192.168.0.0 to 192.168.255.255

Domain Name System

The DNS provides an efficient way to convert human-readable names of IP end systems into machine-readable IP addresses that are necessary for routing.

On TCP/IP networks, hosts are assigned their unique 32-bit IP addresses in the familiar dotted quad notation (x.x.x.x) so that they can send and receive messages over the local network and the Internet. Although not every IPv4 address has been assigned, there are billions of possible destinations. If there were no DNS, you would have to remember the IP address of every host that you would like to reach. Imagine having to remember the IP addresses for even the top 10 websites that you visit.



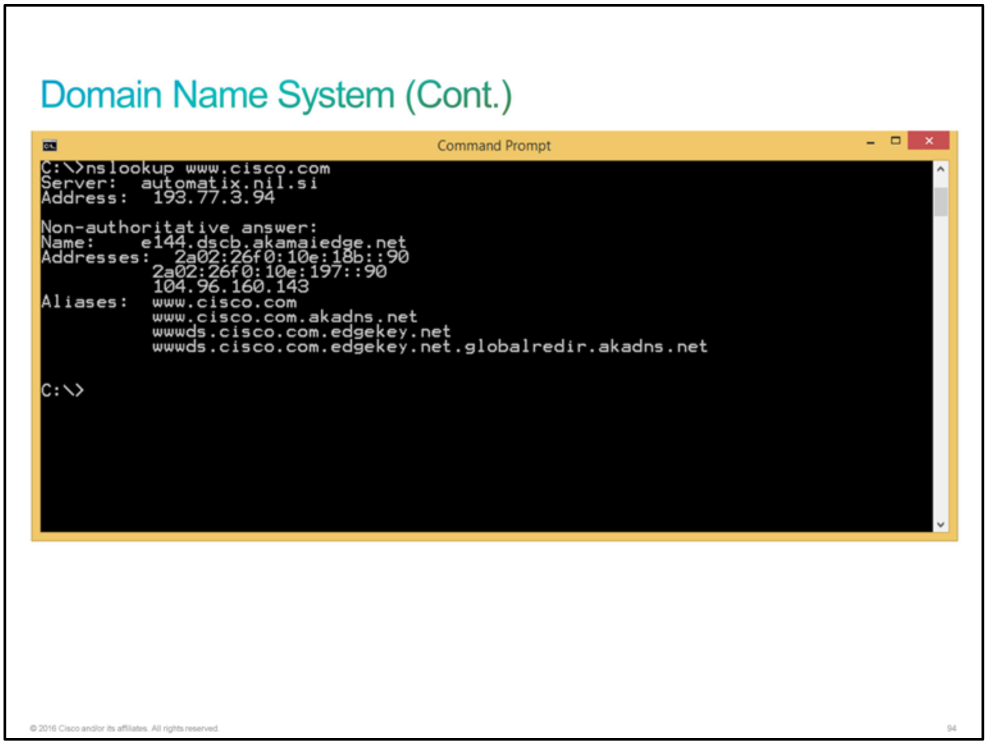
Domain Name System

Domain Name System Examples

DNS Hostname	IP Address
www.cisco.com	184.168.221.96
www.emc.com	184.86.149.199
www.microsoft.com	65.55.57.27
www.netapp.com	63.97.127.59
www.redhat.com	184.86.151.214
www.vmware.com	184.86.147.51
www.gmail.com	74.125.227.118
www.wikipedia.com	208.80.154.225
www.wunderground.com	38.102.136.104
www.thinkgeek.com	74.205.43.152

DNS uses a distributed database that is hosted on several servers, which are located around the world, to resolve the names that are associated with dotted-decimal IP addresses. The DNS protocol defines an automated service that matches resource names with the required numeric network address.

An easy way to observe DNS in action can be performed in a command window in Microsoft Windows, Apple MacOS X, or your favorite Linux distribution. When the command window is open, enter **nslookup www.cisco.com**. This command tells your IP host to make a DNS query. The result will appear below your query.

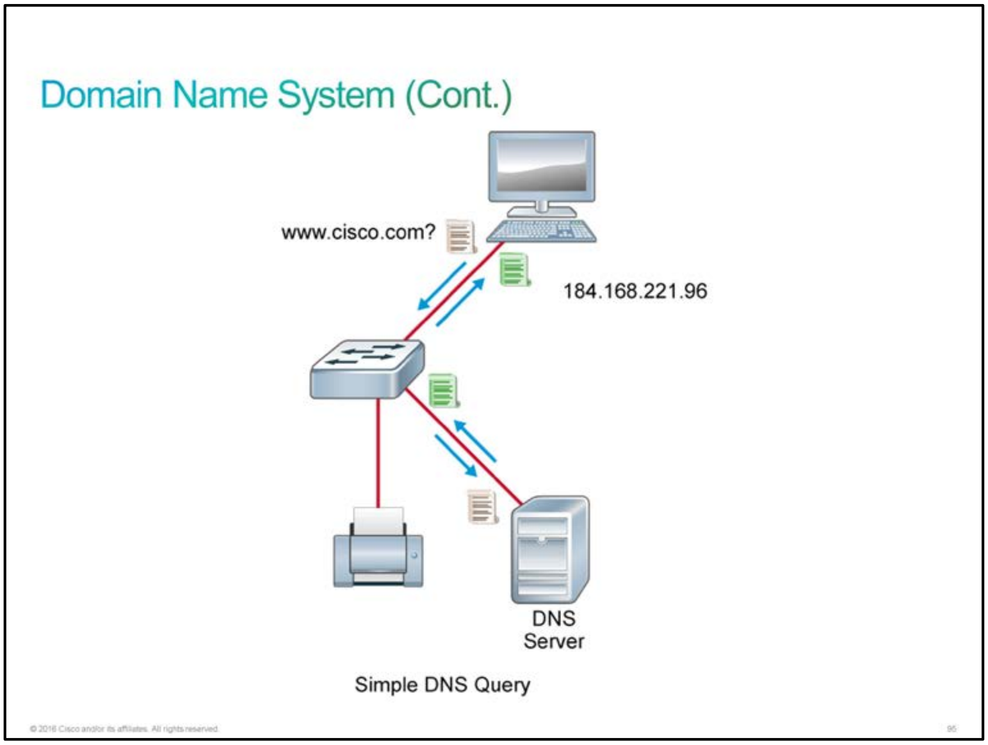


Domain Name System (Cont.)

```
C:\>nslookup www.cisco.com
Server:  automatix.nil.si
Address:  193.77.3.94

Non-authoritative answer:
Name:    e144.dscb.akamaiedge.net
Addresses:  2a02:26f0:10e:18b::90
          2a02:26f0:10e:197::90
          104.96.160.143
Aliases:  www.cisco.com
          www.cisco.com.akadns.net
          wwwds.cisco.com.edgekey.net
          wwwds.cisco.com.edgekey.net.globalredir.akadns.net


C:\>
```

Note	The DNS transaction that is represented in the illustration is a simplification for the purposes of demonstration. In practice, there are DNS transactions, which are external to the local DNS server, that are necessary to receive an answer to the host query.

Your host sends a DNS query for the IP address of www.cisco.com. If your DNS server has the answer cached, it returns the answer directly.



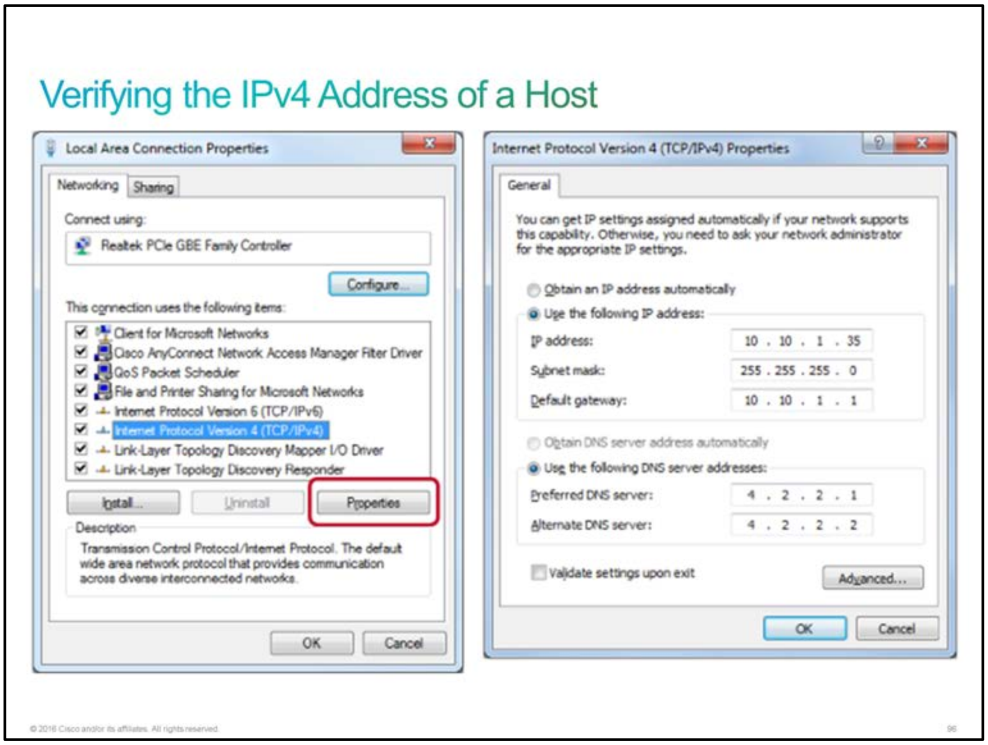
Domain Name System (Cont.)

Simple DNS Query

Verifying the IPv4 Address of a Host

All operating systems that are capable of TCP/IP communications include utilities for configuring, managing, and monitoring the IP networking configuration. Operating systems such as Microsoft Windows, Apple Mac OS X, and most Linux variants include both CLI and GUI tools.

On a PC running Microsoft Windows 7 Enterprise, the Networking tab of a given adapter allows you to view and set the IP address that is associated with that adapter. In this example, the PC is manually configured with a static IP address.



Verifying the IPv4 Address of a Host

Local Area Connection Properties

Networking | Sharing

Connect using:

Realtek PCIe GBE Family Controller

This connection uses the following items:

☑ Client for Microsoft Networks
☑ Cisco AnyConnect Network Access Manager Filter Driver
☑ QoS Packet Scheduler
☑ File and Printer Sharing for Microsoft Networks
☑ Internet Protocol Version 6 (TCP/IPv6)
☑ Internet Protocol Version 4 (TCP/IPv4)
☑ Link-Layer Topology Discovery Mapper I/O Driver
☑ Link-Layer Topology Discovery Responder

Description

Transmission Control Protocol/Internet Protocol. The default wide area network protocol that provides communication across diverse interconnected networks.

Internet Protocol Version 4 (TCP/IPv4) Properties

General

You can get IP settings assigned automatically if your network supports this capability. Otherwise, you need to ask your network administrator for the appropriate IP settings.

○ Obtain an IP address automatically
● Use the following IP address:

IP address: 10 . 10 . 1 . 35
Subnet mask: 255 . 255 . 255 . 0
Default gateway: 10 . 10 . 1 . 1

○ Obtain DNS server address automatically
● Use the following DNS server addresses:

Preferred DNS server: 4 . 2 . 2 . 1
Alternate DNS server: 4 . 2 . 2 . 2

☐ Validate settings upon exit

Note Navigating to the TCP/IP network settings varies widely depending on the operating system that is installed.

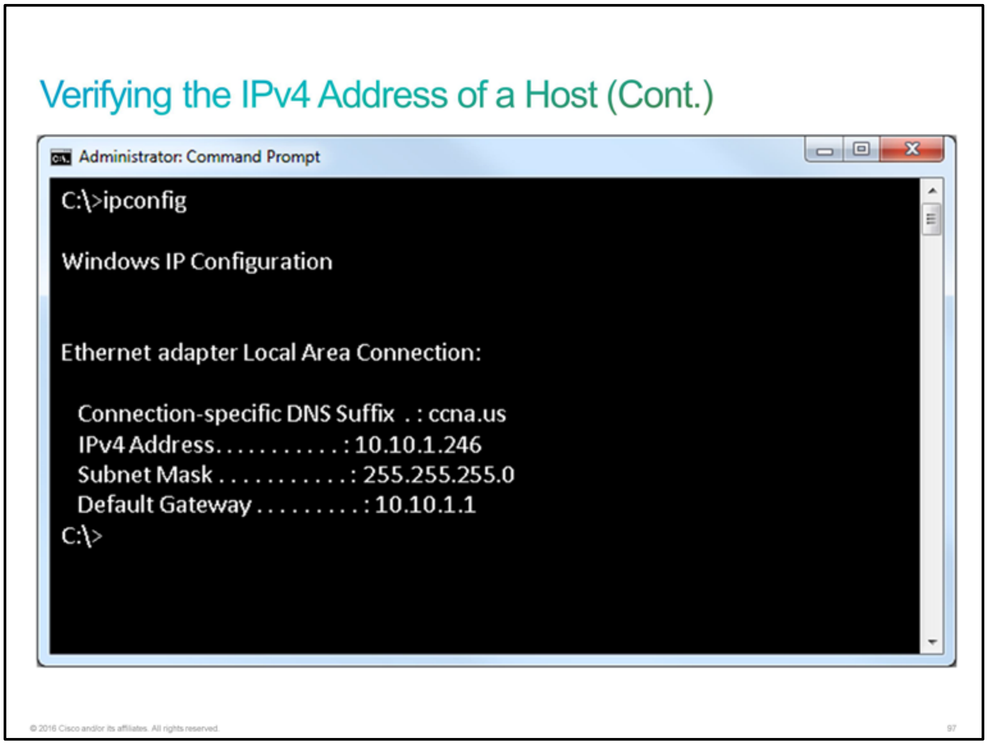
You use the **ipconfig** command to display all current TCP/IP network configuration values at the command line of a Windows computer. Using different command options, you can also use the **ipconfig** command to view and refresh DHCP and DNS settings. Used without command options, the **ipconfig** command displays the IP address, subnet mask, and default gateway for all adapters.

The following is the syntax for the **ipconfig** command:

```
ipconfig [/ all] [/ renew [adapter]] [/ release [adapter]] [/displaydns] [/flushdns]
```

These command options are commonly used:

- **/all**—This option displays the complete TCP/IP configuration for all adapters, including DHCP and DNS configuration. Without this parameter, the **ipconfig** command displays only the IP address, subnet mask, and default gateway values for each adapter. Adapters can represent physical interfaces, such as installed network adapters, or logical interfaces, such as dial-up connections.
- **/renew [***adapter***]**—This option renews DHCP configuration for all adapters (if an adapter is not specified) or for a specific adapter if the adapter parameter is included. This parameter is available only on computers with adapters that are configured to obtain an IP address automatically. To specify an adapter name, enter the adapter name that appears when you use **ipconfig** without parameters.
- **/release** [*adapter*]—This option sends a DHCPRELEASE message to the DHCP server to release the current DHCP configuration and discard the IP address configuration for either all adapters (if an adapter is not specified) or for a specific adapter if the adapter parameter is included. This parameter disables TCP/IP for adapters that are configured to obtain an IP address automatically. To specify an adapter name, enter the adapter name that appears when you use **ipconfig** without parameters.
- **/displaydns**—This option displays the contents of the host DNS cache. When an IP host makes a DNS query for a hostname, it caches the result to avoid unnecessary queries.
- **/flushdns**—This option deletes the host DNS cache. This option is useful if the IP address that is associated with a hostname has changed, but the host is still caching the old IP address.
- **/?**—This option displays help at the command prompt.



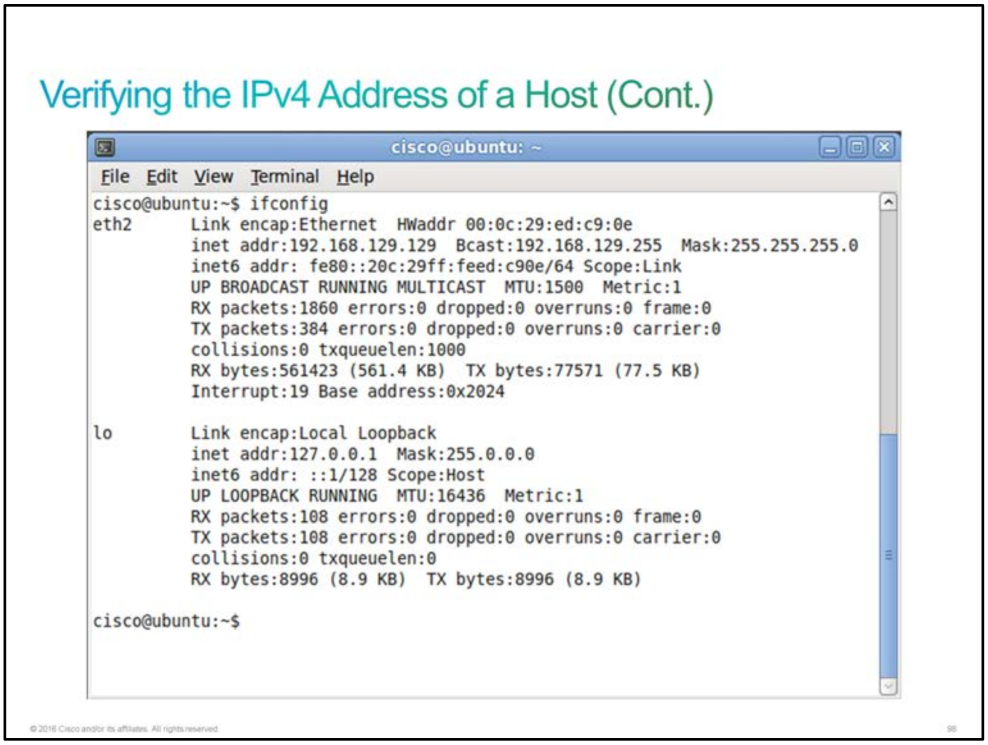
Verifying the IPv4 Address of a Host (Cont.)

Administrator: Command Prompt
© 2016 Cisco and/or its affiliates. All rights reserved.
97

```
C:\>ipconfig

Windows IP Configuration


Ethernet adapter Local Area Connection:

   Connection-specific DNS Suffix  . : ccna.us
   IPv4 Address. . . . . . . . . . . : 10.10.1.246
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . : 10.10.1.1
C:\>
```

Note For additional information about **ipconfig** and the command syntax, use your favorite search engine and search for this string: **microsoft technet dd197434 site:microsoft.com**

On most Linux operating systems, the **ifconfig** command is used to perform the same tasks that **ipconfig** performs on Microsoft Windows operating systems.

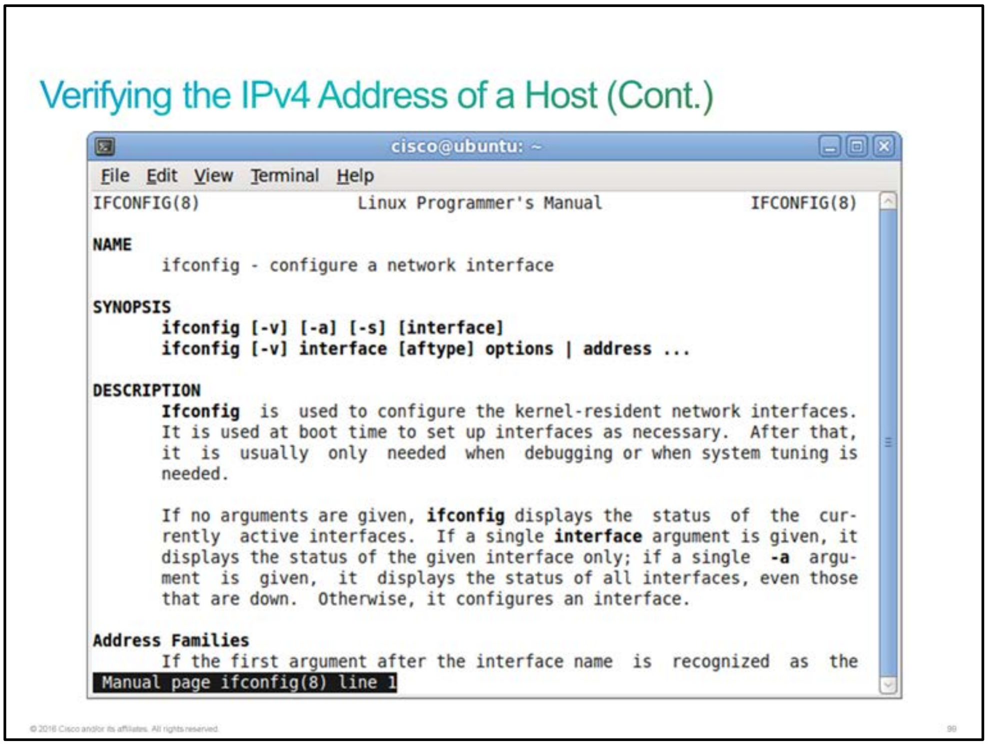


Verifying the IPv4 Address of a Host (Cont.)

```
cisco@ubuntu:~$ ifconfig
eth2      Link encap:Ethernet  HWaddr 00:0c:29:ed:c9:0e
          inet addr:192.168.129.129  Bcast:192.168.129.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:feed:c90e/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:1860 errors:0 dropped:0 overruns:0 frame:0
          TX packets:384 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:561423 (561.4 KB)  TX bytes:77571 (77.5 KB)
          Interrupt:19 Base address:0x2024

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:108 errors:0 dropped:0 overruns:0 frame:0
          TX packets:108 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:8996 (8.9 KB)  TX bytes:8996 (8.9 KB)

cisco@ubuntu:~$
```

© 2016 Cisco and/or its affiliates. All rights reserved. 98

On Linux systems, you can get the details of specific syntax for just about any command using the **man** (manual) command. In this example, **man ifconfig** was entered:



Verifying the IPv4 Address of a Host (Cont.)

```
IFCONFIG(8)              Linux Programmer's Manual              IFCONFIG(8)

NAME
       ifconfig - configure a network interface

SYNOPSIS
       ifconfig [-v] [-a] [-s] [interface]
       ifconfig [-v] interface [aftype] options | address ...

DESCRIPTION
       Ifconfig  is  used to configure the kernel-resident network interfaces.
       It is used at boot time to set up interfaces as necessary.  After that,
       it  is  usually  only  needed  when  debugging or when system tuning is
       needed.

       If no arguments are given, ifconfig displays the  status  of  the  cur-
       rently  active interfaces.  If a single interface argument is given, it
       displays the status of the given interface only; if a single  -a  argu-
       ment  is  given,  it  displays the status of all interfaces, even those
       that are down.  Otherwise, it configures an interface.

Address Families
       If the first argument after the interface name  is  recognized  as  the
 Manual page ifconfig(8) line 1
```

© 2016 Cisco and/or its affiliates. All rights reserved. 99

Challenge

1. Where in the TCP/IP stack does IP operate?

   A. Internet layer
   B. application layer
   C. transport layer
   D. network access layer

2. What are two characteristics of a network ID from an IP address? (Choose two.)

   A. identifies the network of which the host is a part
   B. used by routers to maintain information about routes
   C. identifies the individual host
   D. assigned by organizations to individual devices
   E. identifies the individual subnet

3. What are two characteristics of a host ID from an IP address? (Choose two.)

   A. identifies the network of which the host is a part
   B. used by routers to maintain information about routes
   C. identifies the individual host
   D. assigned by organizations to individual devices
   E. identifies the individual subnet

4. What is the length of the source address in the IPv4 header?

   A. 8 bits
   B. 8 bytes
   C. 32 bytes
   D. 4 bytes

5. What is the correct binary representation of the IPv4 address 192.168.16.101?

   A. 11000000.10101000.00000100.01100101
   B. 11000000.00010000.10101000.01100101
   C. 11000000.10101000.00010000.01100101
   D. 11000000.10101000.00010000.00110101

6. What is the range of the Class C private IP addresses?

   A. 192.0.0.0 to 192.167.255.255
   B. 192.169.0.0 to 223.255.255.255
   C. 192.168.0.0 to 192.168.255.255
   D. 192.168.0.0 to 192.168.0.255

7. What is the range of the Class A public IP addresses?

   A. 1.0.0.0 to 9.255.255.255, 11.0.0.0 to 126.255.255.255
   B. 1.0.0.0 to 126.255.255.255
   C. 0.0.0.0 to 10.255.255.255
   D. 0.0.0.0 to 10.0.0.255

Answer Key

Challenge

1. A
2. A, B, E
3. C, D
4. D
5. C
6. C
7. A

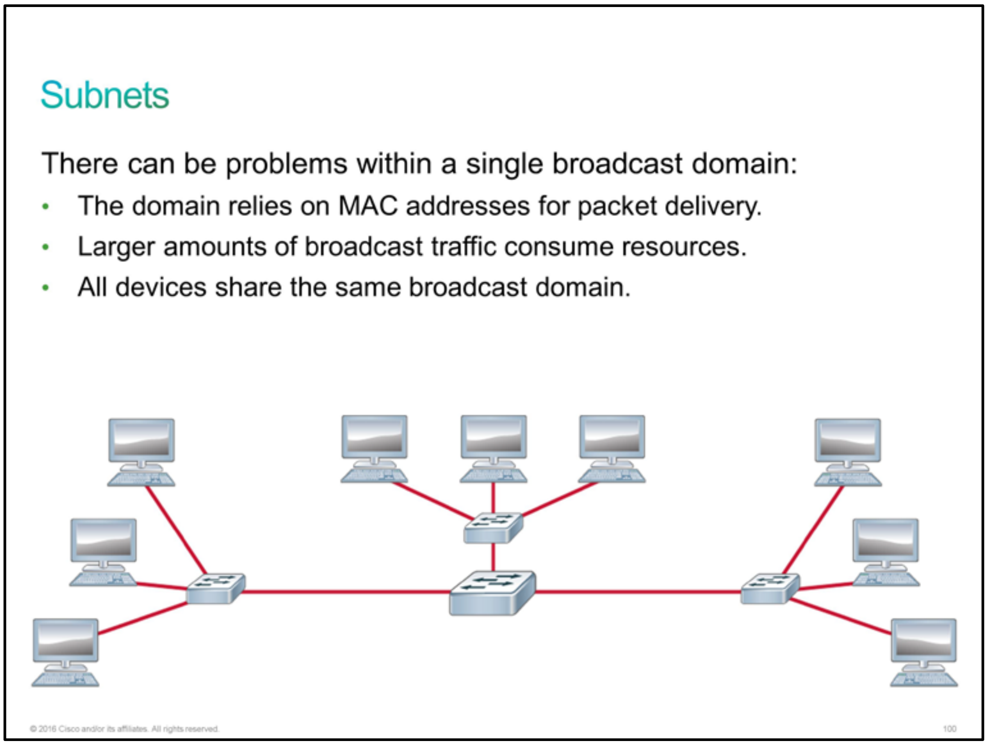
Lesson 2: Understanding IP Addressing and Subnets

Introduction

You did well demonstrating your knowledge on the Internet protocol. But before you are ready to go on deployments involving routers and Internet connectivity, you also need to demonstrate your knowledge of subnetting and VLSM.

Subnets

Network administrators often need to divide networks, especially large networks, into subnetworks, or subnets, to provide scalability.



Subnets

There can be problems within a single broadcast domain:

- The domain relies on MAC addresses for packet delivery.
- Larger amounts of broadcast traffic consume resources.
- All devices share the same broadcast domain.

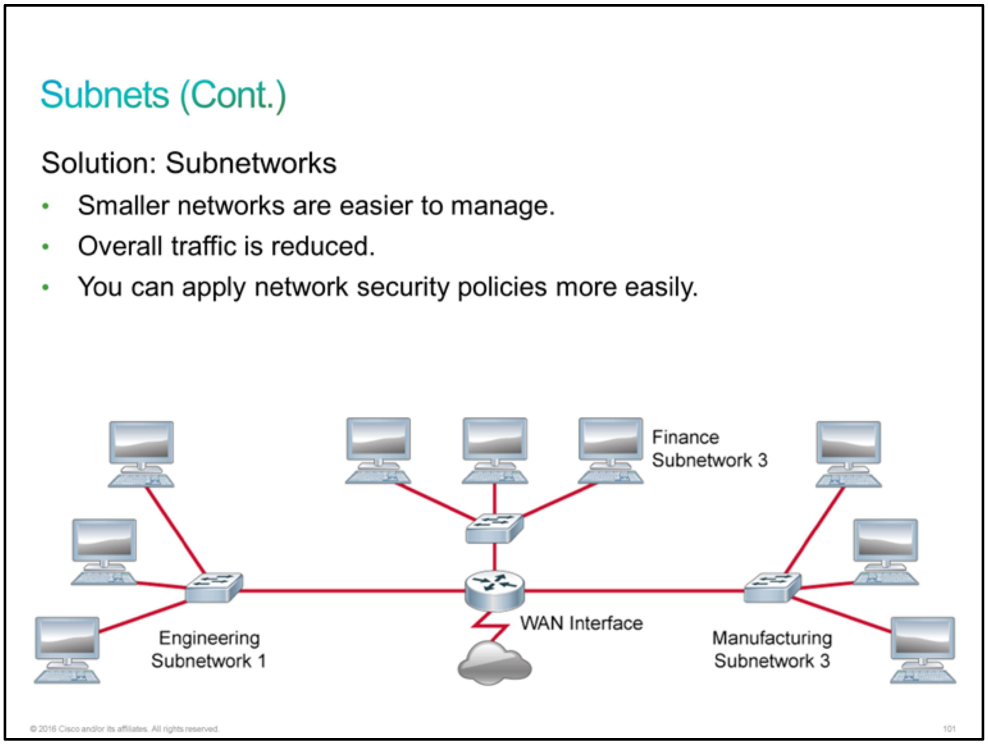
A company that occupies a three-story building might have a network that is divided by floors, with each floor divided into offices. Think of the building as the network, the floors as the three subnets, and the offices as the individual host addresses.

A subnet segments the hosts within the network. Without subnets, the network has a flat topology. A flat topology relies on MAC addresses to deliver packets. MAC addresses have no hierarchical structure. As the network grows, the use of the network bandwidth becomes less efficient.

There are other disadvantages to a flat network. All devices share the same broadcast domain, and it is difficult to apply security policies because there are no boundaries between devices.

Note A broadcast domain is a network in which all devices can reach each other by broadcast.

On a switch-connected network, the host sees all the broadcasts in the broadcast domain. You can use routers to separate networks by breaking the network into multiple subnets or multiple broadcast domains.



Subnets (Cont.)

Solution: Subnetworks

- Smaller networks are easier to manage.
- Overall traffic is reduced.
- You can apply network security policies more easily.

© 2016 Cisco and/or its affiliates. All rights reserved.

101

The advantages of subnetting a network are as follows:

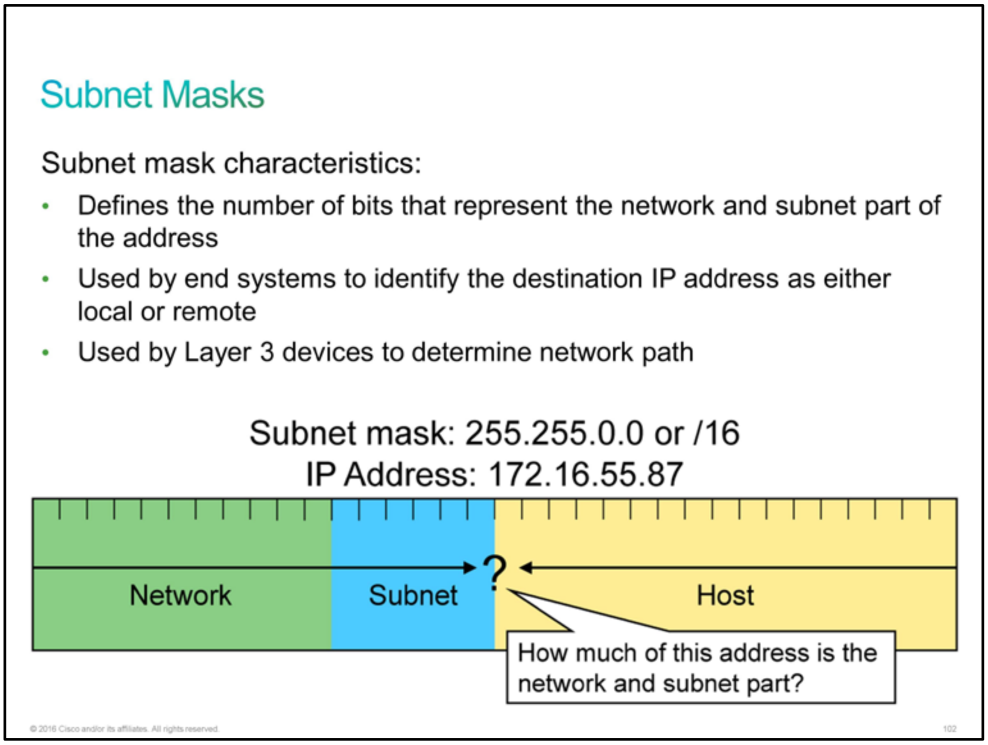
- Smaller networks are easier to manage and map to geographical or functional requirements.
- Subnetting enables you to create multiple logical networks from a single Class A, B, or C network address.
- Overall network traffic is reduced, which can improve performance.
- You can more easily apply network security measures at the interconnections between subnets than within a large single network.

In multiple-network environments, each subnetwork may be connected to the Internet by a single router. The figure shows one router connecting multiple subnetworks to the Internet. The details of the internal network environment and how the network is divided into multiple subnetworks are inconsequential to other IP networks.

The IP addressing that is used in the flat network must be modified to accommodate the required segmentation. A subnet mask identifies the network-significant portion of an IP address. The network-significant portion of an IP address is simply the part that identifies the network that the host device is on. This part is called the *network address* and defines every subnetwork. The use of segmentation is important for the routing operation to be efficient.

Subnet Masks

A subnet mask is a 32-bit combination that is used for routing traffic within a subnet. It describes which portion of an IP address refers to the subnet and which part refers to the host. As you already know, an IP address has two components; the network part and the host part. Subnetting enables the network administrator to further divide the host part. The first part identifies the subnetwork (subnet) to which are device belongs. The other part identifies the host.



Subnet Masks

Subnet mask characteristics:

- Defines the number of bits that represent the network and subnet part of the address
- Used by end systems to identify the destination IP address as either local or remote
- Used by Layer 3 devices to determine network path

Subnet mask: 255.255.0.0 or /16
IP Address: 172.16.55.87

How much of this address is the network and subnet part?

© 2016 Cisco and/or its affiliates. All rights reserved.

102

How do you know how many bits represent the network portion of the address and how many bits represent the host portion? When you express an IPv4 network address, you add a prefix length to the network address. The prefix length is the number of bits in the address that give the network portion. For example, in 172.16.55.87 /20, /20 is the prefix length. It tells you that the first 20 bits are the network address. The remaining 12 bits, the last octet, is the host portion. The entity that is used to specify the network portion of an IPv4 address to the network devices is called the *subnet mask*. The subnet mask consists of 32 bits, just as the address does, and uses 1s and 0s to indicate which bits of the address are network bits and which bits are host bits. You express the subnet mask in the same dotted decimal format as the IPv4 address. The subnet mask is created by placing a binary 1 in each bit position that represents the network portion and placing a binary 0 in each bit position that represents the host portion. A /20 prefix is expressed as a subnet mask of 255.255.240.0 (11111111.11111111.11110000.00000000). The remaining bits (low order) of the subnet mask are zeroes, indicating the host address within the network.

The subnet mask is configured on a host with the IPv4 address to define the network portion of that address.

Networks are not always assigned the same prefix. Depending on the number of hosts on the network, the prefix that is assigned may be different. Having a different prefix number changes the host range and broadcast address for each network.

For example, take a look at the host 10.1.20.70/26:

- Address:
  - 10.1.20.70
  - 00001010.00000001.00010100.01000110
- Subnet mask:
  - 255.255.255.192
  - 11111111.11111111.11111111.11000000
- Network address:
  - 10.1.20.64
  - 00001010.00000001.00010100.01000000

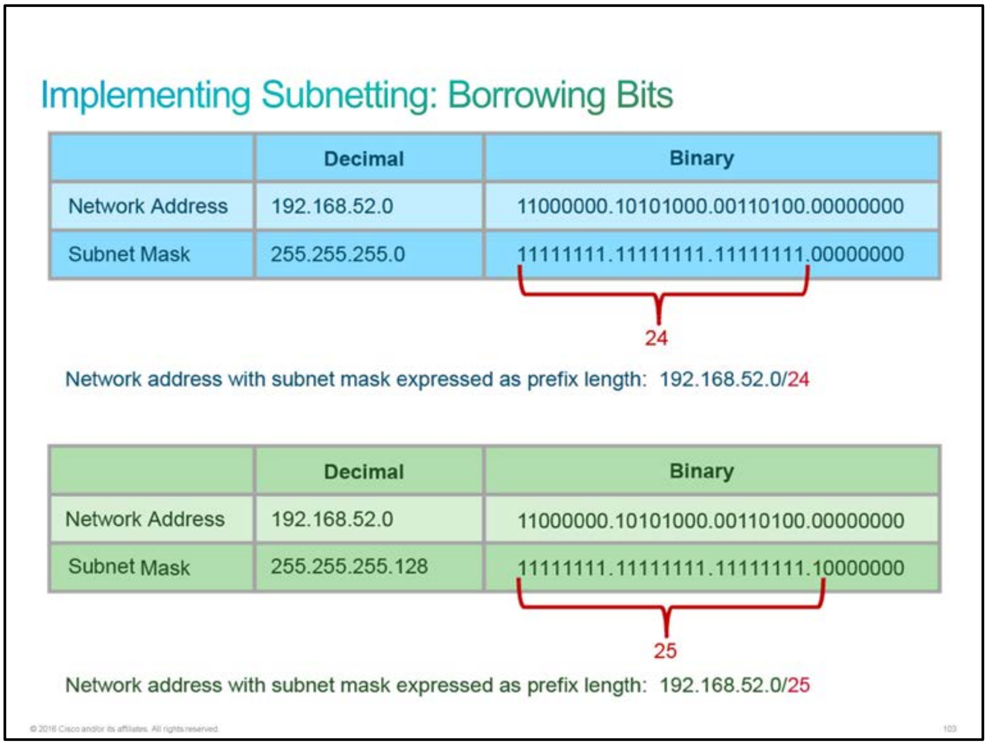
Implementing Subnetting: Borrowing Bits

To implement subnets, follow this procedure:

- Determine the IP address for your network as assigned by the registry authority.
- Based on your organizational and administrative structure, determine the number of subnets that are required for the network. Be sure to plan for growth.
- Based on the address class and required number of subnets, determine the number of bits that you need to borrow from the host ID.
- Determine the binary and decimal value of the new subnet mask that results from borrowing bits from the host ID.
- Apply the subnet mask to the network IP address to determine the subnet and host addresses. Also determine the network and broadcast addresses for each subnet.
- Assign subnet addresses to specific interfaces for all devices that are connected to the network.

To subnet a network address, you must borrow host bits and use them as subnet bits, as the following figure shows. This approach ignores the boundaries of classful networks, on which the predefined classes of IP addresses (A, B, and C) are based, and introduces the classless network. Bits must be borrowed consecutively, starting with the first host bit on the left.

Take a look at the following figure. The top table shows a standard Class C network address that is not subnetted. The bottom table shows the same address after it is subnetted by borrowing one host bit. Notice that the prefix length has changed from 24 to 25. The network IP address itself is unchanged, although it is now considered a subnetwork (subnet) and is one of two subnets that have been created. The subnet mask has changed from 255.255.255.0 in decimal to 255.255.255.128 because the 128 bit is now turned on in the last octet.



Implementing Subnetting: Borrowing Bits

	Decimal	Binary
Network Address	192.168.52.0	11000000.10101000.00110100.00000000
Subnet Mask	255.255.255.0	11111111.11111111.11111111.00000000

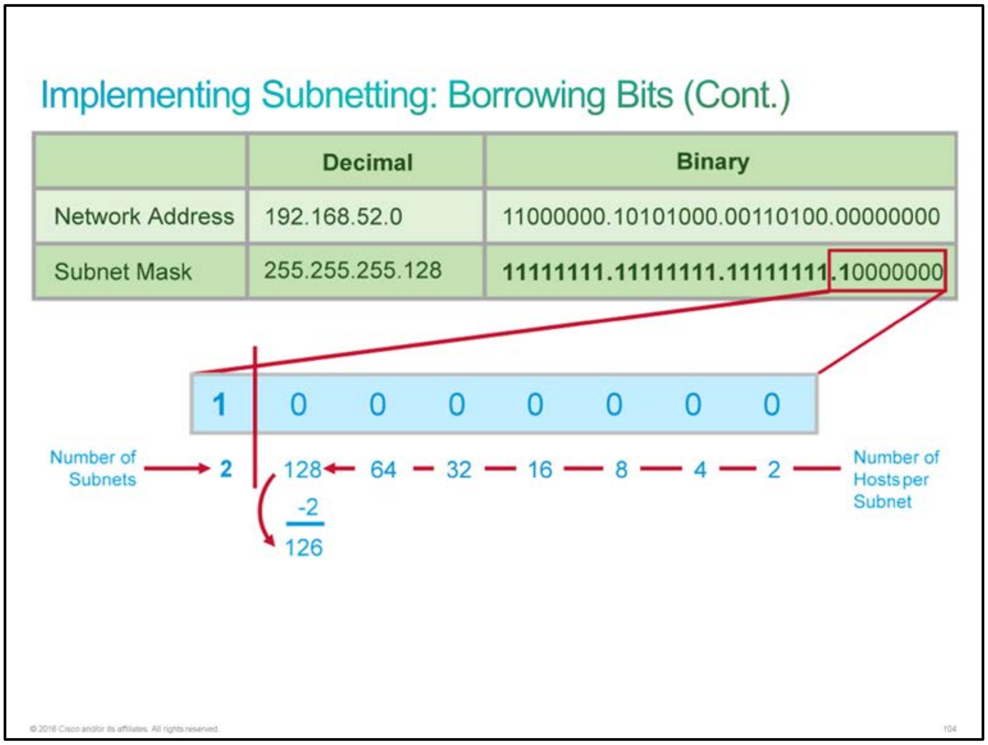
Network address with subnet mask expressed as prefix length: 192.168.52.0/24

	Decimal	Binary
Network Address	192.168.52.0	11000000.10101000.00110100.00000000
Subnet Mask	255.255.255.128	11111111.11111111.11111111.10000000

Network address with subnet mask expressed as prefix length: 192.168.52.0/25

Each time that a bit is borrowed, the number of subnet addresses increases, and the number of host addresses that are available per subnet decreases. The algorithm that is used to compute the number of subnets and hosts uses powers of two. Therefore, borrowing one host bit enables you to create 2^1 (2) subnets, borrowing 2 bits gives you 2^2 (4) subnets, and so on.

As the following figure shows, you can also determine how many host addresses are available when you borrow a given number of bits by counting by powers of two. Starting with the far-right host bit, begin with 2^1 (2) and increase by powers of two. Then, subtract two. The figure shows that borrowing 1 bit for subnetting the address in the example leaves 7 bits for hosts. The formula to determine the number of hosts for this example is 2^7– 2, which calculates to 126 host addresses per subnet.

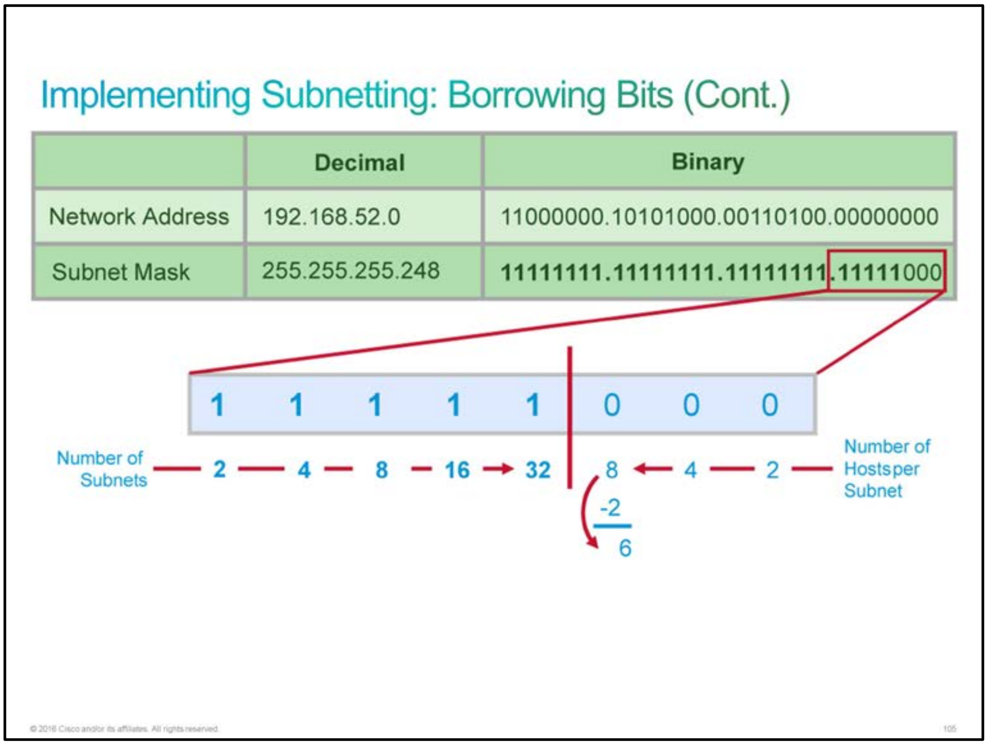


Implementing Subnetting: Borrowing Bits (Cont.)

	Decimal	Binary
Network Address	192.168.52.0	11000000.10101000.00110100.00000000
Subnet Mask	255.255.255.128	**11111111.11111111.11111111.1**0000000

1 0 0 0 0 0 0 0
Number of Subnets
2
128
64
32
16
8
4
2
Number of Hosts per Subnet
-2
126
© 2016 Cisco and/or its affiliates. All rights reserved.
104

Here is another example, in which five host bits are borrowed for subnetting. In this example, 32 subnets are created, and only 6 host addresses are available for each subnet. The new subnet mask is 11111111.11111111.11111111.11111000, which equates to 255.255.255.248 in decimal.



Implementing Subnetting: Borrowing Bits (Cont.)

	Decimal	Binary
Network Address	192.168.52.0	11000000.10101000.00110100.00000000
Subnet Mask	255.255.255.248	**11111111.11111111.11111111.11111**000

© 2016 Cisco and/or its affiliates. All rights reserved.

105

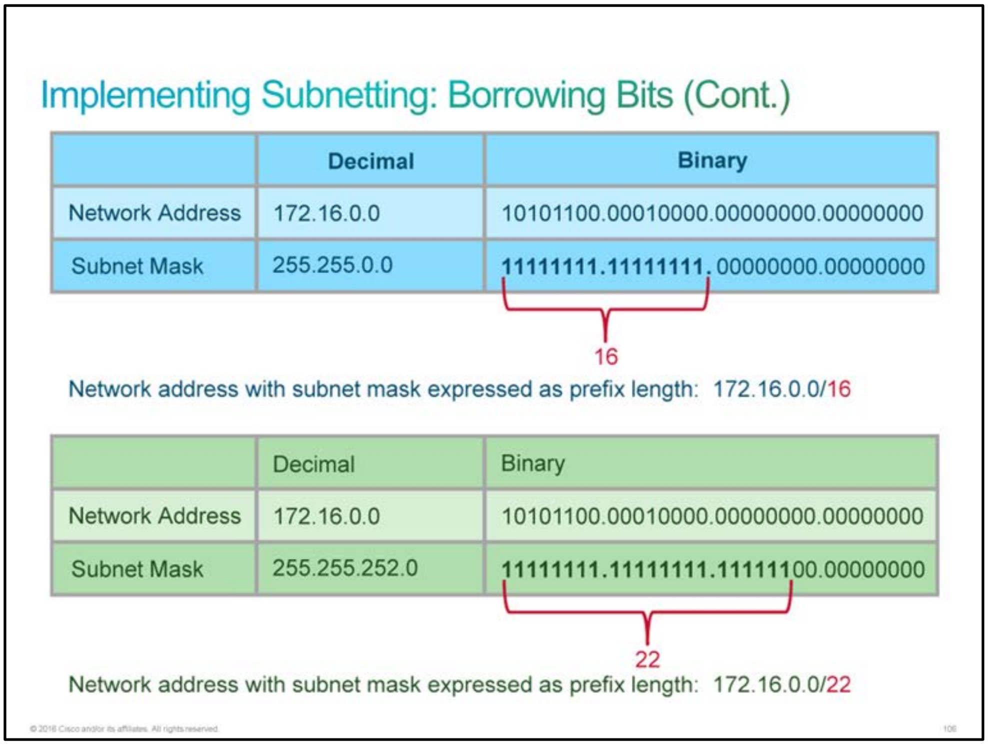
You can use the following formula to calculate the number of subnets that are created by borrowing a given number of host bits:

Number of subnets = 2^s (where *s* is the number of bits borrowed)

You can use a similar formula to calculate the number of host addresses that are available when a given number of host bits are borrowed:

Number of hosts = 2^h (where *h* is the number of host bits remaining after bits are borrowed) – 2

The following figure shows the subnetting of a Class B network address. The top table shows a network address with the default Class B subnet mask, 255.255.0.0. The second table shows the same address after it is subnetted by borrowing six host bits. Notice that the prefix length has changed from 16 to 22. The network IP address itself is unchanged, but the subnet mask has changed from 255.255.0.0 in decimal to 255.255.252.0.



Implementing Subnetting: Borrowing Bits (Cont.)

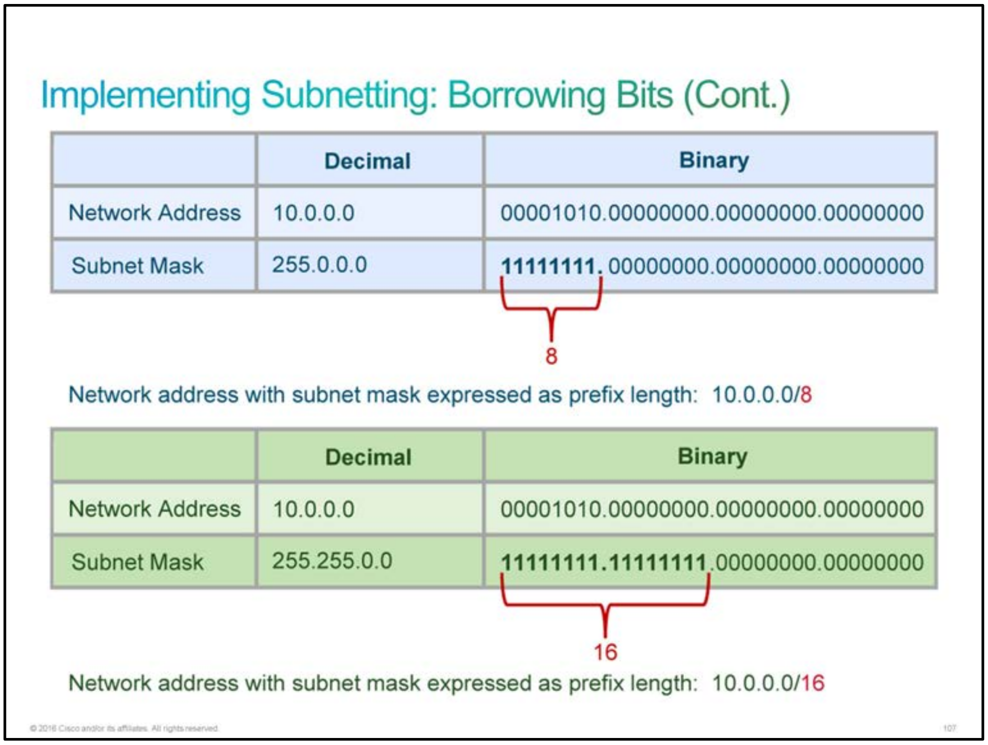
	Decimal	Binary
Network Address	172.16.0.0	10101100.00010000.00000000.00000000
Subnet Mask	255.255.0.0	**11111111.11111111.**00000000.00000000

Network address with subnet mask expressed as prefix length: 172.16.0.0/16

	Decimal	Binary
Network Address	172.16.0.0	10101100.00010000.00000000.00000000
Subnet Mask	255.255.252.0	**11111111.11111111.111111**00.00000000

Network address with subnet mask expressed as prefix length: 172.16.0.0/22

The next figure shows the subnetting of a Class A network address. The top table shows a network address with the default Class A subnet mask, 255.0.0.0. The bottom table shows the same address after it is subnetted by borrowing 8 host bits. Notice that the prefix length has changed from 8 to 16. The network IP address itself is unchanged, but the subnet mask has changed from 255.0.0.0 in decimal to 255.255.0.0.



Implementing Subnetting: Borrowing Bits (Cont.)

	Decimal	Binary
Network Address	10.0.0.0	00001010.00000000.00000000.00000000
Subnet Mask	255.0.0.0	**11111111.** 00000000.00000000.00000000

Network address with subnet mask expressed as prefix length: 10.0.0.0/8

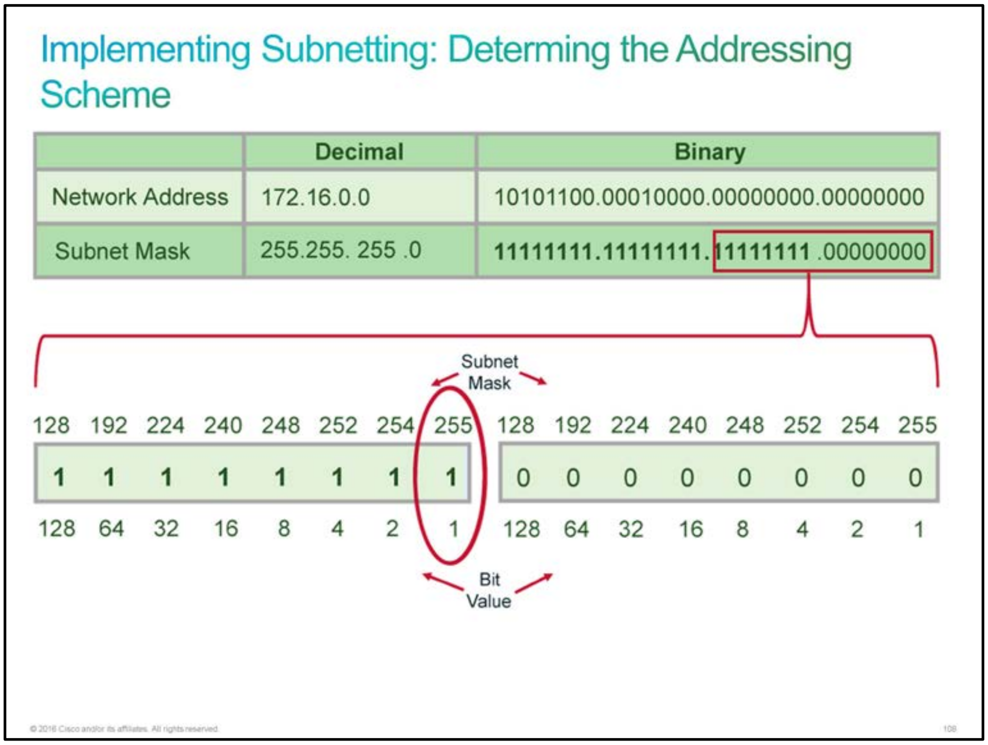
	Decimal	Binary
Network Address	10.0.0.0	00001010.00000000.00000000.00000000
Subnet Mask	255.255.0.0	**11111111.11111111**.00000000.00000000

Network address with subnet mask expressed as prefix length: 10.0.0.0/16

Implementing Subnetting: Determing the Addressing Scheme

If a network address is subnetted, the first subnet that is obtained after subnetting the network address is called subnet zero. To determine each subsequent subnet address, increase the network address by the bit value for the last bit that you borrowed.

In the following example, 8 bits are borrowed for subnetting the same network address, 172.16.0.0. The first subnet address is 172.16.0.0, the zero subnet. The last bit borrowed is the bit with the value of 1, so the next subnet address is 172.16.1.0.



Implementing Subnetting: Determing the Addressing Scheme

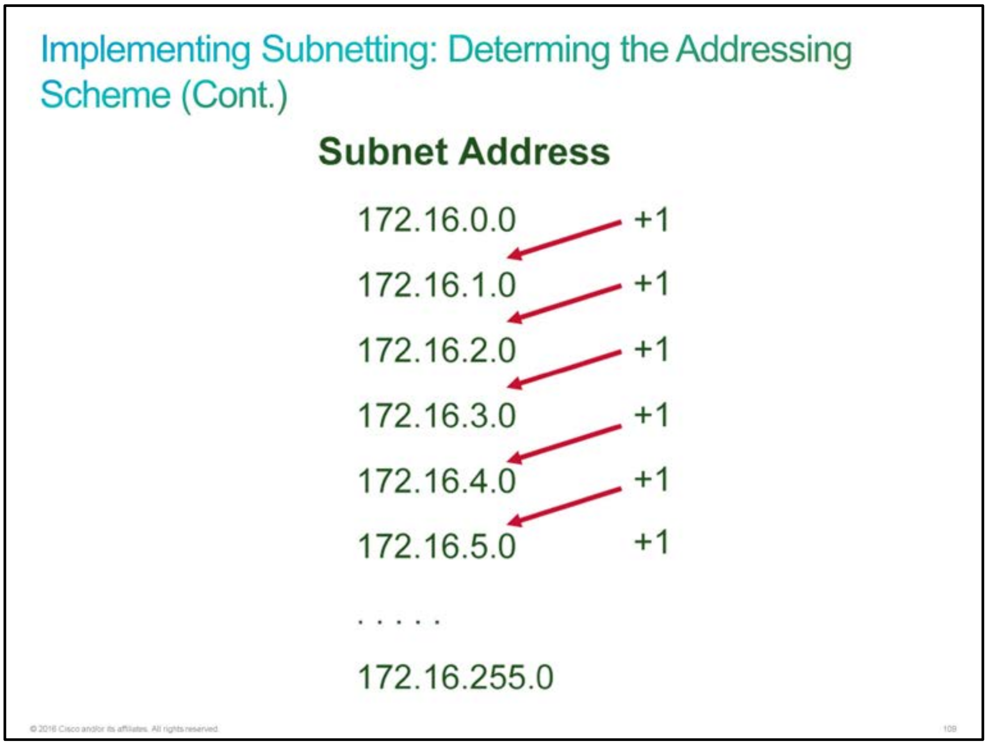
	Decimal	Binary
Network Address	172.16.0.0	10101100.00010000.00000000.00000000
Subnet Mask	255.255. 255 .0	**11111111.11111111.11111111** .00000000

Subnet Mask

128	192	224	240	248	252	254	255	128	192	224	240	248	252	254	255
1	**1**	**1**	**1**	**1**	**1**	**1**	**1**	0	0	0	0	0	0	0	0
128	64	32	16	8	4	2	1	128	64	32	16	8	4	2	1

Bit Value

The following figure shows the first six subnets and the last subnet that are created by borrowing the 8 bits.



Implementing Subnetting: Determing the Addressing Scheme (Cont.)

Subnet Address

172.16.0.0 +1

172.16.1.0 +1

172.16.2.0 +1

172.16.3.0 +1

172.16.4.0 +1

172.16.5.0 +1

. . . . .

172.16.255.0

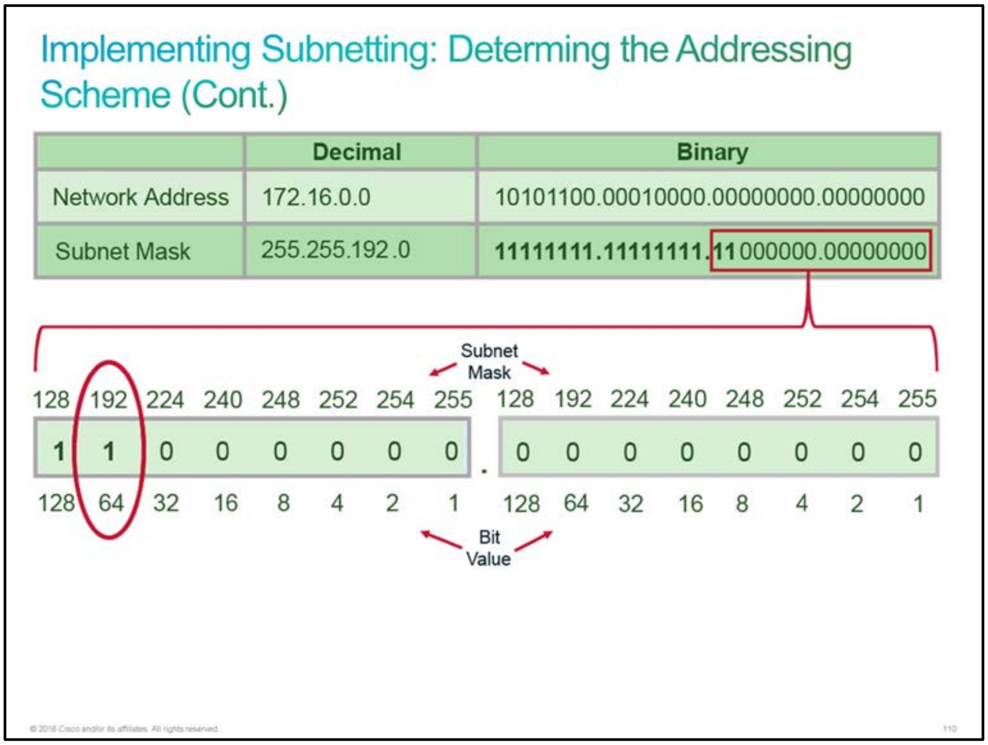
© 2016 Cisco and/or its affiliates. All rights reserved. 109

Here are the host addresses and broadcast addresses for those subnets.

Host Addresses and Broadcast Addresses

Subnet Address	Host Address Range	Broadcast Address
172.16.0.0	172.16.0.1–172.16.0.254	172.16.0.255
172.16.1.0	172.16.1.1–172.16.1.254	172.16.1.255
172.16.2.0	172.16.2.1–172.16.2.254	172.16.2.255
172.16.3.0	172.16.3.1–172.16.3.254	172.16.3.255
172.16.4.0	172.16.4.1–172.16.4.254	172.16.4.255
172.16.5.0	172.16.5.1–172.16.5.254	172.16.5.255
...	...	...
172.16.255.0	172.16.255.1–172.16.255.254	172.16.255.255

In the following figure, Class B network address 172.16.0.0 has been subnetted by borrowing two host bits. The first subnet address is again 172.16.0.0, the zero subnet. The last bit borrowed is the bit with the value of 64, so the next subnet address is 172.16.64.0



Implementing Subnetting: Determing the Addressing Scheme (Cont.)

	Decimal	Binary
Network Address	172.16.0.0	10101100.00010000.00000000.00000000
Subnet Mask	255.255.192.0	**11111111.11111111.11**000000.00000000

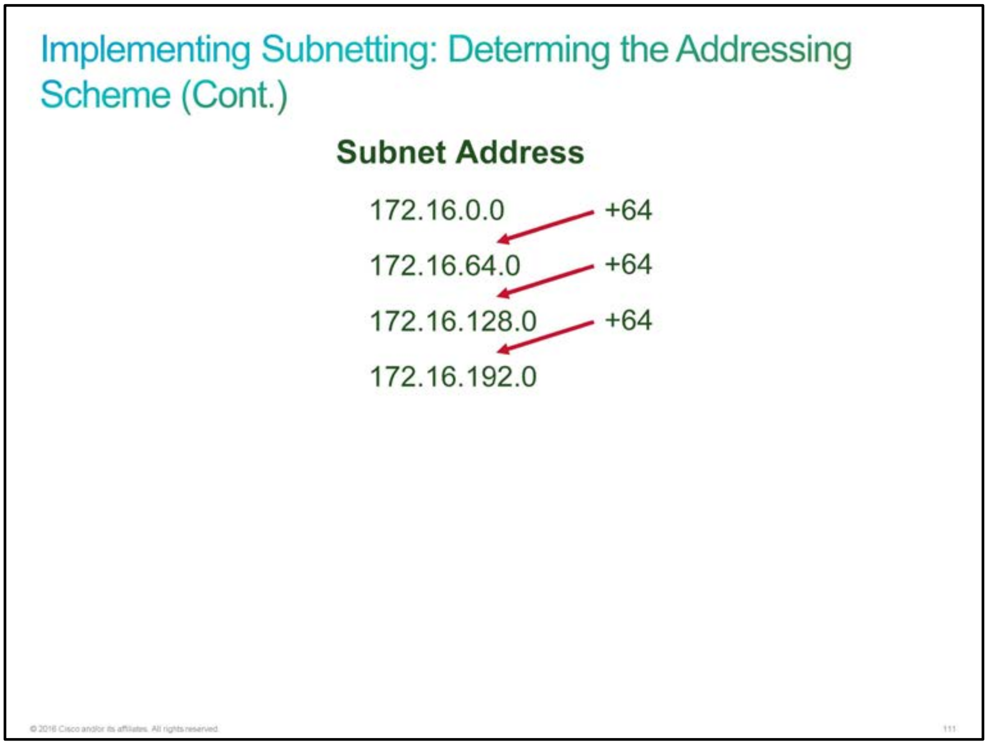
Subnet Mask

128	192	224	240	248	252	254	255		128	192	224	240	248	252	254	255
1	**1**	0	0	0	0	0	0	.	0	0	0	0	0	0	0	0
128	64	32	16	8	4	2	1		128	64	32	16	8	4	2	1

Bit Value

© 2016 Cisco and/or its affiliates. All rights reserved. 110

The following figure shows all the subnets that are created by borrowing the 2 bits. The subnet 172.16.192.0 is the last subnet because 192 + 64 = 256, and the highest possible value for any given octet is 255.



Implementing Subnetting: Determing the Addressing Scheme (Cont.)

Subnet Address

172.16.0.0 +64

172.16.64.0 +64

172.16.128.0 +64

172.16.192.0

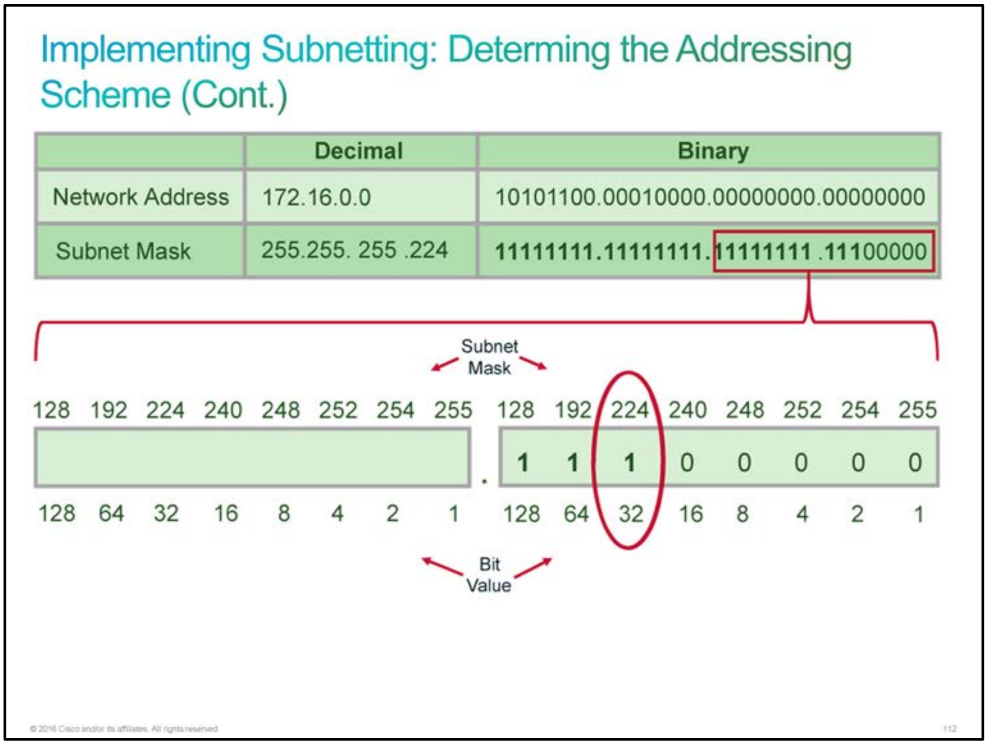
© 2016 Cisco and/or its affiliates. All rights reserved. 111

The following table shows the valid host addresses for each subnet that was created by borrowing 2 bits. You may recall the following statement from a previous topic: "You can determine how many host addresses are available when you borrow a given number of bits by counting by powers of two. Starting with the far-right host bit, begin with 2^1 (2) and increase by powers of two. Then subtract two." The reason that you have to subtract two is to allow for a broadcast address for each subnet and to avoid using the subnet address as a host address. The table shows the valid host IP address range for each network that you examined in the previous practice question.

Valid Host Addresses for Each Subnet Created by Borrowing 2 Bits

Subnet Address	Valid Host Address Range	Broadcast Address
172.16.0.0	172.16.0.1–172.16.63.254	172.16.63.255
172.16.64.0	172.16.64.1–172.16.127.254	172.16.127.255
172.16.128.0	172.16.128.1–172.16.191.254	172.16.191.255
172.16.192.0	172.16.192.1–172.16.255.254	172.16.255.255

Here is one more example of subnetting the same Class B network address, this time borrowing 11 host bits for subnetting. Again, the first subnet address is 172.16.0.0. The second subnet address is 172.16.0.32 because the last bit borrowed has a value of 32. Notice that this time, the last borrowed bit is in the fourth octet. Therefore, the increment of 32 (the value of the last borrowed bit) is applied in the fourth octet.



Implementing Subnetting: Determining the Addressing Scheme (Cont.)

	Decimal	Binary
Network Address	172.16.0.0	10101100.00010000.00000000.00000000
Subnet Mask	255.255. 255 .224	11111111.11111111.11111111 .11100000

The following table shows the first 10 subnet addresses and the last subnet address (with the corresponding host addresses and broadcast addresses) that result from subnetting Class B network 172.16.0.0 by borrowing 11 host bits.

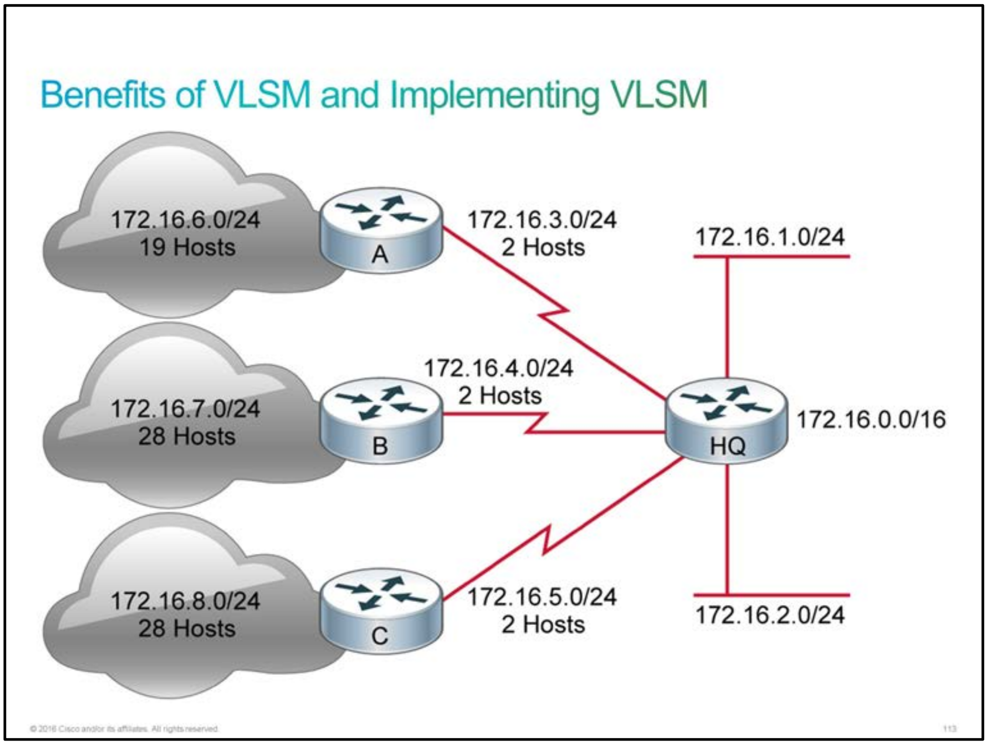
Subnet Address	Host Address Range	Broadcast Address
172.16.0.0	172.16.0.1–172.16.0.30	172.16.0.31
172.16.0.32	172.16.0.33–172.16.0.62	172.16.0.63
172.16.0.64	172.16.0.65–172.16.0.94	172.16.0.95
172.16.0.96	172.16.0.97–172.16.0.126	172.16.0.127
172.16.0.128	172.16.0.129–172.16.0.158	172.16.0.159
172.16.0.160	172.16.0.161–172.16.0.190	172.16.0.191
172.16.0.192	172.16.0.193–172.16.0.222	172.16.0.223
172.16.0.224	172.16.0.225–172.16.0.254	172.16.0.255
172.16.1.0	172.16.1.1–172.16.1.30	172.16.1.31
172.16.1.32	172.16.1.33–172.16.1.62	172.16.1.63
...	...	...
172.16.255.224	172.16.255.225–172.16.255.254	172.16.255.255

Benefits of VLSM and Implementing VLSM

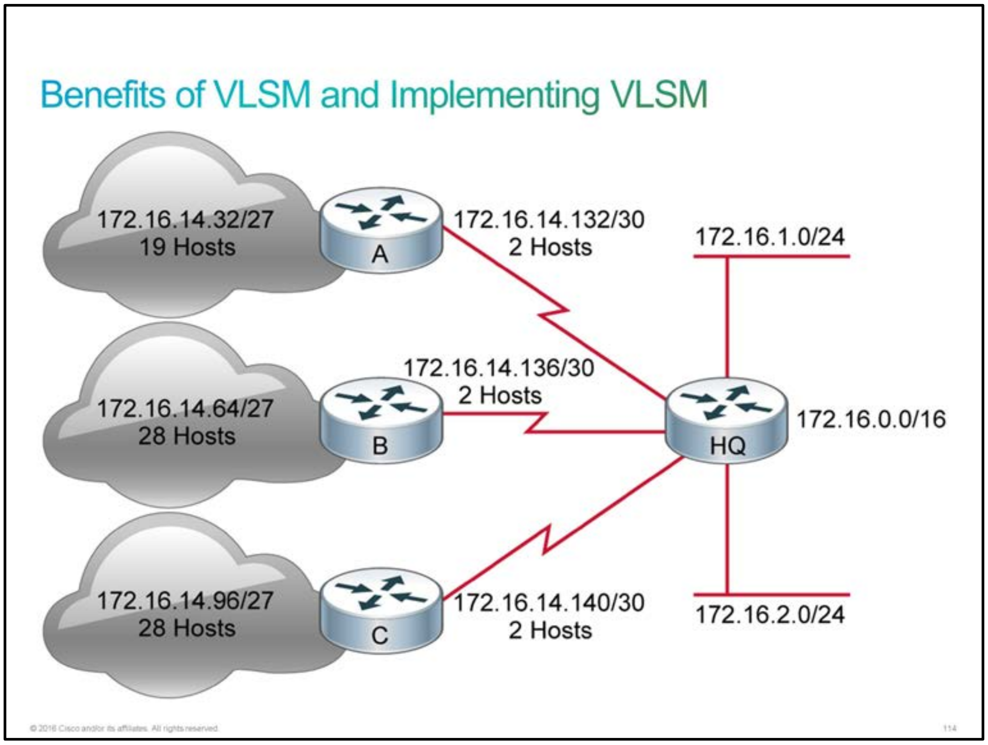
In all the subnetting examples in the previous topics, the same subnet mask was applied for all the subnets. This way, each subnet had the same number of available host addresses. You may need this approach sometimes, but, usually, having the same subnet mask for all subnets ends up wasting address space.

For example, in the following figure, Class B network 172.16.0.0 is subnetted by borrowing 8 host bits and applying a 24-bit subnet mask, which allows for 256 subnets with 254 host addresses each. In this example, many host addresses are wasted. Each WAN link needs only two host addresses, so 252 host addresses are wasted on each WAN link. Many host addresses are also wasted on other subnets. VLSM provides a solution.

VLSM allows you to include more than one subnet mask within a network to achieve more efficient use of IP addresses. Instead of using the same subnet mask for all subnets, you can use the most efficient subnet mask for each subnet. The most efficient subnet mask for a subnet is the mask that provides an appropriate number of host addresses for that individual subnet. For example, subnet 172.16.6.0 has only 19 hosts, so it does not need the 254 host addresses that the 24-bit mask allows. A 27-bit mask would provide 30 host addresses, which is much more appropriate for this subnet.


Benefits of VLSM and Implementing VLSM

In the next figure, the 172.16.0.0/16 network is again divided into subnetworks using a 24-bit subnet mask. However, one of the subnetworks in this range, 172.16.14.0/24, is further divided into smaller subnetworks using a 27-bit mask to accommodate the subnets that have 19 or 28 hosts. These smaller subnetworks range from 172.16.14.0/27 to 172.16.14.224/27. Then, one of these smaller subnets, 172.16.14.128/27, is further divided using a 30-bit mask, which creates subnets with only two hosts to be used on the WAN links. The subnets with the 30-bit mask range from 172.16.14.128/30 to 172.16.14.156/30.



Benefits of VLSM and Implementing VLSM

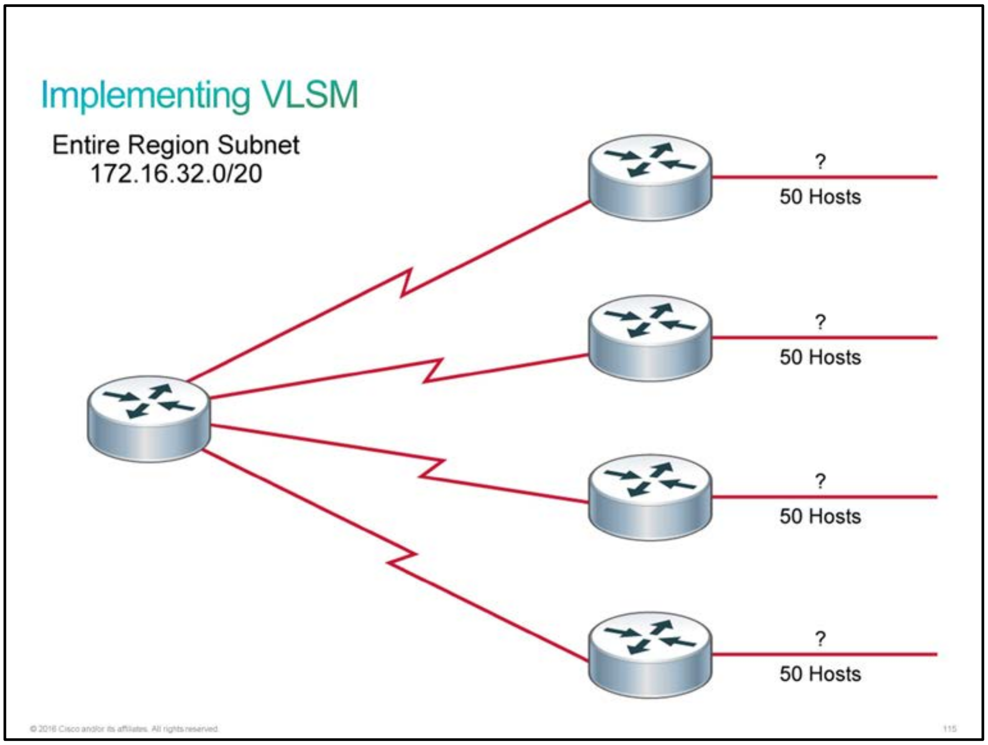
In addition to providing a solution to the problem of wasted IP addresses, VLSM has another important benefit: support for route summarization, which is also called route aggregation. The hierarchical addressing design of VLSM enables easier summarization of network addresses. Route summarization reduces the number of routes in routing tables by representing a range of network subnets in a single summary address. Smaller routing tables require less CPU time for routing lookups.

In the previous figure, the subnet 172.16.14.0/24 describes all the addresses that are further subnets of 172.16.14.0, including those addresses from subnet 172.16.14.0/27 to subnet 172.16.14.128/30.

VLSM is an important technology in large routed networks. It can only be used in networks that run routing protocols that support VLSM. These protocols include RIPv2, OSPF, and EIGRP.

Implementing VLSM

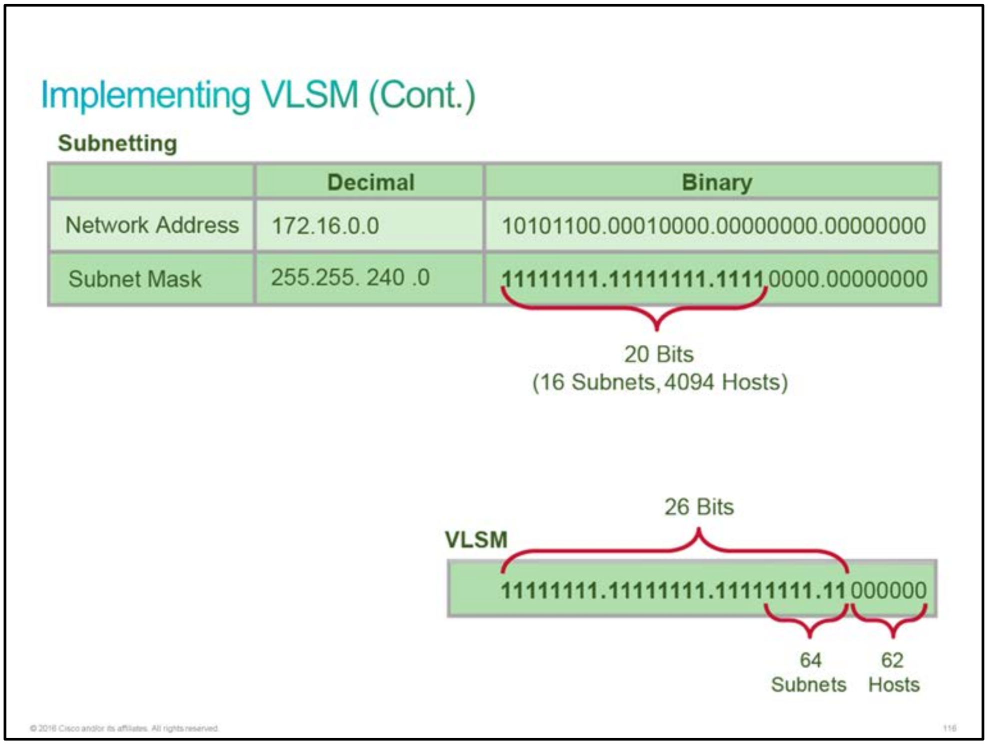
The network 172.16.0.0 has already been subnetted by applying a 20-bit subnet mask. One of the resulting subnet addresses, 172.16.32.0/20, is used for the region of the enterprise network that the following figure shows. This region needs to assign addresses to multiple LANs. Each LAN must have 50 hosts. You can use VLSM to further subnet the address 172.16.32.0/20 to give you more subnet addresses with fewer hosts per subnet.



Implementing VLSM

Entire Region Subnet
172.16.32.0/20

The next figure shows in binary the original subnetting of the 172.16.0.0/16 network by borrowing 4 host bits, which provided 16 subnets with 4094 host addresses each. The figure also shows how further subnetting with VLSM increases the number of subnets and provides the desired number of host addresses per subnet. Counting the additional host bits that are borrowed by powers of 2 from left to right shows that 64 subnets are created. Counting the host bits by powers of 2 from right to left and then subtracting 2 shows that 62 host addresses are available per subnet.



Implementing VLSM (Cont.)

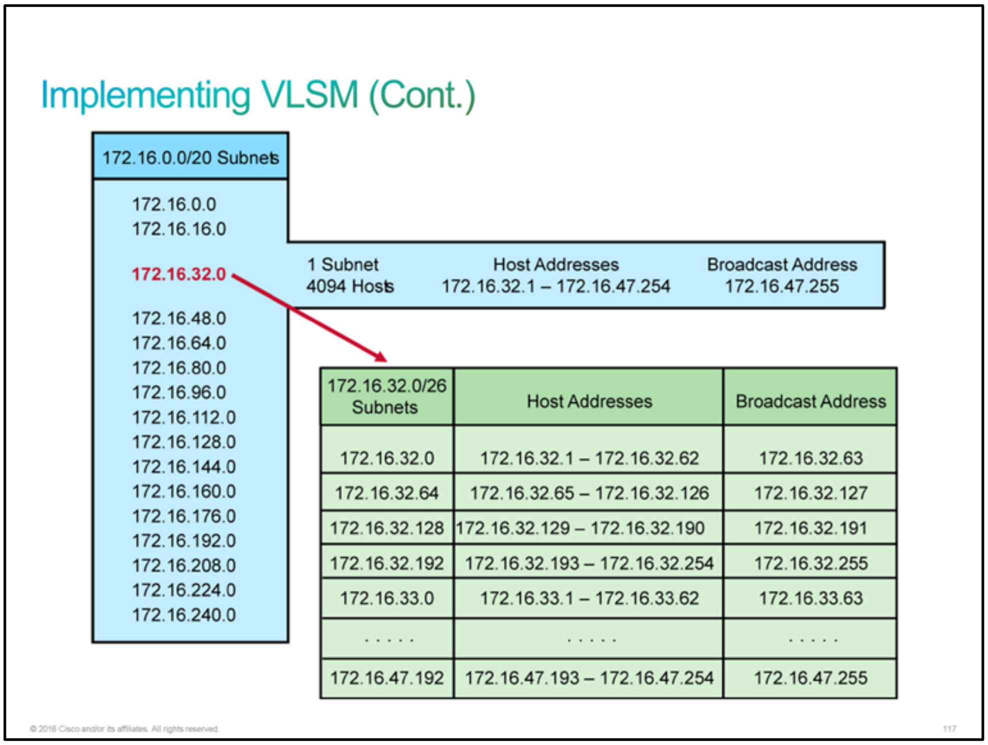
Subnetting

	Decimal	Binary
Network Address	172.16.0.0	10101100.00010000.00000000.00000000
Subnet Mask	255.255. 240 .0	**11111111.11111111.1111**0000.00000000

© 2016 Cisco and/or its affiliates. All rights reserved.

116

The next figure shows the subnet addresses and host addresses that are achieved by using VLSM. The subnet for the region in this example, subnet 172.16.32.0/20, is further subnetted by applying a 26-bit mask as the previous figure shows.



Implementing VLSM (Cont.)

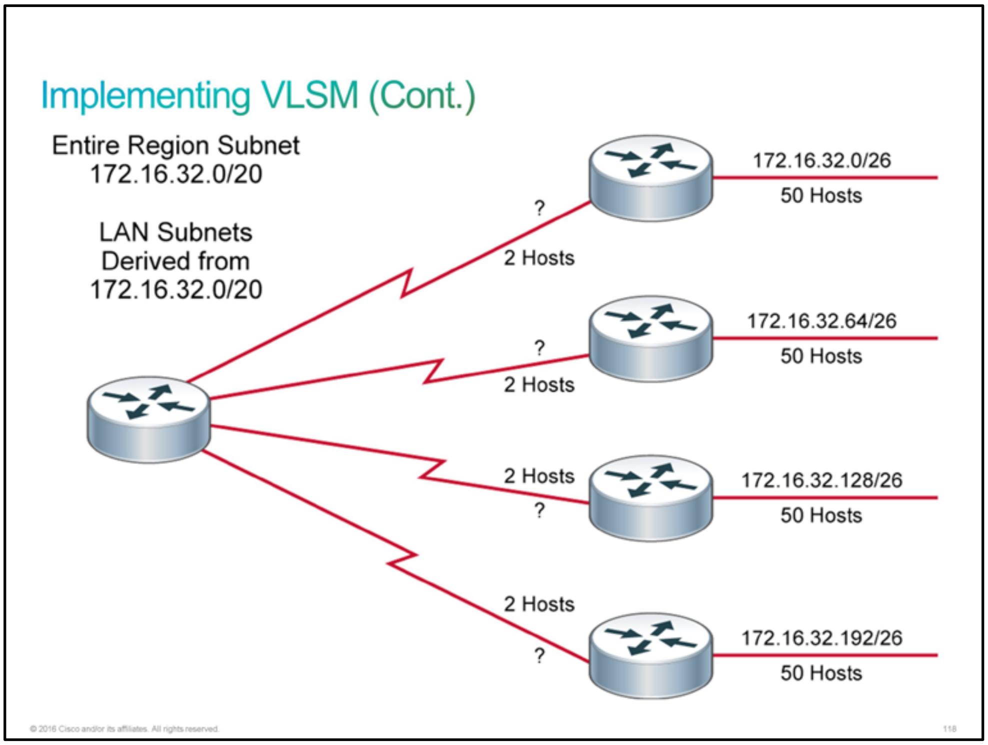
172.16.0.0/20 Subnets
172.16.0.0
172.16.16.0
172.16.32.0
172.16.48.0
172.16.64.0
172.16.80.0
172.16.96.0
172.16.112.0
172.16.128.0
172.16.144.0
172.16.160.0
172.16.176.0
172.16.192.0
172.16.208.0
172.16.224.0
172.16.240.0

1 Subnet	Host Addresses	Broadcast Address
4094 Hosts	172.16.32.1 – 172.16.47.254	172.16.47.255

172.16.32.0/26 Subnets	Host Addresses	Broadcast Address
172.16.32.0	172.16.32.1 – 172.16.32.62	172.16.32.63
172.16.32.64	172.16.32.65 – 172.16.32.126	172.16.32.127
172.16.32.128	172.16.32.129 – 172.16.32.190	172.16.32.191
172.16.32.192	172.16.32.193 – 172.16.32.254	172.16.32.255
172.16.33.0	172.16.33.1 – 172.16.33.62	172.16.33.63
. . . . .	. . . . .	. . . . .
172.16.47.192	172.16.47.193 – 172.16.47.254	172.16.47.255

117

The following figure shows some of the new VLSM subnet addresses applied to the regional network.



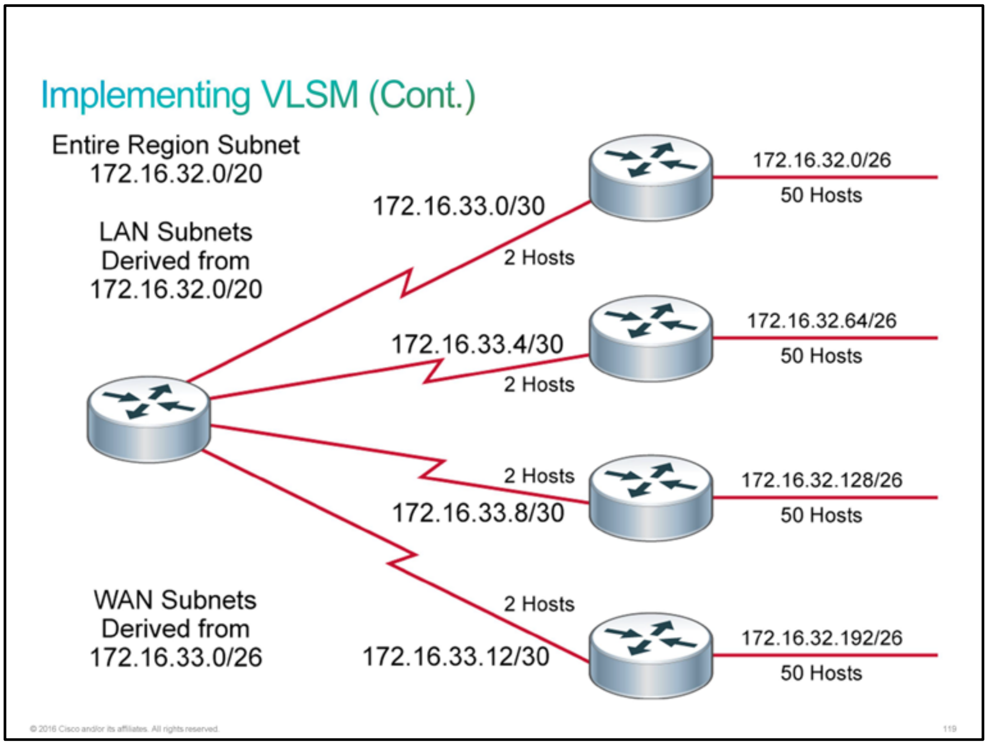
Implementing VLSM (Cont.)

Entire Region Subnet
172.16.32.0/20

LAN Subnets
Derived from
172.16.32.0/20

118

To calculate the subnet addresses for the WAN links, further subnet one of the unused /26 subnets with a 30-bit subnet mask. For this example, subnet 172.16.33.0 will be further subnetted. The 30-bit subnet mask provides 16 (2^4) subnets with 2 ($2^2 - 2$) host addresses each.



Implementing VLSM (Cont.)

As seen in this example, the easiest way to assign the subnets is to assign the largest first.

Challenge

1. What do subnetworks provide?

   A. scalability
   B. reachability
   C. redundancy
   D. load balancing

2. Which two aspects can present a problem with a single broadcast domain? (Choose two.)

   A. Larger amounts of broadcast traffic consume resources.
   B. All devices share the same broadcast domain.
   C. The domain relies on IP addresses for packet delivery.
   D. Larger amounts of multicast traffic consume resources.
   E. All PCs share the same broadcast domain.

3. Which two statements about a network that uses subnetworks are true? (Choose two.)

   A. It is more complex to apply network security policies.
   B. Smaller networks are easier to manage.
   C. Overall traffic is increased.
   D. Smaller networks are harder to manage.
   E. Overall traffic is reduced.

4. What is the decimal equivalent of the binary number 11000000?

   A. 224
   B. 240
   C. 128
   D. 192

5. You have subnetted your 192.168.36.0 network address with a 255.255.255.240 mask. How many usable subnets and hosts per subnet are available?

   A. 2 usable subnets and 126 hosts per subnet
   B. 4 usable subnets and 62 hosts per subnet
   C. 8 usable subnets and 30 hosts per subnet
   D. 16 usable subnets and 14 hosts per subnet
   E. 32 usable subnets and 6 hosts per subnet

6. How many valid host addresses are available for each subnet after subnetting network 192.168.0.0 by borrowing 2 host bits?

   A. 62
   B. 4094
   C. 8190
   D. 16382

7. What does VLSM stand for?

   A. Virtual LAN Subnet Mask
   B. Variable LAN Subnet Mask
   C. Virtual LAN Same Mask
   D. Variable Length Subnet Mask

Answer Key

Challenge

1. A
2. A, B
3. B, E
4. D
5. D
6. A
7. D

Lesson 3: Understanding the TCP/IP Transport Layer

Introduction

Bob notifies you via email that to join CCS, you will also need to demonstrate your understanding of TCP/IP transport layer functionality. Bob wants to validate that you know the differences between TCP and UDP, and common applications that use TCP and UDP as a transport.

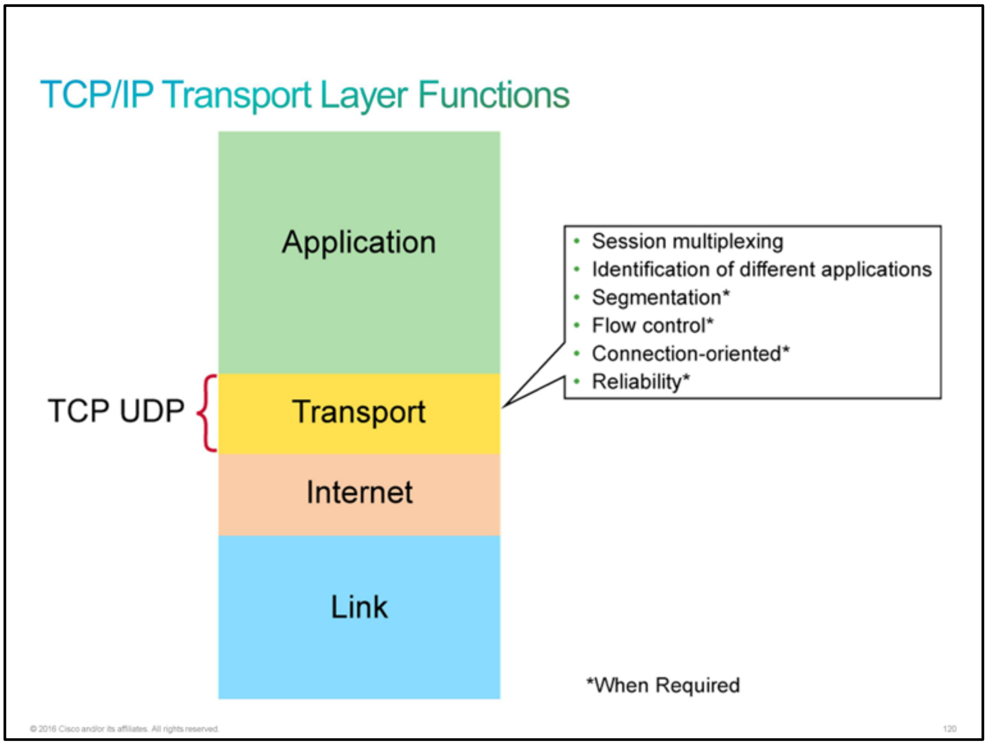
You will first start with the transport layer functions, and contrast reliable and unreliable transport. Compare TCP and UDP side by side. You will need to explain to Bob basic characteristics of UDP and describe its header. You will also discuss with him the TCP three-way handshake.

TCP/IP Transport Layer Functions

Residing between the application and Internet layers of the TCP/IP protocol stack, the transport layer is fundamental to the operation of the TCP/IP layered network architecture. The TCP/IP Internet layer directs information to its destination, but it cannot guarantee that the information will arrive in the correct order, free of errors, or even that it will arrive at all. The two most common transport layer protocols of the TCP/IP protocol suite are TCP and UDP. Both protocols manage the communication of multiple applications and provide communication services directly to the application process on the host.

The basic service that the transport layer provides is tracking individual communication between applications on the source and destination hosts. This service is called session multiplexing, and it is performed by both UDP and TCP. A major difference between TCP and UDP is that TCP can ensure that the data is delivered, while UDP does not.

Note Review of OSI and TCP/IP reference models: The transport layer of the TCP/IP protocol stack maps to the transport layer of the OSI model. The protocols that operate at this layer are said to operate at Layer 4 of the OSI model. If you hear someone use the term "Layer 4," they are referring to the transport layer of the OSI model.



TCP/IP Transport Layer Functions

TCP UDP

Application

Transport

Internet

Link

- Session multiplexing
- Identification of different applications
- Segmentation*
- Flow control*
- Connection-oriented*
- Reliability*

*When Required

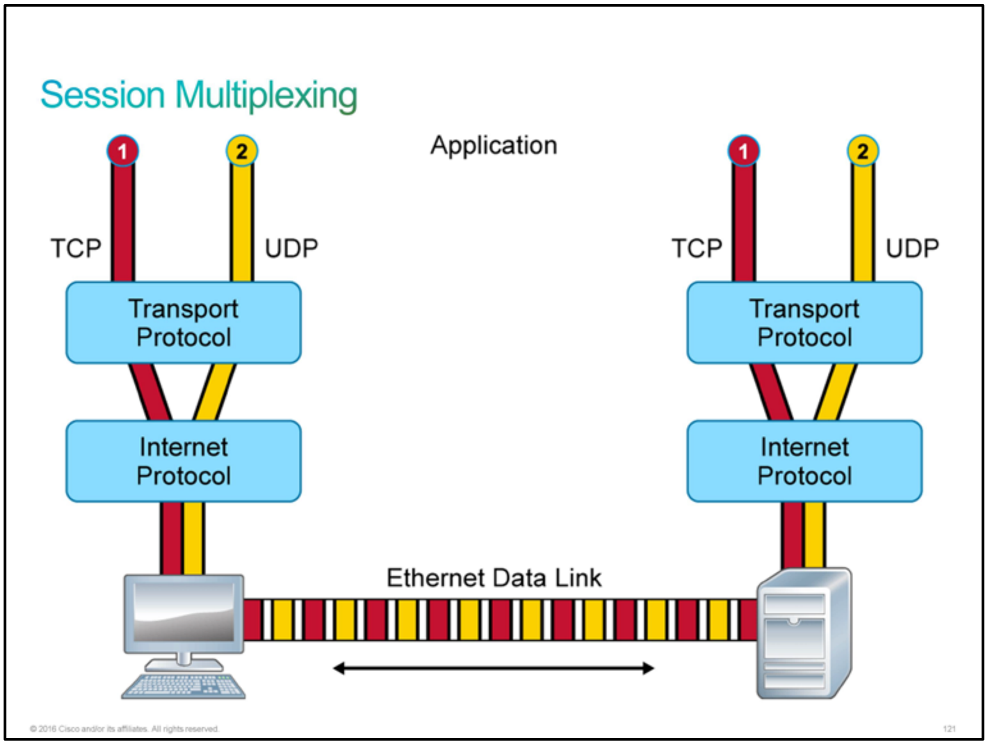
© 2016 Cisco and/or its affiliates. All rights reserved.

120

Multiple communications often occur at once; for instance, you may be searching the web and using FTP to transfer a file at the same time. The transport tracks these communications and keeps them separate. This tracking is provided by both UDP and TCP. To pass data to the proper applications, the transport layer must identify the target application. If TCP is used, the transport layer has the additional responsibilities of establishing end-to-end operations, segmenting data and managing each piece, reassembling the segments into streams of application data, managing flow control, and applying reliability mechanisms.

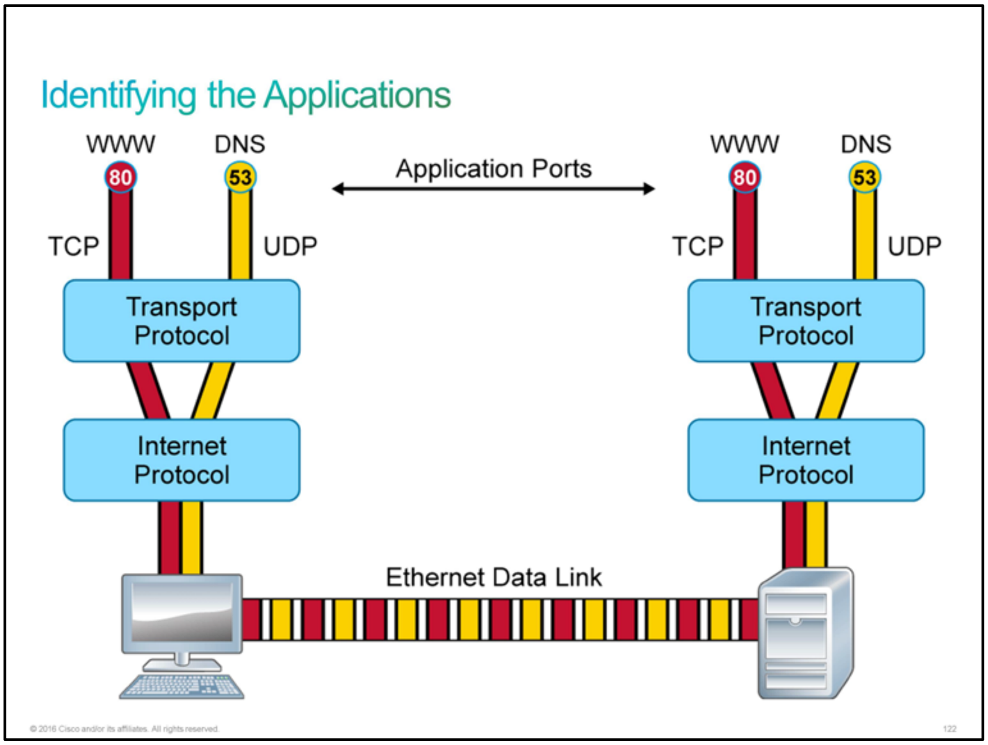
Session Multiplexing

Session multiplexing is the process by which an IP host is able to support multiple sessions simultaneously and manage the individual traffic streams over a single link. A session is created when a source machine needs to send data to a destination machine. Most often, this process involves a reply, but a reply is not mandatory.



Identifying the Applications

To pass data to the proper applications, the transport layer must identify the target application. TCP/IP transport protocols use port numbers to accomplish this task. Each application process that needs to access the network is assigned a port number (often called an application identifier) that is unique in that host. The port number is used in the transport layer header to indicate which application that piece of data is associated with.



Identifying the Applications

122

Segmentation

TCP takes arbitrarily sized data chunks from the application layers and prepares them for transport onto the network. The application relies on TCP to ensure that each chunk is broken up into smaller segments that will fit the MTU of the underlying network layers. UDP does not provide segmentation services. UDP instead expects the application process to perform any necessary segmentation and supply it with data chunks that do not exceed the MTU of lower layers.

Note	The MTU of the IP protocol is 1500 bytes. Larger MTUs are possible, but 1500 bytes is the normal size.

Flow Control

If a sender transmits packets faster than the receiver can receive them, the receiver drops some of the packets and requires them to be retransmitted. TCP is responsible for detecting dropped packets and sending replacements. A high rate of retransmissions introduces latency in the communication channel. To reduce the impact of retransmission-related latency, flow control methods work to maximize the transfer rate and minimize the required retransmissions.

Basic TCP flow control relies on acknowledgments that are generated by the receiver. For every data chunk that is sent, the sender waits for this acknowledgment from the receiver before sending the next part. However, if the RTT is significant, the overall transmission rate may slow to an unacceptable level. To increase network efficiency, a mechanism called *windowing* is combined with basic flow control. Windowing allows a receiving computer to advertise how much data it is able to receive before transmitting an acknowledgment to the sending computer.

Windowing allows avoidance of congestion in the network.

Connection-Oriented Transport Protocol

Within the transport layer, a connection-oriented protocol establishes a session connection between two IP hosts and then maintains the connection during the entire transmission. When the transmission is complete, the session is terminated. The TCP protocol provides connection-oriented reliable transport for application data.

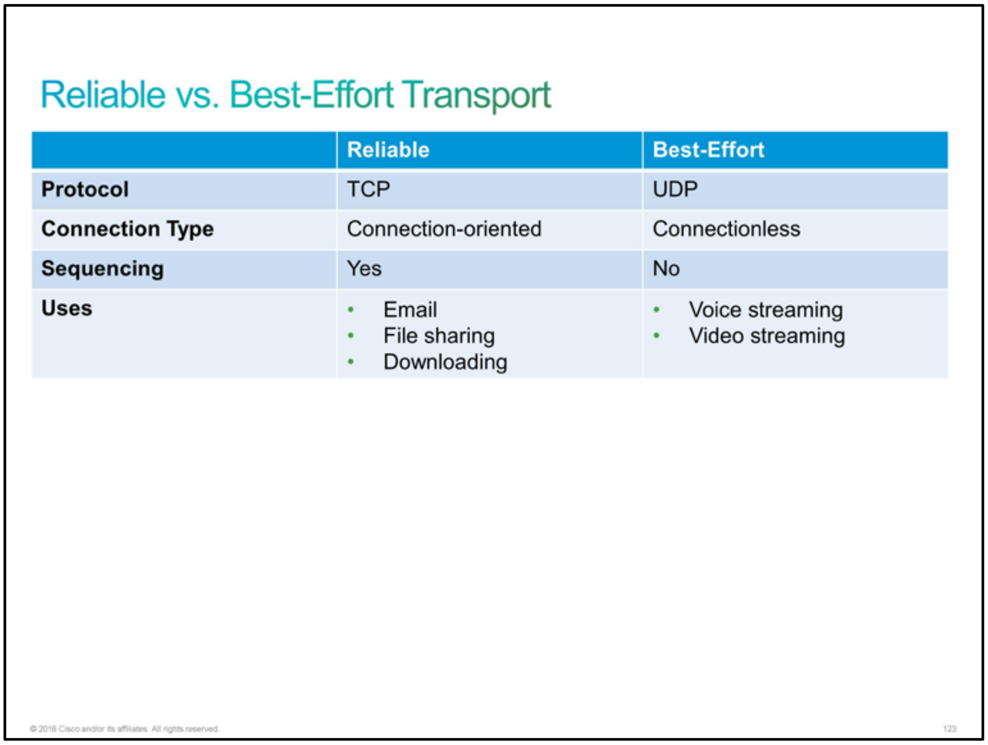
Reliability

TCP reliability has these three main objectives:

- Detection and retransmission of dropped packets
- Detection and remediation of duplicate or out-of-order data
- Avoidance of congestion in the network

Reliable vs. Best-Effort Transport

The terms *reliable* and *best effort* are terms that describe two types of connections between computers. TCP is a connection-oriented protocol that is designed to ensure reliable transport, flow control, and guaranteed delivery of IP packets. For this reason, it is labeled a "reliable" protocol. UDP is a connectionless protocol that relies on the application layer for sequencing and detection of dropped packets and is considered "best effort." Each protocol has strengths that make them useful for particular applications.



Reliable vs. Best-Effort Transport

	Reliable	Best-Effort
Protocol	TCP	UDP
Connection Type	Connection-oriented	Connectionless
Sequencing	Yes	No
Uses	• Email • File sharing • Downloading	• Voice streaming • Video streaming

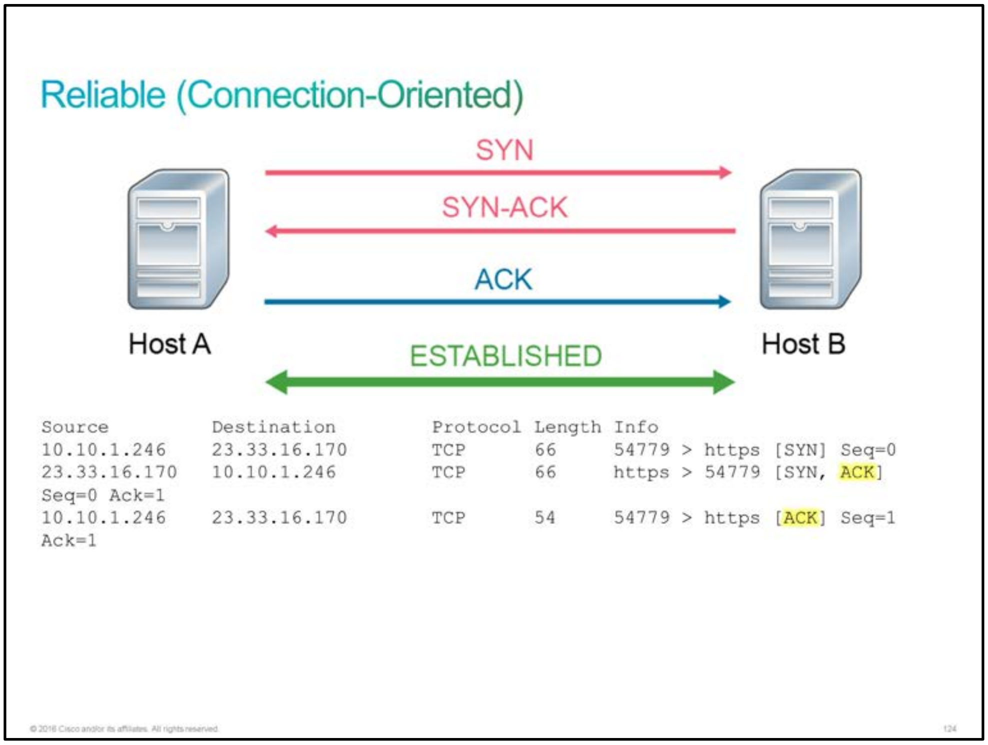
Reliable (Connection-Oriented)

Some types of applications require a guarantee that packets arrive safely and in order. Any missing packets could cause the data stream to be corrupted. Consider the example of using your web browser to download an application. Every piece of that application must be assembled on the receiver in the proper binary order, or it will not execute. File transfer is an application where the use of a connection-oriented protocol like TCP is indicated.

Note TCP employs a three-way handshake that is initiated by the IP host that is making a connection to an application. For more information on TCP session setup, please use your favorite search engine to locate a copy of RFC 793 (Transmission Control Protocol).

TCP uses a three-way handshake when setting up a connection. You can think of it as being similar to a phone call. The phone rings, the called party says "hello," and the caller says "hello." Here are the actual steps:

1. The source of the connection sends a SYN packet to the destination requesting a session. The Sequence Number (or SN) in this case is zero.
2. The destination responds to the SYN with a SYN-ACK and increments the initiator SN by 1.
3. If the source accepts the SYN-ACK, it sends an ACK packet to complete the handshake.



Reliable (Connection-Oriented)

```
Source          Destination          Protocol Length Info
10.10.1.246     23.33.16.170         TCP      66     54779 > https [SYN] Seq=0
23.33.16.170    10.10.1.246          TCP      66     https > 54779 [SYN, ACK]
Seq=0 Ack=1
10.10.1.246     23.33.16.170         TCP      54     54779 > https [ACK] Seq=1
Ack=1
```

124

Here you can see some common applications that use TCP:

- Web browsers
- Email
- FTP
- Network printing
- Database transactions

To support reliability, a connection is established between the IP source and destination to ensure that the application is ready to receive data. During the initial process of connection establishment, information is exchanged about the capabilities of the receiver, and starting parameters are negotiated. These parameters are then used for tracking data transfer during the connection.

When the sending computer transmits data, it assigns a sequence number to each packet. The receiver then responds with an acknowledgment number that is equal to the next expected sequence number. This exchange of sequence and acknowledgment numbers allows the protocol to recognize when data has been lost, or duplicated, or has arrived out of order.

Best Effort (Connectionless)

Reliability (guaranteed delivery) is not always necessary (or even desirable). For example, if one or two segments of a video stream fail to arrive, it would only create a momentary disruption in the stream. This disruption might appear as a momentary distortion of the image, but the user may not notice. In real-time applications, such as voice and video streaming, dropped packets can be tolerated as long as the overall percentage of dropped packets is low.

Here you can see some common applications that use UDP:

- DNS
- Streaming video
- VoIP
- TFTP

UDP provides applications with best-effort delivery and does not need to maintain state information about previously sent data. As a benefit, UDP does not need to establish any connection with the receiver and is termed connectionless. There are many situations in which best-effort delivery is more desirable than reliable delivery. A connectionless protocol is desirable for applications that require faster communication without verification of receipt.

TCP vs. UDP Analogy

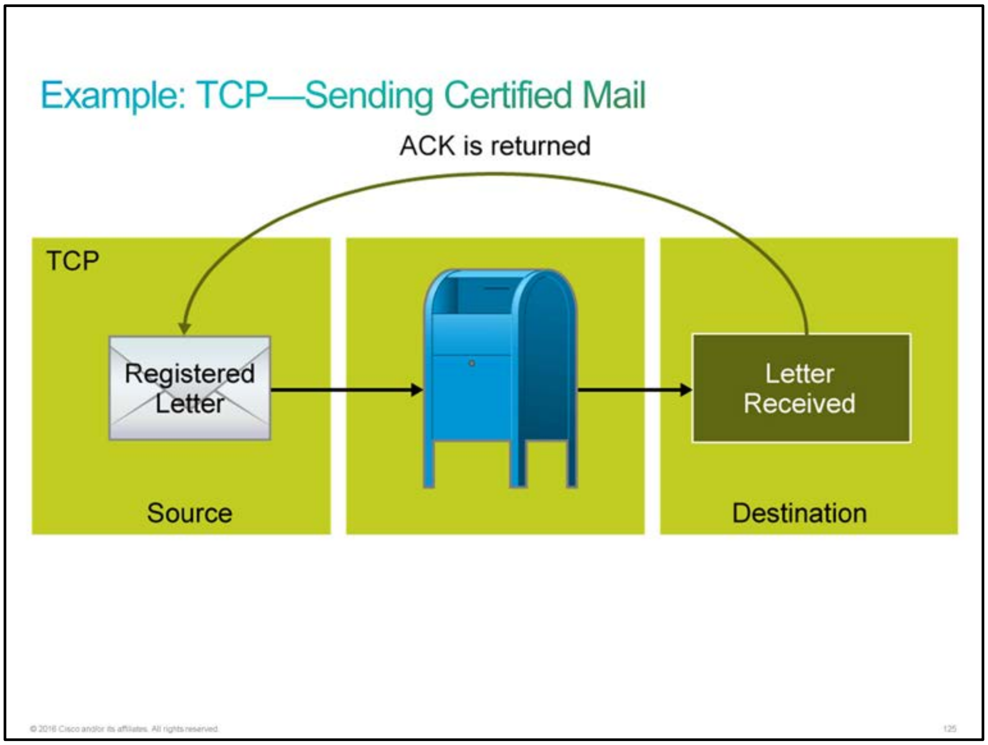
The postal service has been used as an analogy to illustrate the differences between connection-oriented TCP and connectionless services that UDP provides.

Example: TCP—Sending Certified Mail

Imagine that you are a popular author in Seattle. Your editor in Indianapolis is very anxious to publish your next novel and demands that you mail her each page as you finish one. You print each page of the book as you write them and put each page in a separate envelope. To ensure that your editor reassembles the book correctly, you put a page number on each envelope (a sequence number). You address the envelope and send the first one as certified mail. The postal service delivers it by any truck and any route, but because it is certified, the carrier who delivers it must get a signature from your editor and return a certificate of delivery to you.

Your contract with the publisher specifies that each page must be in a separate envelope. But having to go to the post office to send each letter individually is too time-consuming, so you send several envelopes together. The postal service again delivers each envelope by any truck and any route. Your editor signs a separate receipt for each envelope in the batch as she receives them. If one envelope is lost in transit, you will not receive a certificate of delivery for that numbered envelope, and you will need to resend that page. As your editor is receiving your envelopes, she uses the sequence numbers to assemble the book in the proper order.

Like certified mail, TCP offers sequencing, acknowledgements, and retransmission.



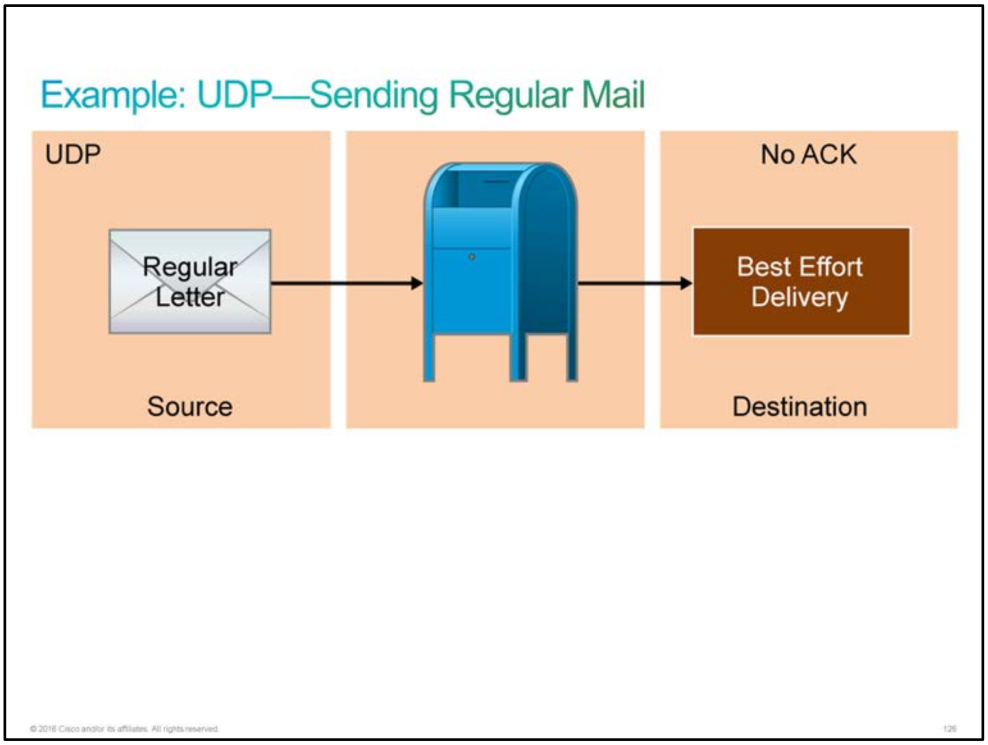
Example: TCP—Sending Certified Mail

125

Example: UDP—Sending Regular Mail

UDP services can be compared to using the postal service to pay your bills. You address each bill payment to a specific company address, stamp the envelope, and include your return address. The postal service guarantees its best effort to deliver each payment. The postal service does not guarantee delivery, and it is not responsible for telling you that delivery was successful or unsuccessful.

Like standard mail, UDP is a simple process that provides basic data-transfer services.



Example: UDP—Sending Regular Mail

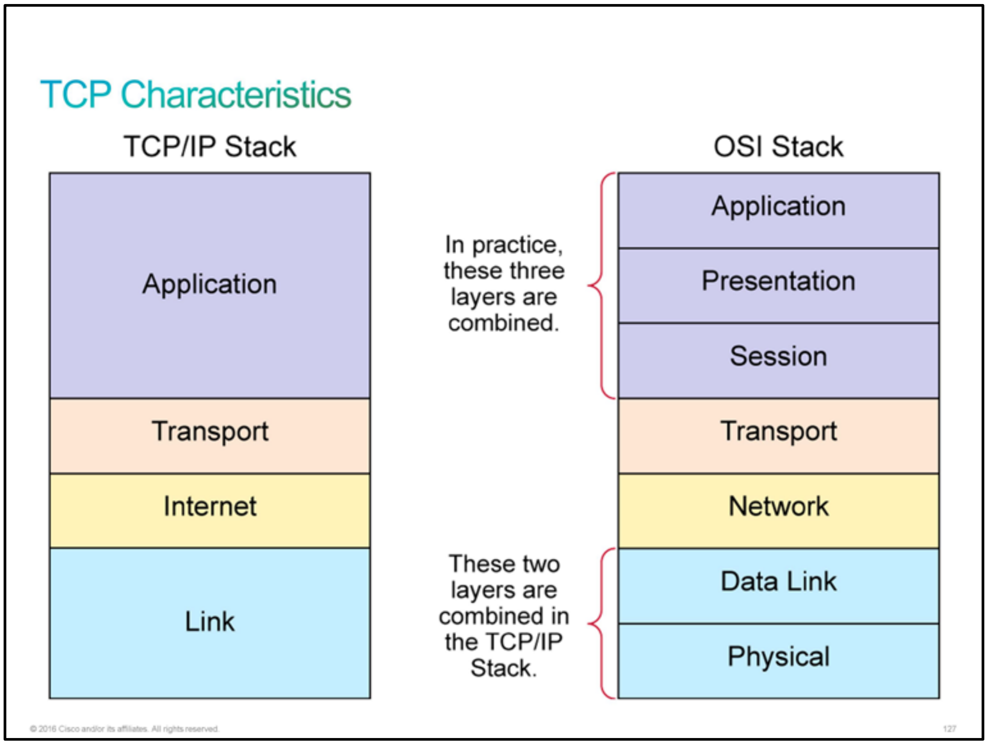
126

TCP Characteristics

TCP is a core protocol in the TCP/IP protocol suite. Applications leverage the connection-oriented services of TCP to provide data reliability between hosts. TCP includes several important features that provide for reliable data transmission.

TCP can be characterized as follows:

- TCP operates at the transport layer (OSI Layer 4) of the TCP/IP stack.
- TCP provides application access to the network layer (OSI Layer 3), where application data is routed from the source IP host to the destination IP host.



TCP Characteristics

TCP/IP Stack

Application

Transport

Internet

Link

OSI Stack

Application

Presentation

Session

Transport

Network

Data Link

Physical

In practice, these three layers are combined.

These two layers are combined in the TCP/IP Stack.

© 2016 Cisco and/or its affiliates. All rights reserved. 127

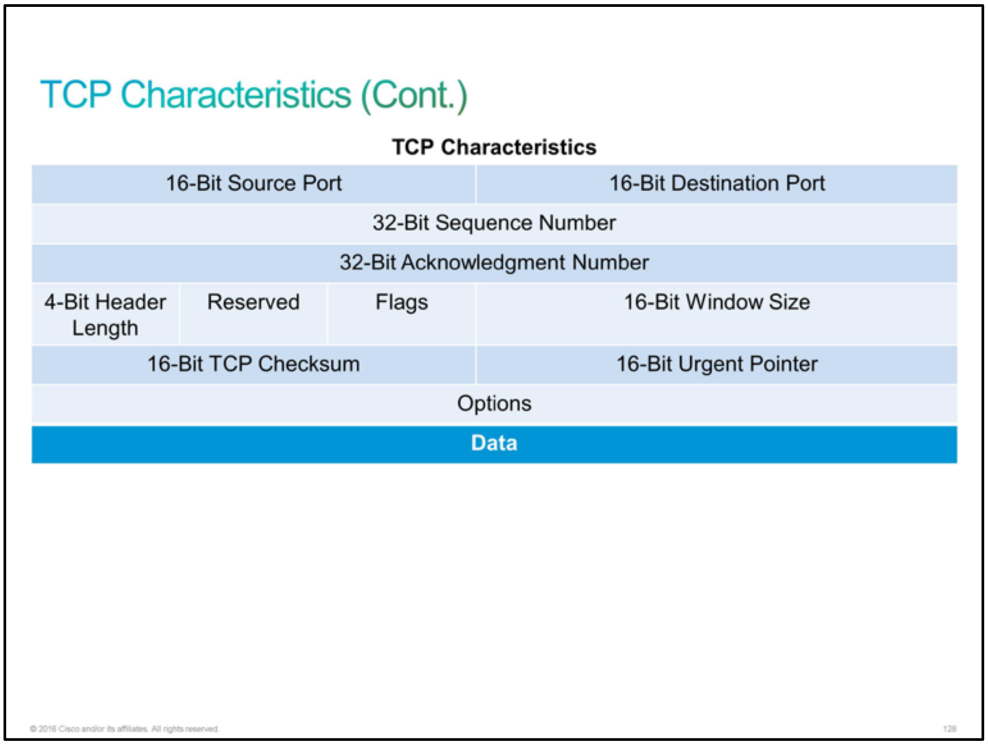
- TCP is connection-oriented and requires that network devices set up a connection to exchange data. The end systems synchronize with one another to manage packet flows and adapt to congestion in the network.
- TCP provides error checking by including a checksum in the IP datagram to verify that the TCP header information is not corrupt.
- A TCP connection is a pair of virtual circuits, one in each direction, so it operates in the full-duplex mode.
- TCP segments are numbered and sequenced so that the destination can reorder segments and determine if data is missing.
- Upon receipt of one or more TCP segments, the receiver returns an acknowledgment to the sender to indicate that it received the segment. Acknowledgments form the basis of reliability within the TCP session. When the source receives acknowledgment, it knows that the data has been successfully delivered. If the source does not receive acknowledgment within a predetermined period, it retransmits that data to the destination. The source may also terminate the connection if it determines that the receiver is no longer on the connection.

- TCP provides recovery services in which the receiver can request retransmission of a segment.
- TCP provides mechanisms for flow control. Flow control assists the reliability of TCP transmission by adjusting the effective rate of data flow between the two services in the session.

Reliable data delivery services are critical for applications such as file transfers, database services, transaction processing, and other mission-critical applications in which delivery of every packet must be guaranteed. TCP segments are sent by using IP packets. The TCP header follows the IP header and supplies information that is specific to the TCP protocol. Flow control, reliability, and other TCP characteristics are achieved by using fields in the TCP header. Each field has a specific function.

The fields of the TCP header include the following:

- **Source Port:** Number of the calling port (16 bits)
- **Destination Port:** Number of the called port (16 bits)
- **Sequence Number** and **Acknowledgment Number:** Used for reliability and congestion avoidance (32 bits each)
- **Header Length:** Size of the TCP header (4 bits)
- **Reserved:** For future use
- **Flags** or control bits (9 bits)
- **Window size:** Number of the window size (16 bits)
- **Checksum:** Calculated checksum of the header and fields that are used for error checking (16 bits)
- **Urgent Pointer:** If the URG flag is set, this field is an offset from the sequence number indicating the last urgent data byte (16 bits)
- **Options:** The length of this field is determined by the data offset field (from 0 to 320 bits)
- **Data:** ULP data (varies in size)



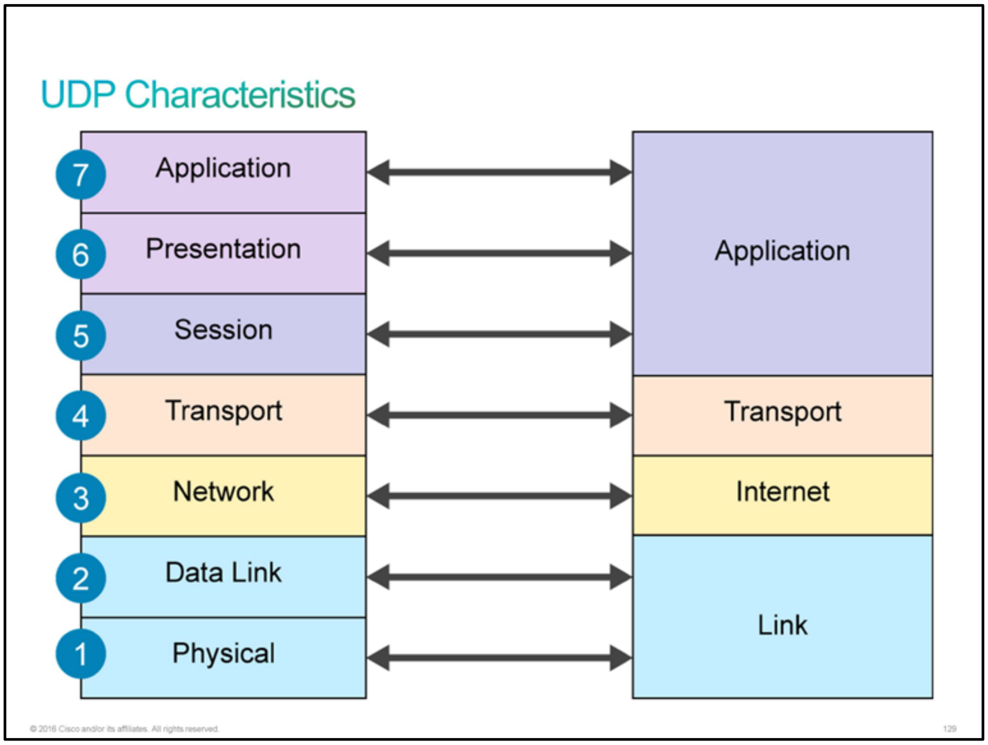
TCP Characteristics (Cont.)

TCP Characteristics

16-Bit Source Port			16-Bit Destination Port
32-Bit Sequence Number			
32-Bit Acknowledgment Number			
4-Bit Header Length	Reserved	Flags	16-Bit Window Size
16-Bit TCP Checksum			16-Bit Urgent Pointer
Options			
Data			

UDP Characteristics

UDP is a core protocol in the TCP/IP protocol suite. Applications leverage the connectionless services of UDP to provide high-performance, low-overhead data communications between hosts. UDP includes several features that provide for low-latency data transmission.



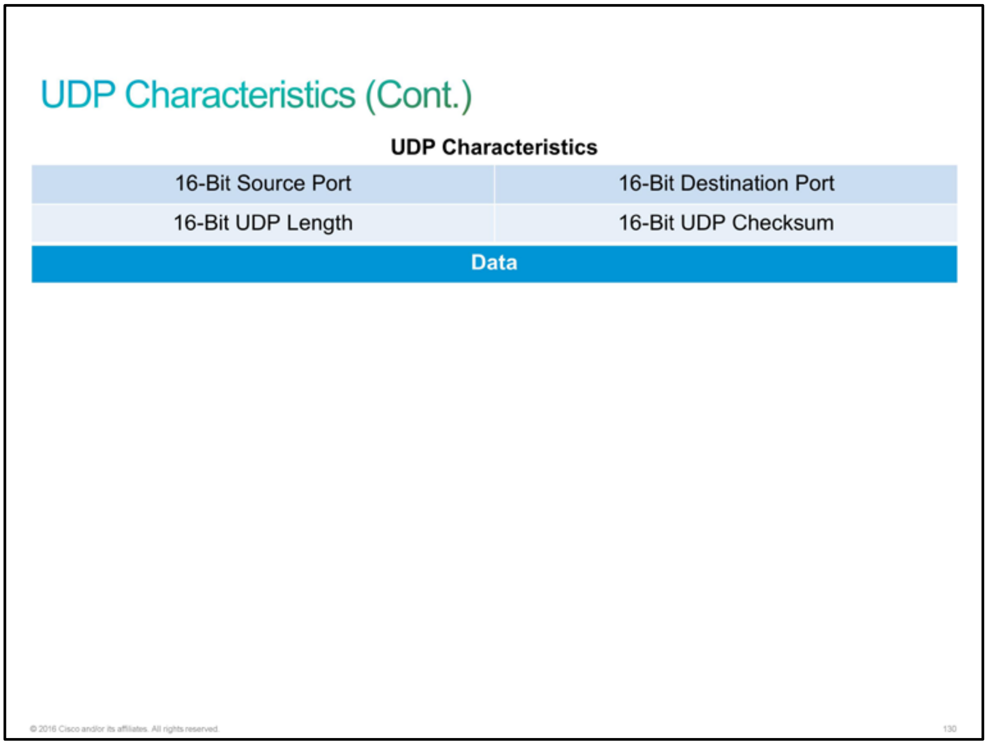
UDP is a simple protocol that provides basic transport layer functions:

- UDP operates at the transport layer of the TCP/IP stack (OSI Layer 4).
- UDP provides applications with access to the network layer without the overhead of reliability mechanisms.
- UDP is a connectionless protocol in which a one-way datagram is sent to a destination without advance notification to the destination device.
- UDP performs only limited error checking. A UDP datagram includes an optional checksum value, which the receiving device can use to test the integrity of the data.
- UDP provides service on a best-effort basis and does not guarantee data delivery, because packets can be misdirected, duplicated, or lost on the way to their destination.
- UDP does not provide any special features that recover lost or corrupted packets. UDP relies on applications that are using its transport services to provide recovery.
- Because of its low overhead, UDP is ideal for applications like DNS and NTP, where there is a simple request-and-response transaction.

An easy way to think of UDP is to use a postal service analogy. You are going to host a three-family garage sale next weekend, and you would like to send postcards that notify neighbors about the event, including the day, time, and location. You address each postcard with the name and address of your neighbors within a 6.2-mile (10-km) radius. The postal service delivers each postcard by any truck and any route. You have the option of paying additional postage for a delivery confirmation, but you decide that this additional expense is unnecessary because it is not important if a postcard is lost in transit, or if a neighbor acknowledges receipt of the message.

The low overhead of UDP is evident when you review the UDP header length of only 64 bits (8 bytes). So, the UDP header length is significantly smaller compared with the TCP minimum header length of 20 bytes. The following list describes the field definitions in the UDP segment:

- **Source Port:** Number of the calling port (16 bits)
- **Destination Port:** Number of the called port (16 bits)
- **Length:** Length of UDP header and UDP data (16 bits)
- **Checksum:** Calculated checksum of the header and data fields (16 bits)
- **Data:** ULP data (varies in size)



UDP Characteristics (Cont.)

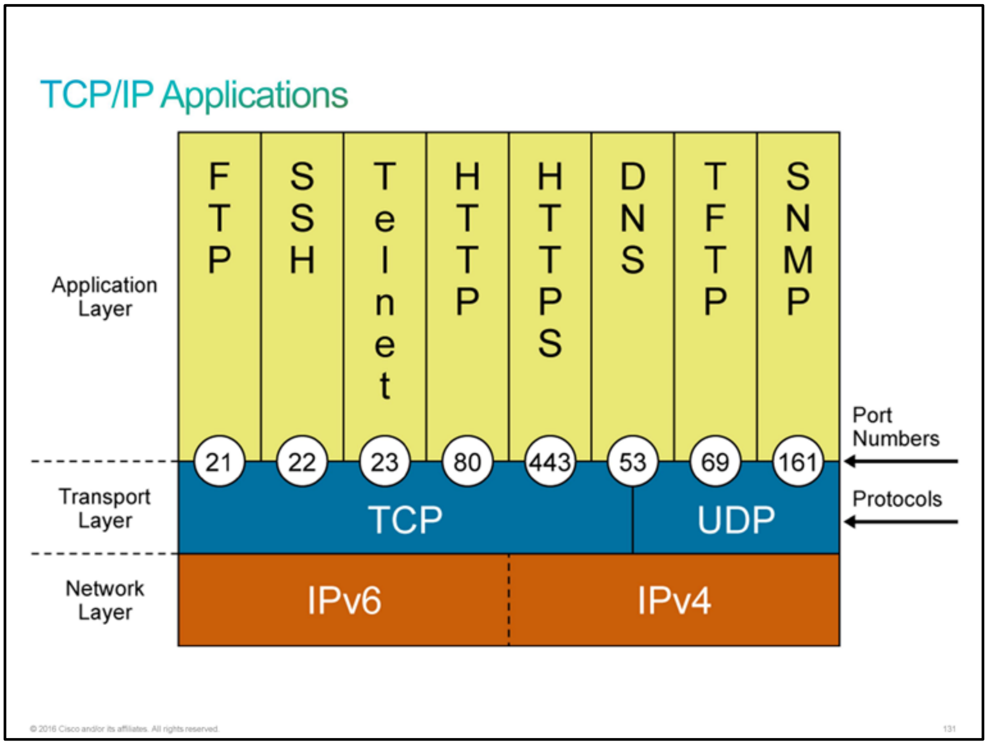
UDP Characteristics

16-Bit Source Port	16-Bit Destination Port
16-Bit UDP Length	16-Bit UDP Checksum
Data	

Application layer protocols that use UDP include DNS, SNMP, DHCP, RIP, TFTP, NFS, online games, and streaming media.

TCP/IP Applications

UDP and TCP use internal software ports to support multiple conversations between various network devices. To differentiate the segments and datagrams for each application, TCP, and UDP both have header fields that uniquely identify these applications. These unique identifiers are the port numbers.



Some of the applications that TCP/IP supports:

- **FTP (port 21, TCP):** FTP is a reliable, connection-oriented service that uses TCP to transfer files between systems that support FTP. FTP supports bidirectional binary and ASCII file transfers. In addition to port 21, which is used for exchange of control, it also uses one additional port for data transmission.
- **SSH (port 22, TCP):** SSH provides the capability to remotely access other computers, servers, and networking devices. SSH enables a user to log in to a remote host and execute commands. SSH messages are encrypted.
- **Telnet (port 23, TCP):** Telnet is a predecessor to SSH. It sends messages in unencrypted cleartext. Most organizations now use SSH for remote communications.
- **HTTP (port 80):** HTTP defines how messages are formatted and transmitted and what actions browsers and web servers take in response to various commands. It uses TCP.
- **HTTPS (port 443, TCP):** HTTPS combines HTTP with a security protocol (SSL/TLS).
- **DNS (port 53, TCP, and UDP):** DNS is used to resolve Internet names to IP addresses. DNS uses a distributed set of servers to resolve names that are associated with numbered addresses.

- **TFTP (port 69, UDP):** TFTP is a connectionless service. Routers use TFTP to transfer configuration files and Cisco IOS images and other files between systems that support TFTP.
- **SNMP (port 161, UDP):** SNMP is an application layer protocol and it facilitates the exchange of management information between network devices. SNMP enables network administrators to manage network performance, find and solve network problems, and plan for network growth.

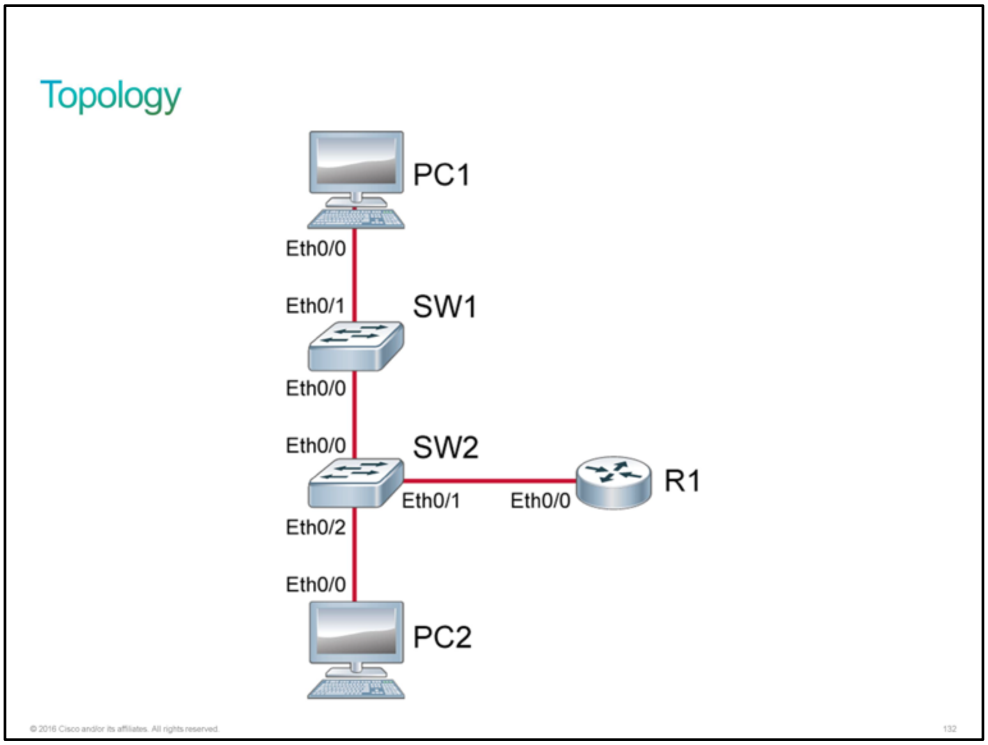
Here, you have seen only some applications with their port numbers. Go to the *Service Name and Transport Protocol Port Number Registry* for a complete list at http://www.iana.org/assignments/service-names-port-numbers/service-names-port-numbers.xhtml.

Discovery 5: Inspect TCP/IP Applications

Introduction

This discovery lab will help you explore TCP and UDP sockets. That is, how TCP and UDP servers listen on particular ports that are made available on particular interfaces, and how clients connect to the servers using their own IP addresses and their own ports. The lab is prepared with the devices represented in the topology diagram with the IP addresses as depicted in the table.

Topology



Job Aid

Device Information

Device Information Table

Device	Characteristic	Value
PC1	Hostname	PC1
PC1	IP address	10.10.1.10/24
PC1	Default gateway	10.10.1.1
PC2	Hostname	PC2

Device	Characteristic	Value
PC2	IP address	10.10.1.20/24
PC2	Default gateway	10.10.1.1
SW1	Hostname	SW1
SW1	VLAN 1 IP address	10.10.1.2/24
SW1	Default gateway	10.10.1.1
SW1	Ethernet0/0 description	Link to SW2
SW1	Ethernet0/1 description	Link to PC1
SW2	Hostname	SW2
SW2	VLAN 1 IP address	10.10.1.3/24
SW2	Default gateway	10.10.1.1
SW2	Ethernet0/0 description	Link to SW1
SW2	Ethernet0/1 description	Link to R1
SW2	Ethernet0/2 description	Link to PC2
R1	Hostname	R1
R1	Ethernet0/0 description	Link to SW2
R1	Ethernet0/0 IP address	10.10.1.1/24
R1	Loopback 0 IP	10.10.3.1/24

PCs in the virtual lab environment are simulated as routers, so you should use Cisco IOS commands to configure them or make verifications.

Task 1: Inspect TCP/IP Applications

Activity

Step 1 R1 has been configured to run several TCP services including Telnet, SSH, HTTP, and HTTPS. It has also been configured to run an NTP service. How and why these services may be configured on a router is beyond the scope of this discovery. For now, verify the services that are running on R1 by viewing its open ports. Access the console of R1 and execute the **show control-plane host open-ports** command.

There are several open TCP ports: 22 for SSH, 23 for Telnet, 80 for HTTP and 443 for HTTPS.

```
R1# show control-plane host open-ports
Active internet connections (servers and established)
Prot               Local Address             Foreign Address
Service    State
 tcp                        *:22                         *:0               SSH-
Server   LISTEN
 tcp                        *:23                         *:0
Telnet   LISTEN
 tcp                        *:80                         *:0
HTTP CORE   LISTEN
 tcp                        *:80                         *:0
HTTP CORE   LISTEN
 tcp                       *:443                         *:0
HTTP CORE   LISTEN
 tcp                       *:443                         *:0
HTTP CORE   LISTEN
 udp                       *:123                         *:0
NTP   LISTEN
```

These ports are in a listening state. That is, no foreign addresses are connected to them, but they are ready for connections to ensue.

Step 2 Access the console of PC1 and use Telnet to connect to R1. The password that is configured on R1 is "Cisco123."

The prompt changes from PC1 to R1 because you are now connected to R1 via Telnet from PC1.

```
PC1# telnet 10.10.1.1
Trying 10.10.1.1 ... Open

User Access Verification

Password: <Cisco123>
R1#
```

Step 3 Return to the console of R1 and review the open ports. You may want to use the Cisco IOS command recall feature to re-enter the command.

There is an additional line in the output when compared with the last execution. It shows a second line that is associated with TCP port 23. In this case, the foreign address is populated. The IP address is 10.10.1.10 (the IP address of PC1). The foreign port number might not be the same as what is shown in the example because it will be an ephemeral port.

```
R1# show control-plane host open-ports
Active internet connections (servers and established)
Prot               Local Address             Foreign Address
Service    State
 tcp                        *:22                         *:0               SSH-
Server   LISTEN
 tcp                        *:23                         *:0
Telnet   LISTEN
 tcp                        *:80                         *:0
HTTP CORE   LISTEN
 tcp                        *:80                         *:0
HTTP CORE   LISTEN
 tcp                       *:443                         *:0
HTTP CORE   LISTEN
 tcp                       *:443                         *:0
HTTP CORE   LISTEN
 tcp                        *:23            10.10.1.10:14044
Telnet ESTABLIS
 udp                       *:123                         *:0
NTP   LISTEN
```

Step 4 Return to the console of PC1 and use the **exit** command to disconnect the telnet session to R1.

The prompt returns to PC1 as you are no longer connected to the R1.

```
R1# exit

[Connection to 10.10.1.1 closed by foreign host]
PC1#
```

Alternatively, you could have used the **logout** command to disconnect from R1.

Step 5 Return to the console of R1 and review the open ports again.

All ports are in a listening state.

```
R1# show control-plane host open-ports
Active internet connections (servers and established)
Prot               Local Address             Foreign Address
Service    State
 tcp                        *:22                         *:0               SSH-
Server   LISTEN
 tcp                        *:23                         *:0
Telnet   LISTEN
 tcp                        *:80                         *:0
HTTP CORE   LISTEN
 tcp                        *:80                         *:0
HTTP CORE   LISTEN
 tcp                       *:443                         *:0
HTTP CORE   LISTEN
 tcp                       *:443                         *:0
HTTP CORE   LISTEN
 udp                       *:123                         *:0
NTP   LISTEN
```

This is the end of the discovery lab.

Challenge

1. What is a major difference between TCP and UDP?

   A. TCP can ensure that the data is delivered, while UDP does not.
   B. UDP can ensure that the data is delivered, while TCP does not.
   C. TCP exists in the transport layer of the TCP/IP model, while UDP exists in the Internet layer of the TCP/IP model.
   D. TCP exists in the Internet layer of the TCP/IP model, while UDP exists in the transport layer of the TCP/IP model.

2. What are the three main objectives of TCP reliability? (Choose three.)

   A. detection and retransmission of dropped packets
   B. detection and remediation of duplicate or out-of-order data
   C. avoidance of congestion in the network
   D. detection of dropped packets. The application does the retransmission.
   E. detection of duplicate or out-of-order data. The application does the remediation.
   F. avoidance of delay in the network

3. What is session multiplexing?

   A. a process by which an IP host is able to support multiple sessions simultaneously and manage the individual traffic streams over a single link.
   B. a process by which an IP host is able to support multiple sessions simultaneously over multiple links.
   C. a process that is used for congestion avoidance.
   D. a process that is used for packet dropping.

4. What are three common applications that use TCP? (Choose three.)

   A. DNS
   B. web browsers
   C. email
   D. streaming video
   E. FTP
   F. VoIP

5. What are three common applications that use UDP? (Choose three.)

   A. network printing
   B. TFTP
   C. database transactions
   D. streaming video
   E. FTP
   F. VoIP

6. Which OSI stack layers match the application layer of the TCP/IP stack?

   A. application layer, presentation layer, and session layer
   B. presentation layer, session layer, and transport layer
   C. application layer, presentation layer, session layer, and transport layer
   D. network layer and transport layer

7. Which OSI stack layers match the link layer of the TCP/IP stack?

   A. network layer, data link layer, and physical layer
   B. data link layer only
   C. data link layer and physical layer
   D. physical layer only

Answer Key

Challenge

1. A
2. A, B, C
3. A
4. B, C, E
5. B, D, F
6. A
7. C

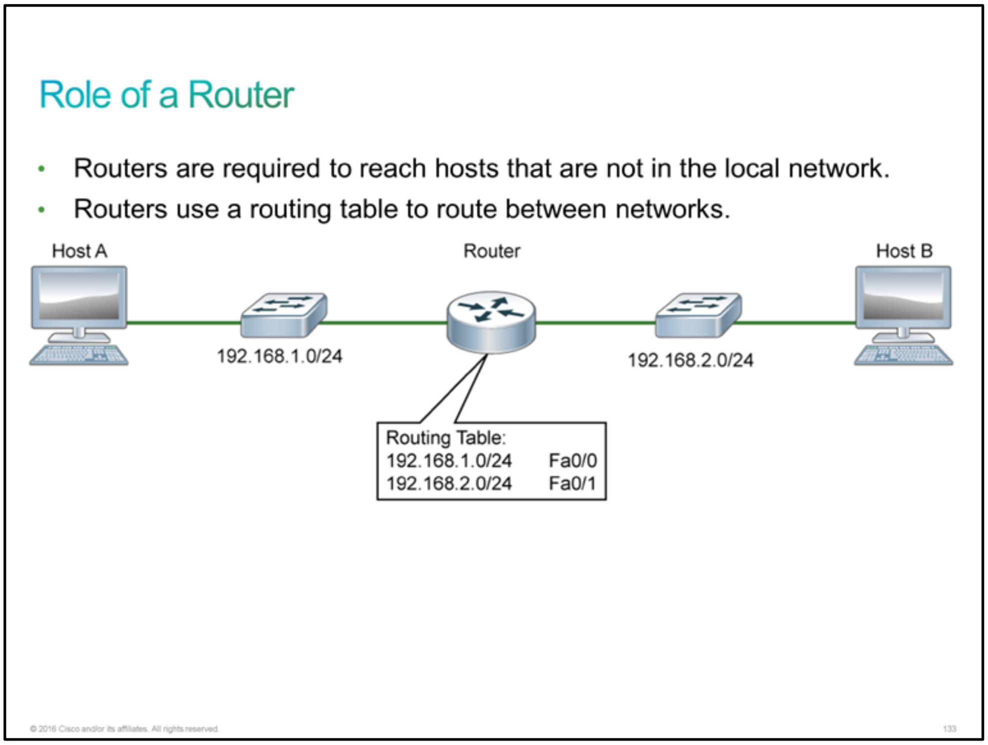
Lesson 4: Exploring the Functions of Routing

Introduction

Bob tells you that you will also need to understand routing functionality, including the types of routes and how dynamic routing protocols work. He tells you not to worry about the differences in link-state and distance vector routing protocols at this point and that you only need a high-level understanding of the functions of dynamic routing protocols.

Role of a Router

A router is a networking device that forwards packets between different networks or LANs.



Role of a Router

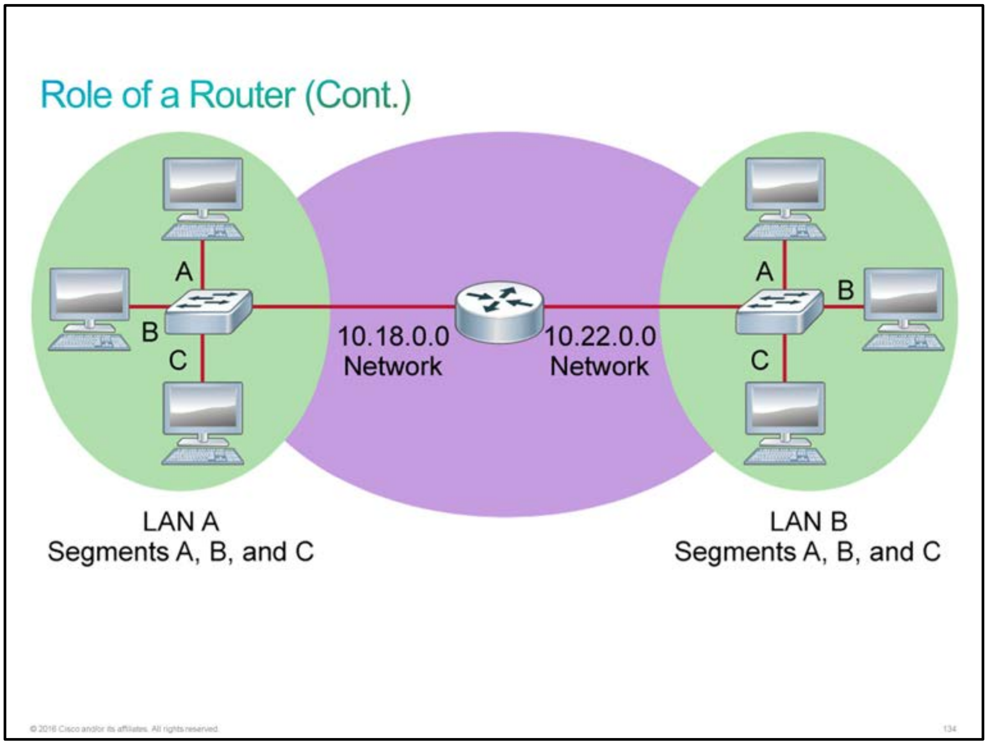
- Routers are required to reach hosts that are not in the local network.
- Routers use a routing table to route between networks.

Routing Table:	
192.168.1.0/24	Fa0/0
192.168.2.0/24	Fa0/1

While switches exchange data frames between segments to enable communication within a single network, routers are required to reach hosts that are not in the local LAN. Routers enable internetwork communication by placing the interface of each router in the network of the other routers. They use routing tables to route traffic between different networks.

In the following figure, LAN A switches data frames between its segments—A, B, and C—to enable communication among hosts on those segments. In other words, the LAN A switch enables communication within a single network, LAN A, whose network IP address is 10.18.0.0/16. Likewise, the LAN B switch enables communication among the hosts on LAN B, whose network IP address is 10.22.0.0/16.

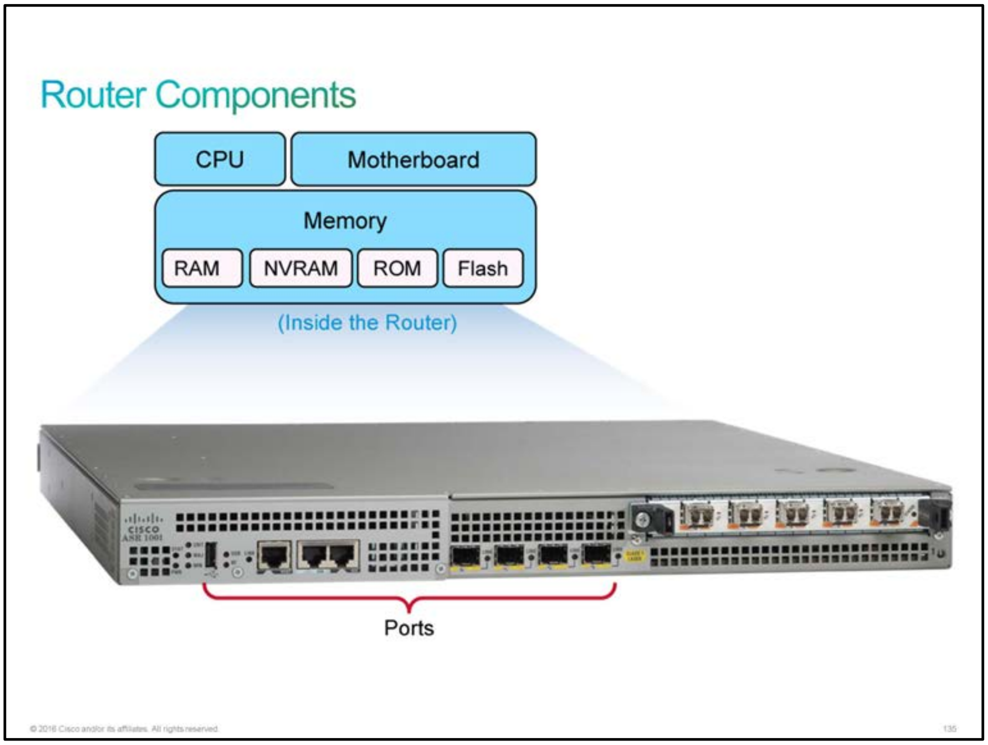
A host in LAN A cannot communicate with a host in LAN B without the router. Routers enable communications between hosts that are not in the same local LAN. Routers are able to do this function because they can be attached to multiple networks and have the ability to route between them. Routers are essential to large networks that use TCP/IP, because they can accommodate growth across wide geographical areas.



Role of a Router (Cont.)

Router Components

Cisco offers many different routers, which come in many shapes and sizes. The various models offer various features that are suitable for an array of different environments. However, the core function of a router is to route packets, and for that reason, all routers have many common components.



Router Components

CPU

Motherboard

Memory

RAM

NVRAM

ROM

Flash

(Inside the Router)

Ports

© 2016 Cisco and/or its affiliates. All rights reserved.

135

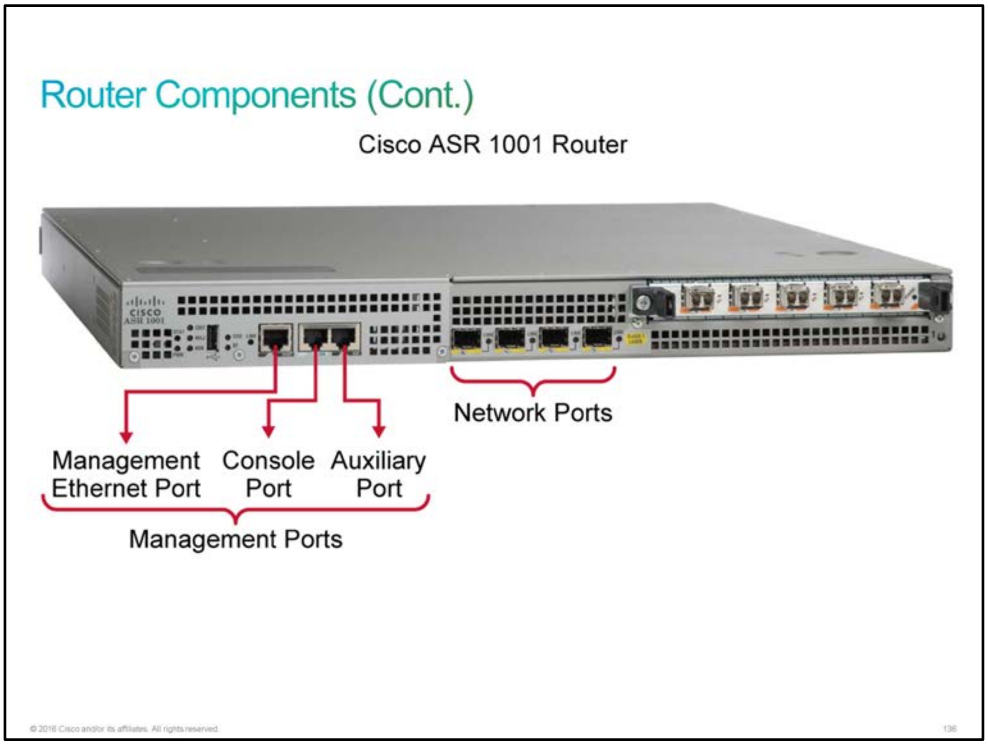
These components are as follows:

- **CPU:** CPU, or processor, is the chip that is installed on the motherboard that carries out the instructions of a computer program. For example, it processes all the information that is gathered from other routers or sent to other routers.
- **Motherboard:** The motherboard is the central circuit board, which holds critical electronic components of the system. The motherboard provides connections to other peripherals and interfaces.
- **Memory:** There are four primary types of memory:
  - **RAM:** RAM is memory on the motherboard that stores data during CPU processing. It is a volatile type of memory in that its information is lost when power is switched off. RAM provides temporary memory for the running configuration of the router while the router is powered on.
  - **NVRAM:** NVRAM retains content when the router is powered down. NVRAM stores the startup configuration file for most router platforms. It also contains the software configuration register, which is used to determine which image to use when booting the router.
  - **ROM:** ROM is read-only memory on the motherboard. The content of ROM is not lost when power is switched off. Data that is stored in ROM cannot be modified, or it can be modified only slowly or with difficulty. ROM sometimes contains a ROM monitor, which provides a user interface when the router cannot find a valid image, and bootloader software, which helps the router boot when it cannot find a valid Cisco IOS image in the flash memory.

  - **Flash:** Flash memory is nonvolatile storage that can be electrically erased and reprogrammed. Flash memory stores the Cisco IOS Software image. On some platforms, it can also store configuration files or boot images.

- **Ports:** Ports are used to connect routers to other devices in the network. Routers can have these types of ports:
  - **Management ports:** Management ports are for the connection of a terminal that is used for management. Routers have a console port that can be used to attach to a terminal that is used for management, configuration, and control. High-end routers may also have a dedicated Ethernet port that can be used only for management. An IP address can be assigned to the Ethernet port, and the router can be accessed from a management subnet. The AUX interface on a router is used for remote management of the router. Typically, a modem is connected to the AUX interface for dial-in access. From a security standpoint, enabling the option to connect remotely to a network device carries with it the responsibility of vigilant device management.
  - **Network ports:** The router has many network ports, including various LAN or WAN media ports, which may be copper or fiber cable. IP addresses are assigned to network ports.

As an example, the following figure shows the ports on a Cisco ASR 1001 Router:



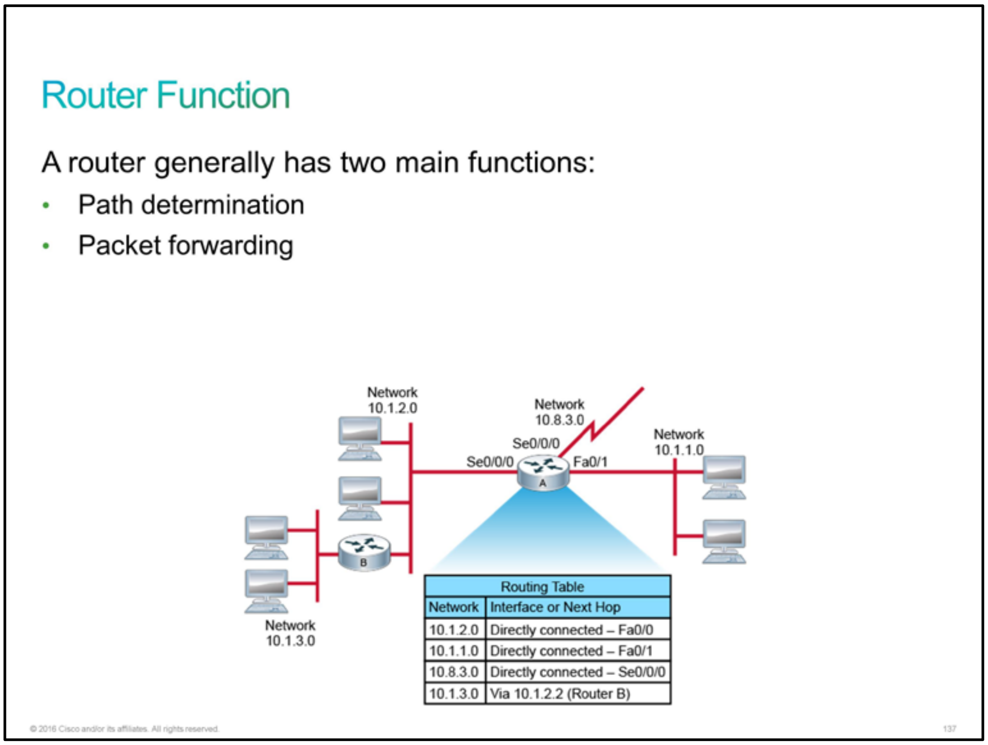
Router Components (Cont.)

Cisco ASR 1001 Router

© 2016 Cisco and/or its affiliates. All rights reserved.

136

Router Function



Router Function

A router generally has two main functions:

- Path determination
- Packet forwarding

Routing Table

Network	Interface or Next Hop
10.1.2.0	Directly connected – Fa0/0
10.1.1.0	Directly connected – Fa0/1
10.8.3.0	Directly connected – Se0/0/0
10.1.3.0	Via 10.1.2.2 (Router B)

Routers have these two important functions:

- **Path determination:** Routers use their routing tables to determine where to forward packets. Each router must maintain its own local routing table, which contains a list of all destinations that are known to the router, and information about how to reach those destinations. When a router receives an incoming packet, it checks the destination IP address in the packet and searches for the best match between the destination address and the network addresses in the routing table. A matching entry may indicate that the destination is directly connected to the router or that it can be reached via another router. This router is called the next-hop router and is on the path to the final destination. If there is no matching entry, the router sends the packet to the default route. If there is no default route, the router drops the packet.
- **Packet forwarding:** After a router determines the appropriate path for a packet, it forwards the packet through a network interface toward the destination network. As shown in the figure, each line of the routing table lists a destination network and its corresponding interface or next-hop address. If there is an interface on the router that has an IP address within the destination network, the destination network is considered "directly connected" to the router. For example, assume that router A receives a packet on its Serial0/0/0 interface that is destined for a host on network 10.1.1.0. Because the routing table indicates that network 10.1.1.0 is directly connected, router A forwards the packet directly to the host via its FastEthernet0/1 interface. If a destination network in the routing table is not directly connected, the packet must reach the destination network via the next-hop router. For example, assume that Router A receives a packet on its Serial0/0/0 interface and the destination host address is on the 10.1.3.0 network. In this case, it must forward the packet to the router B interface with the IP address 10.1.2.2. Routers support three packet-forwarding mechanisms:
  - **Process switching:** Process switching is the oldest forwarding mechanism that is available in Cisco routers. Every packet requires a full lookup in the routing table, which makes this mechanism very slow. It is typically not used in modern networks.

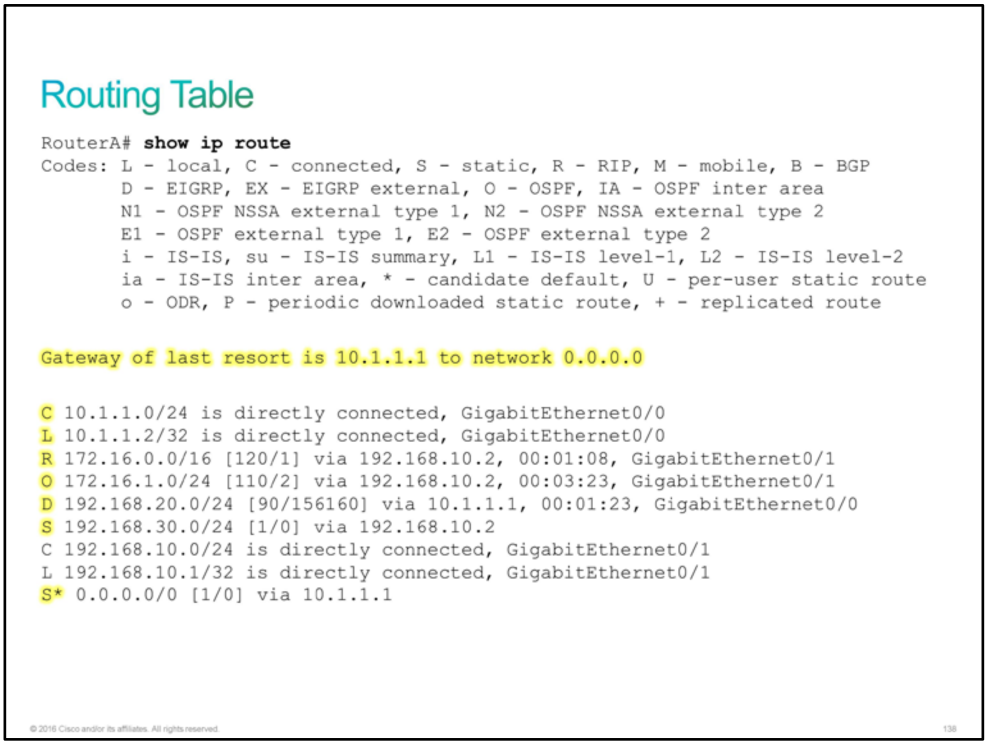
- **Fast switching:** To overcome the slow performance of process switching, Cisco IOS platforms support several switching mechanisms that use a cache to store the most recently used destinations. The first packet whose destination is not found in the fast-switching cache is process-switched, and an entry is created in the cache. Subsequent packets are then able to use fast switching.
- **Cisco Express Forwarding:** Cisco Express Forwarding is the most recent and preferred Cisco IOS packet-forwarding mechanism, which incorporates the best of the previous switching mechanisms. Changes in the network instead of packets trigger the generation of cache table entries. When something changes in the network topology, the change is also reflected in the cache table. All packets are switched using the Cisco Express Forwarding cache, which makes Cisco Express Forwarding the fastest forwarding mechanism and the preferred choice.

Routing Table

A routing table contains a list of all networks that are known to the router and information about how to reach those networks. Each line, or entry, of the routing table lists a destination network and the interface or next-hop address by which that destination network can be reached.

A routing table may contain the following types of entries:

- **Directly connected networks:** All directly connected networks are added to the routing table automatically. A directly connected network is a network that is directly connected to one of the interfaces on the local router. If the interface fails or is administratively shut down, the entry for that network is removed from the routing table.
- **Static routes:** Static routes are entries that you manually enter directly into the configuration of the router. Static routes can be effective for small, simple networks that do not change frequently. However, statically populating routing tables does not scale well and can lead to problems if the network topology changes.
- **Default routes:** A default route is an optional entry that is used when no explicit path to a destination is found in the routing table. You can manually configure the default route as a static route, or a routing protocol can enter it.
- **Dynamic routes:** The router learns dynamic routes automatically when a routing protocol is configured and a neighbor relationship to other routers is established. The information is updated when changes in the network occur. Larger networks require the dynamic routing method because there are usually many addresses and constant changes. These changes require updates to routing tables across all routers in the network, to prevent connectivity loss.



Routing Table

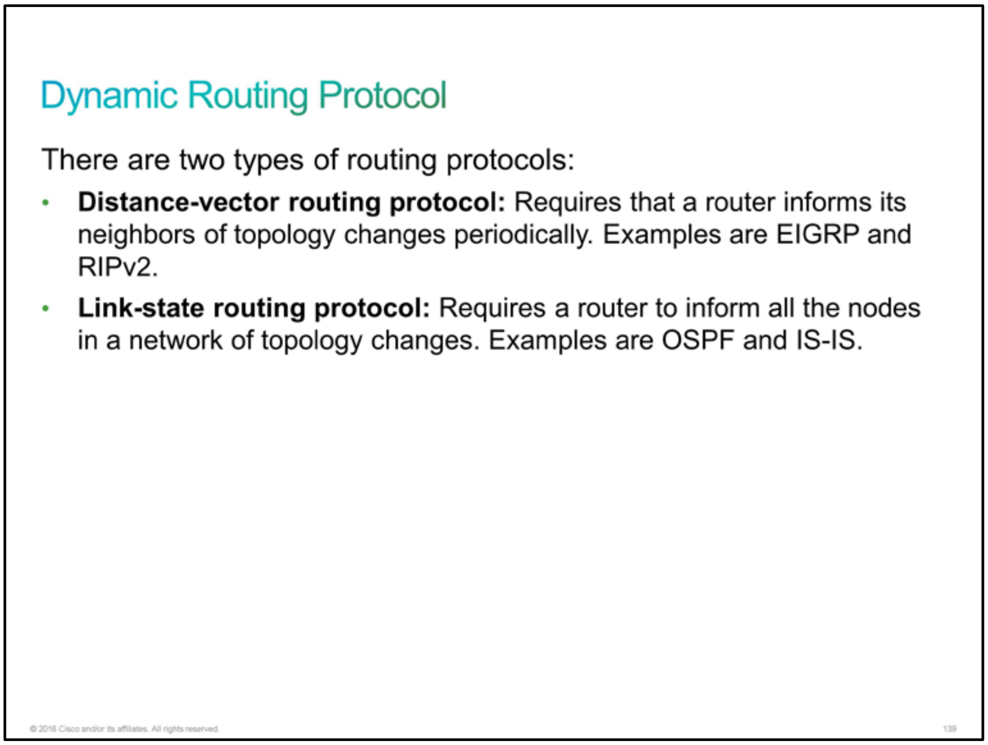
```
RouterA# show ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, + - replicated route

Gateway of last resort is 10.1.1.1 to network 0.0.0.0

C 10.1.1.0/24 is directly connected, GigabitEthernet0/0
L 10.1.1.2/32 is directly connected, GigabitEthernet0/0
R 172.16.0.0/16 [120/1] via 192.168.10.2, 00:01:08, GigabitEthernet0/1
O 172.16.1.0/24 [110/2] via 192.168.10.2, 00:03:23, GigabitEthernet0/1
D 192.168.20.0/24 [90/156160] via 10.1.1.1, 00:01:23, GigabitEthernet0/0
S 192.168.30.0/24 [1/0] via 192.168.10.2
C 192.168.10.0/24 is directly connected, GigabitEthernet0/1
L 192.168.10.1/32 is directly connected, GigabitEthernet0/1
S* 0.0.0.0/0 [1/0] via 10.1.1.1
```

The figure shows the output of the **show ip route** command, which is used to display the contents of the routing table in a router. The first part of the output explains the codes, presenting the letters and the associated sources of the entries in the routing table.

- **The letter C:** Reserved for directly connected networks; labels the first and sixth entries.
- **The letter L:** Reserved for local routes and indicating local interfaces within connected networks; labels the second and seventh entries.
- **The letter S:** Reserved for static routes; labels the fifth and eighth entries.
- **The asterisk (*):** Indicates a default route. In this example command output, the default route is a static route.
- **The letter R:** Reserved for the RIP routing protocol; labels the third entry.
- **The letter O:** Reserved for the OSPF routing protocol; labels the fourth entry.
- **The letter D:** Reserved for EIGRP; labels the fifth entry. The letter D stands for DUAL, which is the update algorithm that EIGRP uses.

Dynamic Routing Protocol

A routing protocol is a set of processes, algorithms, and messages by which routers dynamically share their routing information. Examples of routing protocols include OSPF, EIGRP, RIPv2, and IS-IS.



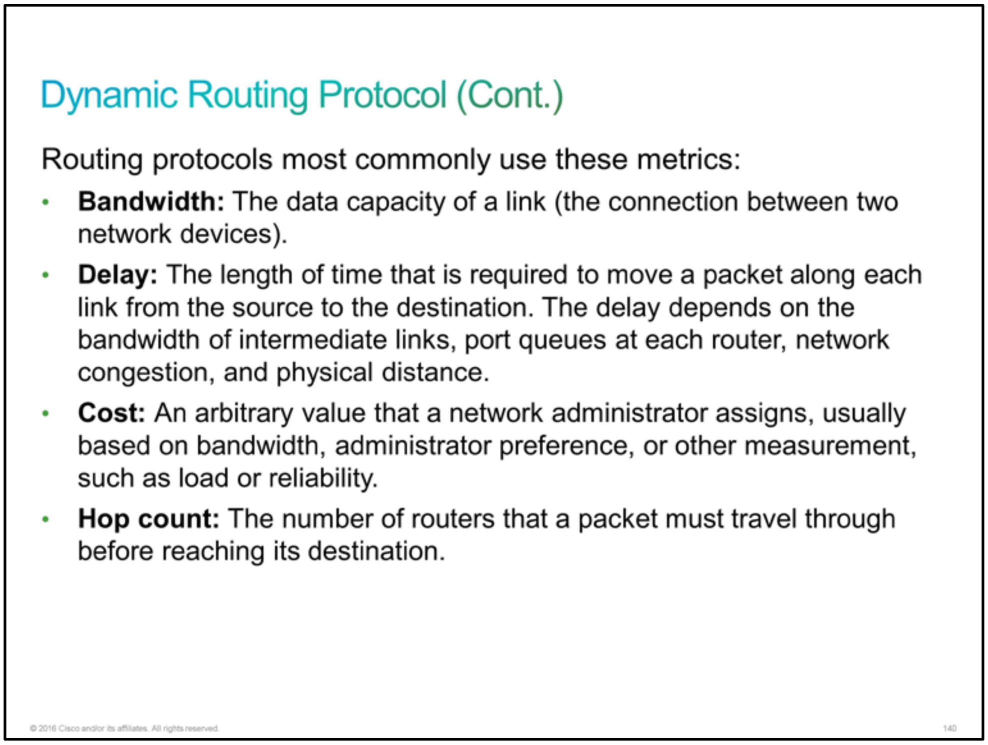
Dynamic Routing Protocol

There are two types of routing protocols:

- **Distance-vector routing protocol:** Requires that a router informs its neighbors of topology changes periodically. Examples are EIGRP and RIPv2.
- **Link-state routing protocol:** Requires a router to inform all the nodes in a network of topology changes. Examples are OSPF and IS-IS.

Routers that are running routing protocols exchange routing update messages to keep their routing tables updated. When a router that is running a routing protocol becomes aware of changes to the network, it passes the information on to other routers that are running the same routing protocol. When a router receives information about new or changed routes, it updates its own routing table and, in turn, passes the information to other routers. In this way, all routers maintain accurate routing tables that are updated dynamically and can learn about routes to remote networks that are many hops away.

Routing protocols not only enable routers to learn about remote networks and to quickly adapt whenever there is a change in the topology; they also enable routers to choose the best path to destination networks. Although there may be multiple paths to a given destination, a routing table holds only one path to any given destination. A routing table holds only one entry for every network.



Dynamic Routing Protocol (Cont.)

Routing protocols most commonly use these metrics:

- **Bandwidth:** The data capacity of a link (the connection between two network devices).
- **Delay:** The length of time that is required to move a packet along each link from the source to the destination. The delay depends on the bandwidth of intermediate links, port queues at each router, network congestion, and physical distance.
- **Cost:** An arbitrary value that a network administrator assigns, usually based on bandwidth, administrator preference, or other measurement, such as load or reliability.
- **Hop count:** The number of routers that a packet must travel through before reaching its destination.

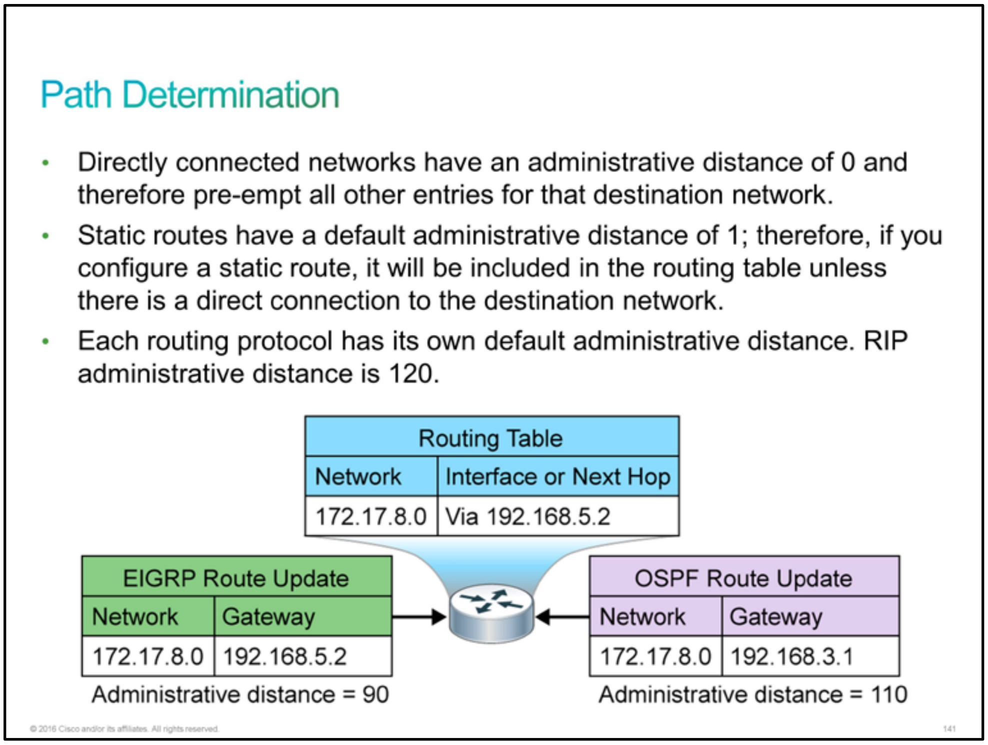
140

To determine the best path to any given destination, routing protocols use a number that is called a metric. The metrics that the various routing protocols use differ, depending on the design of the routing algorithm that is used. The routing algorithm that the protocol uses generates a metric for each path through the network. Metrics can be based on either a single characteristic or on several characteristics of a path. Sophisticated routing protocols can base route selection on multiple metrics, combining them into a single metric. Typically, the lower the metric value, the better the path.

Path Determination

Routing tables can be populated from three types of sources: directly connected networks, static routes, and routing protocols. The router must be able to evaluate the routing information from all the sources and select the best route to populate the routing table.

Routers use a feature called administrative distance to select the best path when they learn two or more different routes to the same destination. Administrative distance defines the reliability of the route source—the smaller the administrative distance value, the more trusted the source. Each source type has a default administrative distance. For example:



Path Determination

- Directly connected networks have an administrative distance of 0 and therefore pre-empt all other entries for that destination network.
- Static routes have a default administrative distance of 1; therefore, if you configure a static route, it will be included in the routing table unless there is a direct connection to the destination network.
- Each routing protocol has its own default administrative distance. RIP administrative distance is 120.

Routing Table	
Network	Interface or Next Hop
172.17.8.0	Via 192.168.5.2

EIGRP Route Update	
Network	Gateway
172.17.8.0	192.168.5.2

Administrative distance = 90

OSPF Route Update	
Network	Gateway
172.17.8.0	192.168.3.1

Administrative distance = 110

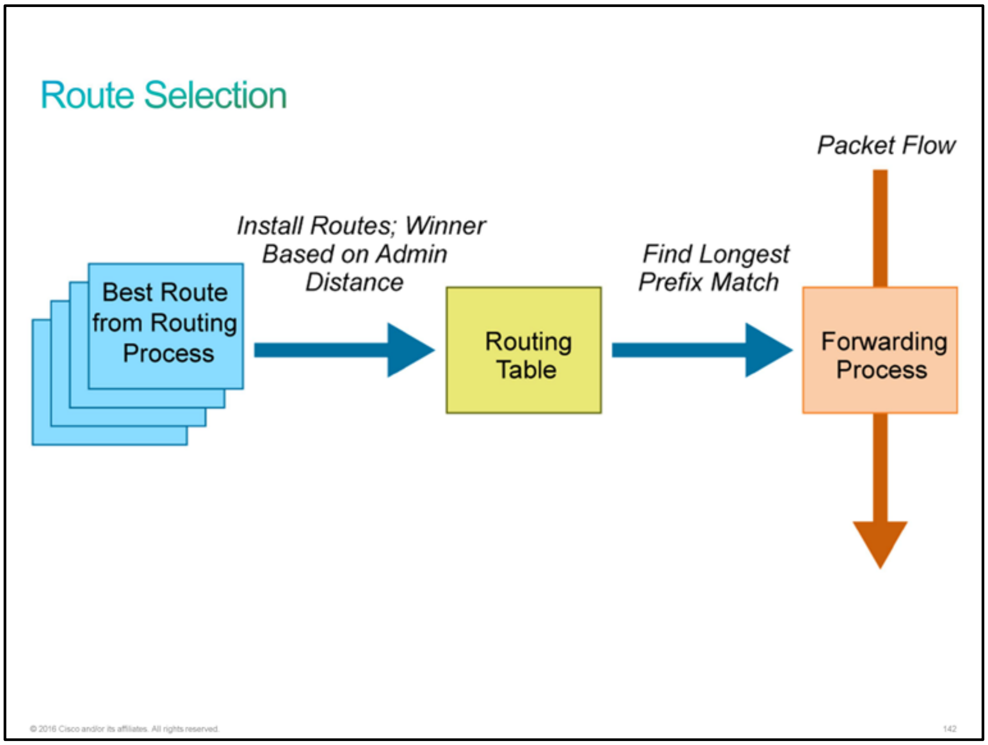
In the figure, the router has received two routing update messages—one from OSPF and one from EIGRP. The metric that EIGRP uses has determined that the best path to network 172.17.8.0 is via 192.168.5.2, but the metric that OSPF uses has determined that the best path to 172.17.8.0 is via 192.168.3.1. The router has used the administrative distance feature to determine which route to install in its routing table. Because the administrative distance for OSPF is 110 and the administrative distance for EIGRP is 90, the router has chosen the EIGRP route and adds only the EIGRP route to its routing table.

Because each entry in a routing table may specify a subnetwork, one destination address may match more than one routing table entry. The most specific of the matching table entries is called the longest prefix match. It is called this way because it is also the entry where the largest number of leading address bits of the destination address matches those addresses in the table entry.

For example, consider a routing table with these entries: 10.1.1.0/24 and 10.1.0.0/16. When the address 10.1.1.1 needs to be looked up, both entries match. In this case, the longest prefix of the candidate routes is 10.1.1.0/24.

Route Selection

Making a forwarding decision actually consists of three sets of processes: the routing protocols, the routing table, and the actual process that makes the forwarding decision and switches packets.



Route Selection

Packet Flow

Install Routes; Winner Based on Admin Distance

Find Longest Prefix Match

Best Route from Routing Process

Routing Table

Forwarding Process

© 2016 Cisco and/or its affiliates. All rights reserved.

142

Three processes are involved in building and maintaining the routing table in a Cisco router:

- Various routing processes, which actually run a routing protocol, such as RIPv2, EIGRP, IS-IS, and OSPF. The best route from the routing process has a potential to be installed into the routing table.
- The routing table itself, which accepts information from the routing processes and also replies to requests for information from the forwarding process.
- The forwarding process, which requests information from the routing table to make a packet forwarding decision.

The longest prefix match always wins among the routes that are actually installed in the routing table, whereas the routing protocol with the lowest administrative distance always wins when installing routes into the routing table.

Challenge

1. Which statement best describes the role of a router?

   A. Routers are responsible only for reaching hosts that are in the local network.
   B. Routers use a MAC table to route between networks.
   C. Routers are required to reach hosts that are not in the local network.
   D. Routers use an ARP table to route between networks.

2. What do you call the router component that holds critical electronic components of the system?

   A. CPU
   B. motherboard
   C. memory
   D. ports

3. In which part of the router memory is the running configuration stored?

   A. RAM
   B. NVRAM
   C. ROM
   D. flash

4. Which two types of ports are available on a router? (Choose two.)

   A. network ports
   B. console ports
   C. AUX ports
   D. debug ports
   E. management ports
   F. monitoring ports

5. In a routing table, what is an optional entry that is used when no explicit path to a destination is found?

   A. static route
   B. directly connected route
   C. OSPF route
   D. default route

6. What does the letter D that is associated with the routing table entry present?

   A. route is learned by OSPF
   B. route is learned by EIGRP
   C. route is static
   D. route is learned by DMVPN

7. What is the administrative distance for OSPF?

A. 110
B. 115
C. 90 for internal route and 170 for external route
D. 170 for internal route and 90 for external route

Answer Key

Challenge

1. C
2. B
3. A
4. A, E
5. D
6. B
7. A

Lesson 5: Configuring a Cisco Router

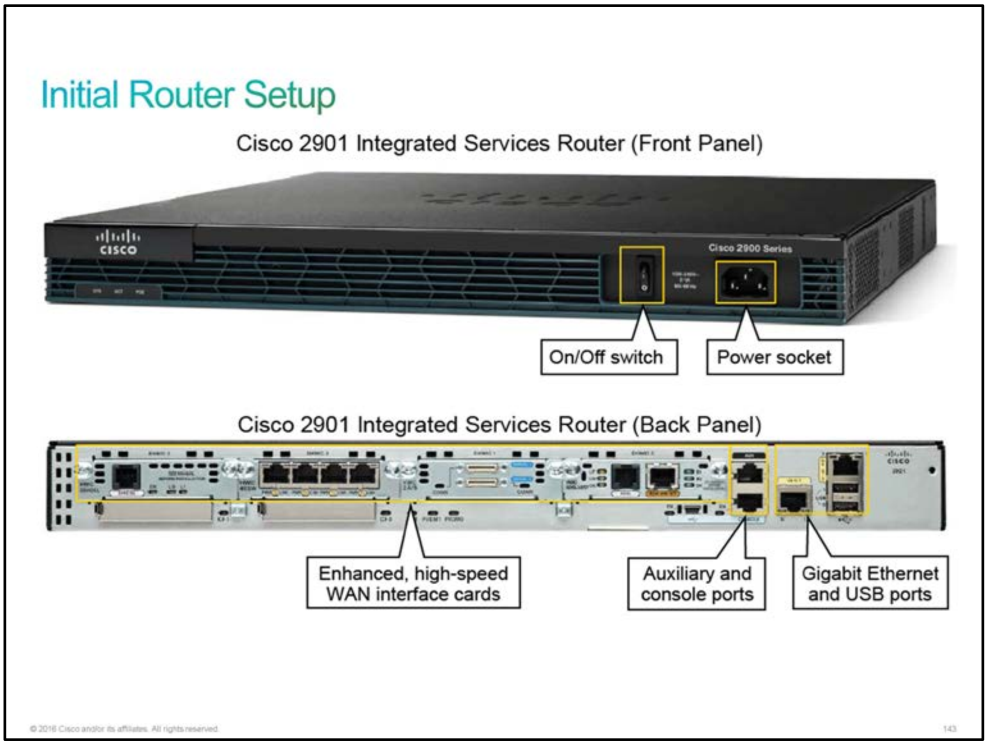
Introduction

Your boss sends you to a customer to install a new router. The router is already physically set, but you will need to configure it. You need to understand the initial configuration steps to properly configure the router. Also, you will configure and verify an interface on the router and use Cisco Discovery Protocol to draw the network topology.

Initial Router Setup

Cisco provides several different types of router hardware, including some routers that do only routing, while other routers offer additional functions. In fact, Cisco has a series of routers that is called ISR, with the name emphasizing the fact that many functions are integrated into a single device.

The following figure shows Cisco 2901 ISR with some of the more important features highlighted.



Initial Router Setup

Cisco 2901 Integrated Services Router (Front Panel)

Cisco 2900 Series
On/Off switch
Power socket

Cisco 2901 Integrated Services Router (Back Panel)

Enhanced, high-speed WAN interface cards
Auxiliary and console ports
Gigabit Ethernet and USB ports

© 2016 Cisco and/or its affiliates. All rights reserved.

143

The startup of a Cisco router requires verifying the physical installation, powering up the router, and viewing the Cisco IOS Software output on the console. To start router operations, the router completes the following tasks:

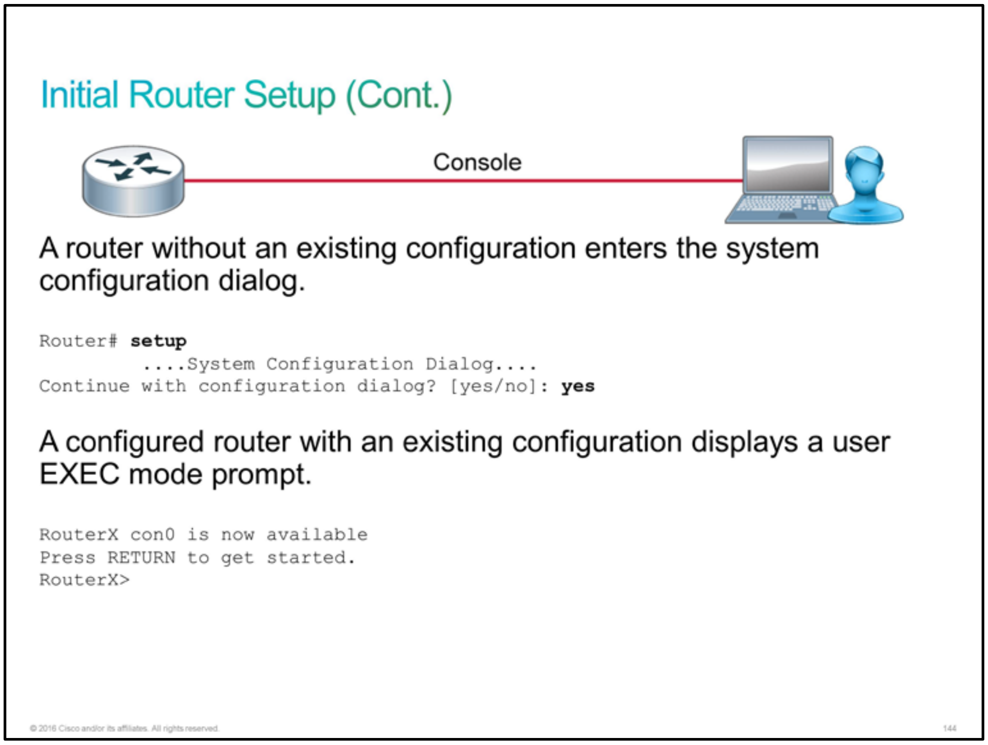
1. Runs the POST to test the hardware
2. Finds and loads the Cisco IOS Software that the router uses for its operating system
3. Finds and applies the configuration statements about router-specific attributes, protocol functions, and interface addresses

When a Cisco router powers on, it performs a POST. During the POST, the router executes diagnostics to verify the basic operation of the CPU, memory, and interface circuitry.

After verifying the hardware functions, the router proceeds with software initialization. During software initialization, the router finds and loads the Cisco IOS image. After the router loads the Cisco IOS image, it finds and loads the configuration file, if one exists.

Note	Before you start the router, verify the power and cooling requirements, cabling, and console connection. Then push the power switch to "On" and observe both the boot sequence and the Cisco IOS Software output on the console.

After a router completes the POST and loads a Cisco IOS image, it looks for a device configuration file in its NVRAM. If the router does not find one, it executes a question-driven, initial configuration routine that is called "setup." Setup is a prompt-driven program that allows a minimal device configuration. If the router has a startup configuration file in NVRAM, the user EXEC mode prompt appears.



Initial Router Setup (Cont.)

A router without an existing configuration enters the system configuration dialog.

```
Router# setup
        ....System Configuration Dialog....
Continue with configuration dialog? [yes/no]: yes
```

A configured router with an existing configuration displays a user EXEC mode prompt.

```
RouterX con0 is now available
Press RETURN to get started.
RouterX>
```

When starting a new Cisco router, there is no configuration file. So the operating system executes the question-driven, initial configuration routine, which is referred to as the *initial configuration dialog* or *setup mode*.

The setup mode is not intended for entering complex protocol features in the router but rather for bringing up a minimal configuration. You do not have to use the setup mode; you can use other configuration modes to configure the router.

The primary purpose of the setup mode is to rapidly bring up a minimal-feature configuration for any router that cannot find its configuration from some other source. In addition to being able to run the setup mode when the router boots, you may also initiate it by entering the **setup** privileged EXEC mode command.

To skip the system configuration dialog and configure the router manually, answer the first question in the system configuration dialog with **no** or press **Ctrl-C**.

To verify the router status, use the **show version** command:

```
R1# show version
Cisco IOS Software, Linux Software (I86BI_LINUX-ADVENTERPRISEK9-M), Version 15.2(4)M3,
DEVELOPMENT TEST SOFTWARE
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2013 by Cisco Systems, Inc.
Compiled Tue 26-Feb-13 19:06 by prod_rel_team

ROM: Bootstrap program is Linux

R1 uptime is 0 minutes
System returned to ROM by reload at 0
System restarted at 03:00:23 PST Wed Oct 7 2015
System image file is "unix:/iou_root/images/IOL/i86bi_linux-adventerprisek9-ms.152-
4.M3"
<... output omitted ...>
```

To verify the running configuration of the router, use the **show running-config** command:

```
R1# show running-config
Building configuration...

Current configuration : 2919 bytes
!
! No configuration change since last restart
version 15.2
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname R1
<... output omitted ...>
```

Configuring Router Interfaces

One of the main functions of a router is to forward packets from one network device to another. For the router to perform this task, you must define the characteristics of the interfaces through which the router receives and sends the packets.

There are two general types of physical interfaces on Cisco routers: *Ethernet interfaces* and *serial interfaces*.

- **Ethernet interfaces:** The term Ethernet interface refers to any type of Ethernet interface. For example, some Cisco routers have an Ethernet interface that is capable of only 10 Mbps, so to configure this type of interface, you would use the **interface ethernet** *slot*/*interface number* configuration command. However, other routers have interfaces that are capable of operating up to 100 Mbps. These interfaces are referred to as Fast Ethernet ports. You use the **interface fastethernet** *slot/interface number* command to configure these types of ports. Similarly, the interfaces that are capable of Gigabit Ethernet speeds are referenced with the **interface gigabitethernet** *slot/interface number* command.
- **Serial interfaces:** Serial interfaces are the second major type of physical interfaces on Cisco routers. To support point-to-point leased lines and Frame Relay access-link standards, Cisco routers use serial interfaces. You can then choose which data link layer protocol to use, such as HDLC or PPP for leased lines or Frame Relay for Frame Relay connections, and configure the router to use the correct data link layer protocol. You use the **interface serial** *slot*/*interface number* command when configuring these types of interfaces.

Note It is appropriate to mention the *loopback interface* here. A loopback interface is a virtual interface that resides on a router. It is not connected to any other device. Loopback interfaces are very useful because they will never go "down," unless the entire router goes down. This helps in managing routers because there will always be at least one active interface on the routers—the loopback interface. To create a loopback interface, all you need to do is enter the configuration mode for the interface. Optionally, you may add an IP address.

```
Router(config)# interface loopback 0
Router(config-if)# ip address 10.0.0.1 255.255.255.255
```

An IP address with all mask bits set to 1 is called the host IP address. The host IP address indicates that only one IP address is used in the subnet and is often used to address loopback interfaces.

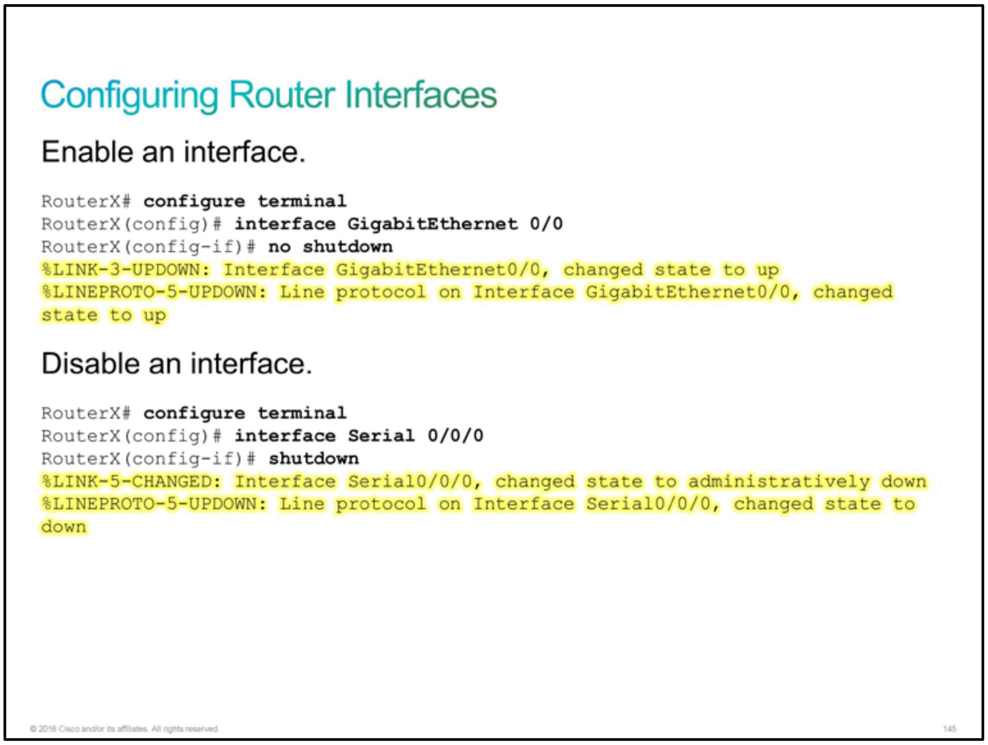
Routers use numbers to distinguish between the different interfaces of the same type. On routers, the interface numbers might be a single number, or two numbers that are separated by a slash, or three numbers that are separated by slashes. For example, all three of the following configuration commands are correct on at least one Cisco router model:

```
interface ethernet 0
interface fastethernet 0/1
interface serial 1/0/1
```

Note The router interface characteristics include, but are not limited to, interface description, the IP address of the interface, the data link encapsulation method, the media type, the bandwidth, and the clock rate. You can enable many features on a per-interface basis.

When you first configure an interface, except in the setup mode, you must administratively enable the interface before the router can use it to transmit and receive packets. Use the **no shutdown** command to allow Cisco IOS Software to use the interface.

You may want to disable an interface to perform hardware maintenance on a specific interface or a segment of a network. You may also want to disable an interface if a problem exists on a specific segment of the network, and you must isolate this segment from the rest of the network. The **shutdown** command administratively turns off an interface. To restart the interface, use the **no shutdown** command.



Configuring Router Interfaces

Enable an interface.

```
RouterX# configure terminal
RouterX(config)# interface GigabitEthernet 0/0
RouterX(config-if)# no shutdown
%LINK-3-UPDOWN: Interface GigabitEthernet0/0, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0, changed
state to up
```

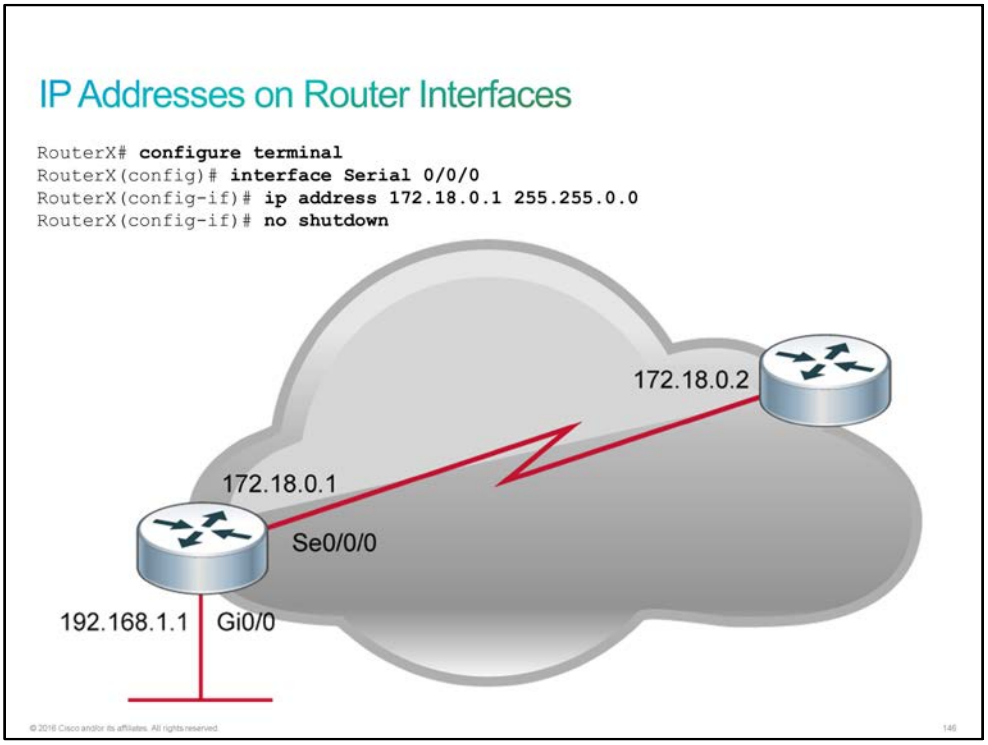
Disable an interface.

```
RouterX# configure terminal
RouterX(config)# interface Serial 0/0/0
RouterX(config-if)# shutdown
%LINK-5-CHANGED: Interface Serial0/0/0, changed state to administratively down
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0/0, changed state to
down
```

IP Addresses on Router Interfaces

You need physical street addresses to identify the locations of specific homes and companies so that mail can reach those real-world locations efficiently. In the same way, each interface on a Cisco router must have its own IP address to uniquely identify it on the network. If no IP address is configured, even if the interface is in the "up/up" state, the router will not attempt to send and receive IP packets on the interface. To attain proper operation, for every interface that a router should use for forwarding IP packets, the router needs an IP address.

The configuration of an IP address on an interface is relatively simple. To configure the address and mask, simply use the **ip address** *ip-address mask* interface subcommand. The following example shows the configuration of an IP address on the serial interface of a router.



IP Addresses on Router Interfaces

```
RouterX# configure terminal
RouterX(config)# interface Serial 0/0/0
RouterX(config-if)# ip address 172.18.0.1 255.255.0.0
RouterX(config-if)# no shutdown
```

The specific steps to configure an interface on a Cisco router are as follows:

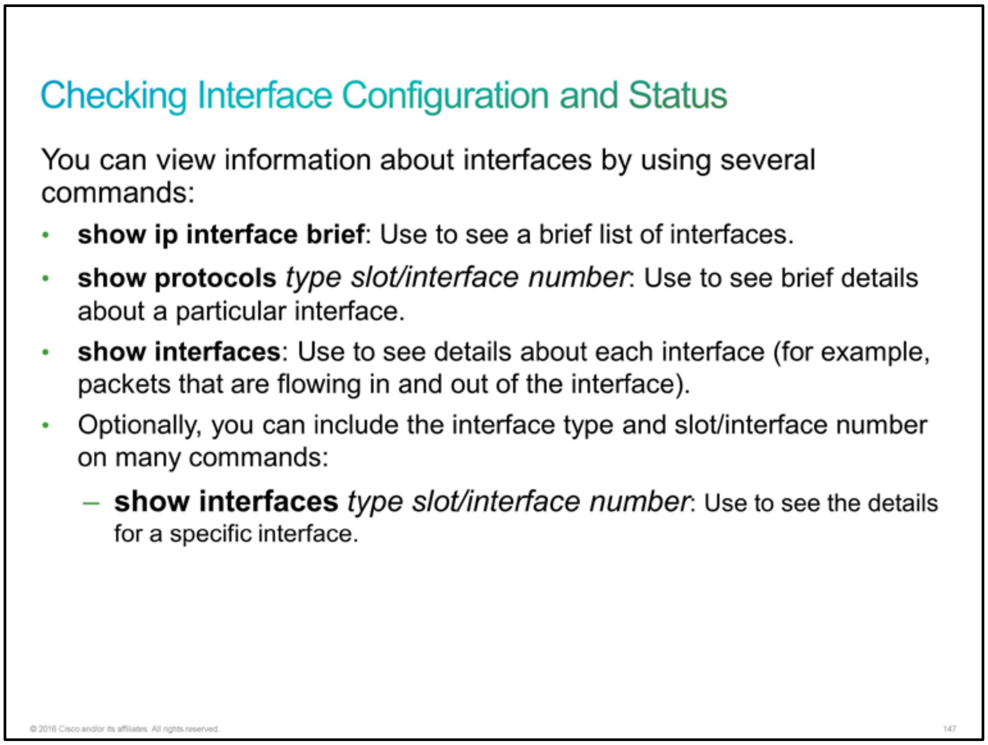
Configuration of an IP Address on the Serial Interface of a Router

Step	Action	Results and Notes
1	Enter the global configuration mode using the **configure terminal** command: Router# **configure terminal**	Displays a new prompt: Router(config)#
2	Identify the specific interface that requires an IP address by using the **interface** *type module*/*slot*/*port* command: Router(config)# **interface Serial 0/0/0**	Displays a new prompt; for example: Router(config-if)#

Step	Action	Results and Notes
3	Set the IP address and subnet mask for the interface by using the **ip address** *ip-address mask* command: Router(config-if)# **ip address 172.18.0.1 255.255.0.0**	Configures the IP address and subnet mask for the selected interface
4	Enable the interface to change the state from "administratively down" to "up" by using the **no shutdown** command: Router(config-if)# **no shutdown**	Enables the current interface
5	Exit the configuration mode of the interface by using the **exit** command: Router(config-if)# **exit**	Displays the global configuration mode prompt: Router(config)#

Checking Interface Configuration and Status

When you have completed the router interface configuration, you can verify the configuration by using various **show** commands.



Checking Interface Configuration and Status

You can view information about interfaces by using several commands:

- **show ip interface brief**: Use to see a brief list of interfaces.
- **show protocols** *type slot/interface number*: Use to see brief details about a particular interface.
- **show interfaces**: Use to see details about each interface (for example, packets that are flowing in and out of the interface).
- Optionally, you can include the interface type and slot/interface number on many commands:
  - **show interfaces** *type slot/interface number*: Use to see the details for a specific interface.

The following examples show sample outputs from the presented commands.

```
RouterY# show ip  interface brief
Interface                    IP-Address      OK? Method Status                Protocol
FastEthernet0/0         10.1.1.1          YES unset     up                         up
FastEthernet0/1         unassigned      YES unset   administratively    down down
Serial0/0/0                 unassigned      YES unset   administratively    down down
Serial0/0/1                 unassigned      YES unset   up                         up
Serial0/1/0                 unassigned      YES unset   up                         up
Serial0/1/1                 unassigned      YES unset   administratively    down down
```

The following table shows the output fields and their meanings.

Output Field	Description
Interface	Type of interface.
IP Address	IP address that is assigned to the interface.
OK?	"Yes" means that the IP address is valid. "No" means that the IP address is not valid.
Method	Describes how the IP address was obtained or configured.

Output Field	Description
Status	Shows the status of the interface.
Protocol	Shows the operational status of the routing protocol on this interface.

```
RouterX# show interfaces
GigabitEthernet0/0 is up, line protocol is up
  Hardware is CN Gigabit Ethernet, address is f866.f231.7250 (bia f866.f231.7250)
  Description: Link to ISP
  Internet address is 192.168.2.1/24
  MTU 1500 bytes, BW 100000 Kbit/sec, DLY 100 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set
  Keepalive set (10 sec)
  Full Duplex, 100Mbps, media type is RJ45
  output flow-control is unsupported, input flow-control is unsupported
  ARP type: ARPA, ARP Timeout 04:00:00
  Last input 00:00:53, output 00:00:09, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
<... output omitted ...>
```

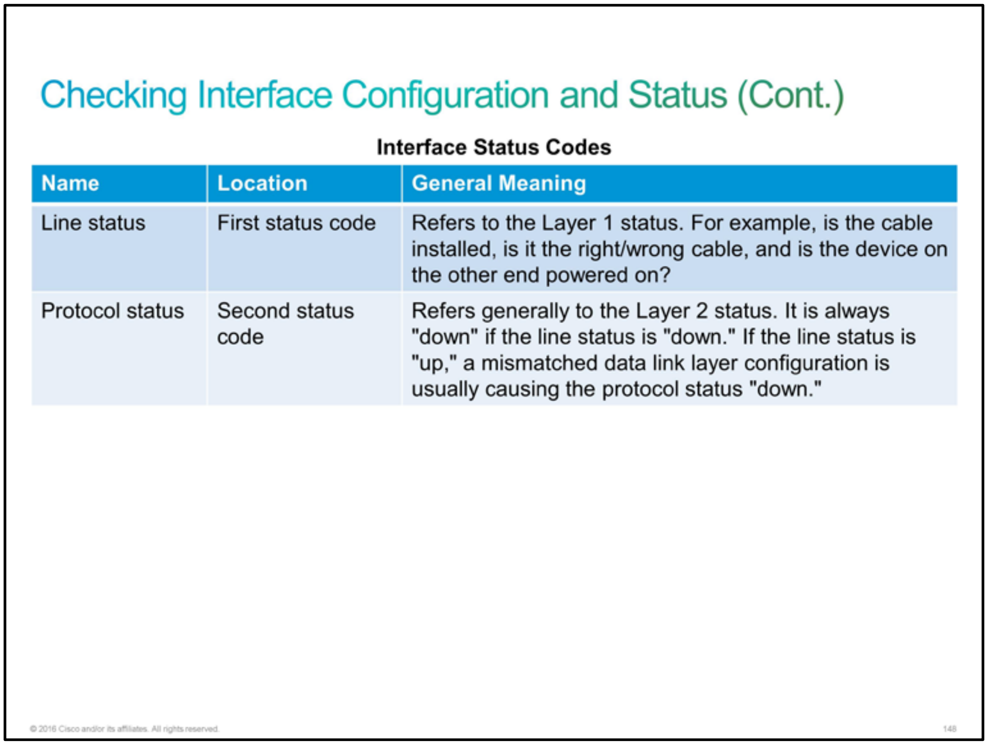
The following table shows some of the output fields for a Gigabit Ethernet interface and their meanings.

Output	Description
GigabitEthernet...is {up | down | administratively down}	Indicates whether the interface hardware is currently active, down, or if an administrator has taken it down.
Line protocol is {up | down}	Indicates whether the software processes that manage the line protocol consider the interface usable (that is, whether keepalives are successful). If the interface misses three consecutive keepalives, the line protocol is marked as down.
Hardware	Displays the hardware type and MAC address.
Description	Displays the configured interface description.
Internet address	Displays the IP address followed by the prefix length (subnet mask).
MTU	Displays the MTU of the interface.
BW	Shows the bandwidth of the interface in kilobits per second. The bandwidth parameter is used to compute routing protocol metrics and other calculations.
DLY	Shows the delay of the interface in microseconds.
Rely	Displays the reliability of the interface as a fraction of 255 (255/255 is 100-percent reliability), which is calculated as an exponential average over 5 minutes.

Output	Description
Load	Displays the load on the interface as a fraction of 255 (255/255 is completely saturated), which is calculated as an exponential average over 5 minutes.
Encapsulation	Shows the encapsulation method that is assigned to an interface.
5 minute input rate, 5 minute output rate	Shows the average number of bits and packets that the interface transmitted per second in the last 5 minutes.

Note By truncating the words, you can significantly shorten the commands that refer to router interfaces. For example, you can use **sh int fa0/0** instead of **show interfaces Fastethernet0/0**.

Each of the command outputs that the example shows lists two interface status codes. For a router to use an interface, the two interface status codes on the interface must be in the "up" state. The first status code refers essentially to whether the Layer 1 is working, and the second status code mainly (but not always) refers to whether the data link layer protocol is working. The following table summarizes these two status codes.



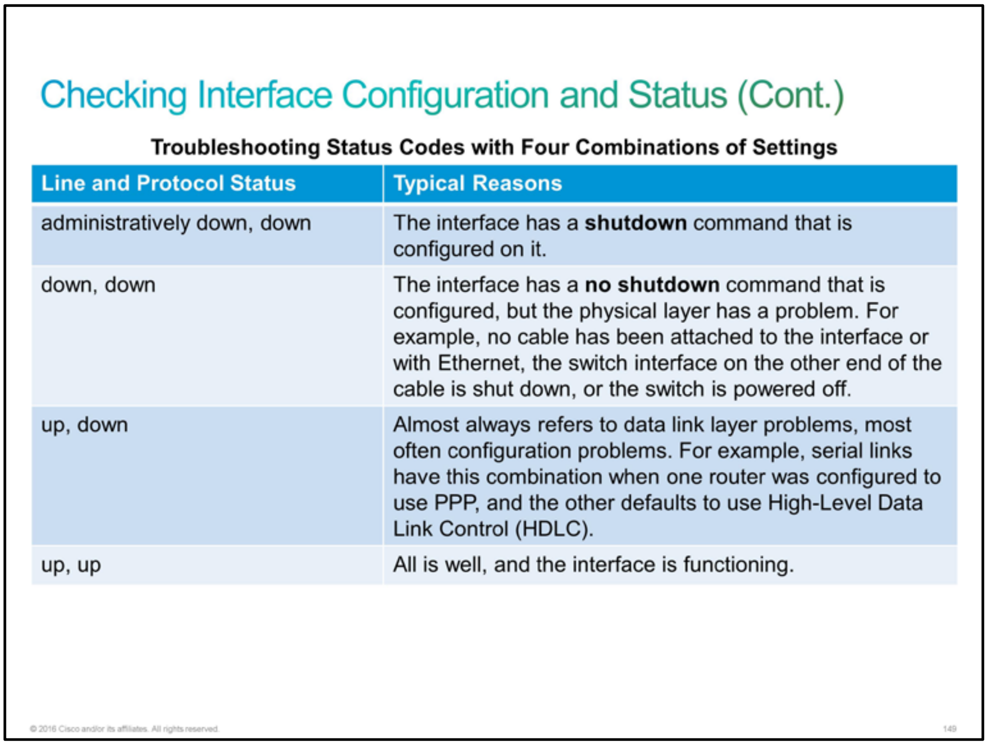
Checking Interface Configuration and Status (Cont.)

Interface Status Codes

Name	Location	General Meaning
Line status	First status code	Refers to the Layer 1 status. For example, is the cable installed, is it the right/wrong cable, and is the device on the other end powered on?
Protocol status	Second status code	Refers generally to the Layer 2 status. It is always "down" if the line status is "down." If the line status is "up," a mismatched data link layer configuration is usually causing the protocol status "down."

© 2016 Cisco and/or its affiliates. All rights reserved. 148

Four combinations of settings exist for the status codes when troubleshooting a network. The following table lists the four combinations, along with an explanation of the typical reasons of why an interface would be in this state. As you review the list, note that if the line status (the first status code) is not "up," the second will always be "down" because the data link layer functions cannot work if the physical layer has a problem.



Checking Interface Configuration and Status (Cont.)

Troubleshooting Status Codes with Four Combinations of Settings

Line and Protocol Status	Typical Reasons
administratively down, down	The interface has a **shutdown** command that is configured on it.
down, down	The interface has a **no shutdown** command that is configured, but the physical layer has a problem. For example, no cable has been attached to the interface or with Ethernet, the switch interface on the other end of the cable is shut down, or the switch is powered off.
up, down	Almost always refers to data link layer problems, most often configuration problems. For example, serial links have this combination when one router was configured to use PPP, and the other defaults to use High-Level Data Link Control (HDLC).
up, up	All is well, and the interface is functioning.

149

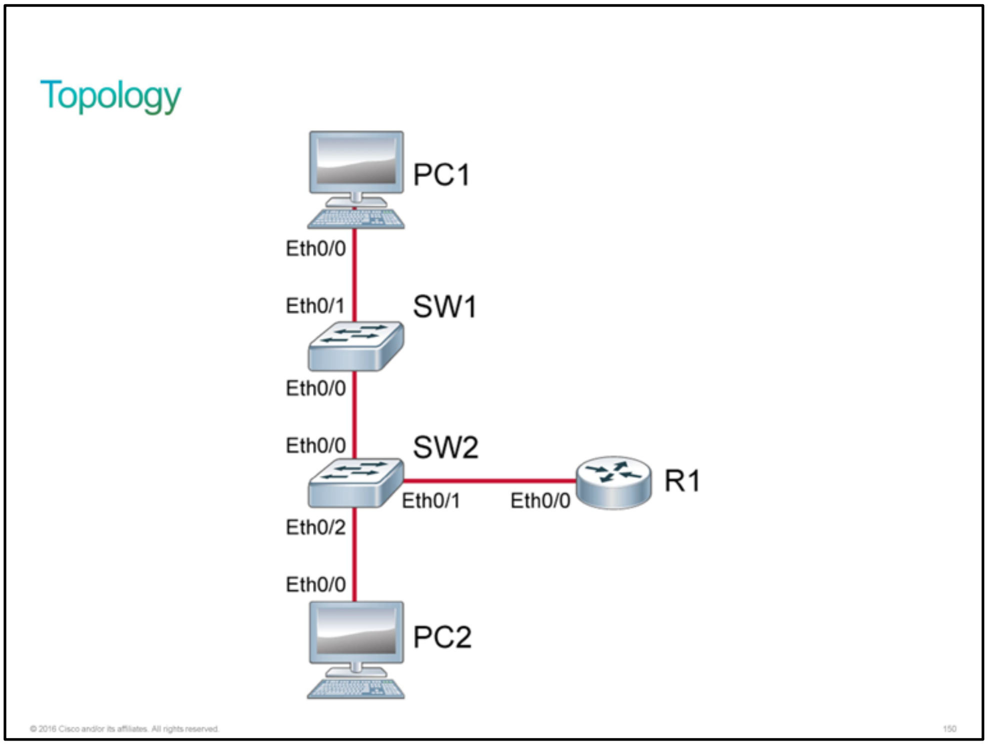
Note Note that the **show protocols** command is not available in all versions of Cisco IOS Software.

Discovery 6: Start with Cisco Router Configuration

Introduction

This discovery lab will guide you through the configuration of an interface on a Cisco IOS router. The lab is prepared with the devices that are represented in the topology diagram and in the connectivity table. In general, the devices are fully configured. An exception is the interface Ethernet0/0 on R1. You will configure that interface now.

Topology



Job Aid

Device Information

Device Information Table

Device	Characteristic	Value
PC1	Hostname	PC1
PC1	IP address	10.10.1.10/24
PC1	Default gateway	10.10.1.1

Device	Characteristic	Value
PC2	Hostname	PC2
PC2	IP address	10.10.1.20/24
PC2	Default gateway	10.10.1.1
SW1	Hostname	SW1
SW1	VLAN 1 IP address	10.10.1.2/24
SW1	Default gateway	10.10.1.1
SW1	Ethernet0/0 description	Link to SW2
SW1	Ethernet0/1 description	Link to PC1
SW2	Hostname	SW2
SW2	VLAN 1 IP address	10.10.1.3/24
SW2	Default gateway	10.10.1.1
SW2	Ethernet0/0 description	Link to SW1
SW2	Ethernet0/1 description	Link to R1
SW2	Ethernet0/2 description	Link to PC2
R1	Hostname	R1
R1	Ethernet0/0 description	Not configured
R1	Ethernet0/0 IP address	Not configured
R1	Loopback 0 IP	10.10.3.1/24

PCs in the virtual lab environment are simulated as routers, so you should use Cisco IOS commands to configure them or make verifications.

Task 1: Configure an IP Address on the Router Interfaces

Activity

Step 1 Access the console of R1 and enter the global configuration mode.

On R1, enter the following command:

```
R1# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#
```

Step 2 Enter the interface configuration mode for Ethernet0/0, configure 10.10.1.1/24 as its IP address, add an interface description, and then enable the interface.

On R1, enter the following commands:

```
R1(config)# interface Ethernet 0/0
R1(config-if)# ip address 10.10.1.1 255.255.255.0
R1(config-if)# description Link to SW2
R1(config-if)# no shutdown
```

Examine the IP routing table on R1. You should see the IP address (L - local) and IP subnet (C - connected) that you have just configured on the Ethernet0/0.

```
R1(config-if)# do show ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       + - replicated route, % - next hop override

Gateway of last resort is not set

      10.0.0.0/8 is variably subnetted, 4 subnets, 2 masks
C        10.10.1.0/24 is directly connected, Ethernet0/0
L        10.10.1.1/32 is directly connected, Ethernet0/0
C        10.10.3.0/24 is directly connected, Loopback0
L        10.10.3.1/32 is directly connected, Loopback0
```

Because you are in the interface configuration mode, you will use the **show** command with the **do** option.

Step 3 Use the **do** command to execute an EXEC mode **ping** command. Attempt to ping PC1 (10.10.1.10). The attempt should succeed.

On R1, enter the following command:

```
R1(config-if)# do ping 10.10.1.10
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.1.10, timeout is 2 seconds:
.!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 1/1/1 ms
R1(config-if)#
```

It is not uncommon for the first one or two ICMP echo requests to time out. It is usually due to delays that are associated with updating the ARP cache with the MAC address of the local peer.

Step 4 Leave the global configuration mode.

On R1, enter the following command:

```
R1(config-if)# end
R1#
```

Task 2: Verify Interface Configuration and Status

Activity

Step 1 On R1, display the running configuration for the Ethernet0/0 interfaces.

On R1, enter the following command:

```
R1# show running-config interface Ethernet 0/0
Building configuration...

Current configuration : 90 bytes
!
interface Ethernet0/0
 description Link to SW2
 ip address 10.10.1.1 255.255.255.0
end
```

Step 2 On R1, display a brief summary of the IP information and the statuses of all interfaces.

On R1, enter the following command:

```
R1# show ip interface brief
Interface                  IP-Address      OK? Method Status
Protocol
Ethernet0/0                10.10.1.1       YES manual up                    up
Ethernet0/1                unassigned      YES NVRAM  administratively down
down
Ethernet0/2                unassigned      YES NVRAM  administratively down
down
Ethernet0/3                unassigned      YES NVRAM  administratively down
down
Serial1/0                  unassigned      YES NVRAM  administratively down
down
Serial1/1                  unassigned      YES NVRAM  administratively down
down
Serial1/2                  unassigned      YES NVRAM  administratively down
down
Serial1/3                  unassigned      YES NVRAM  administratively down
down
Loopback0                  10.10.3.1       YES NVRAM  up                    up
```

Step 3 On the R1 router, display the status and statistics of the Ethernet0/0 interface.

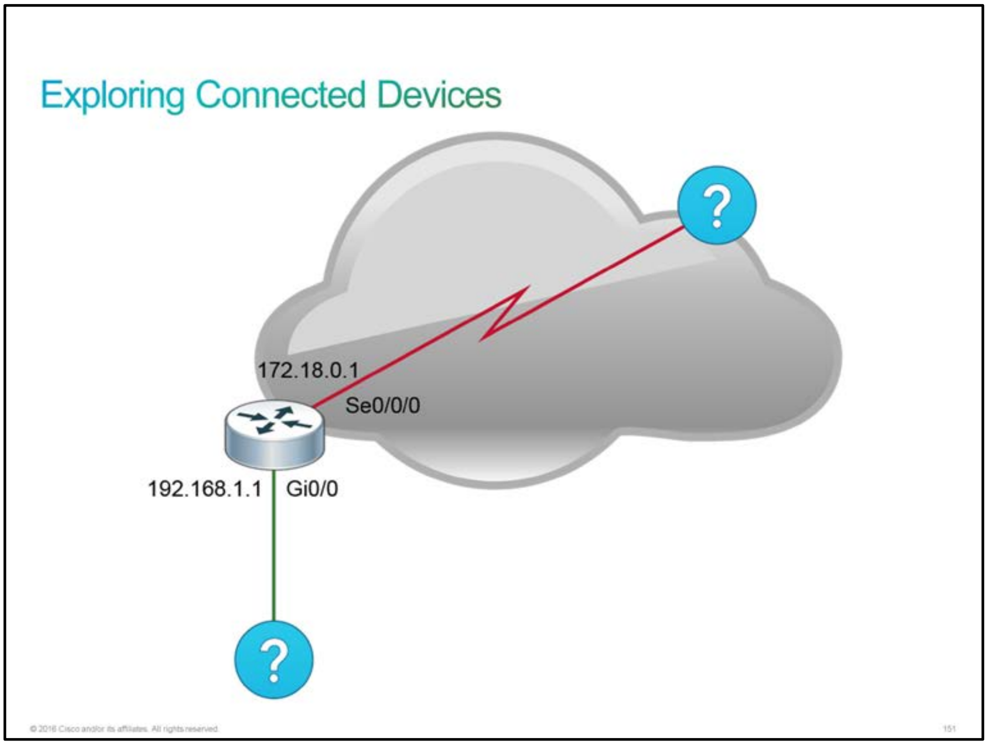
On R1, enter the following command:

```
R1# show interfaces Ethernet 0/0
Ethernet0/0 is up, line protocol is up
  Hardware is AmdP2, address is aabb.cc00.1800 (bia aabb.cc00.1800)
  Description: Link to SW2
  Internet address is 10.10.1.1/24
  MTU 1500 bytes, BW 10000 Kbit/sec, DLY 1000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set
  Keepalive set (10 sec)
  ARP type: ARPA, ARP Timeout 04:00:00
  Last input 00:00:00, output 00:00:06, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     1116 packets input, 71557 bytes, 0 no buffer
     Received 947 broadcasts (0 IP multicasts)
     0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
     0 input packets with dribble condition detected
     218 packets output, 23872 bytes, 0 underruns
     0 output errors, 0 collisions, 2 interface resets
     3 unknown protocol drops
     0 babbles, 0 late collision, 0 deferred
     0 lost carrier, 0 no carrier
     0 output buffer failures, 0 output buffers swapped out
```

This is the end of the discovery lab.

Exploring Connected Devices

Most network devices, by definition, do not work in isolation. A Cisco device frequently has other Cisco devices as neighbors on the network. If you are able to obtain information about those other devices, it can help you with any network design decisions, troubleshooting, and completing equipment changes.



Exploring Connected Devices

If you do not have any documentation about the network topology or if the existing documentation is not up to date, you may find yourself in a position of needing to discover the neighboring devices of a router. You can sometimes do this procedure manually by inspecting the physical wiring if the devices are installed next to each other. When neighboring devices are in other buildings or cities, you must use a different method.

One possibility is to use a dynamic discovery protocol that gathers information about directly connected devices. Cisco devices support Cisco Discovery Protocol, which provides information about directly connected Cisco devices and their functions and capabilities.

Cisco Discovery Protocol is a Cisco proprietary protocol that discovers basic information about neighboring Cisco devices without needing to know the passwords for the neighboring devices. To discover information, routers and switches send Cisco Discovery Protocol messages out each of their interfaces. The messages essentially announce information about the device that sent the Cisco Discovery Protocol message. Devices that support Cisco Discovery Protocol learn information about other devices by listening for the advertisements that these devices send.

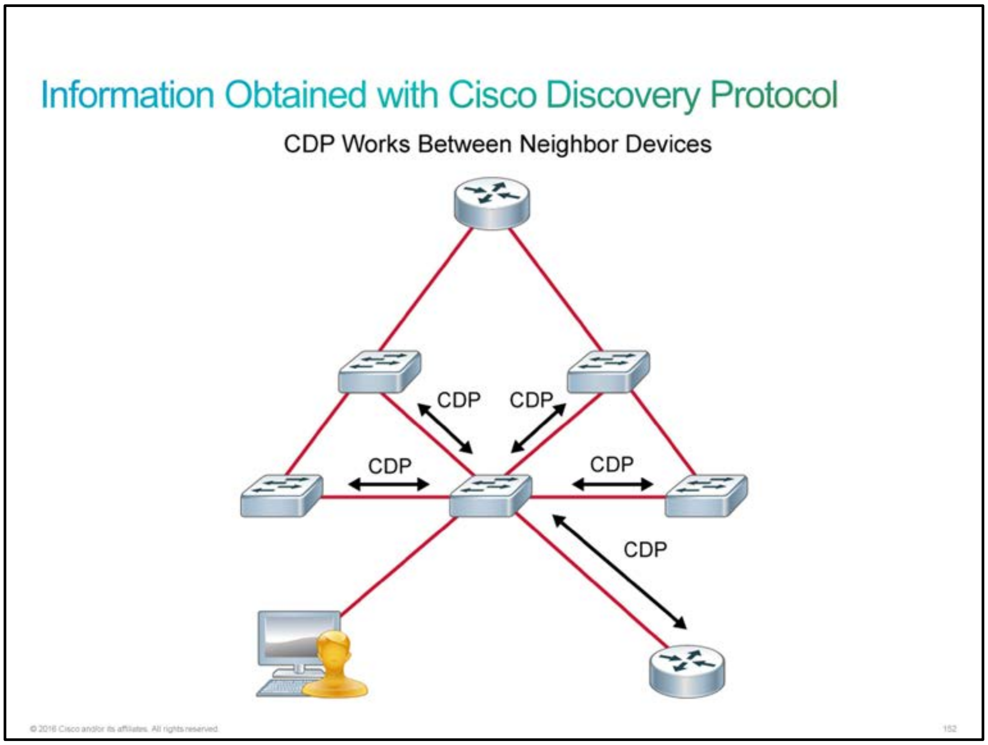
From a troubleshooting perspective, you can use Cisco Discovery Protocol to either confirm or fix the documentation that a network diagram shows or even discover the devices and interfaces that a network uses. Confirming that the network is actually cabled to match the network diagram is a good step to take before trying to predict the normal flow of data in a network.

On media that support multicasts at the data link layer, Cisco Discovery Protocol uses multicast frames; on other media, Cisco Discovery Protocol sends a copy of the Cisco Discovery Protocol update to any known data link addresses. So, any Cisco Discovery Protocol-supporting device that shares a physical medium with another Cisco Discovery Protocol-supporting device can learn about the other device.

Note	Another dynamic discovery protocol is LLDP, which is a standardized, vendor-independent discovery protocol that discovers neighboring devices from different vendors. The IEEE standardized this protocol as the 802.1AB standard. LLDP performs functions that are similar to Cisco Discovery Protocol.

Information Obtained with Cisco Discovery Protocol

The following figure displays an example of how Cisco Discovery Protocol (CDP) exchanges information with its directly connected neighbors. You can display the results of this information exchange on a console that is connected to a network device that is configured to run Cisco Discovery Protocol on its interfaces.



Information Obtained with Cisco Discovery Protocol

CDP Works Between Neighbor Devices

Cisco Discovery Protocol provides the following information about each neighboring device:

- **Device identifiers:** For example, the configured host name of the switch
- **Address list:** Up to one network layer address for each protocol that is supported
- **Port identifier:** The name of the local port and remote port, in the form of an ASCII character string such as Ethernet0
- **Capabilities list:** Supported features—for example, the device acting as a source-route bridge and also as a router
- **Platform:** The hardware platform of the device; for example, Cisco 7200 Series Routers

Notice that the upper router in the previous figure is not connected directly to the console of the administrator. To obtain Cisco Discovery Protocol information about this upper router from the console of the administrator, network staff could use Telnet to connect to a switch that is connected directly to this target device.

Using Cisco Discovery Protocol

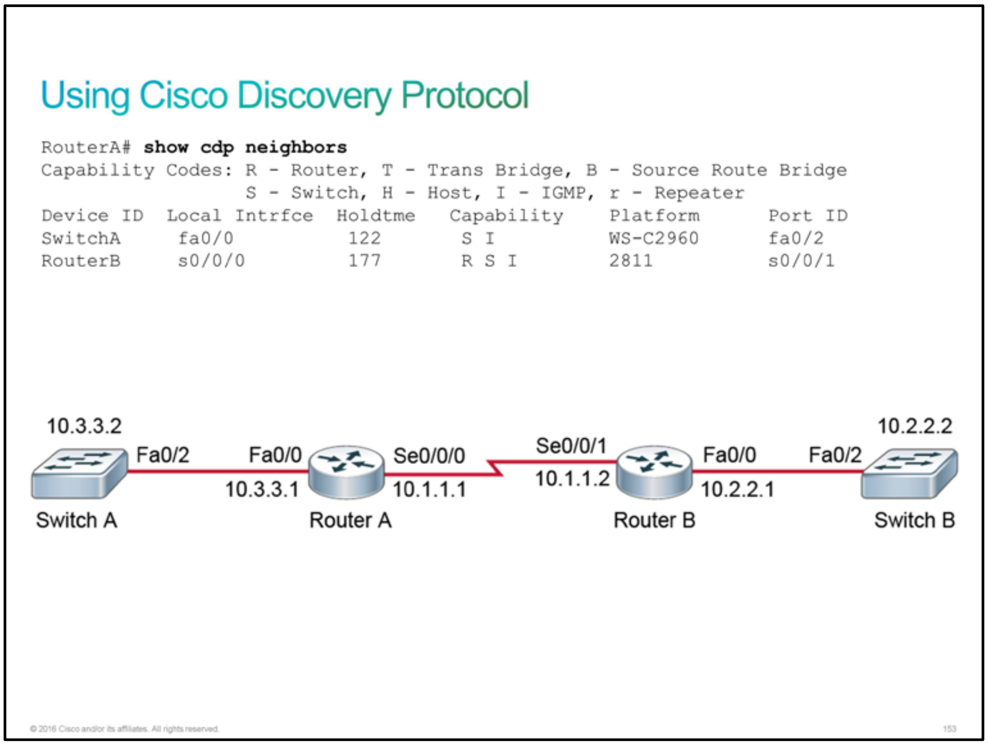
You can enable or disable Cisco Discovery Protocol on a router as a whole (global) or on a port-by-port (interface) basis. You can also view Cisco Discovery Protocol information with the **show cdp** command. Cisco Discovery Protocol has several keywords that enable access to different types of information and different levels of detail. The following example shows the different **show cdp** options.

```
RouterA# show cdp ?
  entry      Information for specific neighbor entry
  interface  CDP interface status and configuration
  neighbors  CDP neighbor entries
  traffic    CDP statistics
```

The Cisco Discovery Protocol functionality is enabled by default on all interfaces (except for Frame Relay multipoint subinterfaces), but you can disable this functionality at the device level. However, some interfaces, such as ATM interfaces, do not support Cisco Discovery Protocol. To prevent other Cisco Discovery Protocol-capable devices from accessing information about a specific device, you use the **no cdp run** global configuration command. To disable Cisco Discovery Protocol on an interface, use the **no cdp enable** command. To enable Cisco Discovery Protocol on an interface, use the **cdp enable** interface configuration command.

```
RouterA(config)# no cdp run
! Disable CDP Globally
RouterA(config)# interface serial0/0/0
RouterA(config-if)# no cdp enable
! Disable CDP on just this interface
```

The **show cdp neighbors** command displays information about Cisco Discovery Protocol neighbors. The following example shows the Cisco Discovery Protocol output for Router A.



Using Cisco Discovery Protocol

```
RouterA# show cdp neighbors
Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater
Device ID  Local Intrfce  Holdtme   Capability    Platform      Port ID
SwitchA     fa0/0          122       S I          WS-C2960      fa0/2
RouterB     s0/0/0         177       R S I        2811          s0/0/1
```

For each Cisco Discovery Protocol neighbor, the interface displays the following information:

- Device ID
- Local interface
- Holdtime value, in seconds
- Device capability code
- Hardware platform
- Remote port ID

The holdtime value indicates how long the receiving device should hold the Cisco Discovery Protocol packet before discarding it.

The format of the **show cdp neighbors** output varies among different types of devices, but the available information is generally consistent across devices.

You can use the **show cdp neighbors** command on a Cisco Catalyst switch to display the Cisco Discovery Protocol updates that the switch receives on the local interfaces. Note that on a switch, the local interface is referred to as the local port.

If you add the **detail** argument to the **show cdp neighbors** command, the resulting output includes additional information, such as the network layer addresses of neighboring devices. The output from the **show cdp neighbors detail** command is identical to the one that the **show cdp entry *** command produces, as shown here.

```
Device ID: RouterB
Entry address(es):
  IP address: 10.1.1.2
Platform: Cisco 2811,  Capabilities: Router Switch IGMP
Interface: Serial0/0/0,  Port ID (outgoing port): Serial0/0/1
Holdtime : 155 sec
Version :
Cisco IOS Software, 2800 Software (C2800NM-ADVIPSERVICESK9-M), Version 12.4(12),
RELEASE
SOFTWARE (fc1)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2006 by Cisco Systems, Inc.
Compiled Fri 17-Nov-06 12:02 by prod_rel_team
```

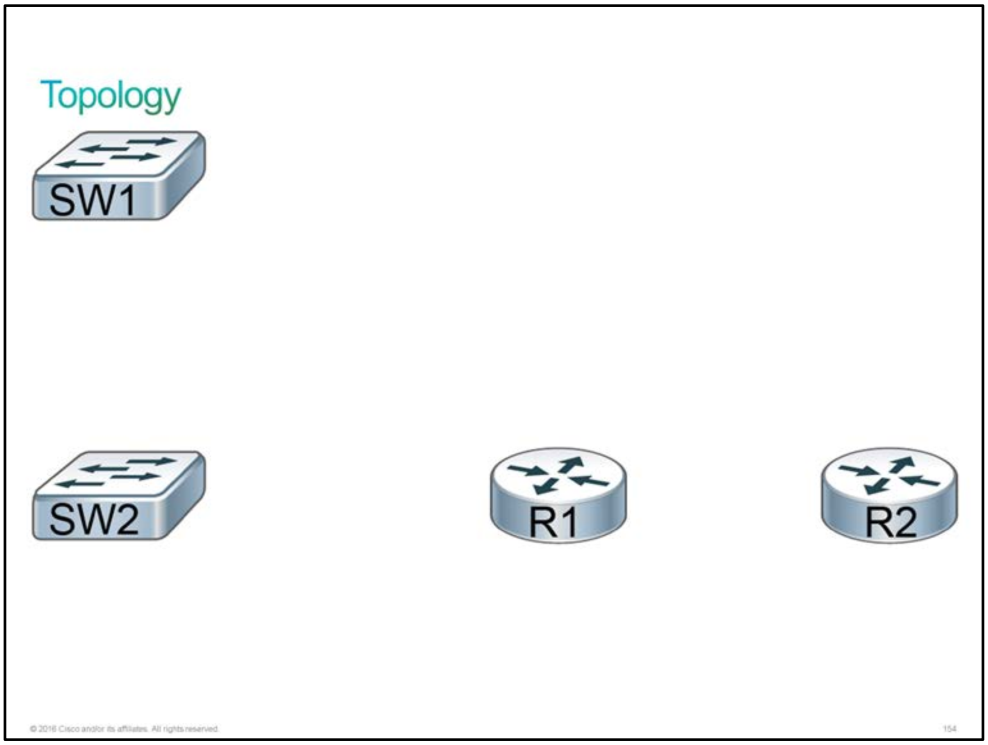
Note Cisco Discovery Protocol is limited to gathering information about the directly connected Cisco neighbors. Other tools, such as Telnet, are available for gathering information about remote devices that are not directly connected.

Discovery 7: Configure Cisco Discovery Protocol

Introduction

During this discovery lab, you will use Cisco Discovery Protocol to map the connectivity within an unfamiliar network. There are four devices in the topology and you have access to their console ports, but you do not know how they are connected. Using Cisco Discovery Protocol commands, you will determine the actual topology.

Topology



Job Aid

There is no Job Aid available for this lab exercise because the objective of the lab is to map the connectivity within an unfamiliar network.

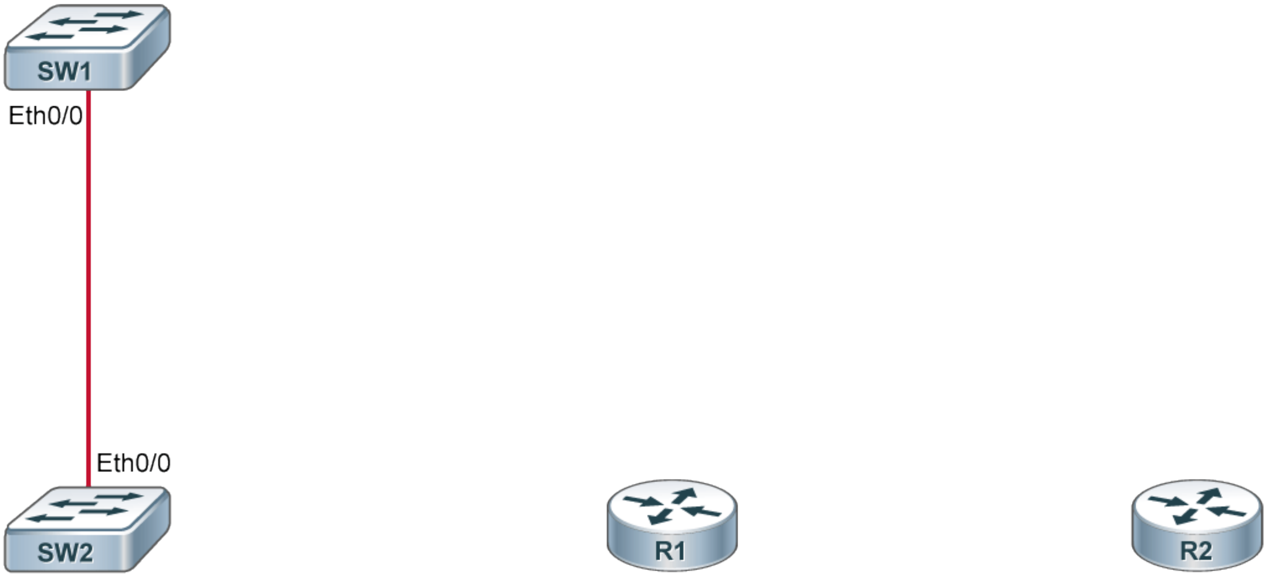
Task 1: Discover Neighbors Using Cisco Discovery Protocol

Activity

Step 1 Before accessing the console of SW1, wait 60 second for Cisco Discovery Protocol to populate its database. On SW1, use the **show cdp neighbor** command to determine the devices to which SW1 is connected. Note both the local port and the port on the remote device.

On SW1, enter the following command:

```
SW1# show cdp neighbor
Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater, P - Phone,
                  D - Remote, C - CVTA, M - Two-port Mac Relay

Device ID        Local Intrfce     Holdtme    Capability  Platform  Port ID
SW2              Eth 0/0           153              S I   Linux Uni Eth 0/0
```

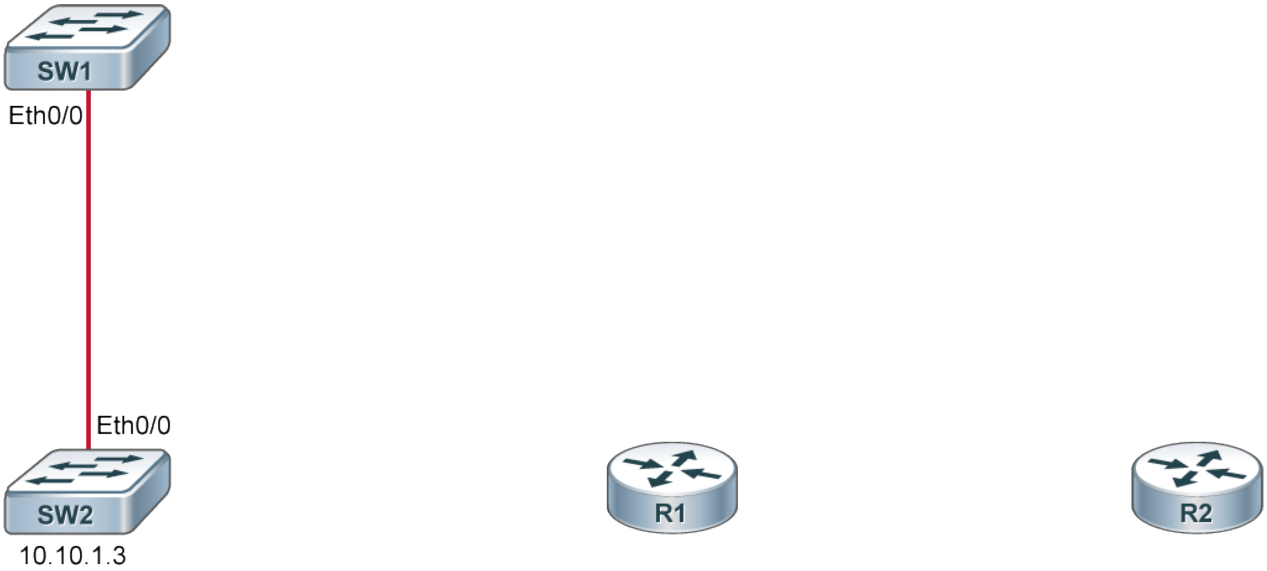


Step 2 Execute the **show cdp neighbor** command again, but this time using the **detail** argument. What is the IP address of SW2?

On SW1, enter the following command:

```
SW1# show cdp neighbor detail
-------------------------
Device ID: SW2
Entry address(es):
  IP address: 10.10.1.3
Platform: Linux Unix,  Capabilities: Switch IGMP
Interface: Ethernet0/0,  Port ID (outgoing port): Ethernet0/0
Holdtime : 147 sec

Version :
Cisco IOS Software, Solaris Software (I86BI_LINUXL2-ADVENTERPRISEK9-M),
Experimental Version 15.1(20130919:231344) [dstivers-sept19-2013pm-team_track
107]
Copyright (c) 1986-2013 by Cisco Systems, Inc.
Compiled Thu 19-Sep-13 22:38 by dstivers

advertisement version: 2
VTP Management Domain: ''
Duplex: half
Management address(es):
  IP address: 10.10.1.3
```

The IP address of SW2 is 10.10.1.3.

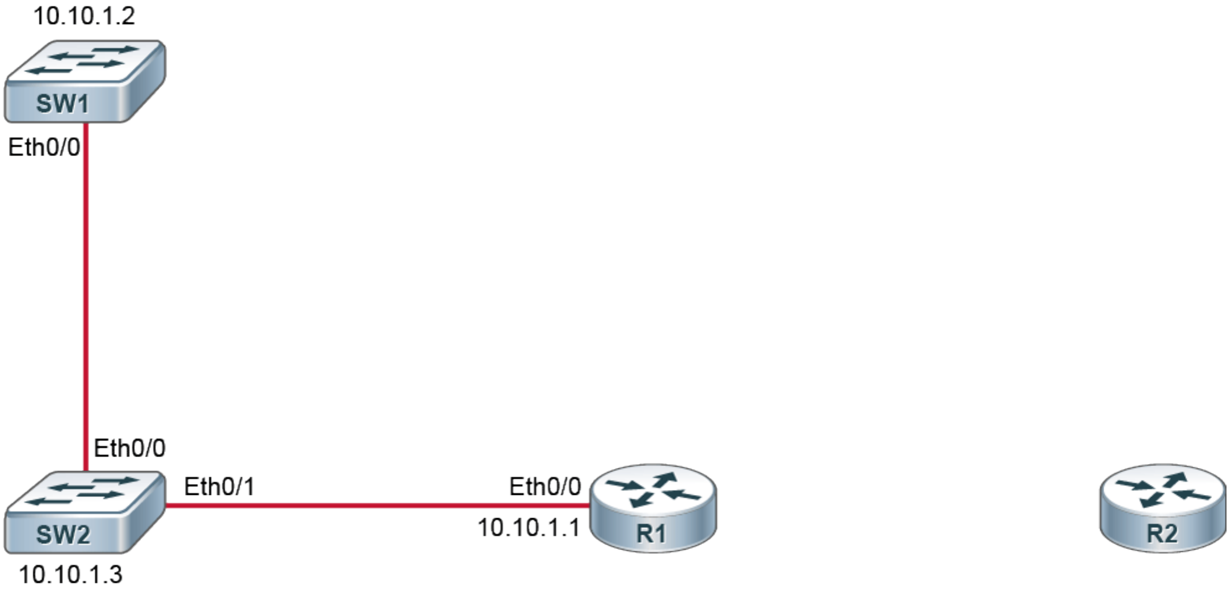


Step 3 Continue the topology inspection from SW2. You know that SW1 is one of the neighbors of SW2. What are the other neighbors of SW2?

On SW2, enter the following command:

```
SW2# show cdp neighbors
Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater, P - Phone,
                  D - Remote, C - CVTA, M - Two-port Mac Relay

Device ID        Local Intrfce     Holdtme    Capability  Platform  Port ID
SW1              Eth 0/0           130              S I   Linux Uni Eth 0/0
R1               Eth 0/1           153               R    Linux Uni Eth 0/0
SW2# show cdp neighbors detail
-------------------------
Device ID: SW1
Entry address(es):
  IP address: 10.10.1.2
Platform: Linux Unix,  Capabilities: Switch IGMP
Interface: Ethernet0/0,  Port ID (outgoing port): Ethernet0/0
Holdtime : 124 sec

Version :
Cisco IOS Software, Solaris Software (I86BI_LINUXL2-ADVENTERPRISEK9-M),
Experimental Version 15.1(20130919:231344) [dstivers-sept19-2013pm-team_track
107]
Copyright (c) 1986-2013 by Cisco Systems, Inc.
Compiled Thu 19-Sep-13 22:38 by dstivers

advertisement version: 2
VTP Management Domain: ''
Duplex: half
Management address(es):
  IP address: 10.10.1.2

-------------------------
Device ID: R1
Entry address(es):
  IP address: 10.10.1.1
Platform: Linux Unix,  Capabilities: Router
Interface: Ethernet0/1,  Port ID (outgoing port): Ethernet0/0
Holdtime : 147 sec

Version :
Cisco IOS Software, Linux Software (I86BI_LINUX-ADVENTERPRISEK9-M), Version
15.2(4)M3, DEVELOPMENT TEST SOFTWARE
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2013 by Cisco Systems, Inc.
Compiled Tue 26-Feb-13 19:06 by prod_rel_team

advertisement version: 2
Duplex: half
Management address(es):
```

The IP address of SW1 is 10.10.1.2 and the IP address of R1 is 10.10.1.1.

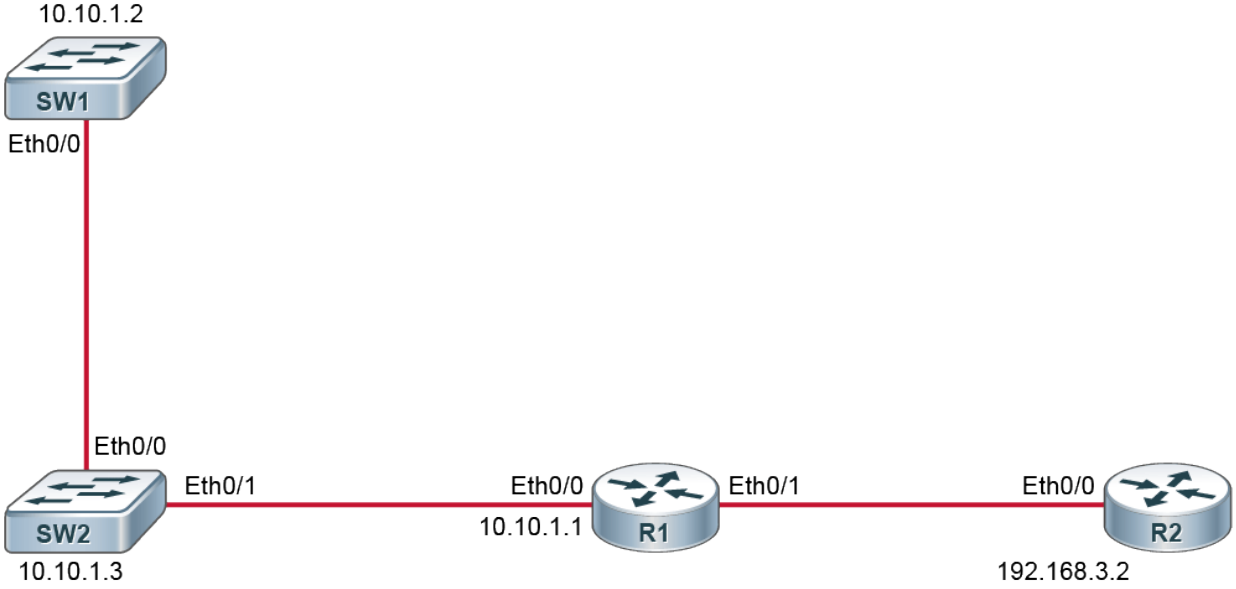


Step 4 Continue the topology inspection from R1. What are the other neighbors of R1?

On R1, enter the following commands:

```
R1# show cdp neighbors
Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater, P - Phone,
                  D - Remote, C - CVTA, M - Two-port Mac Relay

Device ID        Local Intrfce     Holdtme    Capability  Platform  Port ID
SW2              Eth 0/0            164            S I    Linux Uni Eth 0/1
R2               Eth 0/1            173             R     Linux Uni Eth 0/0
R1# show cdp neighbors  detail
-------------------------
Device ID: SW2
Entry address(es):
  IP address: 10.10.1.3
Platform: Linux Unix,  Capabilities: Switch IGMP
Interface: Ethernet0/0,  Port ID (outgoing port): Ethernet0/1
Holdtime : 161 sec

Version :
Cisco IOS Software, Solaris Software (I86BI_LINUXL2-ADVENTERPRISEK9-M),
Experimental Version 15.1(20130919:231344) [dstivers-sept19-2013pm-team_track
107]
Copyright (c) 1986-2013 by Cisco Systems, Inc.
Compiled Thu 19-Sep-13 22:38 by dstivers

advertisement version: 2
VTP Management Domain: ''
Native VLAN: 1
Duplex: half

-------------------------
Device ID: R2
Entry address(es):
  IP address: 192.168.3.2
Platform: Linux Unix,  Capabilities: Router
Interface: Ethernet0/1,  Port ID (outgoing port): Ethernet0/0
Holdtime : 170 sec

Version :
Cisco IOS Software, Linux Software (I86BI_LINUX-ADVENTERPRISEK9-M), Version
15.2(4)M3, DEVELOPMENT TEST SOFTWARE
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2013 by Cisco Systems, Inc.
Compiled Tue 26-Feb-13 19:06 by prod_rel_team

advertisement version: 2
Duplex: half
```

The IP address of R2 is 192.168.3.2.

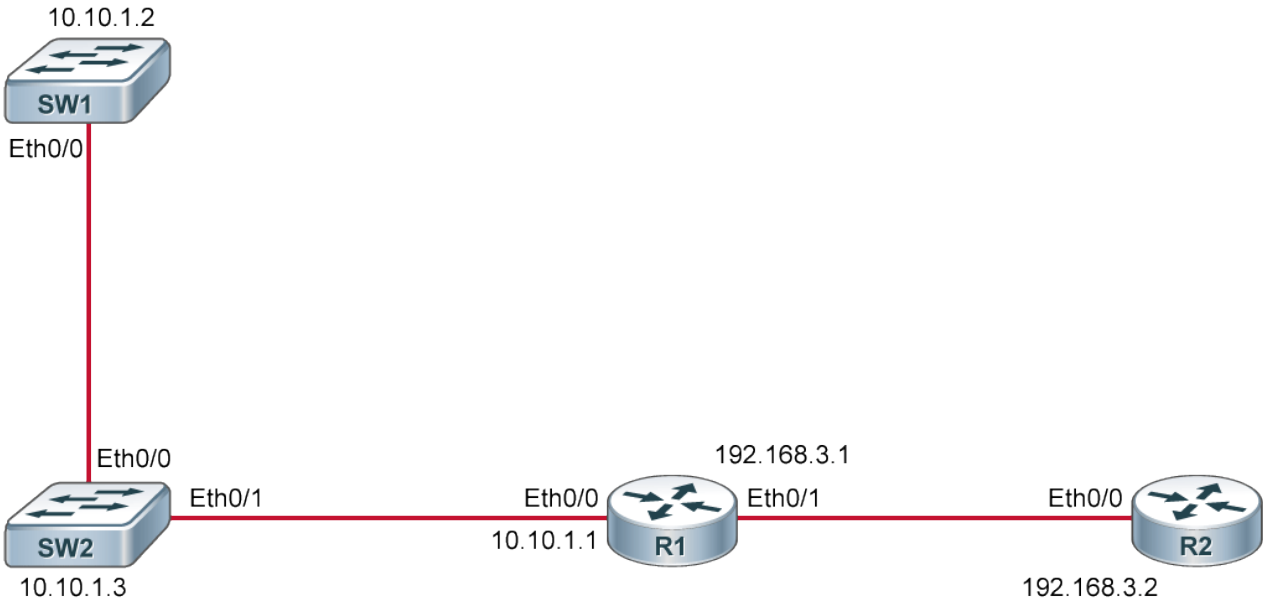


Step 5 Continue the topology inspection from R2.

On R2, enter the following commands:

```
R2# show cdp neighbors
Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater, P - Phone,
                  D - Remote, C - CVTA, M - Two-port Mac Relay

Device ID        Local Intrfce     Holdtme    Capability  Platform  Port ID
R1               Eth 0/0            176              R    Linux Uni Eth 0/1
R2# show cdp neighbors detail
-------------------------
Device ID: R1
Entry address(es):
  IP address: 192.168.3.1
Platform: Linux Unix,  Capabilities: Router
Interface: Ethernet0/0,  Port ID (outgoing port): Ethernet0/1
Holdtime : 174 sec

Version :
Cisco IOS Software, Linux Software (I86BI_LINUX-ADVENTERPRISEK9-M), Version
15.2(4)M3, DEVELOPMENT TEST SOFTWARE
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2013 by Cisco Systems, Inc.
Compiled Tue 26-Feb-13 19:06 by prod_rel_team

advertisement version: 2
Duplex: half
```

R2 has no additional neighbors.



This is the end of the discovery lab.

Configuring LLDP

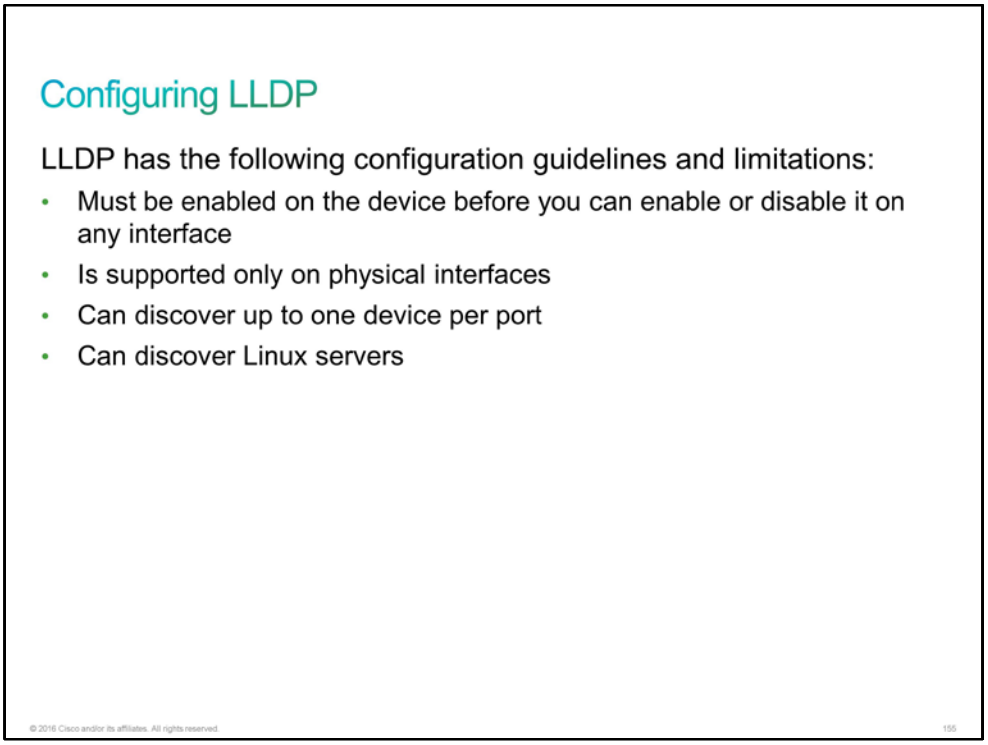
To permit the discovery of non-Cisco devices, the switch also supports LLDP, which is a vendor-neutral device discovery protocol that IEEE 802.1AB standard defines. LLDP allows network devices to advertise information about themselves to other devices on the network. This protocol runs over the data link layer, which allows two systems that are running different network layer protocols to learn about each other.

LLDP is a one-way protocol that transmits information about the capabilities and the status of a device and its interfaces. LLDP devices use the protocol to solicit information only from other LLDP devices.

LLDP supports a set of attributes that it uses to discover other devices. These attributes contain TLV descriptions. LLDP devices can use TLVs to send and receive information to other devices on the network. Using this protocol, devices can advertise details such as configuration information, device capabilities, and device identity.

LLDP advertises the following TLVs by default:

- DCBXP
- Management address
- Port description
- Port VLAN
- System capabilities
- System description
- System name

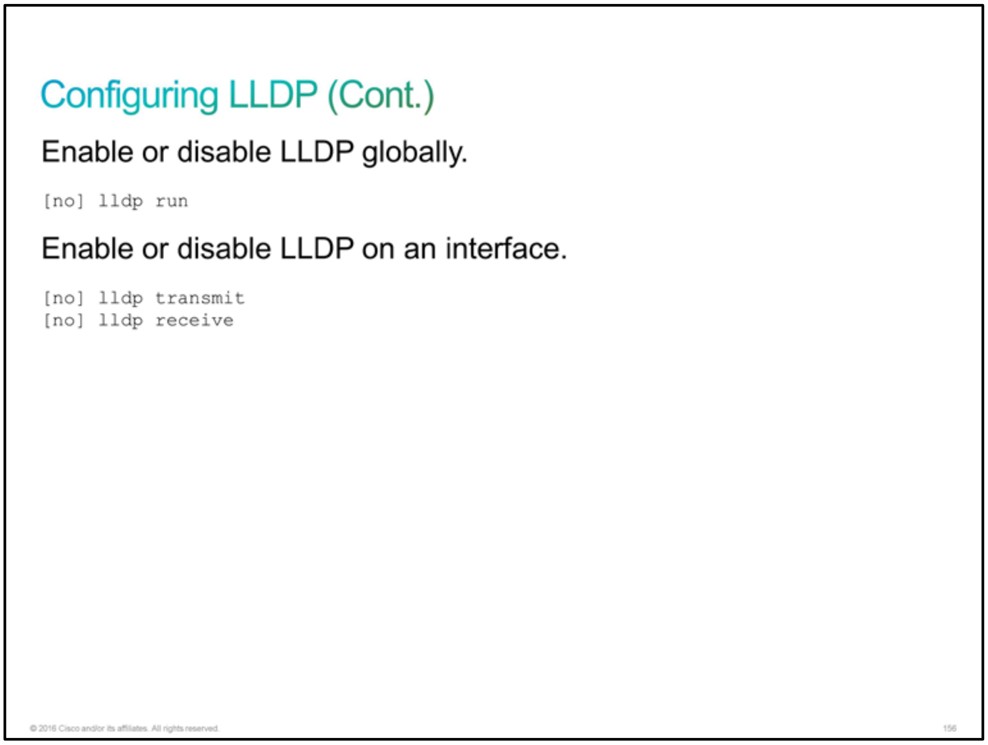


Configuring LLDP

LLDP has the following configuration guidelines and limitations:

- Must be enabled on the device before you can enable or disable it on any interface
- Is supported only on physical interfaces
- Can discover up to one device per port
- Can discover Linux servers

155



Configuring LLDP (Cont.)

Enable or disable LLDP globally.

```
[no] lldp run
```

Enable or disable LLDP on an interface.

```
[no] lldp transmit
[no] lldp receive
```

156

After you globally enable LLDP, it is enabled on all supported interfaces by default. The **lldp transmit** command enables the transmission of LLDP packets on an interface. The **lldp receive** command enables the reception of LLDP packets on an interface.

Challenge

1. Which of the following does the router first perform upon being booted up?

   A. Finds and loads IOS
   B. Finds and applies the configuration statements
   C. Runs POST to test the hardware
   D. Displays command line prompt

2. If POST completes during the startup of a Cisco Device, and there is no configuration file, what does the router do?

   A. It reboots
   B. It initiates initial configuration through 'setup'
   C. It creates a configuration file with default settings and boots that.
   D. It shutdowns the router

3. **show ip interface brief** command displays the packets that are flowing in and out of the interface. True or False.?

   A. True
   B. False

4. Match the correct description about the line and protocol status on the right with conditions listed on the left.

Serial 0/0/0 is up, line protocol is up	Interface is shutdown condition
Serial0/0/0 is up, line protocol is down	Interface is in no shutdown condition, but cable is not connected to ethernet interface
Ethernet0/0 is administratively down, line protocol is down	Cable is connected to the interface and placed in no shutdown condition, but encapsulation mismatch encountered on the serial links
Ethernet0/0 is down, line protocol is down	Interface is functioning well with no issues

5. Which command would you use to enable LLDP globally on a router?

   A. **lldp transmit**
   B. **lldp run**
   C. **lldp receive**
   D. **cdp run**
   E. **cdp enable**

6. CDP can determine the platform version of the neighboring devices. True or False?

   A. True
   B. False

7. Which command displays the IP address of a neighboring device?

   A. **show cdp neighbors**
   B. **show ip interface brief**
   C. **show interfaces**
   D. **show cdp neighbors detail**

Answer Key

Challenge

1. C
2. B
3. B
4.

Ethernet0/0 is administratively down, line protocol is down	Interface is shutdown condition
Ethernet0/0 is down, line protocol is down	Interface is in no shutdown condition, but cable is not connected to ethernet interface
Serial0/0/0 is up, line protocol is down	Cable is connected to the interface and placed in no shutdown condition, but encapsulation mismatch encountered on the serial links
Serial 0/0/0 is up, line protocol is up	Interface is functioning well with no issues

5. B
6. A
7. D

Lesson 6: Exploring the Packet Delivery Process

Introduction

Your boss sends you to your customer to debug problems with undelivered IP packets. You will need to illustrate the role of the Layer 2 address and Layer 3 address in the packet delivery process. You will investigate the role of ARP. At this point, you will need to understand all individual pieces of the packet delivery process.

Address Resolution Protocol

Because a frame must contain a MAC address, there must be a way to resolve an IP address to a MAC address. For example, if you issue the **ping 10.1.1.3** command, the MAC address of 10.1.1.3 must be included in the MAC destination field of the frame. To determine the MAC address of 10.1.1.3, a process is performed by a Layer 2 protocol called ARP.

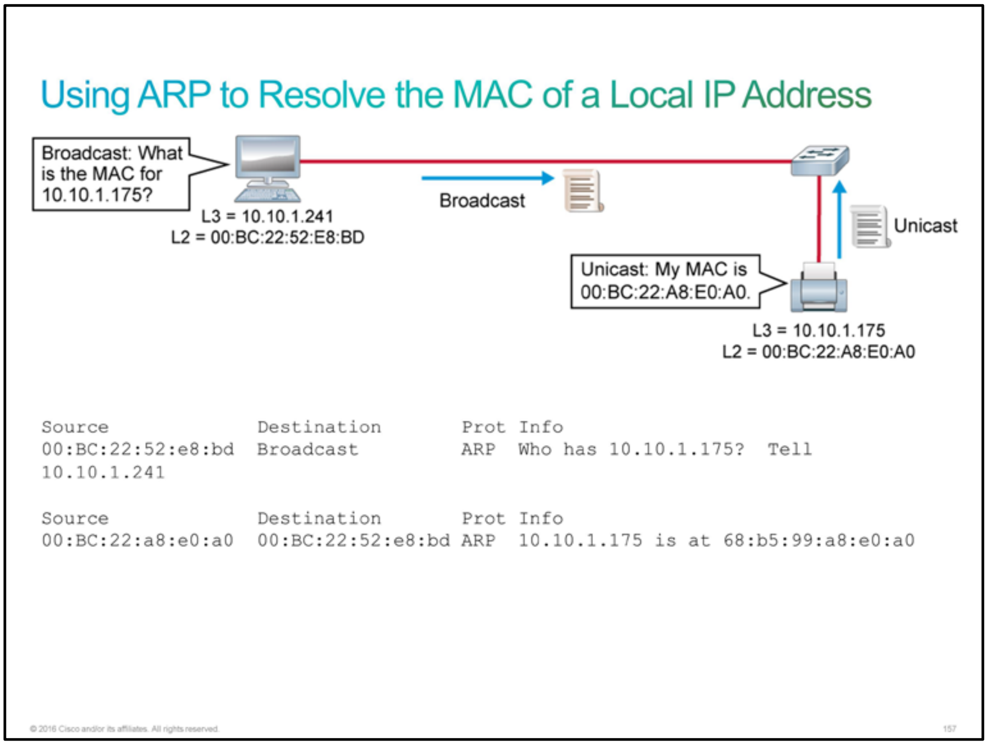
ARP provides two essential services:

- **Address resolution:** Mapping IP addresses to MAC addresses on a network
- **Caching:** Locally storing MAC addresses that are learned via ARP

The term *address resolution* refers to the process of binding the IP address of a remote device to its MAC address. ARP sends a broadcast message to all devices on the local network. This message includes its own IP address and the destination IP address. The message is basically asking the device on which the destination IP address resides to respond with its MAC address. The address resolution procedure is completed when the originator receives the reply packet, which contains the required MAC address, from the target and updates the table containing all the current bindings.

Using ARP to Resolve the MAC of a Local IP Address

Because ARP is a Layer 2 protocol, its scope is limited to the local LAN. An IP host can tell if the IP host that it wants to communicate with is on the same network by comparing the destination IP address against its configured subnet mask. For example, IP host 10.10.1.241/24 is on the 10.10.1.0 network. If the IP host that it wants to communicate with is 10.10.1.175, it knows that this IP host is also on the local 10.10.1.0 network, and can request an ARP for its MAC address directly.



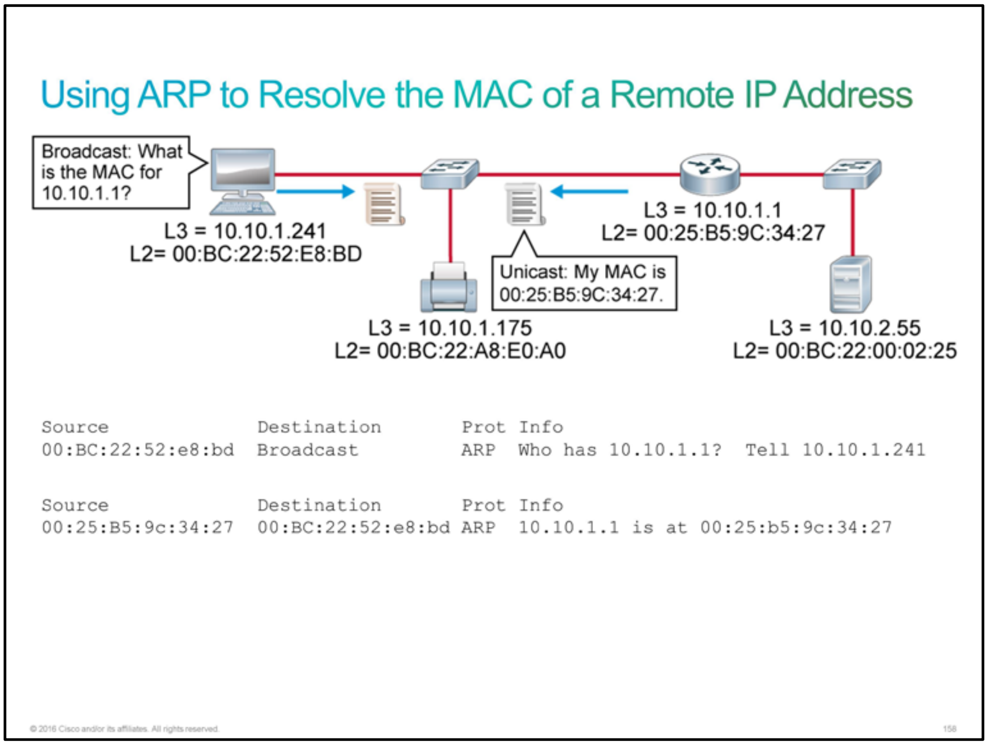
Using ARP to Resolve the MAC of a Local IP Address

Broadcast: What is the MAC for 10.10.1.175?
L3 = 10.10.1.241
L2 = 00:BC:22:52:E8:BD
Broadcast
Unicast
Unicast: My MAC is 00:BC:22:A8:E0:A0.
L3 = 10.10.1.175
L2 = 00:BC:22:A8:E0:A0
© 2016 Cisco and/or its affiliates. All rights reserved.
157

```
Source              Destination       Prot Info
00:BC:22:52:e8:bd   Broadcast         ARP  Who has 10.10.1.175?  Tell
10.10.1.241

Source              Destination       Prot Info
00:BC:22:a8:e0:a0   00:BC:22:52:e8:bd ARP  10.10.1.175 is at 68:b5:99:a8:e0:a0
```

Using ARP to Resolve the MAC of a Remote IP Address

When the IP host 10.10.1.241 wants to communicate with the destination IP host 10.10.2.55, it compares this IP address against its subnet mask and discovers that the host is on a different IP network (10.10.2.0). You know that when a PC wants to send data to a device that is on another network, it sends the data to the default gateway. So the destination MAC address in the frame needs to be the MAC address of the default gateway. In this situation, the IP source must request an ARP for its default gateway. The default gateway is the IP address of the router interface on the local subnet. In the example, IP host 10.10.1.241 sends an ARP broadcast for the MAC address of 10.10.1.1.



Using ARP to Resolve the MAC of a Remote IP Address

```
Source             Destination        Prot Info
00:BC:22:52:e8:bd  Broadcast          ARP  Who has 10.10.1.1?  Tell 10.10.1.241

Source             Destination        Prot Info
00:25:B5:9c:34:27  00:BC:22:52:e8:bd ARP  10.10.1.1 is at 00:25:b5:9c:34:27
```

158

Understanding the ARP Cache

Each IP device on a network segment maintains a table in memory—the ARP table. The purpose of this table is to cache recent IP addresses and MAC address bindings. When a host wants to transmit data to another host on the same network, it searches the ARP table to see if there is an entry. If there is an entry, the host uses it. If there is no entry, the IP host sends an ARP broadcast requesting resolution.

Note By caching recent bindings, ARP broadcasts can be avoided for any mappings in the cache. Without the ARP cache, each IP host would have to send an ARP broadcast each time that it wanted to communicate with another IP host.

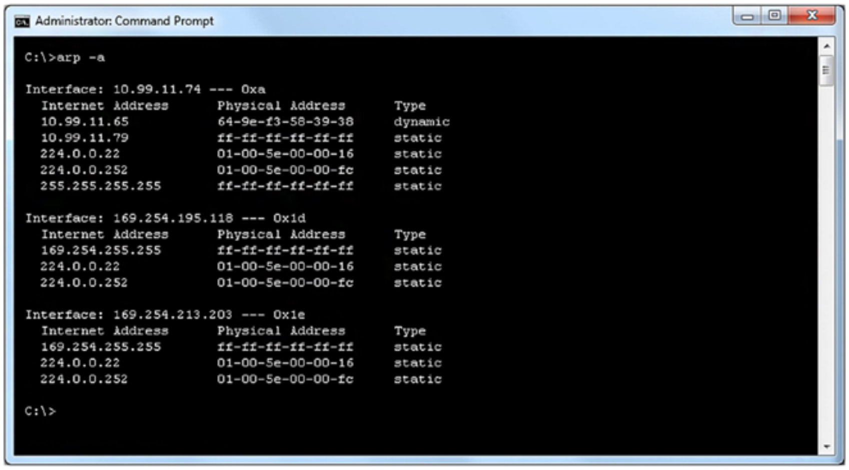
Each entry, or row, of the ARP table has a pair of values—an IP address and a MAC address. The relationship between the two values is a map, which simply means that you can locate an IP address in the table and discover the corresponding MAC address. The ARP table caches the mapping for the devices on the local LAN.

The device creates and maintains the ARP table dynamically. It adds and changes address relationships as they are used on the local host. The entries in an ARP table usually expire after 300 seconds, which is the default value. This short timeout ensures that the table does not contain information for systems that may be switched off or that have been moved. When the local host wants to transmit data again, the entry in the ARP table is regenerated through the ARP process.

If no device responds to the ARP request, the packet is dropped because a frame cannot be created without the destination MAC address.

Understanding the ARP Cache

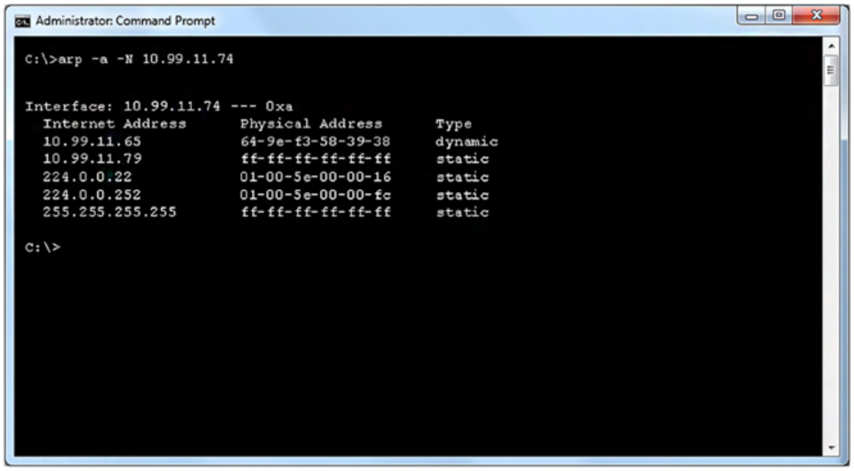
The **arp -a** command displays the current ARP table for all interfaces on a PC using the Microsoft Windows operating system.



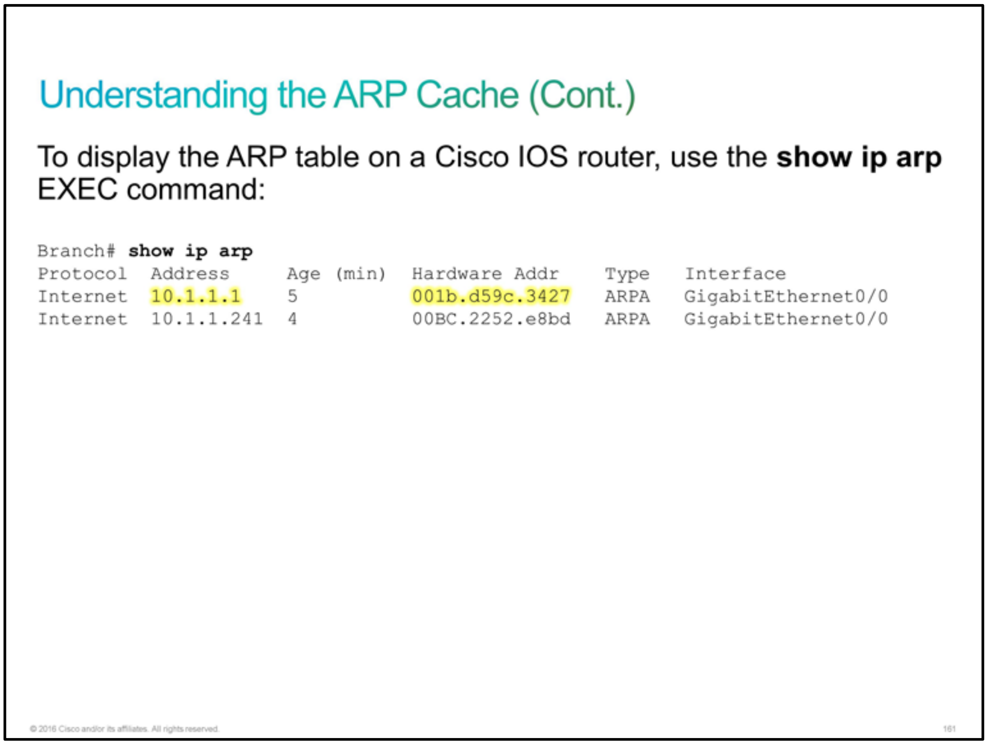
```
C:\>arp -a

Interface: 10.99.11.74 --- 0xa
  Internet Address      Physical Address      Type
  10.99.11.65           64-9e-f3-58-39-38     dynamic
  10.99.11.79           ff-ff-ff-ff-ff-ff     static
  224.0.0.22            01-00-5e-00-00-16     static
  224.0.0.252           01-00-5e-00-00-fc     static
  255.255.255.255       ff-ff-ff-ff-ff-ff     static

Interface: 169.254.195.118 --- 0x1d
  Internet Address      Physical Address      Type
  169.254.255.255       ff-ff-ff-ff-ff-ff     static
  224.0.0.22            01-00-5e-00-00-16     static
  224.0.0.252           01-00-5e-00-00-fc     static

Interface: 169.254.213.203 --- 0x1e
  Internet Address      Physical Address      Type
  169.254.255.255       ff-ff-ff-ff-ff-ff     static
  224.0.0.22            01-00-5e-00-00-16     static
  224.0.0.252           01-00-5e-00-00-fc     static

C:\>
```

Understanding the ARP Cache (Cont.)

To limit the output of the **arp** command to a single interface, use the **arp -a -N** *ip_address* command.



```
C:\>arp -a -N 10.99.11.74

Interface: 10.99.11.74 --- 0xa
  Internet Address      Physical Address      Type
  10.99.11.65           64-9e-f3-58-39-38     dynamic
  10.99.11.79           ff-ff-ff-ff-ff-ff     static
  224.0.0.22            01-00-5e-00-00-16     static
  224.0.0.252           01-00-5e-00-00-fc     static
  255.255.255.255       ff-ff-ff-ff-ff-ff     static

C:\>
```



Understanding the ARP Cache (Cont.)

To display the ARP table on a Cisco IOS router, use the **show ip arp** EXEC command:

```
Branch# show ip arp
Protocol  Address      Age (min)  Hardware Addr    Type   Interface
Internet  10.1.1.1     5          001b.d59c.3427   ARPA   GigabitEthernet0/0
Internet  10.1.1.241   4          00BC.2252.e8bd   ARPA   GigabitEthernet0/0
```

The proper syntax to display the ARP table is **show ip arp** [*ip-address*] [*host-name*] [*mac-address*] [*interface type number*].

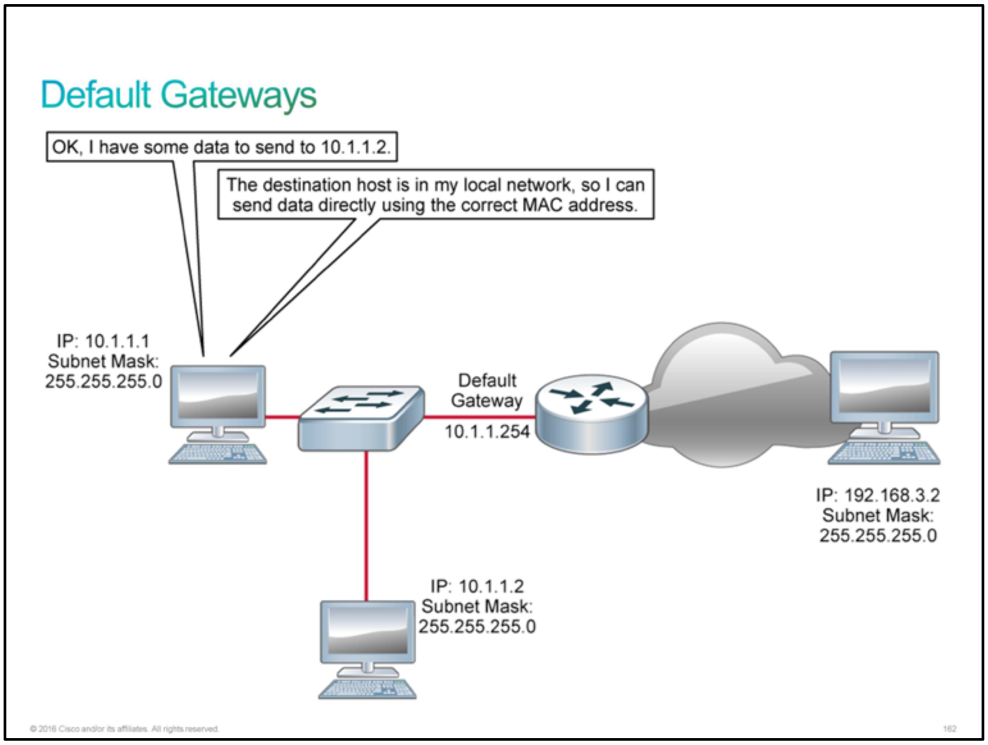
Syntax Description

Parameter	Description
ip-address	(Optional) Displays ARP entries matching this IP address
host-name	(Optional) Hostname
mac-address	(Optional) 48-bit MAC address
interface type number	(Optional) Displays ARP entries that are learned via this interface type and number

Address Resolution Protocol

Default Gateways

The source host is able to communicate directly with the destination host only if the two hosts are on the same network. If the two hosts are on different networks, the sending host must send the data to the default gateway, which will forward the data to the destination.



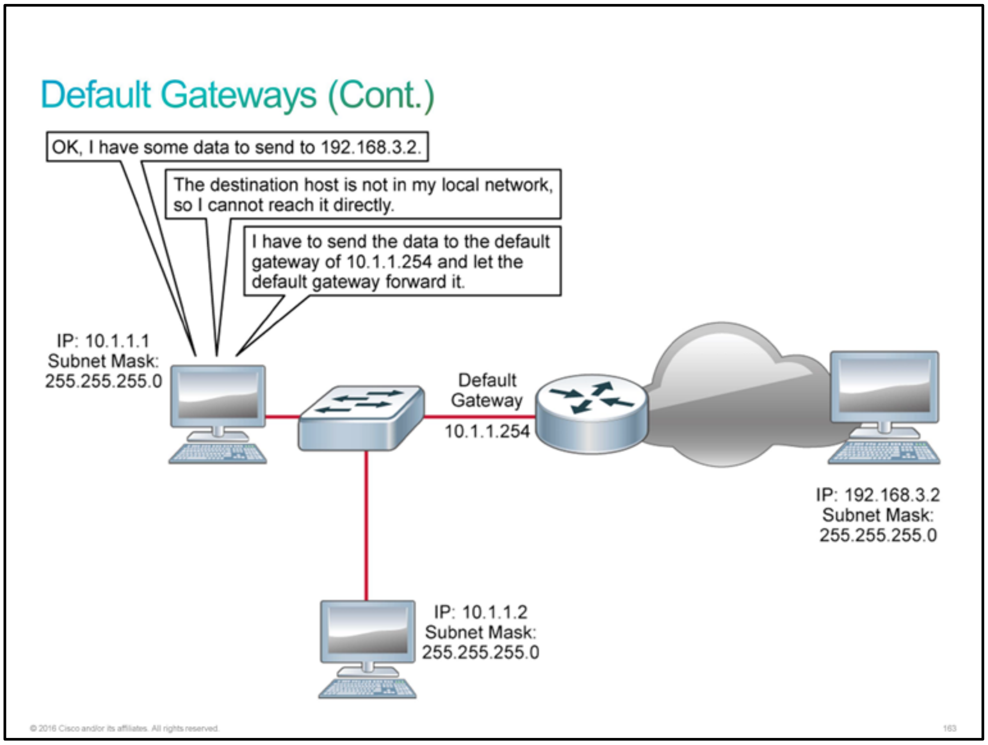
Default Gateways

OK, I have some data to send to 10.1.1.2.

The destination host is in my local network, so I can send data directly using the correct MAC address.

© 2016 Cisco and/or its affiliates. All rights reserved. 162

Before an end system can send a packet to its destination, it must first determine if the destination address is in the local network. The subnet mask defines the network part of the IP address. The end system compares the network portion of the local network address with the destination network address of the packet to be sent. If the network portion of the local network address is the same as the destination network address, the end system can deliver packets directly. If the network portion of the local network address is not the same as the destination network address, the packets must be forwarded to some other network.



Default Gateways (Cont.)

OK, I have some data to send to 192.168.3.2.
The destination host is not in my local network, so I cannot reach it directly.
I have to send the data to the default gateway of 10.1.1.254 and let the default gateway forward it.
IP: 10.1.1.1
Subnet Mask:
255.255.255.0
Default
Gateway
10.1.1.254
IP: 192.168.3.2
Subnet Mask:
255.255.255.0
IP: 10.1.1.2
Subnet Mask:
255.255.255.0
© 2016 Cisco and/or its affiliates. All rights reserved.
163

The default gateway is needed to send a packet out of the local network. If the network portion of the destination address of the packet is different from the network of the originating host, the packet has to be routed outside of the original network. To do this, the packet is sent to the default gateway. This default gateway is a router interface that is connected to the local network. The default gateway interface has a network layer address that matches the network address of the hosts. The hosts are configured to recognize that address as the default gateway.

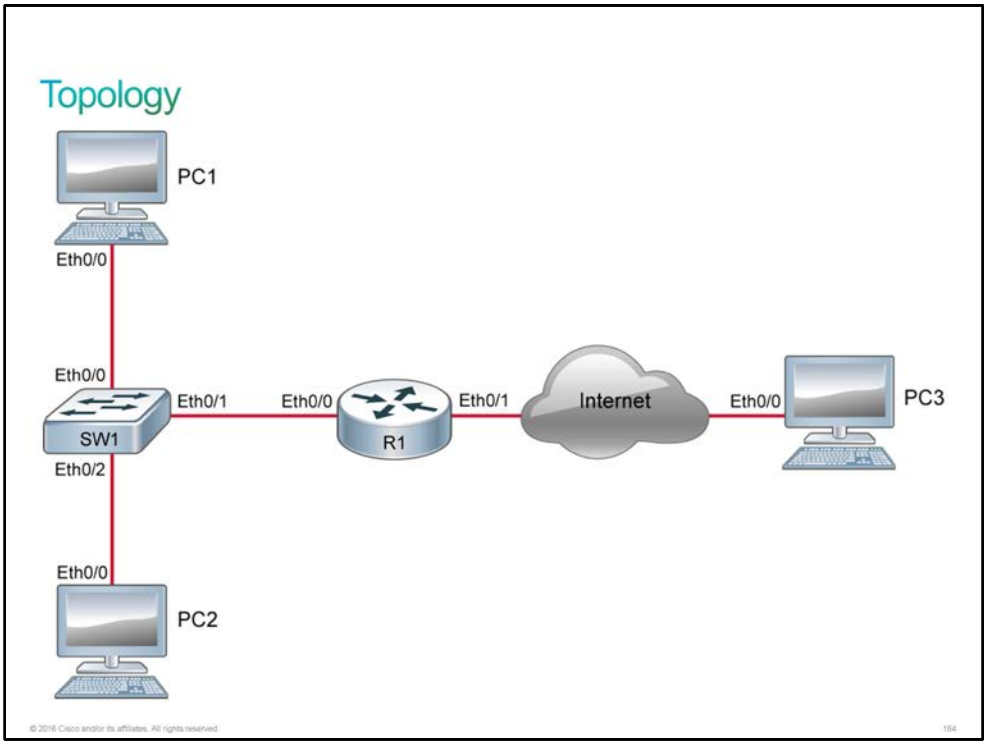
On a Windows computer, the Internet Protocol (TCP/IP) Properties tools are used to enter the default gateway IP address. The host IP address and the default gateway address must have the same network portion of their respective addresses.

Discovery 8: Configure Default Gateway

Introduction

This discovery lab will help you explore how ARP maps IP addresses to MAC addresses and how default gateways allow access to hosts on remote subnets. The lab is prepared with the devices represented in the topology diagram with the IP addresses as depicted in the table. Note that PC1, PC2, PC3, SW1, and R1 are fully configured.

Topology



Job Aid

Device Information

Device Information Table

Device	Characteristic	Value
PC1	Hostname	PC1
PC1	IP address	10.10.1.10/24
PC1	Default gateway	Not configured
PC2	Hostname	PC2

Device	Characteristic	Value
PC2	IP address	10.10.1.20/24
PC2	Default gateway	10.10.1.1
PC3	Hostname	PC3
PC3	IP address	192.168.3.2/24
PC3	Default gateway	192.168.3.1
SW1	Hostname	SW1
SW1	VLAN 1 IP address	10.10.1.2/24
SW1	Default gateway	10.10.1.1
SW1	Ethernet0/0 description	Link to PC1
SW1	Ethernet0/1 description	Link to R1
SW1	Ethernet0/2 description	Link to PC2
R1	Hostname	R1
R1	Ethernet0/0 description	Link to SW1
R1	Ethernet0/0 IP address	10.10.1.1/24
R1	Ethernet0/1 IP address	192.168.3.1/24
R1	Loopback 0 IP	10.10.3.1/24

PCs in the virtual lab environment are simulated as routers, so you should use Cisco IOS commands to configure them or make verifications.

Task 1: Configure Default Gateway

Activity

Step 1 All networked devices with IP addresses maintain an ARP cache. Via the ARP process, devices learn the MAC address of other hosts on their local subnet for which they need to communicate. Access the console of PC1 and execute the **show arp** command.

PC1 should have an entry for itself (10.10.1.10).

```
PC1# show arp
Protocol  Address          Age (min)  Hardware Addr   Type   Interface
Internet  10.10.1.10              -   aabb.cc00.2200  ARPA   Ethernet0/0
```

If there was activity within the discovery before executing the **show arp** command, you may find that there are other entries in the table.

The command **show arp** does not work on PCs. It is used here because the actual device that is used to simulate a PC is a router.

MAC addresses in your output may be different.

Step 2 To initiate communication between PC1 and other devices on the subnet, which will initiate the ARP process to learn the appropriate MAC addresses, use the **ping** command. Ping PC2 (10.10.1.20), R1 (10.10.1.1) and SW1 (10.10.1.2).

Sometimes the first ping times out due to the delay that the ARP process caused.

```
PC1# ping 10.10.1.20
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.1.20, timeout is 2 seconds:
.!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 1/1/1 ms
PC1# ping 10.10.1.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.1.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/201/1003 ms
PC1# ping 10.10.1.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.1.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/203/1004 ms
```

Step 3 Examine the ARP cache on PC1 again.

The ARP cache is now populated with all four hosts that have IP addresses on the 10.10.1.0/24 subnet.

```
PC1# show arp
Protocol  Address          Age (min)  Hardware Addr   Type   Interface
Internet  10.10.1.1               2   aabb.cc00.2100  ARPA   Ethernet0/0
Internet  10.10.1.2               0   aabb.cc80.2a00  ARPA   Ethernet0/0
Internet  10.10.1.10              -   aabb.cc00.2200  ARPA   Ethernet0/0
Internet  10.10.1.20              2   aabb.cc00.2800  ARPA   Ethernet0/0
```

Step 4 From PC1, ping 192.168.3.2, which is a PC on a different subnet.

From PC1, ping PC3:

```
PC1# ping 192.168.3.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.3.2, timeout is 2 seconds:
.!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 1/1/1 ms
```

View the ARP cache on PC1.

```
PC1# show arp
Protocol  Address          Age (min)  Hardware Addr   Type   Interface
Internet  10.10.1.1               1   aabb.cc00.2100  ARPA   Ethernet0/0
Internet  10.10.1.2               0   aabb.cc80.2a00  ARPA   Ethernet0/0
Internet  10.10.1.10              -   aabb.cc00.2200  ARPA   Ethernet0/0
Internet  10.10.1.20              1   aabb.cc00.2800  ARPA   Ethernet0/0
Internet  192.168.3.2             0   aabb.cc00.2100  ARPA   Ethernet0/0
```

There is an ARP cache entry for 192.168.3.2. The MAC address for 192.168.3.2 and 10.10.1.1 are identical. This behavior is the result of the Proxy ARP feature which is enabled on IOS routers by default. PC1 does not have a default gateway that is configured, so it attempts to ARP for all addresses. R1 saw the ARP request for a remote address which was available in its routing table, and sent an ARP reply with its own MAC address. PC1 can then forward traffic that is destined to 192.168.3.2 to R1’s MAC address and R1 will forward as necessary. While Proxy ARP can be helpful as a last resort, properly configuring a default gateway is a better practice.

Step 5 Verify that PC1 does not have a default route in its routing table.

On PC1, enter the following command:

```
PC1# show ip route
Default gateway is not set

Host               Gateway           Last Use    Total Uses  Interface
ICMP redirect cache is empty
```

Step 6 Configure R1 as the default gateway for PC1.

On PC1, enter the following command:

```
PC1# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
PC1(config)# ip default-gateway 10.10.1.1
PC1(config)# end
PC1#
```

Verify the routing table on PC1 again:

```
PC1# show ip route
Default gateway is 10.10.1.1

Host               Gateway           Last Use    Total Uses  Interface
ICMP redirect cache is empty
```

Step 7 Remove the entry for 192.168.3.2 from the ARP cache of PC1 and verify that the entry has been removed.

On PC1, enter the following commands:

```
PC1# clear ip arp 192.168.3.2
PC1# show arp
Protocol  Address          Age (min)  Hardware Addr   Type   Interface
Internet  10.10.1.1               6   aabb.cc00.2100  ARPA   Ethernet0/0
Internet  10.10.1.2               6   aabb.cc80.2a00  ARPA   Ethernet0/0
Internet  10.10.1.10              -   aabb.cc00.2200  ARPA   Ethernet0/0
Internet  10.10.1.20              6   aabb.cc00.2800  ARPA   Ethernet0/0
```

Step 8 Ping 192.168.3.2 and verify that there is no entry for 192.168.3.2 in the ARP cache of PC1.

On PC1, enter the following commands:

```
PC1# ping 192.168.3.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.3.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
PC1# show arp
Protocol  Address          Age (min)  Hardware Addr   Type   Interface
Internet  10.10.1.1               8   aabb.cc00.2100  ARPA   Ethernet0/0
Internet  10.10.1.2               8   aabb.cc80.2a00  ARPA   Ethernet0/0
Internet  10.10.1.10              -   aabb.cc00.2200  ARPA   Ethernet0/0
Internet  10.10.1.20              8   aabb.cc00.2800  ARPA   Ethernet0/0
```

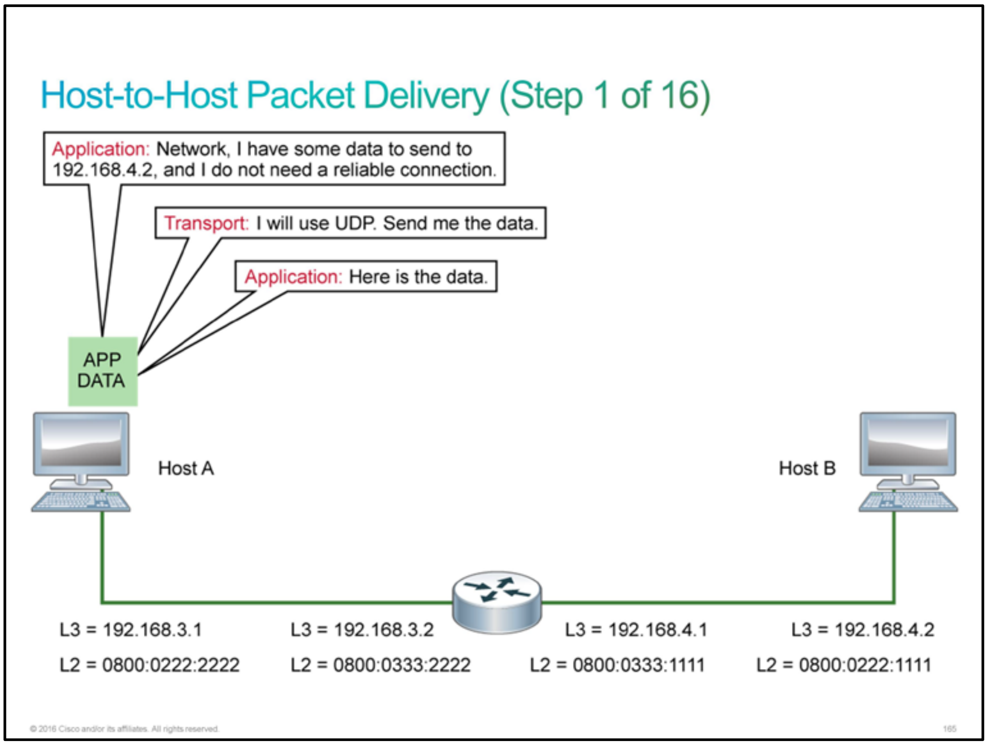
For all addresses outside of the 10.10.1.0/24 subnet, PC1 will use the destination MAC address of 10.10.1.1 (R1, its default gateway). R1 will then forward the packet appropriately due to its routing table.

This is the end of the discovery lab.

Host-to-Host Packet Delivery

Host-to-host packet delivery consists of an interesting series of processes. In this multipart example, you will discover what happens "behind the scenes" when an IP host communicates with another IP host. The IP host 192.168.3.1 needs to send arbitrary application data to the IP host 192.168.4.2, which is located on another subnet.

Host-to-Host Packet Delivery (Step 1 of 16)



Host-to-Host Packet Delivery (Step 1 of 16)

Application: Network, I have some data to send to 192.168.4.2, and I do not need a reliable connection.

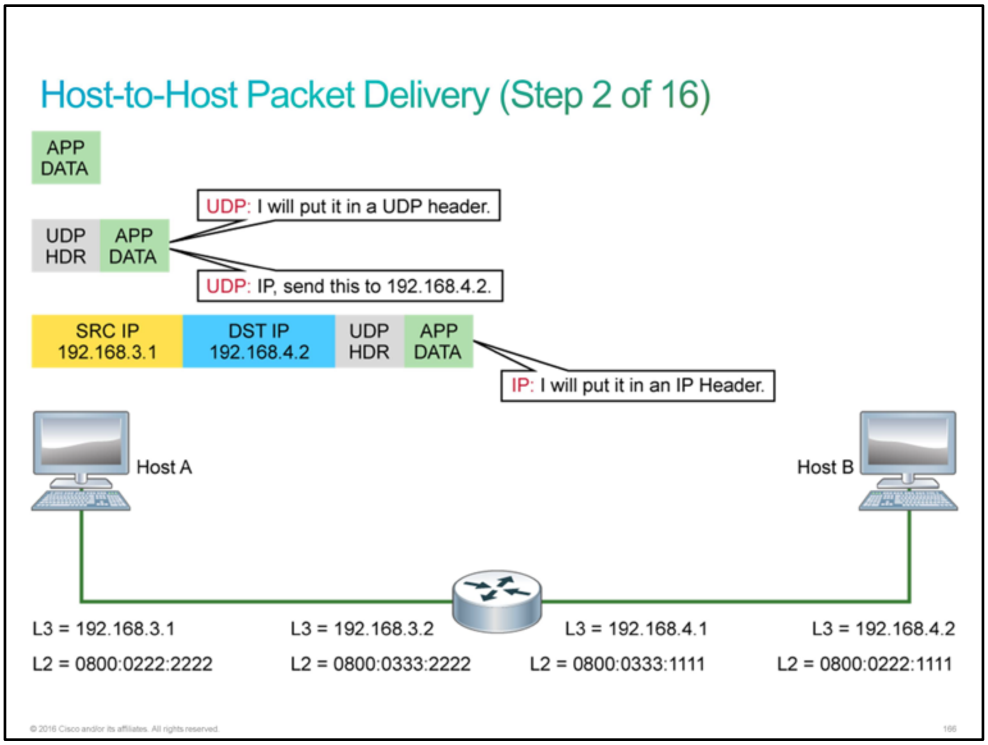
Transport: I will use UDP. Send me the data.

Application: Here is the data.

165

In this example, the host 192.168.3.1 has data that it wants to send to the host 192.168.4.2. The application does not need a reliable connection, so it uses UDP. Because it is not necessary to set up a session, the application can start sending data.

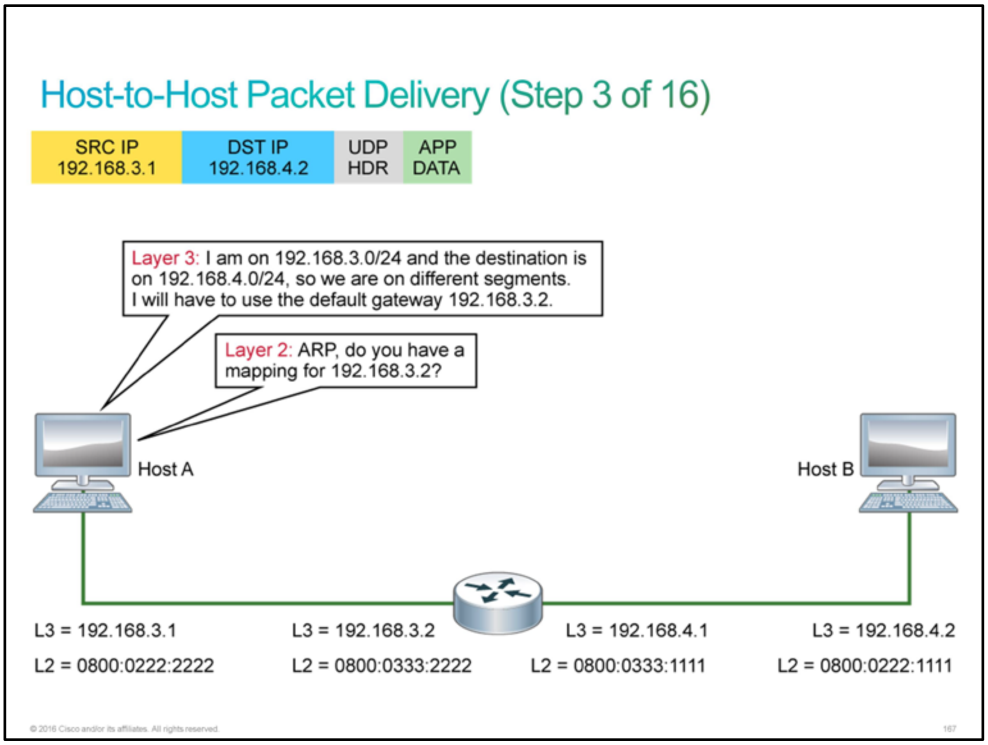
Host-to-Host Packet Delivery (Step 2 of 16)



Host-to-Host Packet Delivery (Step 2 of 16)

UDP prepends a UDP header (UDP HDR) and passes the PDU to the IP (Layer 3) with an instruction to send the PDU to 192.168.4.2. IP encapsulates the PDU in a Layer 3 packet, setting the source IP address (SRC IP) of the packet to 192.168.3.1, while the destination IP address (DST IP) is set to 192.168.4.2.

Host-to-Host Packet Delivery (Step 3 of 16)



Host-to-Host Packet Delivery (Step 3 of 16)

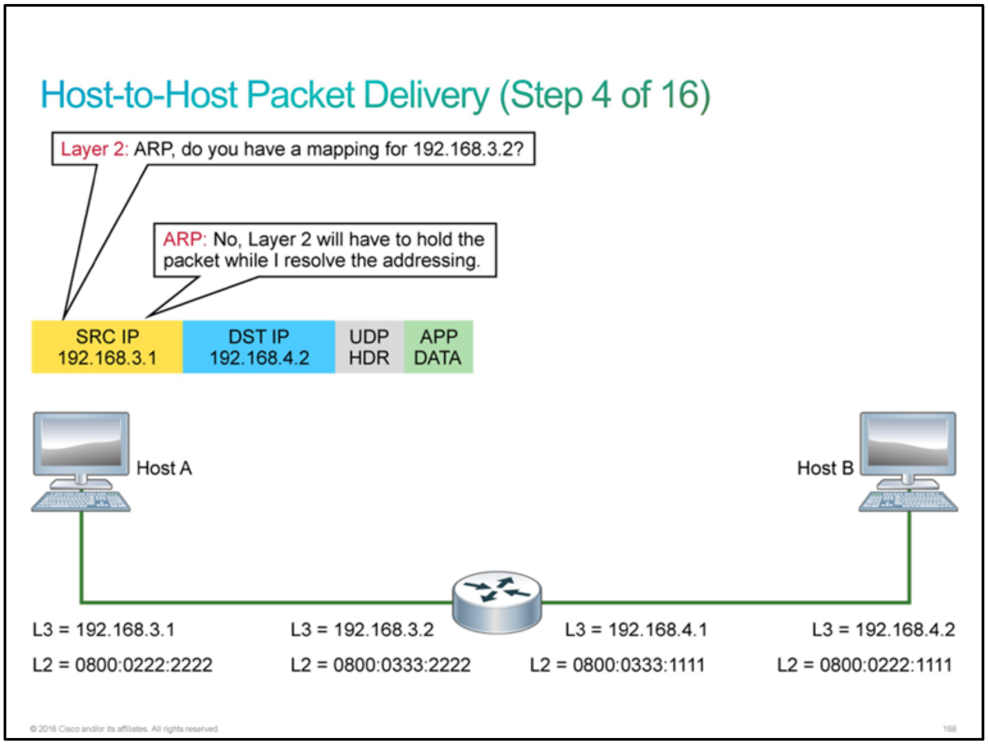
SRC IP 192.168.3.1	DST IP 192.168.4.2	UDP HDR	APP DATA

Layer 3: I am on 192.168.3.0/24 and the destination is on 192.168.4.0/24, so we are on different segments. I will have to use the default gateway 192.168.3.2.

Layer 2: ARP, do you have a mapping for 192.168.3.2?

When Host A analyzes the destination address, it finds that the destination address is on a different network. The host sends any packet that is not destined for the local IP network to the default gateway. The default gateway is the address of the local router, which must be configured on hosts (PCs, servers, and so on). IP encapsulates the PDU in a Layer 3 packet and passes it to Layer 2 with instructions to forward it to the default gateway. Host A must place the packet in its parking lot until it can obtain the needed information that is related to the default network.

Host-to-Host Packet Delivery (Step 4 of 16)



Host-to-Host Packet Delivery (Step 4 of 16)

Layer 2: ARP, do you have a mapping for 192.168.3.2?

ARP: No, Layer 2 will have to hold the packet while I resolve the addressing.

SRC IP 192.168.3.1	DST IP 192.168.4.2	UDP HDR	APP DATA

Host A

Host B

L3 = 192.168.3.1
L2 = 0800:0222:2222

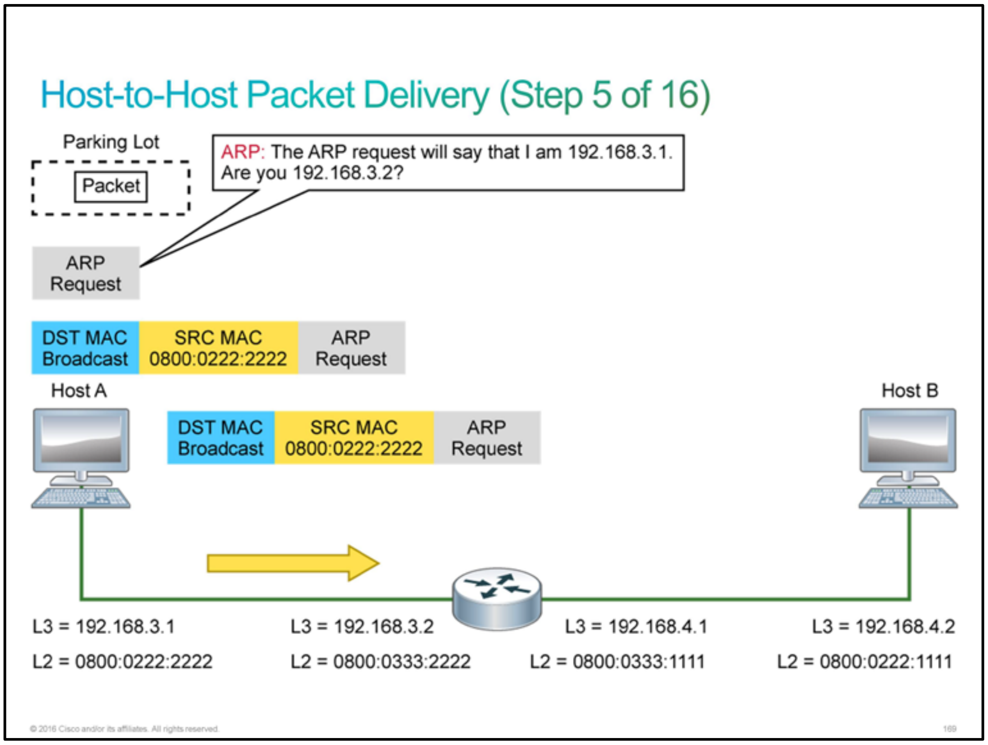
L3 = 192.168.3.2
L2 = 0800:0333:2222

L3 = 192.168.4.1
L2 = 0800:0333:1111

L3 = 192.168.4.2
L2 = 0800:0222:1111

To deliver the packet, the host needs the Layer 2 information of the next-hop device. The ARP table in the host does not have an entry and must resolve the Layer 2 address (MAC address) of the default gateway. The default gateway is the next hop for the packet. The packet waits while the host resolves the Layer 2 information.

Host-to-Host Packet Delivery (Step 5 of 16)

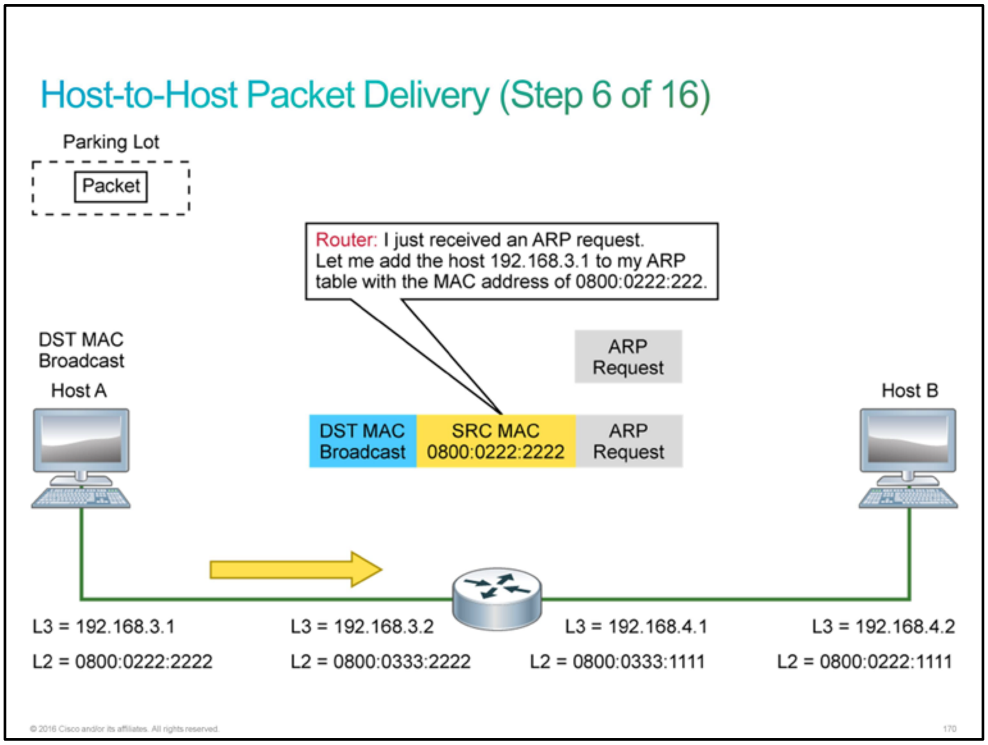


Host-to-Host Packet Delivery (Step 5 of 16)

ARP: The ARP request will say that I am 192.168.3.1. Are you 192.168.3.2?

Because the host does not have a mapping of Layer 2 and Layer 3 addresses for the default gateway, the host uses the standard ARP process to obtain the mapping. The host sends an ARP request to the router.

Host-to-Host Packet Delivery (Step 6 of 16)



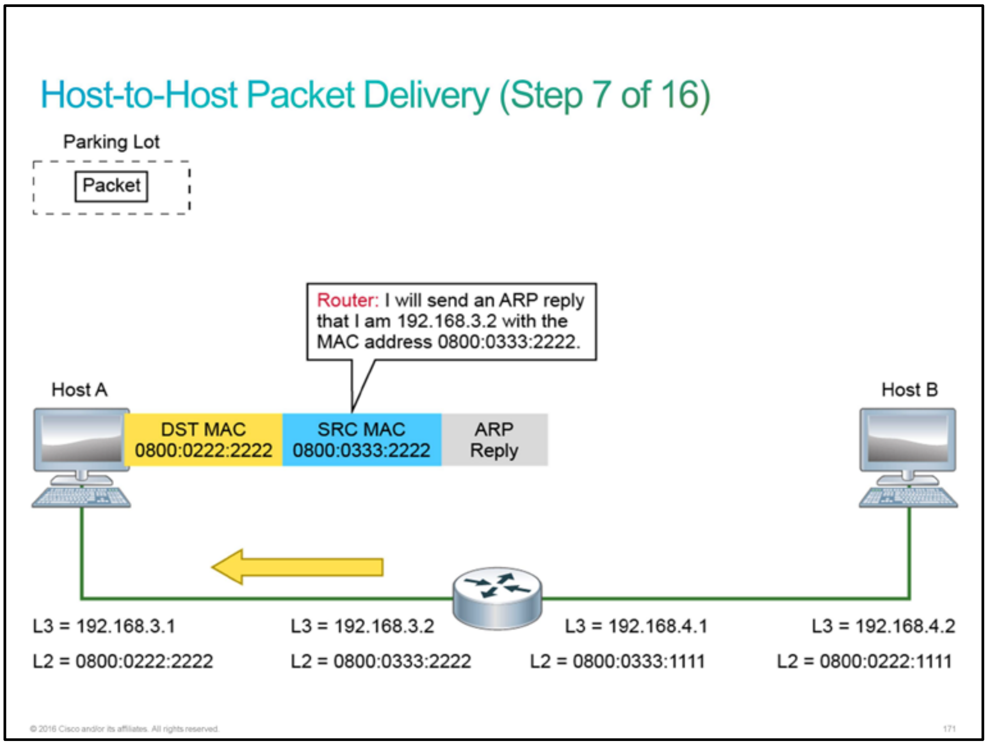
Host-to-Host Packet Delivery (Step 6 of 16)

Router: I just received an ARP request. Let me add the host 192.168.3.1 to my ARP table with the MAC address of 0800:0222:222.

170

The user has programmed the IP address of 192.168.3.2 as the default gateway. The host 192.168.3.1 sends out the ARP request, and the router receives it. The ARP request contains information about the host, and the router adds the information to its ARP table.

Host-to-Host Packet Delivery (Step 7 of 16)

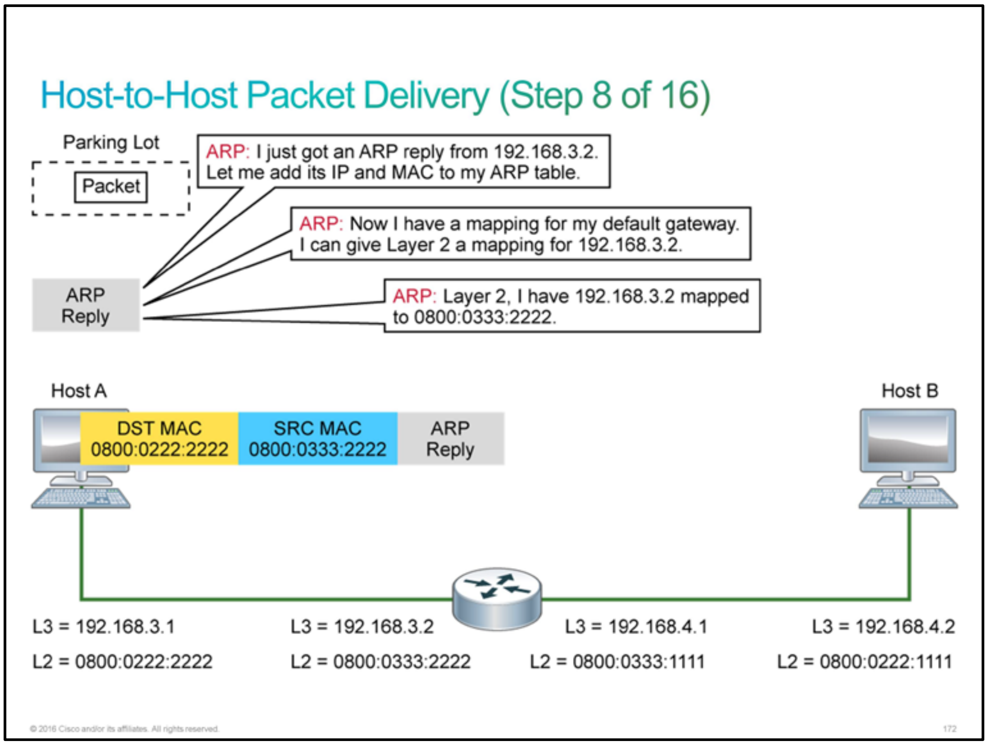


Host-to-Host Packet Delivery (Step 7 of 16)

Router: I will send an ARP reply that I am 192.168.3.2 with the MAC address 0800:0333:2222.

The router processes the ARP request like any other host and sends the ARP reply with its own information.

Host-to-Host Packet Delivery (Step 8 of 16)



Host-to-Host Packet Delivery (Step 8 of 16)

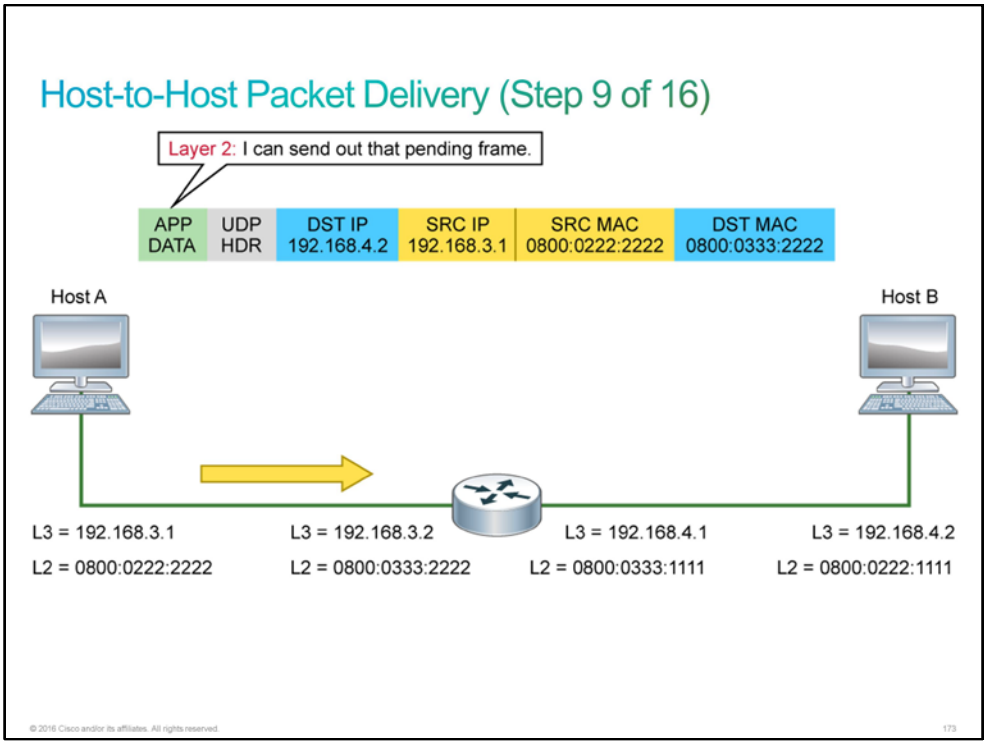
ARP: I just got an ARP reply from 192.168.3.2. Let me add its IP and MAC to my ARP table.

ARP: Now I have a mapping for my default gateway. I can give Layer 2 a mapping for 192.168.3.2.

ARP: Layer 2, I have 192.168.3.2 mapped to 0800:0333:2222.

The host receives an ARP reply to the ARP request and enters the information to its local ARP table.

Host-to-Host Packet Delivery (Step 9 of 16)

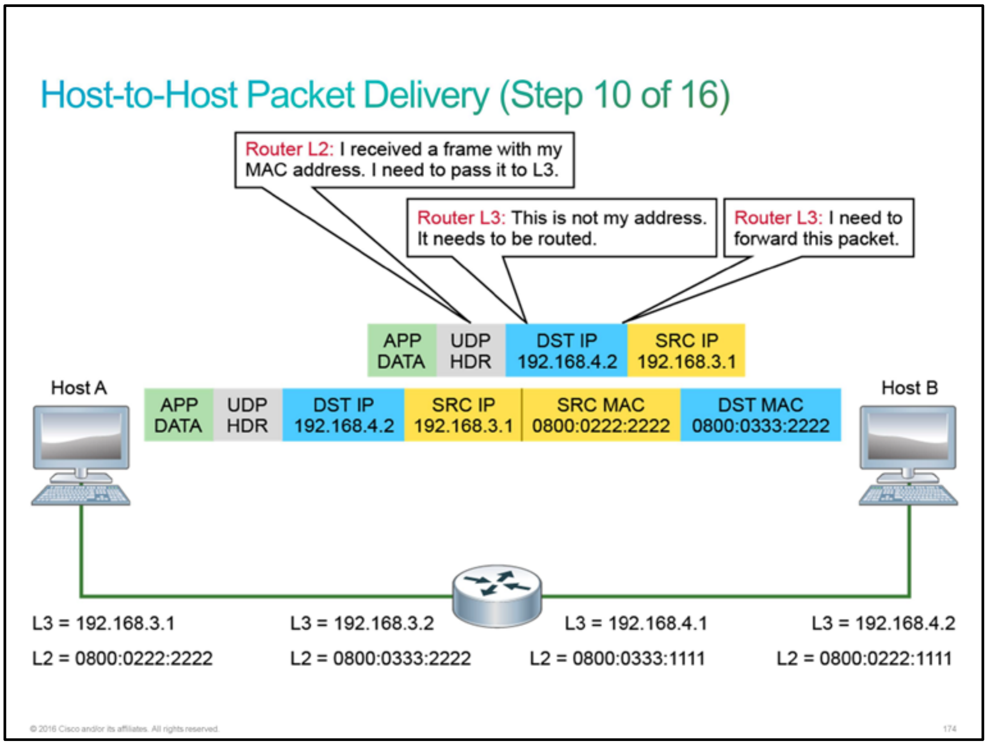


Host-to-Host Packet Delivery (Step 9 of 16)

Layer 2: I can send out that pending frame.

Now the Layer 2 frame with the application data can be sent to the default gateway. The pending frame is sent with the local host IP address and MAC address as the source. However, the destination IP address is that of the remote host, but the destination MAC address is that of the default gateway.

Host-to-Host Packet Delivery (Step 10 of 16)



Host-to-Host Packet Delivery (Step 10 of 16)

Router L2: I received a frame with my MAC address. I need to pass it to L3.

Router L3: This is not my address. It needs to be routed.

Router L3: I need to forward this packet.

APP DATA	UDP HDR	DST IP 192.168.4.2	SRC IP 192.168.3.1

APP DATA	UDP HDR	DST IP 192.168.4.2	SRC IP 192.168.3.1	SRC MAC 0800:0222:2222	DST MAC 0800:0333:2222

L3 = 192.168.3.1
L2 = 0800:0222:2222

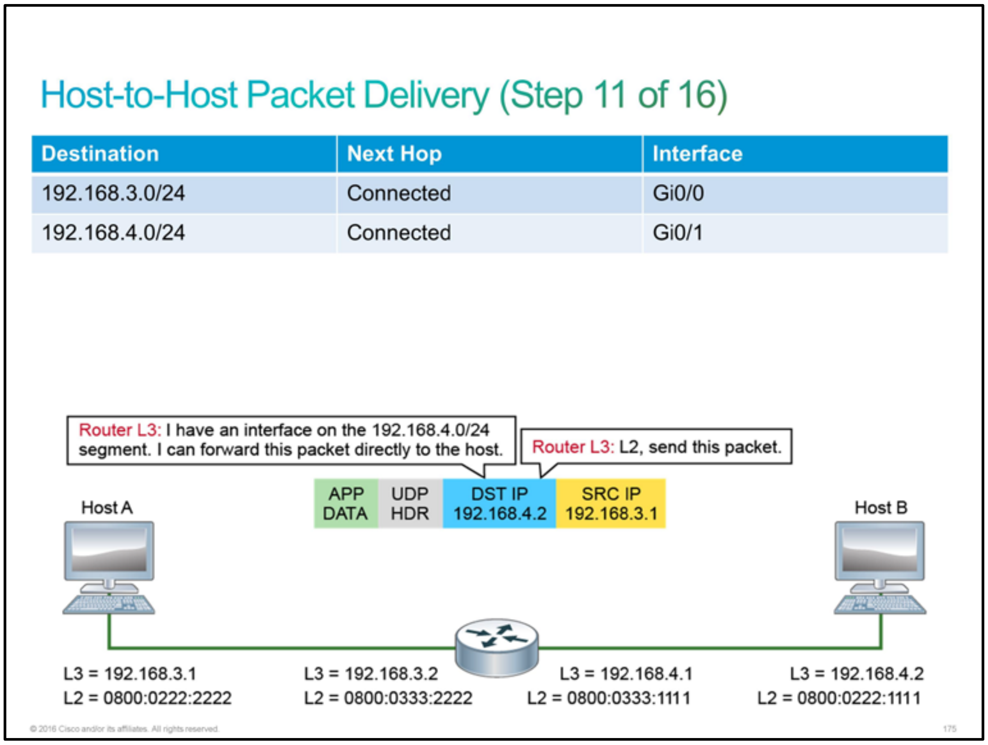
L3 = 192.168.3.2
L2 = 0800:0333:2222

L3 = 192.168.4.1
L2 = 0800:0333:1111

L3 = 192.168.4.2
L2 = 0800:0222:1111

When the router receives the frame, the router recognizes its MAC address and processes the frame. At Layer 3, the router sees that the destination IP address is not its address. A host Layer 3 device would discard the frame. However, because this device is a router, it passes all packets that are for unknown destinations to the routing process. The routing process determines where to send the packet.

Host-to-Host Packet Delivery (Step 11 of 16)



Host-to-Host Packet Delivery (Step 11 of 16)

Destination	Next Hop	Interface
192.168.3.0/24	Connected	Gi0/0
192.168.4.0/24	Connected	Gi0/1

Router L3: I have an interface on the 192.168.4.0/24 segment. I can forward this packet directly to the host.

Router L3: L2, send this packet.

APP DATA	UDP HDR	DST IP 192.168.4.2	SRC IP 192.168.3.1

Host A

Host B

L3 = 192.168.3.1
L2 = 0800:0222:2222

L3 = 192.168.3.2
L2 = 0800:0333:2222

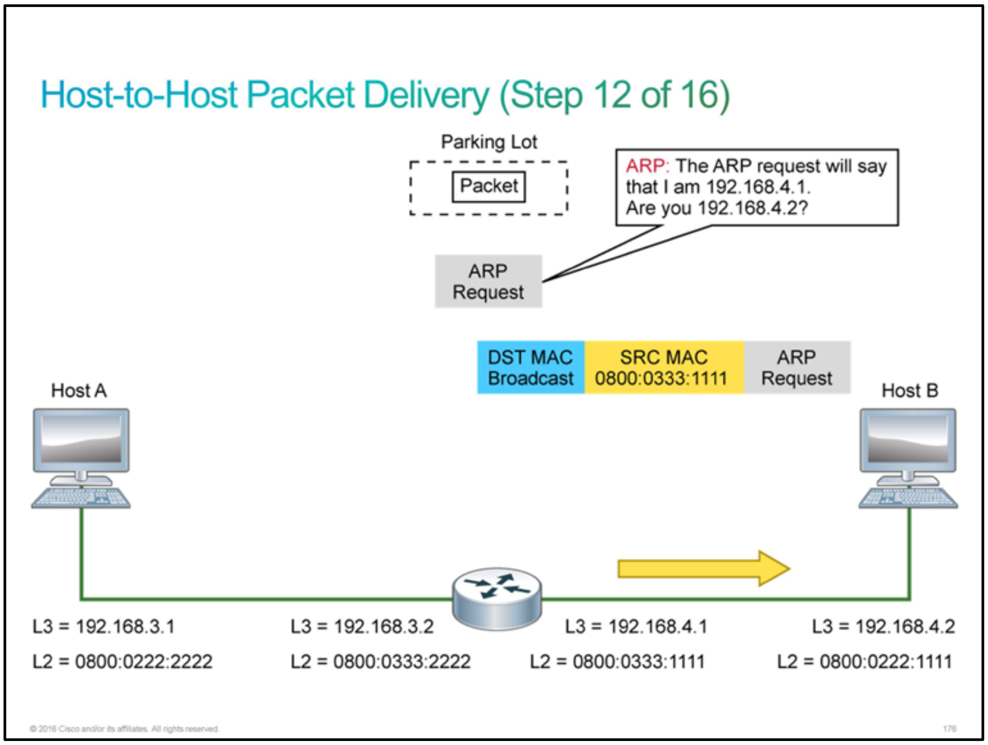
L3 = 192.168.4.1
L2 = 0800:0333:1111

L3 = 192.168.4.2
L2 = 0800:0222:1111

175

The routing process looks up the destination IP address in its routing table. In this example, the destination segment is directly connected. Therefore, the routing process can pass the packet directly to Layer 2 for the appropriate interface.

Host-to-Host Packet Delivery (Step 12 of 16)



Host-to-Host Packet Delivery (Step 12 of 16)

Parking Lot

Packet

ARP: The ARP request will say that I am 192.168.4.1. Are you 192.168.4.2?

ARP Request

DST MAC Broadcast	SRC MAC 0800:0333:1111	ARP Request

Host A

Host B

L3 = 192.168.3.1
L2 = 0800:0222:2222

L3 = 192.168.3.2
L2 = 0800:0333:2222

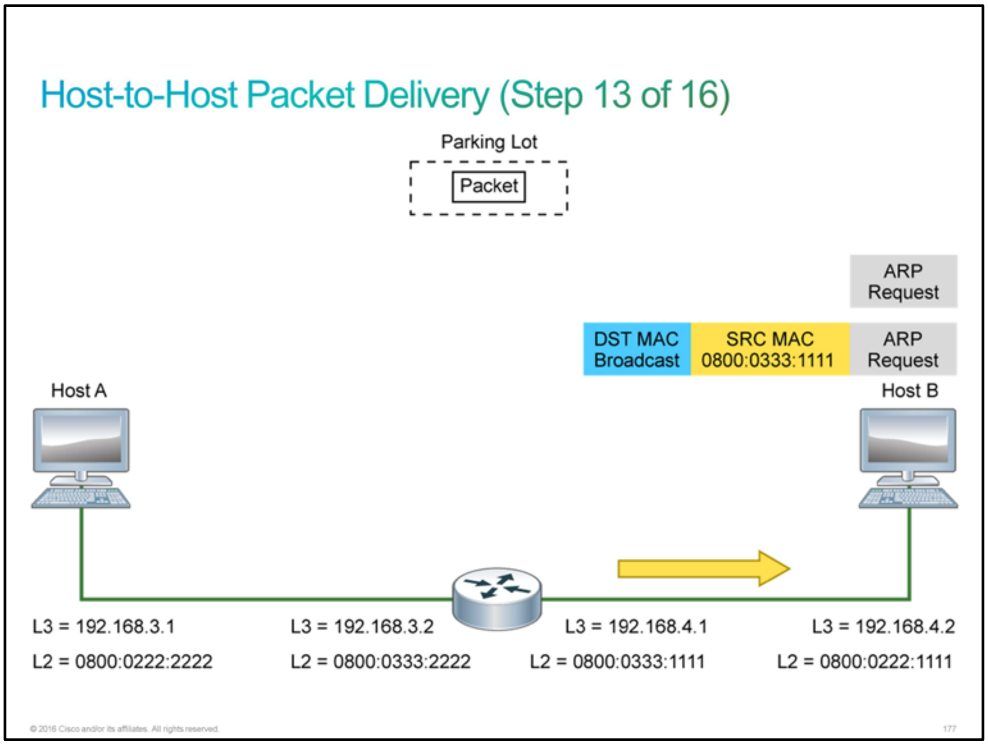
L3 = 192.168.4.1
L2 = 0800:0333:1111

L3 = 192.168.4.2
L2 = 0800:0222:1111

176

Layer 2 uses the ARP process to obtain the mapping for the IP address and the MAC address. The router asks for the Layer 2 information in the same way as the hosts. An ARP request for the destination Layer 3 address is sent to the link.

Host-to-Host Packet Delivery (Step 13 of 16)



Host-to-Host Packet Delivery (Step 13 of 16)

Parking Lot

Packet

ARP Request

DST MAC Broadcast	SRC MAC 0800:0333:1111	ARP Request

Host A

Host B

L3 = 192.168.3.1

L2 = 0800:0222:2222

L3 = 192.168.3.2

L2 = 0800:0333:2222

L3 = 192.168.4.1

L2 = 0800:0333:1111

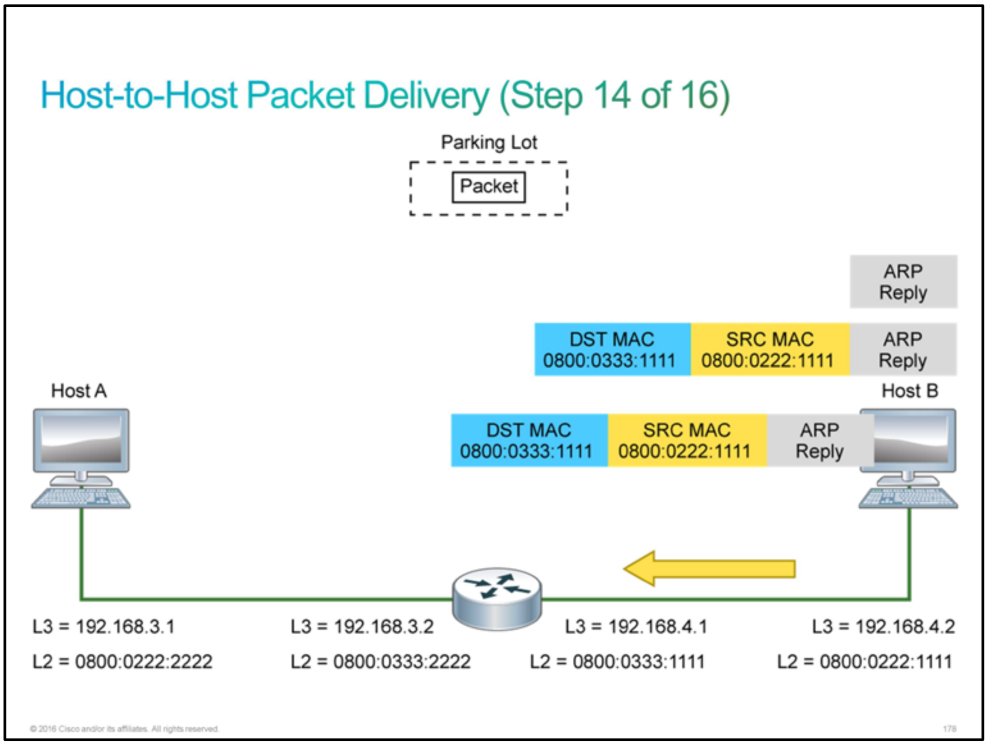
L3 = 192.168.4.2

L2 = 0800:0222:1111

177

The destination receives and processes the ARP request.

Host-to-Host Packet Delivery (Step 14 of 16)



Host-to-Host Packet Delivery (Step 14 of 16)

Parking Lot

Packet

ARP Reply

DST MAC 0800:0333:1111

SRC MAC 0800:0222:1111

ARP Reply

Host A

Host B

DST MAC 0800:0333:1111

SRC MAC 0800:0222:1111

ARP Reply

L3 = 192.168.3.1

L2 = 0800:0222:2222

L3 = 192.168.3.2

L2 = 0800:0333:2222

L3 = 192.168.4.1

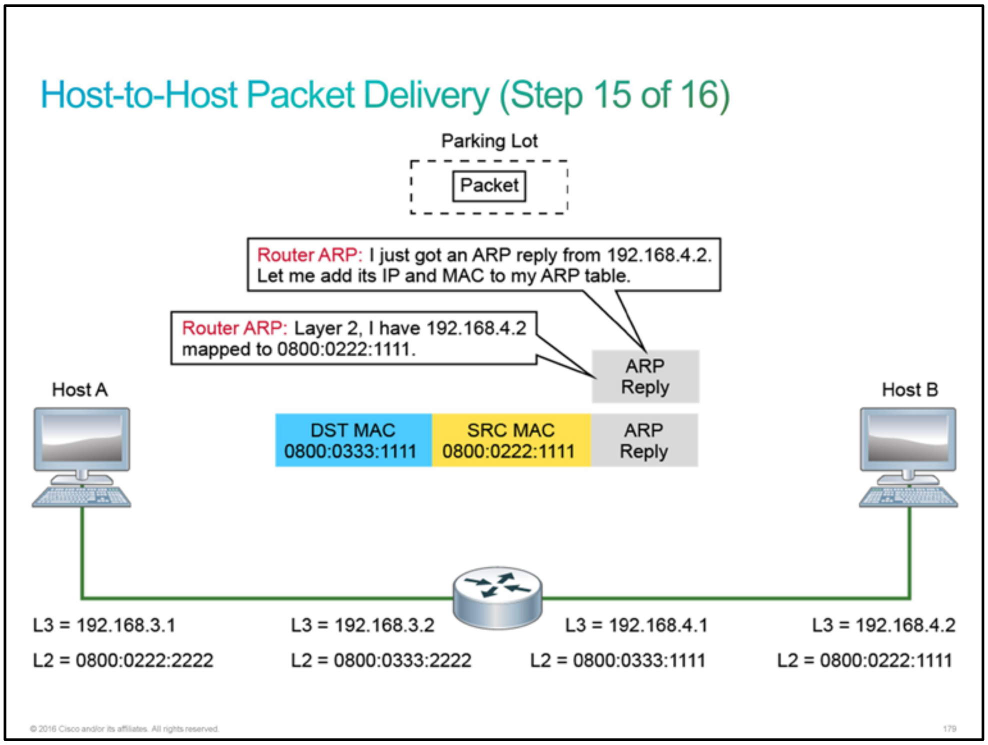
L2 = 0800:0333:1111

L3 = 192.168.4.2

L2 = 0800:0222:1111

The host receives the frame that contains the ARP request and passes the request to the ARP process. The ARP process takes the information about the router from the ARP request and places the information in its local ARP table. The ARP process generates the ARP reply and sends it back to the router.

Host-to-Host Packet Delivery (Step 15 of 16)



Host-to-Host Packet Delivery (Step 15 of 16)

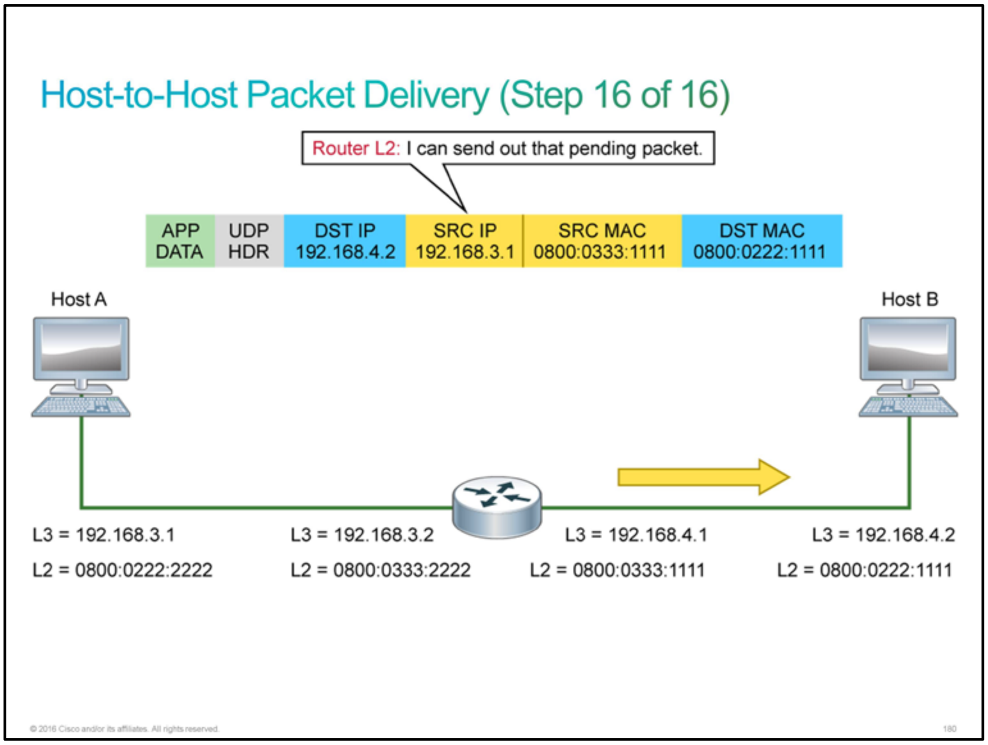
Router ARP: I just got an ARP reply from 192.168.4.2. Let me add its IP and MAC to my ARP table.

Router ARP: Layer 2, I have 192.168.4.2 mapped to 0800:0222:1111.

DST MAC	SRC MAC	ARP Reply
0800:0333:1111	0800:0222:1111	

The router receives the ARP reply and takes the information that is required for forwarding the packet to the next hop. The router populates its local ARP table and starts the packet-forwarding process.

Host-to-Host Packet Delivery (Step 16 of 16)



Host-to-Host Packet Delivery (Step 16 of 16)

Router L2: I can send out that pending packet.

APP DATA	UDP HDR	DST IP 192.168.4.2	SRC IP 192.168.3.1	SRC MAC 0800:0333:1111	DST MAC 0800:0222:1111

The frame is forwarded to the destination. Note that the router changed the Layer 2 address as needed, but it will not change the Layer 3 address.

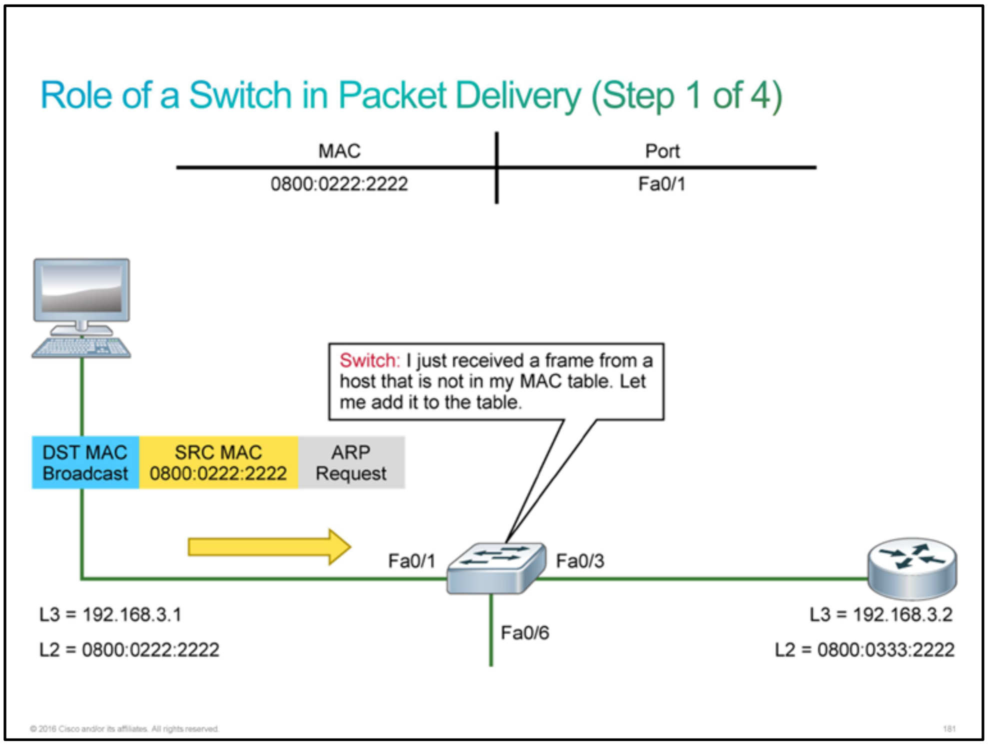
Note The router changes source and destination MAC addresses, while source and destination IP addresses remain the same.

Note Remember that a switch does not change the frame in any way. When the switch receives the frame, it needs to forward it out the proper port according to the MAC address table.

Role of a Switch in Packet Delivery

This example focuses on the role of a switch in the host-to-host packet delivery process. An application on host PC A wishes to send data to a distant network. Before an IP packet can be forwarded to the default gateway, its MAC address needs to be obtained. ARP on PC A creates an ARP request and sends it out. Before the ARP request reaches other devices on a network, the switch receives it.

Role of a Switch in Packet Delivery (Step 1 of 4)



Role of a Switch in Packet Delivery (Step 1 of 4)

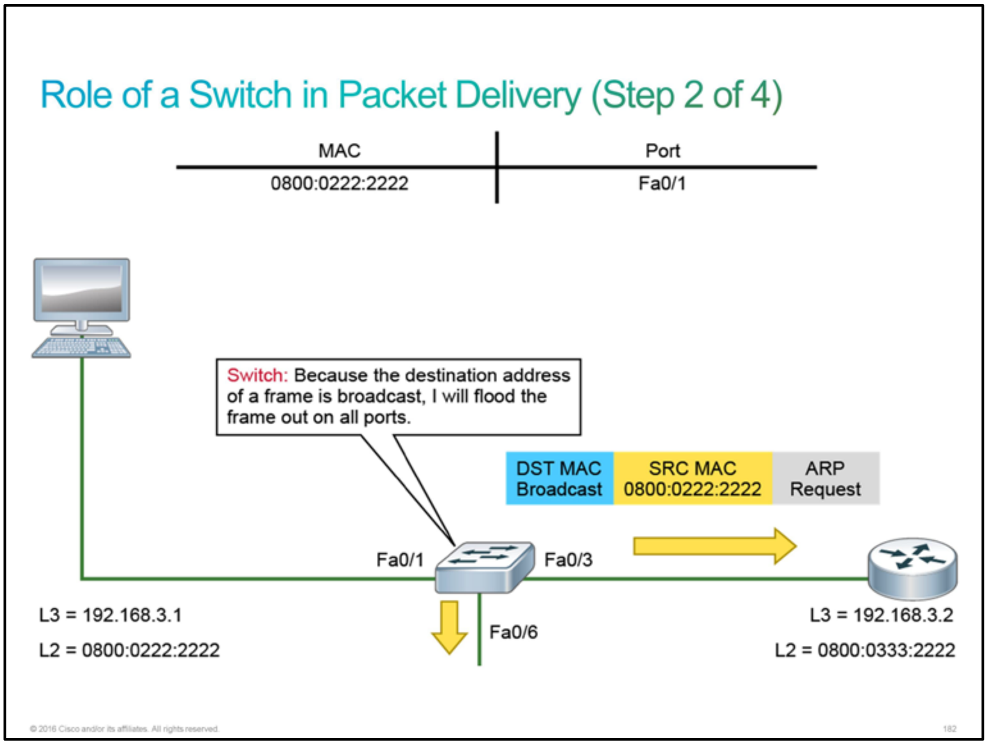
MAC	Port
0800:0222:2222	Fa0/1

Switch: I just received a frame from a host that is not in my MAC table. Let me add it to the table.

DST MAC	SRC MAC	ARP
Broadcast	0800:0222:2222	Request

When the switch receives the frame, it needs to forward it out on the proper port. However, in this example, the source and destination MAC addresses are not in the MAC address table of the switch. The switch can learn the port mapping for the source host from the source MAC address in the frame, so the switch will add it to the table (0800:0222:2222 = port FastEthernet0/1).

Role of Switch in Packet Delivery (Step 2 of 4)



Role of a Switch in Packet Delivery (Step 2 of 4)

MAC	Port
0800:0222:2222	Fa0/1

Switch: Because the destination address of a frame is broadcast, I will flood the frame out on all ports.

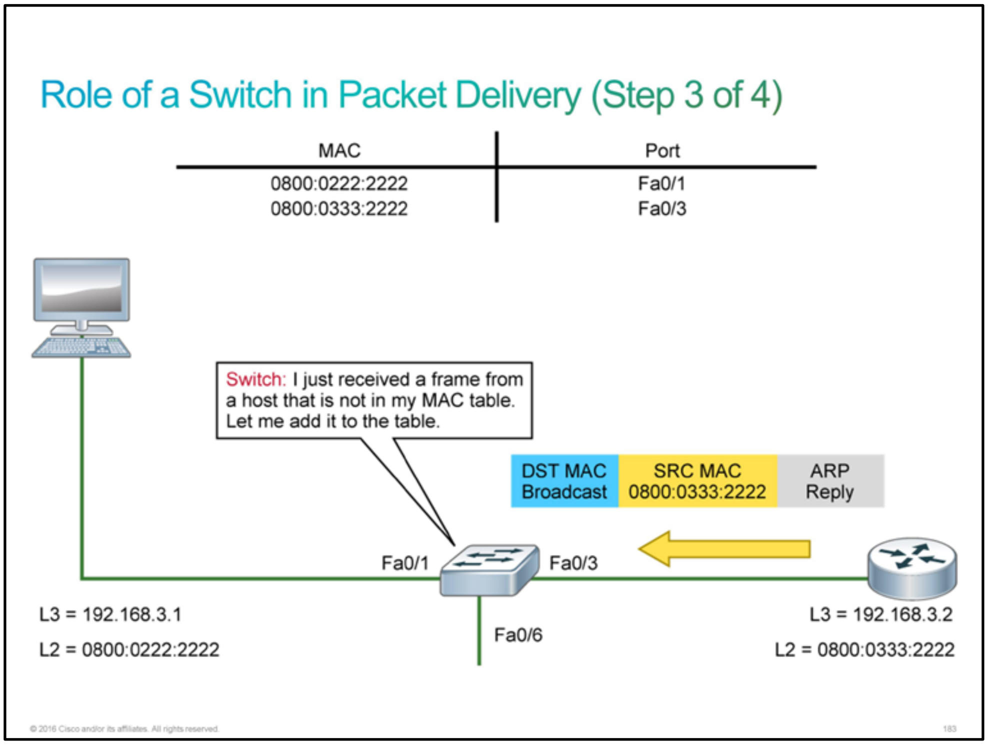
DST MAC Broadcast	SRC MAC 0800:0222:2222	ARP Request

© 2016 Cisco and/or its affiliates. All rights reserved.

182

Because the destination address of the frame is a broadcast, the switch has to flood the packet out to all the ports. The only exception is the port on which the switch received the broadcast frame.

Role of Switch in Packet Delivery (Step 3 of 4)



Role of a Switch in Packet Delivery (Step 3 of 4)

MAC	Port
0800:0222:2222	Fa0/1
0800:0333:2222	Fa0/3

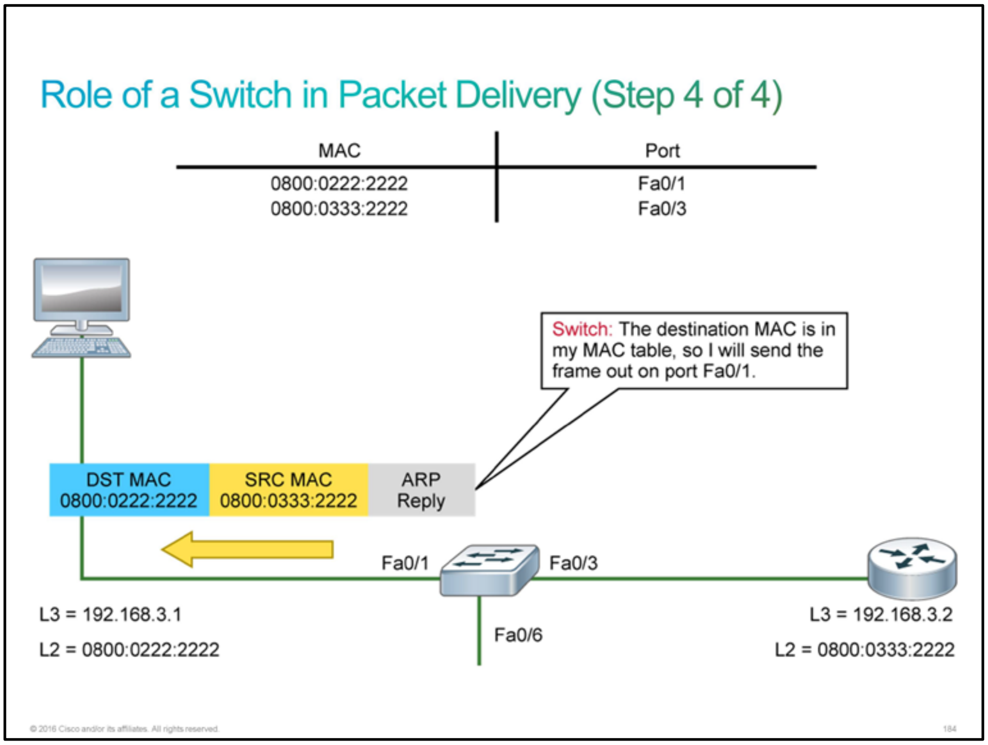
Switch: I just received a frame from a host that is not in my MAC table. Let me add it to the table.

DST MAC	SRC MAC	ARP
Broadcast	0800:0333:2222	Reply

The router replies to the ARP request and sends an ARP reply packet back to the sender as a unicast frame.

The switch learns the port mapping for the new source host from the source MAC address in the frame. The switch adds it to the MAC address table (0800:0333:2222 = port FastEthernet0/3).

Role of Switch in Packet Delivery (Step 4 of 4)



Role of a Switch in Packet Delivery (Step 4 of 4)

MAC	Port
0800:0222:2222	Fa0/1
0800:0333:2222	Fa0/3

Switch: The destination MAC is in my MAC table, so I will send the frame out on port Fa0/1.

DST MAC	SRC MAC	ARP
0800:0222:2222	0800:0333:2222	Reply

L3 = 192.168.3.1

L2 = 0800:0222:2222

L3 = 192.168.3.2

L2 = 0800:0333:2222

© 2016 Cisco and/or its affiliates. All rights reserved.

184

The destination address of the frame is found in the MAC address table, so the switch can forward out the frame on port FastEthernet0/1. If the destination address is not found in the MAC address table, the switch would need to flood out the frame on all ports.

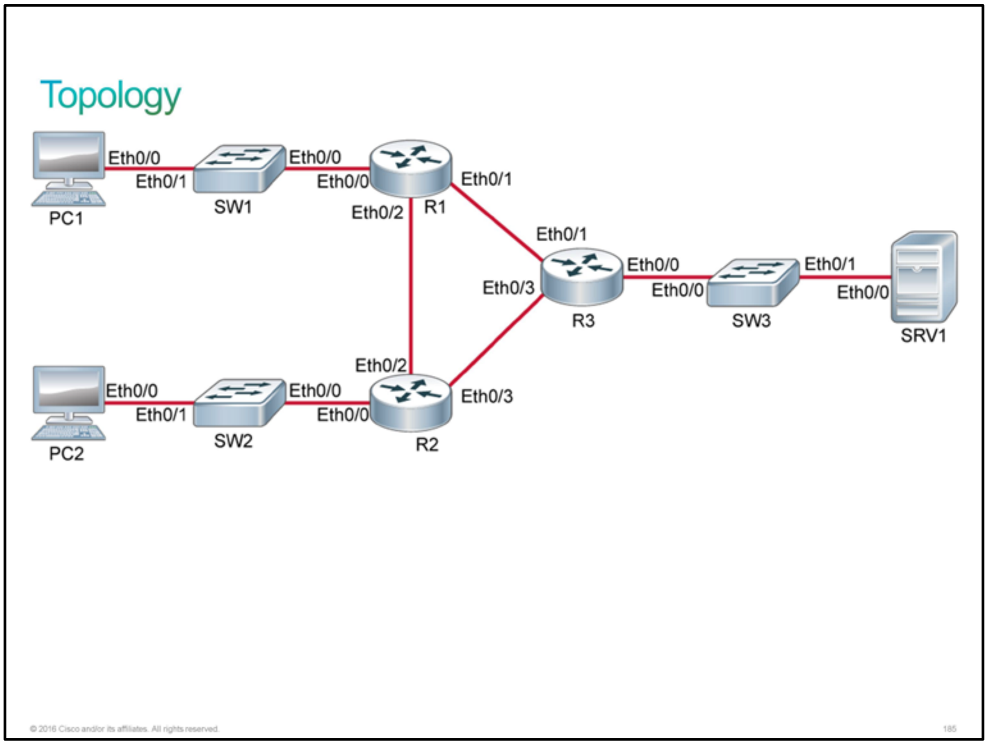
All frames pass through the switch unchanged. When the switch builds the MAC address table, it sends all unicast frames directly to a destination host based on the destination MAC address and data that are stored in the MAC address table.

Discovery 9: Exploration of Packet Forwarding

Introduction

This discovery lab will guide you through the exploration of packet forwarding. The lab is prepared with the devices as represented in the topology diagram. The devices are fully configured, including static routing on the routers.

Topology



Job Aid

Device Information

Device Information Table

Device	Characteristic	Value
PC1	Hostname	PC1
PC1	IP address	10.10.1.10/24
PC1	Default gateway	10.10.1.1
PC2	Hostname	PC2
PC2	IP address	10.10.2.20/24

Device	Characteristic	Value
PC2	Default gateway	10.10.2.1
SRV1	Hostname	SRV1
SRV1	IP address	10.10.3.30/24
SRV1	Default gateway	10.10.3.1
SW1	Hostname	SW1
SW1	VLAN 1 IP address	10.10.1.4/24
SW1	Default gateway	10.10.1.1
SW1	Ethernet0/0 description	Link to R1
SW1	Ethernet0/1 description	Link to PC1
SW2	Hostname	SW2
SW2	VLAN 1 IP address	10.10.2.4/24
SW2	Default gateway	10.10.2.1
SW2	Ethernet0/0 description	Link to R2
SW2	Ethernet0/1 Description	Link to PC2
SW3	Hostname	SW3
SW3	VLAN 1 IP address	10.10.3.4/24
SW3	Default gateway	10.10.3.1
SW3	Ethernet0/0 description	Link to R3
SW3	Ethernet0/1 description	Link to SRV1
R1	Hostname	R1
R1	Ethernet0/0 description	Link to SW1
R1	Ethernet0/0 IP address	10.10.1.1/24
R1	Ethernet0/1 description	Link to R3
R1	Ethernet0/1 IP address	10.1.1.2/30
R1	Ethernet0/2 description	Link to R2

Device	Characteristic	Value
R1	Ethernet0/2 IP address	10.1.1.10/30
R2	Hostname	R2
R2	Ethernet0/0 description	Link to SW2
R2	Ethernet0/0 IP address	10.10.2.1/24
R2	Ethernet0/2 description	Link to R1
R2	Ethernet0/2 IP address	10.1.1.9/30
R2	Ethernet0/3 description	Link to R3
R2	Ethernet0/3 IP address	10.1.1.6/30
R3	Hostname	R3
R3	Ethernet0/0 description	Link to SW3
R3	Ethernet0/0 IP address	10.10.3.1/24
R3	Ethernet0/1 description	Link to R1
R3	Ethernet0/1 IP address	10.1.1.1/30
R3	Ethernet0/3 description	Link to R2
R3	Ethernet0/3 IP address	10.1.1.5/30

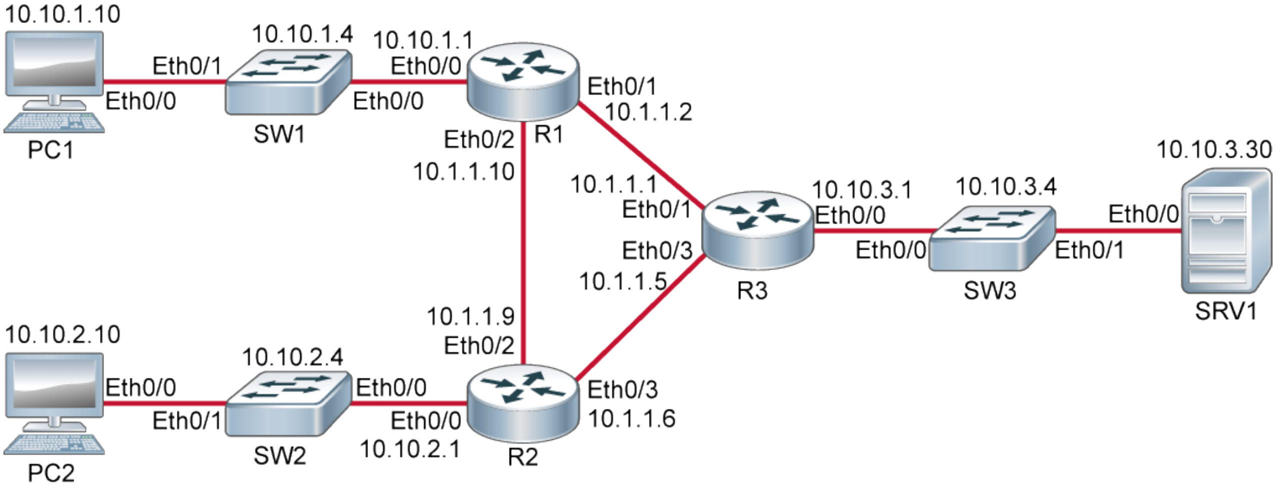
PCs and SRV in the virtual lab environment are simulated as routers, so you should use Cisco IOS commands to configure them or make verifications.

Task 1: Exploration of Packet Forwarding

Activity

Step 1 Consult the topology diagram. This discovery will focus on the forwarding of packets from PC1 to SRV1.

The devices in the path between these two hosts are SW1, R1, R3 and SW3.



Step 2 Access the console of PC1 and verify connectivity to SRV1 using the **ping** and **traceroute** commands.

```
PC1# ping 10.10.3.30
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.3.30, timeout is 2 seconds:
.!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 1/1/1 ms
PC1# traceroute 10.10.3.30
Type escape sequence to abort.
Tracing the route to 10.10.3.30
VRF info: (vrf in name/id, vrf out name/id)
  1 10.10.1.1 1 msec 0 msec 0 msec
  2 10.1.1.1 1 msec 0 msec 1 msec
  3 10.10.3.30 0 msec *  1 msec
```

The **traceroute** output shows 10.10.1.1 and 10.1.1.1 being in the forwarding path to SRV1. These addresses belong to Ethernet0/0 on R1 and Ethernet0/1 on R3. The interfaces Ethernet0/1 on R1 and Ethernet0/0 on R3 are also involved in the forwarding process, as are the switches SW1 and SW3.

Step 3 One at a time, access the consoles of PC1, R1, R3, and SRV1 and use the **show interfaces** command to inventory the IP addresses and MAC addresses on the interfaces that are involved in the forwarding process.

The information that you need is in the output of the **show interfaces** command, but to focus explicitly on the data that you are interested in it would be useful to send the output through the include filter and only display lines that contain the string **address**.

```
PC1# show interfaces Ethernet0/0
Ethernet0/0 is up, line protocol is up
  Hardware is AmdP2, address is aabb.cc00.1500 (bia aabb.cc00.1500)
  Internet address is 10.10.1.10/24
  MTU 1500 bytes, BW 10000 Kbit/sec, DLY 1000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set
  Keepalive set (10 sec)
  ARP type: ARPA, ARP Timeout 04:00:00
  Last input 00:00:01, output 00:00:07, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     1470 packets input, 93664 bytes, 0 no buffer
     Received 1229 broadcasts (0 IP multicasts)
     0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
     0 input packets with dribble condition detected
     311 packets output, 34770 bytes, 0 underruns
     0 output errors, 0 collisions, 1 interface resets
     2 unknown protocol drops
     0 babbles, 0 late collision, 0 deferred
     0 lost carrier, 0 no carrier
     0 output buffer failures, 0 output buffers swapped out
```

This example illustrated the full command syntax of the **show interfaces** command. The examples that follow will utilize command abbreviation.

The string that is passed to the include filter cannot be abbreviated, per se. That is, Cisco IOS cannot determine that when you use the string **add** that you intend for it to be an abbreviation of **address**. But the only appearance of the string **add** in the command's output is as a substring of **address**, therefore it is an acceptable string to use for this purpose.

```
R1# sh int e0/0 | inc add
  Hardware is AmdP2, address is aabb.cc00.1200 (bia aabb.cc00.1200)
  Internet address is 10.10.1.1/24
R1# sh int e0/1 | inc add
  Hardware is AmdP2, address is aabb.cc00.1210 (bia aabb.cc00.1210)
  Internet address is 10.1.1.2/30
```

The main purpose of MAC addresses is to differentiate between hosts on a multiple access network. With point to point serial links, there are exactly two systems on the network. Packets that one sends the other one receives, and vice versa. Because of this reciprocity, there is no need for MAC addresses to differentiate hosts on this type of network.

```
R3# sh int e0/0 | inc add
  Hardware is AmdP2, address is aabb.cc00.1400 (bia aabb.cc00.1400)
  Internet address is 10.10.3.1/24
R3# sh int e0/1 | inc add
  Hardware is AmdP2, address is aabb.cc00.1410 (bia aabb.cc00.1410)
  Internet address is 10.1.1.1/30
```

```
SRV1# sh int e0/0 | inc add
  Hardware is AmdP2, address is aabb.cc00.2e00 (bia aabb.cc00.2e00)
  Internet address is 10.10.3.30/24
```

MAC addresses in your output may be different.

Step 4 The output of the **show interfaces** commands can be compiled for reference into a table.

The table would appear as follows:

Device	Interface	MAC Address	IP Address
PC1	Ethernet0/0	aabb.cc00.1500	10.10.1.10
R1	Ethernet0/0	aabb.cc00.1200	10.10.1.1
R1	Ethernet0/1	aabb.cc00.1210	10.10.1.2
R3	Ethernet0/1	aabb.cc00.1410	10.10.1.10
R3	Ethernet0/0	aabb.cc00.1400	10.10.3.1
SRV1	Ethernet0/0	aabb.cc00.2e00	10.10.3.30

Step 5 When PC1 generates an IP packet for SRV1, it will encapsulate the data with an IP header specifying 10.10.3.30 as the destination IP address and 10.10.1.10 as the source IP address. It will then encapsulate the IP packet with an Ethernet header specifying aabb.cc00.1200 (the Ethernet0/0 MAC address in R1) as the destination MAC address and aabb.cc00.1500 (its own MAC address) as the source. PC1 obtains the MAC address of R1 from its ARP cache. Access the console of PC1 and display its ARP cache.

The entry for 10.10.1.1 was populated in the ARP table when you performed the **ping** operation at the beginning of this discovery.

```
PC1# show ip arp
Protocol  Address          Age (min)  Hardware Addr   Type   Interface
Internet  10.10.1.1              45   aabb.cc00.1200  ARPA   Ethernet0/0
Internet  10.10.1.10              -   aabb.cc00.1500  ARPA   Ethernet0/0
```

Step 6 Execute the following sequence of commands to observe the behavior of the ARP process. You will enable debugging of ARP packets and you will use **show** commands to provide visibility into the process. You will shut down the Ethernet0/0 interface of PC1, which will clear the ARP cache entries that are associated with the interface. You will then re-enable the interface and initiate connectivity, both actions stimulating ARP activity. The informational notes that are imbedded in the directions further explain the operations.

On PC1, enable debugging of ARP packets:

```
PC1# debug arp
ARP packet debugging is on
```

Be very careful when using **debug** commands in production environments. Depending on the circumstances, they can have a catastrophic effect on router performance. Until you have experience with **debug** commands, it is best to consult a senior engineer within your organization on their use.

```
PC1# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
PC1(config)# interface Ethernet 0/0
PC1(config-if)# do show ip arp
Protocol  Address          Age (min)  Hardware Addr   Type   Interface
Internet  10.10.1.1              51   aabb.cc00.1200  ARPA   Ethernet0/0
Internet  10.10.1.10              -   aabb.cc00.1500  ARPA   Ethernet0/0
```

The **do** command allows access to EXEC mode commands from within the configuration mode. The **show ip arp** command verifies that the two entries are still in the ARP cache.

```
PC1(config-if)# shutdown
PC1(config-if)#
*Oct  9 12:40:03.589: %LINK-5-CHANGED: Interface Ethernet0/0, changed state to
administratively down
*Oct  9 12:40:04.589: %LINEPROTO-5-UPDOWN: Line protocol on Interface
Ethernet0/0, changed state to down
PC1(config-if)# do show ip arp
```

This time there is no output from the **show ip arp** command. The ARP cache on R1 is currently empty. The entries that are associated with Ethernet0/0 were cleared when the interface was shut down.

```
PC1(config-if)# no shutdown
PC1(config-if)#
*Oct  9 12:41:24.437: IP ARP: sent rep src 10.10.1.10 aabb.cc00.1500,
                 dst 10.10.1.10 ffff.ffff.ffff Ethernet0/0
*Oct  9 12:41:24.437: IP ARP: sent rep src 10.10.1.10 aabb.cc00.1500,
                 dst 10.10.1.10 ffff.ffff.ffff Ethernet0/0
```

The two messages above are debug messages. They both indicate that R1 sent ARP packets. The destination IP address is 10.10.1.10. R1 is sending this ARP broadcast asking any hosts that have the IP address 10.10.1.10 to respond back with an ARP reply. Cisco IOS sends this ARP broadcast automatically when interfaces are brought online. It is an attempt to recognize when there are duplicate IP addresses on the network. If any responses are received, syslog messages would be generated to alert the network administrator about the situation. No replies were received, which is normal.

```
*Oct  9 12:41:26.434: %LINK-3-UPDOWN: Interface Ethernet0/0, changed state to
up
*Oct  9 12:41:27.434: %LINEPROTO-5-UPDOWN: Line protocol on Interface
Ethernet0/0, changed state to up
```

These two messages are the normal syslog messages, which are generated when interfaces change their state.

```
PC1(config-if)# end
PC1#
```

You just left configuration mode. The rest of this exploration will be completed from privileged EXEC.

```
PC1# ping 10.10.3.30
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.3.30, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
```

For this ping operation from PC1 to SRV1 to complete, PC1 must send packets to R1 for forwarding. PC1 needs to know the MAC address of R1 to send the packets to R1.

```
*Oct  9 12:41:27.434: IP ARP: creating incomplete entry for IP address:
10.10.1.1 interface Ethernet0/0
```

This debug message indicates that PC1 recognizes that it needs the MAC address for 10.10.1.1 (R1, its default gateway). PC1 creates an entry in its ARP cache and starts the ARP process.

```
*Oct  9 12:41:27.434: IP ARP: sent req src 10.10.1.10 aabb.cc00.1500,
                 dst 10.10.1.1 0000.0000.0000 Ethernet0/0
```

This debug message indicates that PC1 sent an ARP request specifying 10.10.1.1 as the destination. This request is broadcast to all hosts within the broadcast domain. PC1 is requesting any system with the IP address 10.10.1.1 to respond with an ARP reply.

```
*Oct  9 12:41:27.435: IP ARP: rcvd rep src 10.10.1.1 aabb.cc00.1200, dst
10.10.1.10 Ethernet0/0
```

This debug message indicates that PC1 received an ARP reply from 10.10.1.1, indicating that its MAC address is aabb.cc00.1200.

Step 7 View the ARP cache on PC1 resulting from the exchange of ARP packets that you just witnessed.

The ARP cache of PC1 now has an entry mapping the IP address of R1 to the MAC address of R1.

```
PC1# show ip arp
Protocol  Address          Age (min)  Hardware Addr   Type   Interface
Internet  10.10.1.1               8   aabb.cc00.1200  ARPA   Ethernet0/0
Internet  10.10.1.10              -   aabb.cc00.1500  ARPA   Ethernet0/0
```

Step 8 The close inspection of the ARP process is complete. Turn off the debug operations.

Debug can be turned off on a per classification basis. That is, you could have used **undebug arp** to turn off the debug process that you started with the **debug arp** command.

```
PC1# undebug all
All possible debugging has been turned off
```

A common abbreviation that is used for **undebug all** is **u all**.

The lab environment does not support capturing packets on the interface links, but the results above support the following extrapolation which describes how a packet is forwarded from PC1 to SRV1:

- The IP header remains constant across the entire path, the IP header will specify 10.10.3.30 as the destination IP address and 10.10.1.10 as the source IP address.
- A unique Layer 2 header is used to traverse each network segment.
- PC1 and R1 learn each other's MAC addresses via ARP.
- R1 and R3 learn each other's MAC addresses via ARP.
- SRV1 and R3 learn each other's MAC addresses via ARP.
- PC1 will encapsulate the IP packet with an Ethernet header that specifies aabb.cc00.1200 (R1 Ethernet0/0) as the destination MAC address and aabb.cc00.1500 (PC1) as the source MAC address.
- PC1 will send this packet out its Ethernet0/0 interface, and R1 will receive it on its Ethernet0/0 interface.
- R1 will strip the Ethernet header and replace it with another Ethernet header that specifies aabb.cc00.1410 (R3 Ethernet 0/1) as the destination MAC address and aabb.cc00.1210 (R1 Ethernet0/1) as the source MAC address.
- R3 will strip the Ethernet header and replace it with another Ethernet header that specifies aabb.cc00.2e00 (SRV1) as the destination MAC address and aabb.cc00.1400 as the source MAC address.
- R3 will send this packet out its Ethernet0/0 interface, and SRV1 will receive it on its Ethernet0/0 interface.

Step 9 The previous steps did not depict how SW1 supports the forwarding of packets between PC1 and R1 and how SW3 supports the forwarding of packets between R3 and SRV1. Switches learn which ports connect to which MAC addresses based on examination of the source MAC address on incoming frames. When they know which ports lead to which MAC address, they can forward to those MAC addresses based on the destination MAC address in frames. Access the console of SW1 and view its MAC address table.

The MAC address of PC1 is associated with the port Ethernet0/1 and the MAC address of R1 is associated with the port Ethernet0/0.

```
SW1# show mac address-table
          Mac Address Table
-------------------------------------------

Vlan    Mac Address       Type        Ports
----    -----------       --------    -----
   1    aabb.cc00.1200    DYNAMIC     Et0/0
   1    aabb.cc00.1500    DYNAMIC     Et0/1
Total Mac Addresses for this criterion: 2
```

Step 10 Clear the MAC address table on SW1 and display it again to verify that it is empty.

The example shows an empty MAC address table, but when you attempt this step, you may see that the entries have already repopulated. If so, simply repeat the last two commands as quickly as possible (use the **Up Arrow** key for command recall) until you see the empty MAC address table.

```
SW1# clear mac address-table dynamic
SW1# show mac address-table
          Mac Address Table
-------------------------------------------

Vlan    Mac Address       Type        Ports
----    -----------       --------    -----
```

Step 11 Wait at least 10 seconds after viewing the empty MAC address table before continuing. Display the MAC address table one more time.

The MAC address table has again been repopulated with the MAC addresses of PC1 and R1.

```
SW1# show mac address-table
          Mac Address Table
-------------------------------------------

Vlan    Mac Address       Type        Ports
----    -----------       --------    -----
   1    aabb.cc00.1200    DYNAMIC     Et0/0
   1    aabb.cc00.1500    DYNAMIC     Et0/1
Total Mac Addresses for this criterion: 2
```

Step 12 How did the table get repopulated? When the switch receives a packet of any kind, it examines the source MAC address to determine if it needs to add it to the MAC address table. By default, with Cisco IOS, Ethernet interfaces send packets to their own MAC address every 10 seconds as a keepalive mechanism. Verify this setting by accessing the console of PC1 and use the **show interface** command to view the status of Ethernet0/0.

The keepalive value is set to 10 seconds. Also make note of the number of packets output from the interface.

```
PC1# sh int e0/0
Ethernet0/0 is up, line protocol is up
  Hardware is AmdP2, address is aabb.cc00.1500 (bia aabb.cc00.1500)
  Internet address is 10.10.1.10/24
  MTU 1500 bytes, BW 10000 Kbit/sec, DLY 1000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set
  Keepalive set (10 sec)
  ARP type: ARPA, ARP Timeout 04:00:00
  Last input 00:00:00, output 00:00:09, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     3583 packets input, 225244 bytes, 0 no buffer
     Received 3014 broadcasts (0 IP multicasts)
     0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
     0 input packets with dribble condition detected
     708 packets output, 76818 bytes, 0 underruns
     0 output errors, 0 collisions, 2 interface resets
     2 unknown protocol drops
     0 babbles, 0 late collision, 0 deferred
     0 lost carrier, 0 no carrier
     0 output buffer failures, 0 output buffers swapped out
```

Step 13 Wait 10 seconds and repeat the **show interface** command. Verify that the number of packets output has increased by at least 1.

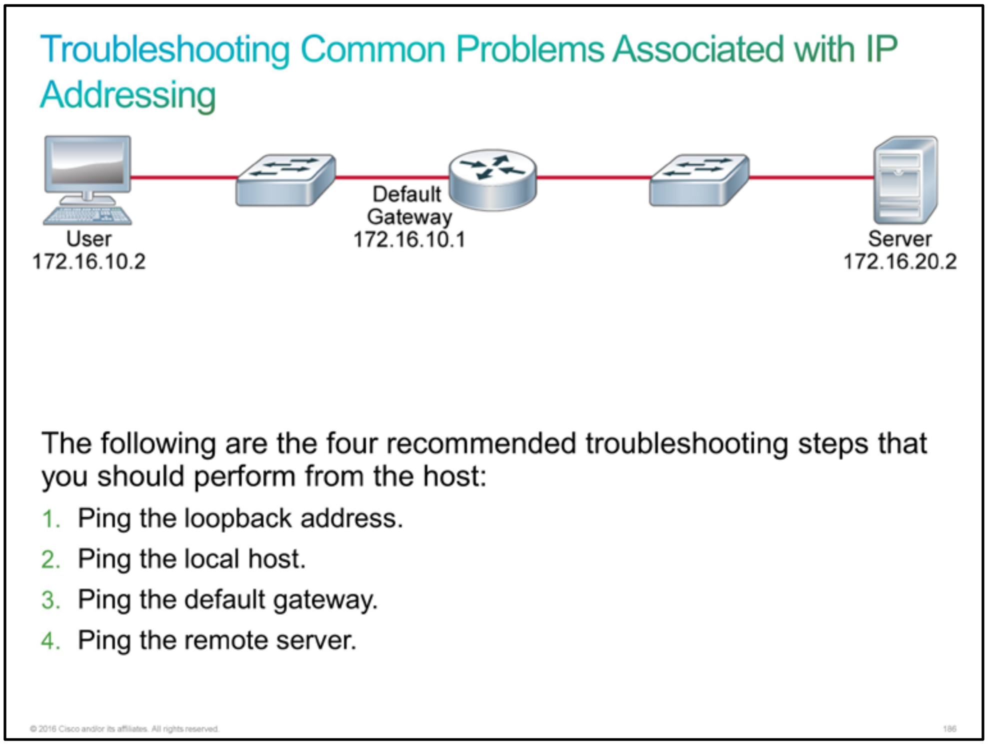
You now have some experience with the forwarding of packets between IP hosts, including the ARP process and the use of MAC addresses on Ethernet networks. You also investigated in how switches populate and use the MAC address tables. Feel free to continue exploring independently within the lab environment.

```
PC1# sh int e0/0
Ethernet0/0 is up, line protocol is up
  Hardware is AmdP2, address is aabb.cc00.1500 (bia aabb.cc00.1500)
  Internet address is 10.10.1.10/24
  MTU 1500 bytes, BW 10000 Kbit/sec, DLY 1000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set
  Keepalive set (10 sec)
  ARP type: ARPA, ARP Timeout 04:00:00
  Last input 00:00:00, output 00:00:00, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     3624 packets input, 227780 bytes, 0 no buffer
     Received 3048 broadcasts (0 IP multicasts)
     0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
     0 input packets with dribble condition detected
     716 packets output, 77591 bytes, 0 underruns
     0 output errors, 0 collisions, 2 interface resets
     2 unknown protocol drops
     0 babbles, 0 late collision, 0 deferred
     0 lost carrier, 0 no carrier
     0 output buffer failures, 0 output buffers swapped out
```

This is the end of the discovery lab.

Troubleshooting Common Problems Associated with IP Addressing

Troubleshooting IP addressing is an important skill and will prove valuable when resolving several network issues. Assume that a host cannot communicate to a server that is on a remote network.


Troubleshooting Common Problems Associated with IP Addressing

The following are the four recommended troubleshooting steps that you should perform from the host:

1. Ping the loopback address.
2. Ping the local host.
3. Ping the default gateway.
4. Ping the remote server.

© 2016 Cisco and/or its affiliates. All rights reserved. 186

1. Ping the Loopback Address

Access the command prompt and ping 127.0.0.1. This address is the diagnostic or loopback address. If you get a successful ping, your IP stack is considered to be initialized. If it fails, you have an IP stack failure and you need to reinstall TCP/IP on the host.

```
C:\> ping 127.0.0.1
Pinging 127.0.0.1 with 32 bytes of data:
Reply from 127.0.0.1: bytes=32 time<1ms TTL=128
Reply from 127.0.0.1: bytes=32 time<1ms TTL=128
Reply from 127.0.0.1: bytes=32 time<1ms TTL=128
Reply from 127.0.0.1: bytes=32 time<1ms TTL=128
Ping statistics for 127.0.0.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

2. Ping the Local Host

From the command prompt, ping the IP address of the local host. If the ping is successful, your NIC is functioning. If it fails, there is a problem with the NIC. If the ping is successful, it does not mean that a cable is plugged into the NIC, but only that the IP protocol stack on the host can communicate to the NIC (via the LAN driver).

```
C:\> ping 172.16.10.2
Pinging 172.16.10.2 with 32 bytes of data:
Reply from 172.16.10.2: bytes=32 time<1ms TTL=128
Reply from 172.16.10.2: bytes=32 time<1ms TTL=128
Reply from 172.16.10.2: bytes=32 time<1ms TTL=128
Reply from 172.16.10.2: bytes=32 time<1ms TTL=128
Ping statistics for 172.16.10.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

3. Ping the Default Gateway

From the command prompt, ping the default gateway (router). If the ping works, it means that the NIC is plugged into the network and can communicate on the local network. If it fails, you have a local physical network problem that could be anywhere from the NIC to the router.

```
C:\> ping 172.16.10.1
Pinging 172.16.10.1 with 32 bytes of data:
Reply from 172.16.10.1: bytes=32 time<1ms TTL=128
Reply from 172.16.10.1: bytes=32 time<1ms TTL=128
Reply from 172.16.10.1: bytes=32 time<1ms TTL=128
Reply from 172.16.10.1: bytes=32 time<1ms TTL=128
Ping statistics for 172.16.10.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

4. Ping the Remote Server

If Steps 1 through 3 are successful, try to ping the remote server. If the ping works, then you know that you have IP communication between the local host and the remote server. You also know that the remote physical network is working.

```
C:\> ping 172.16.20.2
Pinging 172.16.20.2 with 32 bytes of data:
Reply from 172.16.20.2: bytes=32 time<1ms TTL=128
Reply from 172.16.20.2: bytes=32 time<1ms TTL=128
Reply from 172.16.20.2: bytes=32 time<1ms TTL=128
Reply from 172.16.20.2: bytes=32 time<1ms TTL=128
Ping statistics for 172.16.20.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

If you still cannot communicate with the server after Steps 1 through 4 are successful, you probably have some type of name resolution problem and need to check your DNS settings. But if the ping to the remote server fails, then you know that you have some type of remote physical network problem and need to go to the server and work on Steps 1 through 3 until you find the issue.

Important Troubleshooting Commands

As a network engineer, your primary goal is to make sure that your network equipment is always operating properly. This section will cover the most important commands that you will find helpful and perhaps even mandatory throughout your networking career and specifically during network troubleshooting situations.

The commands, which are truly invaluable, as are follows:

- **ping—ping** uses the IP ICMP echo request and echo reply messages to test reachability to a remote system. In its simplest form, **ping** simply confirms that an IP packet is capable of getting to and getting back from a destination IP address. This tool generally returns two pieces of information: whether the source can reach the destination (and, by inference, vice versa) and the RTT (typically in milliseconds). If **ping** fails or returns an unusual RTT, you can use the **traceroute** command to help narrow down the problem. You can also vary the size of the ICMP echo payload to test problems that are related to the MTU.
- **traceroute**—**traceroute** can return useful information about TCP/IP connectivity across your network. The **traceroute** utility sends out either ICMP echo request (Windows) or UDP (most implementations) messages with gradually increasing IP TTL values to probe the path by which a packet traverses the network. The first packet with the TTL set to 1 will be discarded by the first hop. Also, the first hop will send back an ICMP TTL exceeded message that is sourced from its IP address facing the source of the packet. When the machine running the **traceroute** receives the ICMP TTL exceeded message, it can determine the hop via the source IP address. This process continues until the request or message reaches the destination. The destination will return either an ICMP echo reply (Windows) or an ICMP port unreachable, indicating that the request or message has reached the destination. The Cisco implementation of **traceroute** sends out three packets at each TTL value, allowing **traceroute** to report routers that have multiple, equal-cost paths to the destination.
- **tracert—**This is the same command as **traceroute**, but it is a Microsoft Windows command and will not work on a Cisco router.
- **arp -a** —The device uses ARP to perform IP address resolution that is the linking of IP addresses to MAC addresses. ARP uses a broadcast to do this action by asking the host that has the given IP address to respond to the broadcast with its MAC address. The **arp -a** command displays IP-to-MAC-address mappings on a Windows PC.

```
C:\Windows\system32> arp -a
Interface: 10.1.10.100 --- 0xd
  Internet Address      Physical Address      Type
   10.1.10.1             54-75-d0-8e-9a-d8     dynamic
  224.0.0.22            01-00-5e-00-00-16     static
  224.0.0.252           01-00-5e-00-00-fc     static
  255.255.255.255       ff-ff-ff-ff-ff-ff     static
```

- **show ip arp**—This is the same command as **arp -a,** but it displays the ARP table on a Cisco router. Like the **traceroute** and **tracert** commands, the two are not interchangeable through DOS and Cisco.
- **ipconfig /all—ipconfig**—This is a command-line utility that is available on all versions of Microsoft Windows starting with Windows NT. You run the **ipconfig** utility from the Windows command prompt. This utility allows you to get the IP address information of a Windows computer. The **ipconfig /all** option displays the IP address, network mask, and gateway for all physical and virtual network adapters. Also, it displays DNS and WINS settings for each adapter.

- **telnet**—You will find Telnet or SSH applications useful for connecting to remote devices. One way to obtain information about a remote network device is to connect to it using either the Telnet or SSH applications. Telnet and SSH are virtual terminal protocols that are part of the TCP/IP suite. The protocols allow connections and remote console sessions from one network device to one or more remote devices. To log on to a host that supports Telnet, use the **telnet** EXEC command:

```
RouterA# telnet host
```

(where *host* is an IP address or hostname of a remote system)

- **ssh**—Telnet is the most common method of accessing a network device. However, Telnet is an insecure way of accessing a network. SSH is a secure replacement for Telnet that gives the same type of access. Communication between the client and server is encrypted in both SSHv1 and SSHv2. Implement SSHv2 when possible because it uses a more enhanced security encryption algorithm. To start an encrypted session with a remote networking device, use the **ssh user** EXEC command:

```
RouterA# ssh ip address
```

Challenge

1. Which two statements about ARP are accurate? (Choose two.)

   A. A device can use ARP to assign MAC addresses to dynamic NICs.
   B. ARP entries are not permanent.
   C. Devices can exchange entire ARP tables.
   D. ARP tables are created and maintained dynamically.

2. Which two features does ARP provide? (Choose two.)

   A. mapping IP addresses to DLCIs on a network
   B. mapping IP addresses to MAC addresses on a network
   C. locally storing DLCIs that are learned via ARP
   D. mapping MAC addresses to DLCIs on a network
   E. locally storing MAC addresses that are learned via ARP

3. Which Cisco IOS command do you use to display the ARP table?

   A. **show arp table**
   B. **show -a**
   C. **show ip arp**
   D. **arp -a**

4. Which Microsoft Windows command do you use to display the ARP table?

   A. **show arp table**
   B. **show arp**
   C. **show ip arp**
   D. **arp -a**

5. Which of the following troubleshooting steps would you take if you wanted to check the connectivity to the local network?

   A. Ping the loopback address.
   B. Ping the local host.
   C. Ping the default gateway.
   D. Ping the remote server.

6. Which of the following troubleshooting steps would you take if you wanted to check the IP Stack on your own device?

   A. Ping the loopback address.
   B. Ping the local host.
   C. Ping the default gateway.
   D. Ping the remote server.

7. Which of the following troubleshooting steps would you take if you wanted to check the NIC on your own device?

   A. Ping the loopback address.
   B. Ping the local host.
   C. Ping the default gateway.
   D. Ping the remote server.

Answer Key

Challenge

1. B, D
2. B, E
3. C
4. D
5. C
6. A
7. B

Lesson 7: Enabling Static Routing

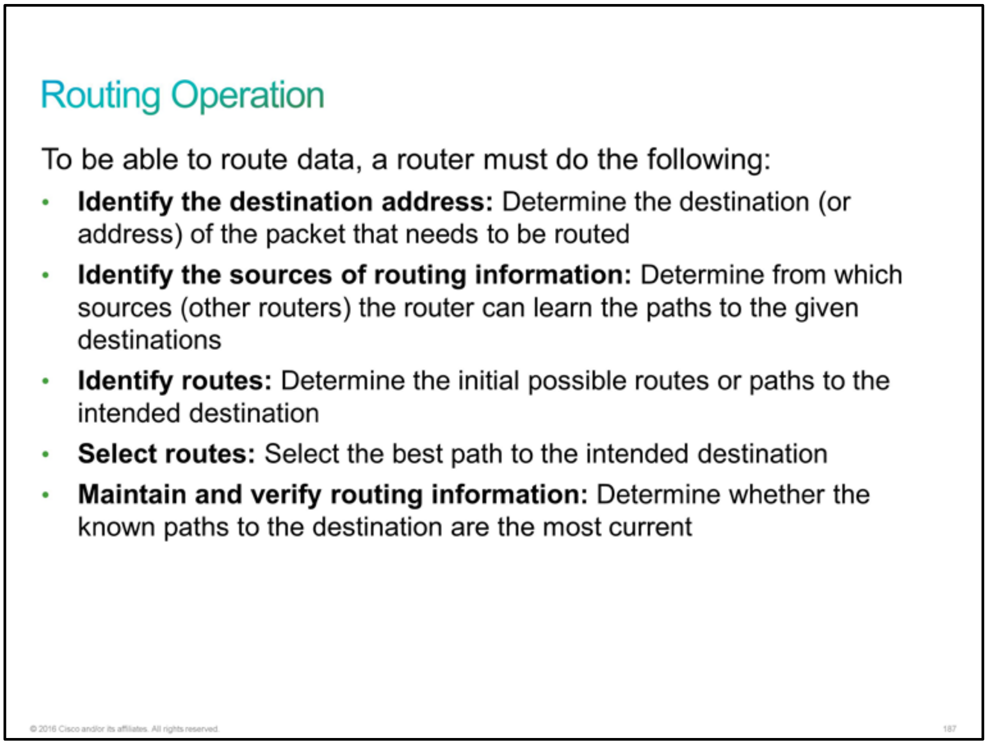
Introduction

Your boss sends you to your customer to enable basic static IP routing. You should be able to present to the customer the difference between static and dynamic routing. If there is a problem with several static routes and a static default route, you should be able to understand what the solution is. You should be able to configure and verify both static and default static routes.

Routing Operation

Routing is the process of determining where to send data packets that are destined for addresses outside the local network. Routers gather and maintain routing information to enable the transmission and receipt of such data packets.

Conceptually, routing information takes the form of entries in a routing table, with one entry for each identified route. You can manually configure the entries in the routing table, or the router can use a routing protocol to create and maintain the routing table dynamically to accommodate network changes when they occur.

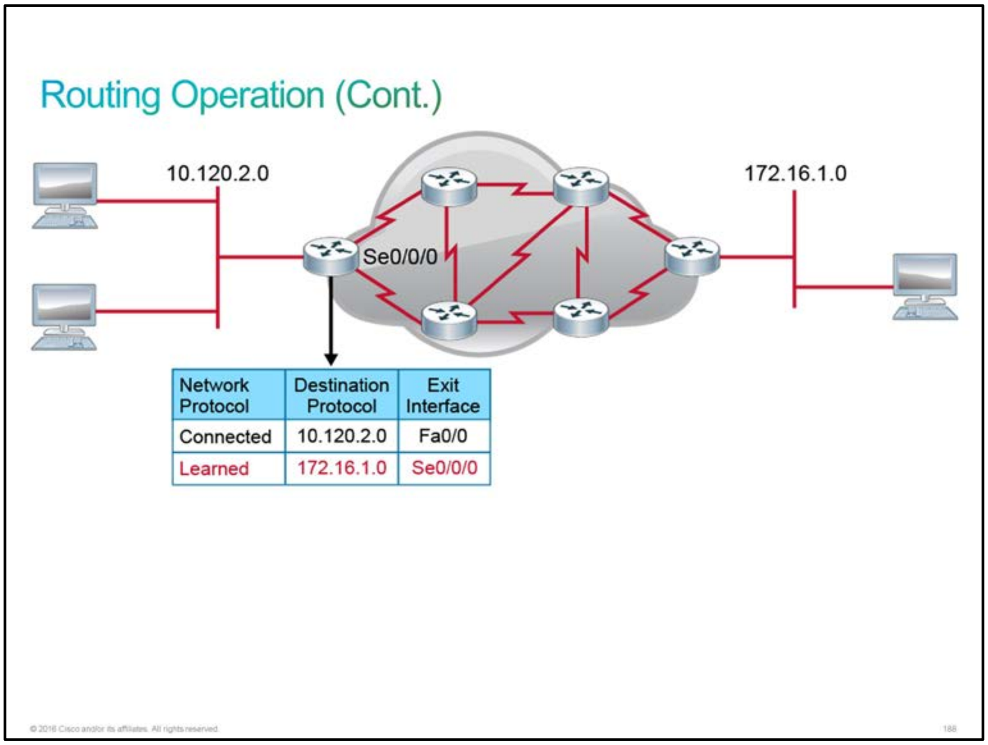


Routing Operation

To be able to route data, a router must do the following:

- **Identify the destination address:** Determine the destination (or address) of the packet that needs to be routed
- **Identify the sources of routing information:** Determine from which sources (other routers) the router can learn the paths to the given destinations
- **Identify routes:** Determine the initial possible routes or paths to the intended destination
- **Select routes:** Select the best path to the intended destination
- **Maintain and verify routing information:** Determine whether the known paths to the destination are the most current

The routing information that a router obtains from other routers is placed in its routing table. The router relies on this table to tell it which interfaces to use when forwarding packets. The following figure shows that the router on the left uses interface Serial0/0/0 to get to the 172.16.1.0 subnet.



Routing Operation (Cont.)

Network Protocol	Destination Protocol	Exit Interface
Connected	10.120.2.0	Fa0/0
Learned	172.16.1.0	Se0/0/0

If the destination network is directly connected, that is, if there is an interface on the router that belongs to that network, the router already knows which interface to use when forwarding packets. If destination networks are not directly attached, the router must learn the best route to use when forwarding packets.

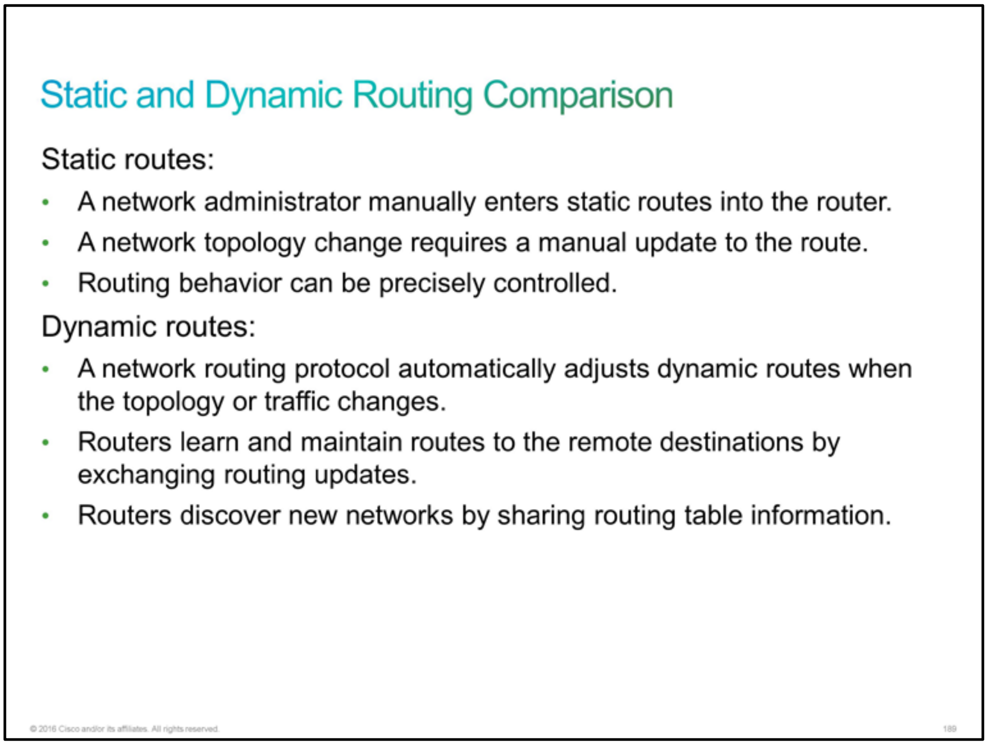
The destination information can be learned in two ways:

- You can enter routing information manually, also known as a static route.
- You can collect routing information through the dynamic routing process that runs in the routers.

Static and Dynamic Routing Comparison

There are two ways that a router can learn where to forward packets to destination networks that are not directly connected.

- **Static routing:** The router learns routes when an administrator manually configures the static route. The administrator must manually update this static route entry whenever an internetwork topology change requires an update. Static routes are user-defined routes that specify the path that packets take when moving between a source and a destination. These administrator-defined routes allow very precise control over the routing behavior of the IP internetwork.
- **Dynamic routing:** The router dynamically learns routes after an administrator configures a routing protocol that helps determine routes. Unlike the situation with static routes, after the network administrator enables dynamic routing, the routing process automatically updates route knowledge whenever the device receives new topology information. The router learns and maintains routes to the remote destinations by exchanging routing updates with other routers in the internetwork.



Static and Dynamic Routing Comparison

Static routes:

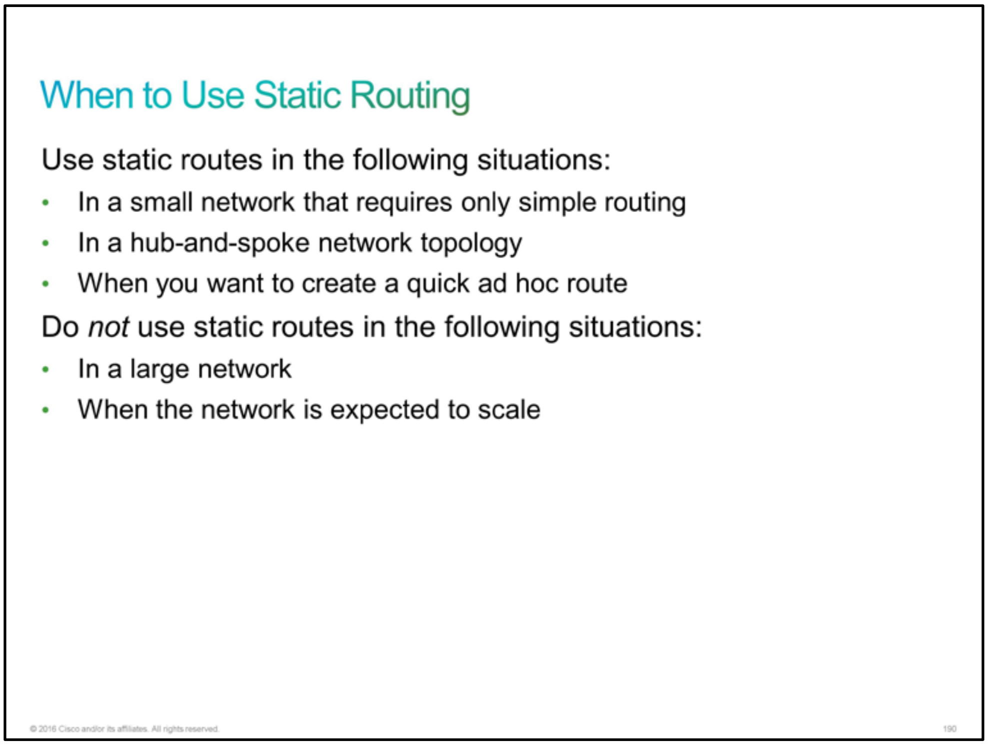
- A network administrator manually enters static routes into the router.
- A network topology change requires a manual update to the route.
- Routing behavior can be precisely controlled.

Dynamic routes:

- A network routing protocol automatically adjusts dynamic routes when the topology or traffic changes.
- Routers learn and maintain routes to the remote destinations by exchanging routing updates.
- Routers discover new networks by sharing routing table information.

When to Use Static Routing

Static routes are best suited for small networks, such as LANs, where routes rarely change. If routes change, you need to manually update your routes to reflect the new data transmission paths.



When to Use Static Routing

Use static routes in the following situations:

- In a small network that requires only simple routing
- In a hub-and-spoke network topology
- When you want to create a quick ad hoc route

Do *not* use static routes in the following situations:

- In a large network
- When the network is expected to scale

© 2016 Cisco and/or its affiliates. All rights reserved. 190

Some of the advantages of using static routes are as follows:

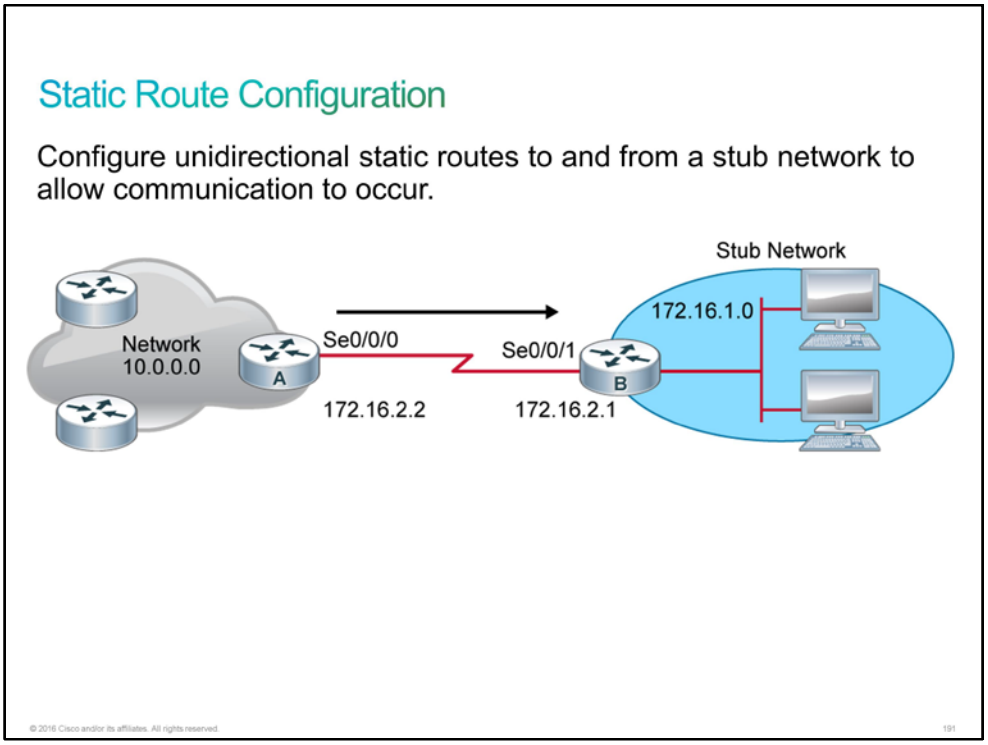
- **Conserving router resources**: Static routing does not consume network bandwidth and the CPU resources of the router. When you use a routing protocol, the traffic between routers adds some overhead as the routers exchange routing updates about remote networks. Depending on the size of the network, a router requires some CPU cycles to compute the best way to remote networks.
- **Simple to configure in a small network**: Static routes are commonly used in small networks that have few routers. Many small networks are designed as stub networks; for these types of networks, static routes are the most appropriate solution. Also, most of these networks are designed in a hub-and-spoke topology, where you can use default routes for branches that are pointing to the hub, which is the gateway to other networks.
- **Security**: Sometimes, you may want to define static routes to control the data transmission paths that are used by your data. This option may be useful in highly secure environments.

Some of the disadvantages of using static routes are as follows:

- **Scalability**: Static routing might be appropriate for networks that have fewer than four or five routers. Dynamic routing is more appropriate for large networks to reduce the probability of errors in routing configuration.
- **Accuracy**: If your network changes and you do not update the static routes, your router does not have accurate knowledge of your network. Not having accurate knowledge of your network can result in lost or delayed data transmissions.
- **High maintenance**: When the number of routers increases, the number of static routes also increases. In large networks, adding even one router with only one new network means that in addition to configuring the newly added router with static routes to other networks, you must configure all existing routers in the network with static routes to the new network.

Static Route Configuration

Static routes are commonly used when you are routing from a network to a stub network (a network that is accessed by a single link). Static routes can also be useful for specifying a "gateway of last resort" to which all packets with an unknown destination address are sent.



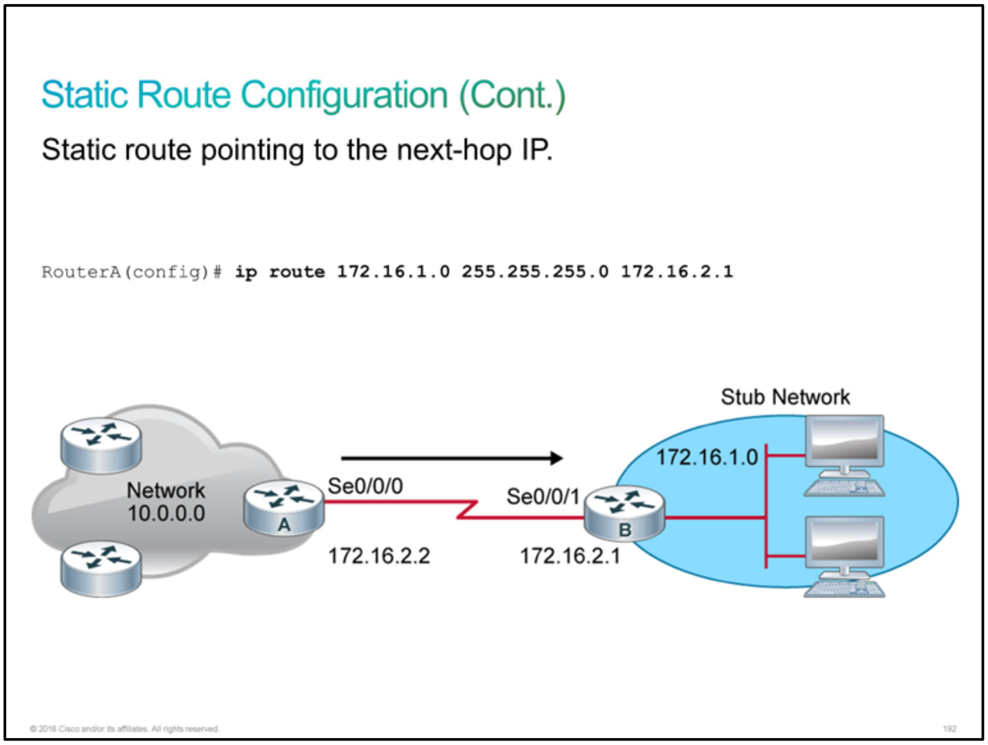
Static Route Configuration

Configure unidirectional static routes to and from a stub network to allow communication to occur.

191

Static route configuration steps:

- Define a path to an IP destination network (172.16.1.0 255.255.255.0).
- Use the IP address of the next-hop router (172.16.2.1).
- Or, use the outbound interface of the local router (Serial0/0/0).



Static Route Configuration (Cont.)

Static route pointing to the next-hop IP.

```
RouterA(config)# ip route 172.16.1.0 255.255.255.0 172.16.2.1
```

192

In the figure, router A is configured with a static route to reach the 172.16.1.0 subnet via the next hop IP 172.16.2.1 using the **ip route** command.

Alternatively, you can configure the static route by pointing to the exit interface instead of using the next-hop IP address.

```
RouterA(config)# ip route 172.16.1.0 255.255.255.0 serial0/0/0
```

The table lists the **ip route** command parameters for this example.

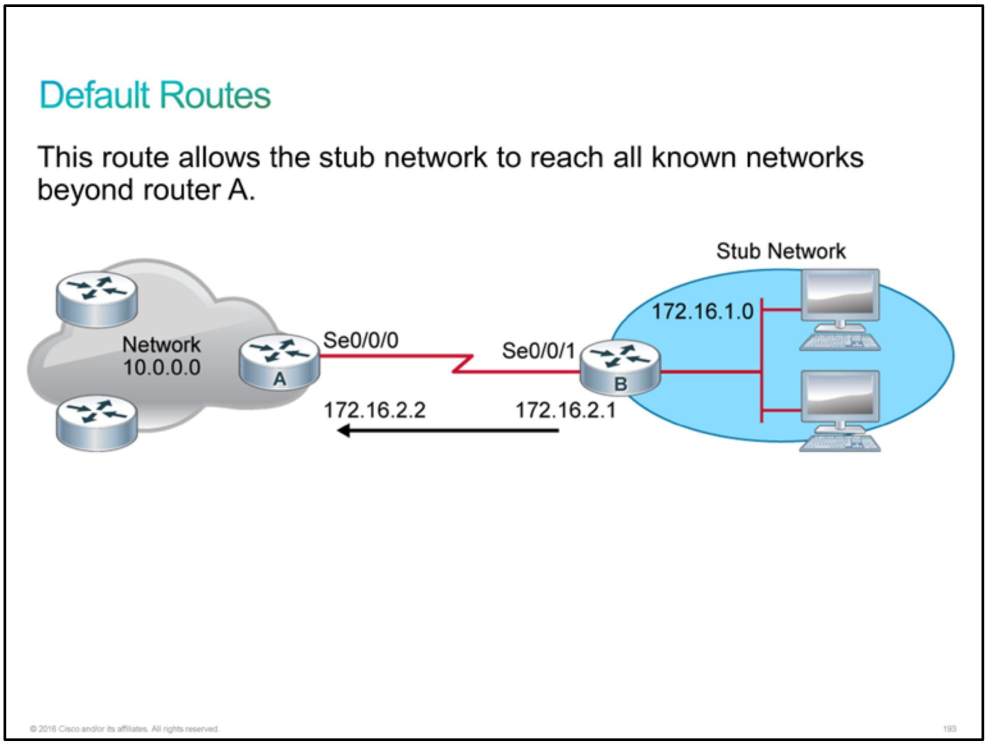
Command Parameters	Description
ip route	Identifies the static route
172.16.1.0	IP address of a static route to the destination subnetwork
255.255.255.0	Indicates the subnet mask—there are 8 bits of subnetting in effect
172.16.2.1	IP address of the next-hop router in the path to the destination
Serial 0/0/0	Identifies the interface that will be used to reach the next-hop router

In the figure, you would also need to configure router B with a static or default route to reach the networks behind router A via the serial interface of router B.

Note A static route is configured for connectivity to remote networks that are not directly connected to your router. For end-to-end connectivity, you must configure a static route in both directions.

Default Routes

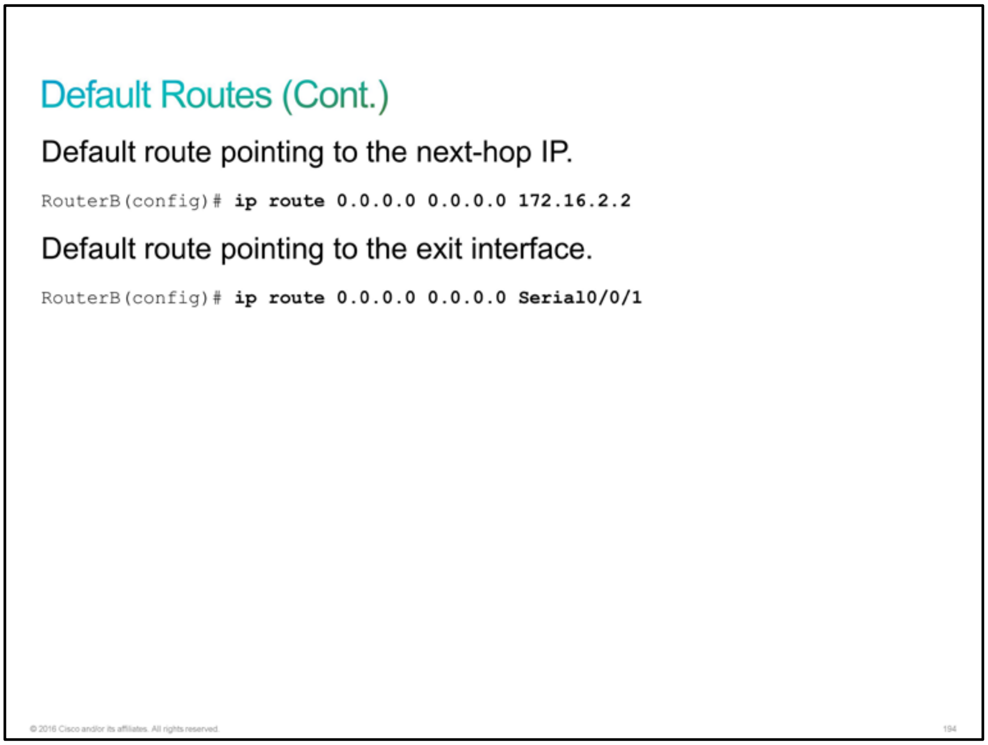
Use a default route when the route from a source to a destination is not known or when it is not feasible for the router to maintain many routes in its routing table.



Default Routes

This route allows the stub network to reach all known networks beyond router A.

193



Default Routes (Cont.)

Default route pointing to the next-hop IP.

```
RouterB(config)# ip route 0.0.0.0 0.0.0.0 172.16.2.2
```

Default route pointing to the exit interface.

```
RouterB(config)# ip route 0.0.0.0 0.0.0.0 Serial0/0/1
```

194

A default static route is a route that matches all packets. Default static routes are used in these instances:

- When no other routes in the routing table match the destination IP address of the packet, or when a more specific match does not exist. A common use for a default static route is to connect the edge router of a company to an ISP network.
- When a router has only one other router to which it is connected. This condition is known as a stub router.

The syntax for a default static route is like the one that is demonstrated for any other static route, except that the network address is 0.0.0.0 and the subnet mask is 0.0.0.0.

```
RouterB(config)# ip route 0.0.0.0 0.0.0.0 172.16.2.2
```

or

```
RouterB(config)# ip route 0.0.0.0 0.0.0.0 serial0/0/1
```

The 0.0.0.0 network address and 0.0.0.0 mask is called a *quad-zero* route.

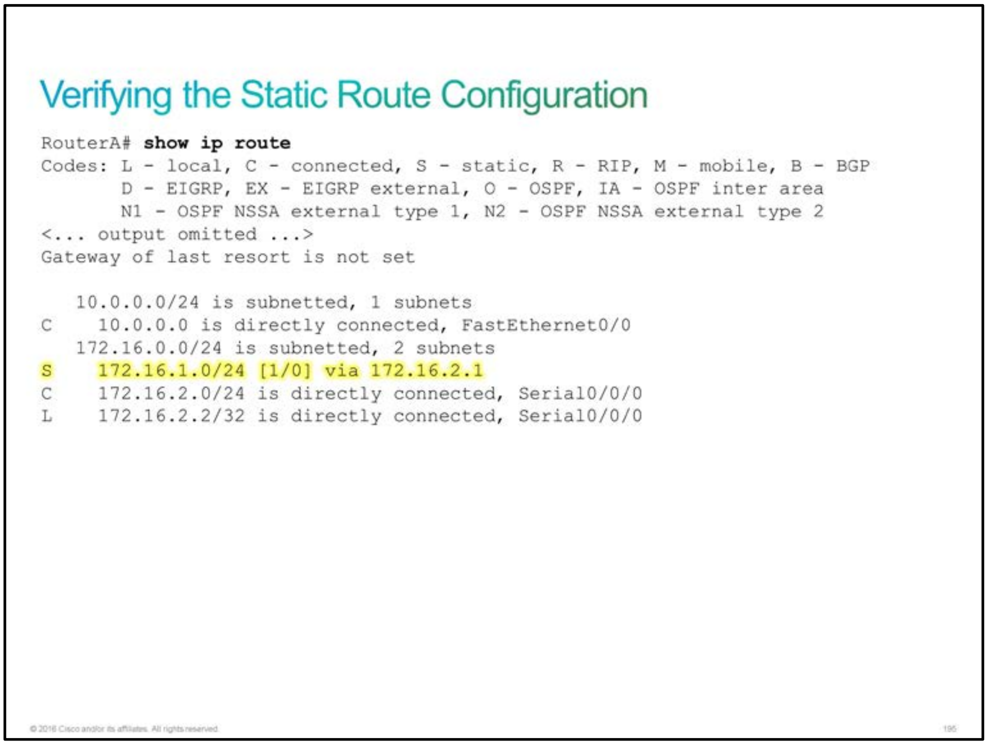
In the figure, router B is configured to forward to router A all packets that do not have the destination network that is listed in the router B routing table.

This table lists the **ip route** command parameters for this example.

Command Parameters	Description
ip route	Identifies the static route
0.0.0.0	Routes to networks that are not in the routing table
0.0.0.0	Special mask that indicates the default route
172.16.2.2	IP address of the next-hop router to be used as the default for packet forwarding

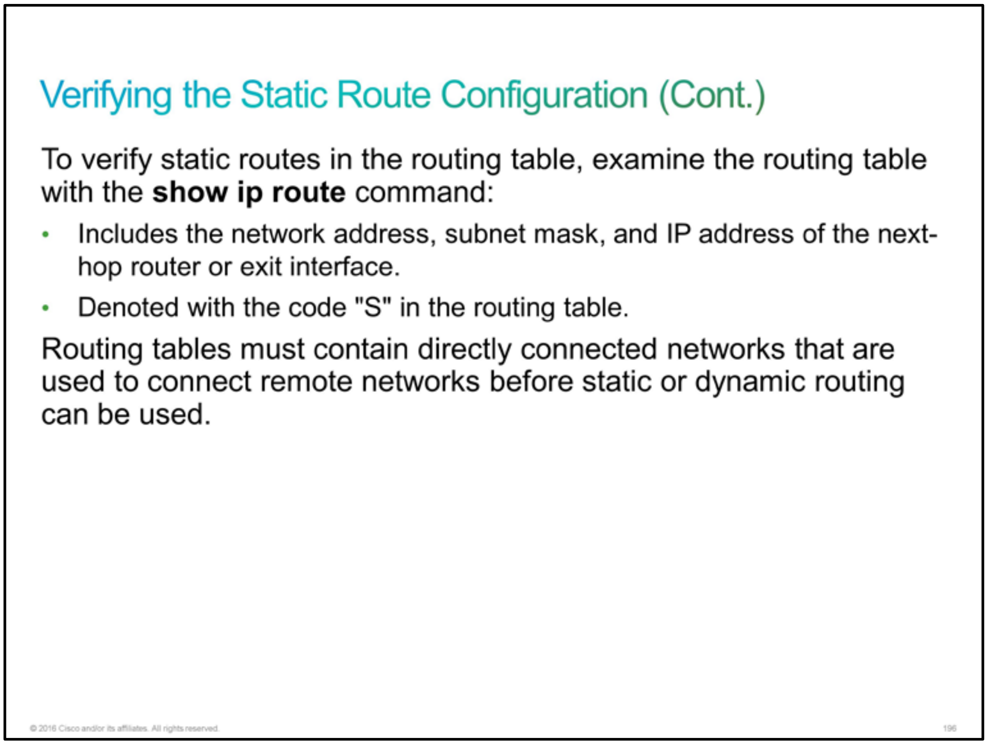
Verifying the Static Route Configuration

Most routing tables contain a combination of static routes and dynamic routes. However, the routing table must first contain the directly connected networks that are used to access the remote networks before any static or dynamic routing can be used.



Verifying the Static Route Configuration

```
RouterA# show ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
<... output omitted ...>
Gateway of last resort is not set

   10.0.0.0/24 is subnetted, 1 subnets
C    10.0.0.0 is directly connected, FastEthernet0/0
   172.16.0.0/24 is subnetted, 2 subnets
S    172.16.1.0/24 [1/0] via 172.16.2.1
C    172.16.2.0/24 is directly connected, Serial0/0/0
L    172.16.2.2/32 is directly connected, Serial0/0/0
```



Verifying the Static Route Configuration (Cont.)

To verify static routes in the routing table, examine the routing table with the **show ip route** command:

- Includes the network address, subnet mask, and IP address of the next-hop router or exit interface.
- Denoted with the code "S" in the routing table.

Routing tables must contain directly connected networks that are used to connect remote networks before static or dynamic routing can be used.

A static route includes the network address and subnet mask of the remote network, along with the IP address of the next-hop router or exit interface. Static routes are denoted with the code "S" in the routing table, as shown in the figure.

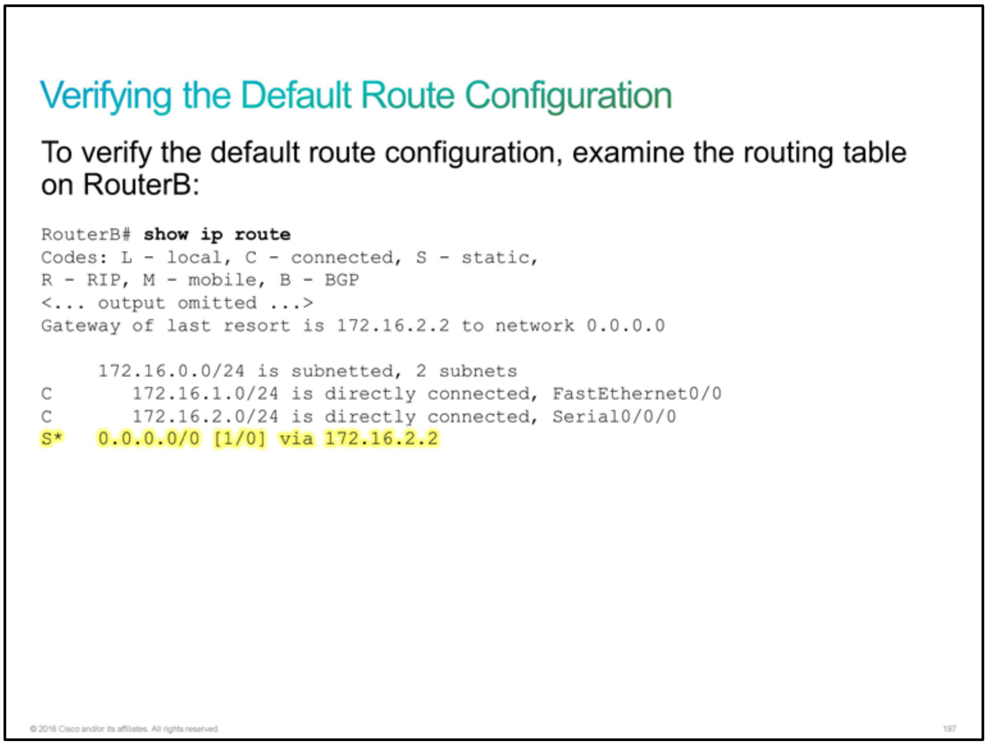
When you configure a static route to use an exit interface instead of a next-hop IP address, the routing table entry is changed as follows:

```
RouterB# show ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       + - replicated route, % - next hop override

Gateway of last resort is 172.16.2.2 to network 0.0.0.0

S*    0.0.0.0/0 [1/0] via 172.16.2.2
      172.16.0.0/16 is variably subnetted, 4 subnets, 2 masks
C        172.16.1.0/24 is directly connected, Ethernet0/0
L        172.16.1.1/32 is directly connected, Ethernet0/0
C        172.16.2.0/24 is directly connected, Serial2/0
L        172.16.2.1/32 is directly connected, Serial2/0
RouterB#
```

Note that the entry in the routing table no longer refers to the next-hop IP address but refers directly to the exit interface. This exit interface is the same one to which the static route was resolved when it used the next-hop IP address. Now that the routing table process has a match for a packet and this static route, it is able to resolve the route to an exit interface in a single lookup.

Note The static route displays the route as directly connected. It is important to understand that this does not mean that this route is a directly connected network or a directly connected route. This route is still a static route.

Verifying the Default Route Configuration



Verifying the Default Route Configuration

To verify the default route configuration, examine the routing table on RouterB:

```
RouterB# show ip route
Codes: L - local, C - connected, S - static,
R - RIP, M - mobile, B - BGP
<... output omitted ...>
Gateway of last resort is 172.16.2.2 to network 0.0.0.0

     172.16.0.0/24 is subnetted, 2 subnets
C       172.16.1.0/24 is directly connected, FastEthernet0/0
C       172.16.2.0/24 is directly connected, Serial0/0/0
S*   0.0.0.0/0 [1/0] via 172.16.2.2
```

197

The example in the figure shows the RouterB routing table after configuration of the default route. The asterisk (*) indicates the last path option that the router will use when forwarding a packet.

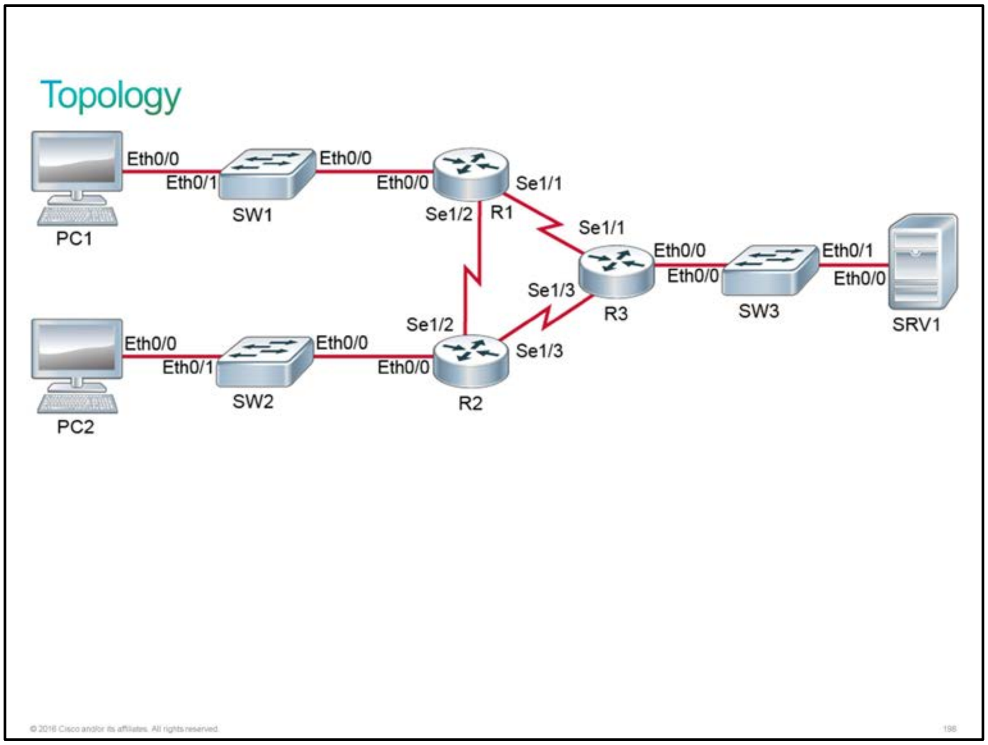
Discovery 10: Configure and Verify Static Routes

Introduction

In this discovery lab, you will explore IP routing, focusing on static routing. You will configure and verify static routes and observe the packet forwarding behavior that is associated with various routing configurations, including the use of a statically defined default route.

The lab is prepared with the devices as represented in the topology diagram and connectivity table. All devices have their basic configurations in place including hostnames and IP addresses. Default gateways are defined on PC1, PC2, and SRV1, but no other routing has been configured.

Topology



Job Aid

Device Information

Device Information Table

Device	Characteristic	Value
PC1	Hostname	PC1
PC1	IP address	10.10.1.10/24
PC1	Default gateway	10.10.1.1

Device	Characteristic	Value
PC2	Hostname	PC2
PC2	IP address	10.10.2.20/24
PC2	Default gateway	10.10.2.1
SRV1	Hostname	SRV1
SRV1	IP address	10.10.3.30/24
SRV1	Default gateway	10.10.3.1
SW1	Hostname	SW1
SW1	VLAN 1 IP address	10.10.1.4/24
SW1	Default gateway	10.10.1.1
SW1	Ethernet0/0 description	Link to R1
SW1	Ethernet0/1 description	Link to PC1
SW2	Hostname	SW2
SW2	VLAN 1 IP address	10.10.2.4/24
SW2	Default gateway	10.10.2.1
SW2	Ethernet0/0 description	Link to R2
SW2	Ethernet0/1 description	Link to PC2
SW3	Hostname	SW3
SW3	VLAN 1 IP address	10.10.3.4/24
SW3	Default gateway	10.10.3.1
SW3	Ethernet0/0 description	Link to R3
SW3	Ethernet0/1 description	Link to SRV1
R1	Hostname	R1
R1	Ethernet0/0 description	Link to SW1
R1	Ethernet0/0 IP address	10.10.1.1/24
R1	Serial1/1 description	Link to R3

Device	Characteristic	Value
R1	Serial1/1 IP address	10.1.1.2/30
R1	Serial1/2 description	Link to R2
R1	Serial1/2 IP address	10.1.1.10/30
R2	Hostname	R2
R2	E0/0 description	Link to SW2
R2	E0/0 IP address	10.10.2.1/24
R2	S1/2 description	Link to R1
R2	S1/2 IP address	10.1.1.9/30
R2	S1/3 description	Link to R3
R2	S1/3 IP address	10.1.1.6/30
R3	Hostname	R3
R3	Ethernet0/0 description	Link to SW3
R3	Ethernet0/0 IP address	10.10.3.1/24
R3	Serial1/1 description	Link to R1
R3	Serial1/1 IP address	10.1.1.1/30
R3	Serial1/3 description	Link to R2
R3	Serial1/3 IP address	10.1.1.5/30

PCs and SRV in the virtual lab environment are simulated as routers, so you should use Cisco IOS commands to configure them or make verifications.

Task 1: Verify Devices Reachability

Activity

Step 1 Before getting into the configuration of static routes, observe the connectivity when routing is not yet configured on any of the routers.

Access the console of PC1 and ping SW1 and R1.

```
PC1# ping 10.10.1.4
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.1.4, timeout is 2 seconds:
.!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 1/1/1 ms
PC1# ping 10.10.1.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.1.1, timeout is 2 seconds:
.!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 1/1/2 ms
```

Be sure to take advantage of the Cisco IOS command recall feature when entering similar commands. Use the **Up Arrow** key to scroll through the command history and use the **Right** and **Left Arrow** and **Backspace** keys to edit commands that are similar to what you need to enter.

Consult the topology diagram whenever it is helpful to clarify the physical layout of the lab.

You probably expected to be able to ping these addresses. They are on the same subnet as PC1. So, routing is not required. The ARP protocol resolves the MAC address of the peer and communication ensues at Layer 2.

Step 2 Personal computers, and IP end hosts in general, normally have routing tables. They usually consist of a single entry, a default route to their default gateway. View the routing table on PC1 to verify that R1 is its default gateway.

On PC1, enter following command:

```
PC1# show ip route
Default gateway is 10.10.1.1

Host               Gateway           Last Use    Total Uses  Interface
ICMP redirect cache is empty
```

Step 3 From PC1, ping the IP addresses of the remote Serial1/1 and Serial1/2 interfaces of R1.

On PC1, enter following commands:

```
PC1# ping 10.1.1.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.1.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/201/1002 ms
PC1# ping 10.1.1.10
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.1.10, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
```

The ping worked without any routes configured on R1. PC1 was configured to send all remote traffic to R1, and R1 has all the respective subnets (including the subnet of PC1) in its routing table as directly connected networks.

Step 4 Try to ping R2’s Serial1/2 interface, which is a point-to-point neighbor to the Serial1/2 interface of R1.

On PC1, enter following command:

```
PC1# ping 10.1.1.9
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.1.9, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)
```

This ping attempt fails. Interestingly, the ICMP echo requests actually do make it to R2. PC1 is configured to use R1 as its default gateway and R1 has 10.1.1.0/30 as a directly connected network in its routing table. So, the forwarding to R2 will function. The problem is that R2 does not have a route back to 10.10.1.0/24 and as a result cannot forward the replies to R1. R2 drops the ICMP packet.

Task 2: Configure and Verify Static Routes

Activity

Step 1 Now it is time to configure some static routes. On the R1 router, configure routes to 10.10.2.0/24 and 10.10.3.0/24 through R2 and R3 respectively.

On R1, enter the following commands:

```
R1# conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)# ip route 10.10.2.0 255.255.255.0 10.1.1.9
R1(config)# ip route 10.10.3.0 255.255.255.0 10.1.1.1
R1(config)# end
R1#
```

Step 2 On R2, configure routes to 10.10.1.0/24 and 10.10.3.0/24 through R1 and R3 respectively.

On R2, enter the following commands:

```
R2# conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)# ip route 10.10.1.0 255.255.255.0 10.1.1.10
R2(config)# ip route 10.10.3.0 255.255.255.0 10.1.1.5
R2(config)# end
R2#
```

Step 3 Consult the topology diagram and consider the static routes that you just configured. Should PC1 be able to ping PC2? How about SRV1? And how about 10.1.1.6 or 10.1.1.5 (the IP addresses on the subnet between R2 and R3)? Access the console of PC1 and explore the current connectivity.

On PC1, enter the following commands:

```
PC1# ping 10.10.2.20
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.2.20, timeout is 2 seconds:
.!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 1/1/1 ms
```

PC1 can ping PC2. This fact implies bidirectional connectivity. The forwarding ICMP echoes from PC1 to PC2 was successful and the forwarding of ICMP echo replies from PC2 to PC1 was successful.

```
PC1# ping 10.10.3.30
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.3.30, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)
```

This ping attempt was not successful. With the current route configuration, the ICMP echoes will actually reach SRV1. R1 has a route to 10.10.3.0/24 using R3, and R3 has an interface that is directly connected to 10.10.3.0/24. But, the ICMP echo replies that SRV1 generated will be sent to R3 (the default gateway of SRV1), but R3 does not have a route back to 10.10.1.0/24. So, it will drop the echo replies.

The "." characters in the ping output indicate timeouts on the reply

```
PC1# ping 10.1.1.6
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.1.6, timeout is 2 seconds:
U.U.U
Success rate is 0 percent (0/5)
```

This ping attempt fails for a different reason. The subnet 10.1.1.4/30 is the point to point link between R2 and R3. R1 does not have a route to that subnet. Hence, it must drop the packets that are destined for that subnet.

The "U" characters in the ping output indicate that a router in the forwarding path returned ICMP Unreachable messages to PC1.

Step 4 Consult the topology diagram. There are six subnets. Each router has direct connectivity to three of those subnets with the remaining three subnets being remote to the router. For full connectivity, each router must have a route defined for each of the three remote subnets. Configure the third static route on both R1 and R2, and configure all three routes on R3.

On R1, enter the following commands:

```
R1# conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)# ip route 10.1.1.4 255.255.255.252 10.1.1.9
R1(config)# end
R1#
```

On R1 and R2, if your login session has not timed out, command recall will still function when you enter configuration mode, providing access to the previously entered route commands. But, be careful! The routes have different subnet masks, so you must change them along with the IP addresses!

On R2, enter the following commands:

```
R2# conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)# ip route 10.1.1.0 255.255.255.252 10.1.1.10
R2(config)# end
R2#
```

The next hop specified (10.1.1.10) is an interface on R1. 10.1.1.5 on R3 would have been an equivalent option for the next hop. The choice to use R1 as the next hop was arbitrary.

On R3, enter the following commands:

```
R3# conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R3(config)# ip route 10.10.1.0 255.255.255.0 10.1.1.2
R3(config)# ip route 10.10.2.0 255.255.255.0 10.1.1.6
R3(config)# ip route 10.1.1.8 255.255.255.252 10.1.1.2
R3(config)# end
R3#
```

Step 5 Now is a good time to verify the routing tables on all three routers.

This example shows the routing table and configuration on R1. R2 and R3 should have similar, complimentary routes configured.

```
R1# show ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       + - replicated route, % - next hop override

Gateway of last resort is not set

      10.0.0.0/8 is variably subnetted, 9 subnets, 3 masks
C        10.1.1.0/30 is directly connected, Serial1/1
L        10.1.1.2/32 is directly connected, Serial1/1
S        10.1.1.4/30 [1/0] via 10.1.1.9
C        10.1.1.8/30 is directly connected, Serial1/2
L        10.1.1.10/32 is directly connected, Serial1/2
C        10.10.1.0/24 is directly connected, Ethernet0/0
L        10.10.1.1/32 is directly connected, Ethernet0/0
S        10.10.2.0/24 [1/0] via 10.1.1.9
S        10.10.3.0/24 [1/0] via 10.1.1.1
R1# show running-config | include route
ip route 10.1.1.4 255.255.255.252 10.1.1.9
ip route 10.10.2.0 255.255.255.0 10.1.1.9
ip route 10.10.3.0 255.255.255.0 10.1.1.1
```

Step 6 At this point, all three routers have routes (either directly connected or statically defined) to all six subnets. Full connectivity should be available.

Access the console of PC1 and verify that IP addresses from the different subnets are reachable with the **ping** command.

```
PC1# ping 10.10.2.20
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.2.20, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
PC1# ping 10.10.3.30
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.3.30, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
PC1# ping 10.1.1.6
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.1.6, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/2 ms
PC1# ping 10.1.1.5
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.1.5, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/2 ms
```

Task 3: Demonstrate Static Route Drawback

Activity

Step 1 The pings demonstrate that there is connectivity. Use the **traceroute** command to verify the paths that are in place.

On PC1, enter the following commands:

```
PC1# traceroute 10.10.2.20
Type escape sequence to abort.
Tracing the route to 10.10.2.20
VRF info: (vrf in name/id, vrf out name/id)
  1 10.10.1.1 1 msec 1 msec 1 msec
  2 10.1.1.9 1 msec 0 msec 1 msec
  3 10.10.2.20 2 msec *  2 msec
```

The path from PC1 to PC2 goes through R1 and R2.

```
PC1# traceroute 10.10.3.30
Type escape sequence to abort.
Tracing the route to 10.10.3.30
VRF info: (vrf in name/id, vrf out name/id)
  1 10.10.1.1 1 msec 1 msec 1 msec
  2 10.1.1.1 1 msec 0 msec 0 msec
  3 10.10.3.30 1 msec *  1 msec
```

It is normal in the lab environment for the middle attempt to the final destination to time out with the **traceroute** command.

The path from PC1 to SRV1 goes through R1 and R3.

At this point, one of the limitations of static routes should be apparent. They do not scale well. In the lab, there are only six subnets and three routers with no path being longer than two hops. In this simple environment, it required nine static routes for full connectivity. As the network complexity grows, the number of required static routes grows very fast and quickly becomes unwieldy.

In the next series of steps, you will experience another limitation of static routes. They do not provide redundancy. Introduce an interface fault into the network.

Step 2 On R3, disable the interface Serial1/1, which connects R3 to R1.

On R3, enter the following commands:

```
R3# conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R3(config)# interface Serial 1/1
R3(config-if)# shutdown
R3(config-if)# end
R3#
*Oct 15 07:04:28.078: %SYS-5-CONFIG_I: Configured from console by console
R3#
*Oct 15 07:04:29.292: %LINK-5-CHANGED: Interface Serial1/1, changed state to
administratively down
*Oct 15 07:04:30.296: %LINEPROTO-5-UPDOWN: Line protocol on Interface
Serial1/1, changed state to down
```

Step 3 Shutting down Serial1/1 on R3 will have effects on R1 and R3. Access the console of R1 and verify that a syslog message is displayed, indicating that the interface Serial1/1 has changed status to "down."

In the lab environment, this status change may take a minute to propagate.

```
R1#
*Oct 15 07:04:57.975: %LINEPROTO-5-UPDOWN: Line protocol on Interface
Serial1/1, changed state to down
```

Step 4 View the interface status and routing table on R1.

On R1, enter the following command:

```
R1# show ip interface brief
Interface                  IP-Address      OK? Method Status
Protocol
Ethernet0/0                10.10.1.1       YES NVRAM  up                    up
Ethernet0/1                unassigned      YES NVRAM  administratively down
down
Ethernet0/2                unassigned      YES NVRAM  administratively down
down
Ethernet0/3                unassigned      YES NVRAM  administratively down
down
Serial1/0                  unassigned      YES NVRAM  administratively down
down
Serial1/1                  10.1.1.2        YES NVRAM  up
down
Serial1/2                  10.1.1.10       YES NVRAM  up                    up
Serial1/3                  unassigned      YES NVRAM  administratively down
down
```

The protocol status of Serial1/1 is "down."

```
R1# show ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       + - replicated route, % - next hop override

Gateway of last resort is not set

      10.0.0.0/8 is variably subnetted, 6 subnets, 3 masks
S        10.1.1.4/30 [1/0] via 10.1.1.9
C        10.1.1.8/30 is directly connected, Serial1/2
L        10.1.1.10/32 is directly connected, Serial1/2
C        10.10.1.0/24 is directly connected, Ethernet0/0
L        10.10.1.1/32 is directly connected, Ethernet0/0
S        10.10.2.0/24 [1/0] via 10.1.1.9
```

There are only two static routes in the routing table. With Serial1/1 being down, there is no path to 10.1.1.1 on the 10.1.1.0/30 subnet. So, the route to 10.10.3.0/24 that uses 10.1.1.1 as the next hop is invalid and has been removed from the routing table.

The **static route** command still exists in the configuration.

Step 5 Explore the connectivity from the perspective of PC1. Access the console of PC1 and attempt a **ping** and a **traceroute** to 10.10.3.30.

On PC1, enter the following command:

```
PC1# ping 10.10.3.30
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.3.30, timeout is 2 seconds:
U.U.U
Success rate is 0 percent (0/5)
```

The ping is not successful. The "U" characters indicate that a router in the path (in this case R1) is sending an ICMP unreachable message back to PC1.

```
PC1# traceroute 10.10.3.30
Type escape sequence to abort.
Tracing the route to 10.10.3.30
VRF info: (vrf in name/id, vrf out name/id)
  1 10.10.1.1 1 msec 1 msec 0 msec
  2 10.10.1.1 !H  *  !H
```

The path gets to R1 (10.10.1.1), but then gets stuck.

Step 6 Repair the interface fault by returning to R3 and enabling Serial1/1.

On R3, enter the following commands:

```
R3# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R3(config)# interface Serial 1/1
R3(config-if)# no shutdown
R3(config-if)# end
R3#
*Oct 15 07:13:12.022: %LINK-3-UPDOWN: Interface Serial1/1, changed state to up
R3#
*Oct 15 07:13:12.747: %SYS-5-CONFIG_I: Configured from console by console
*Oct 15 07:13:13.027: %LINEPROTO-5-UPDOWN: Line protocol on Interface
Serial1/1, changed state to up
```

Step 7 Return to the console of R1 and verify that the display of the syslog message is indicating that Serial1/1 has changed back to the "up" state.

```
R1#
*Oct 15 07:13:18.148: %LINEPROTO-5-UPDOWN: Line protocol on Interface
Serial1/1, changed state to up
```

Step 8 The real proof comes by verifying end to end connectivity. Return to the console of PC1 and execute a **ping** and a **traceroute** command to SRV1.

On PC1, enter the following commands:

```
PC1# ping 10.10.3.30
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.3.30, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 10/10/11 ms
PC1# traceroute 10.10.3.30
Type escape sequence to abort.
Tracing the route to 10.10.3.30
VRF info: (vrf in name/id, vrf out name/id)
  1 10.10.1.1 1 msec 1 msec 1 msec
  2 10.1.1.1 10 msec 10 msec 9 msec
  3 10.10.3.30 11 msec *  9 msec
```

The path from PC1 to SRV1 now, again travels through R1 and R3.

Task 4: Configure and Verify the Backup Static Route

Activity

What has been demonstrated so far in this discovery covers the typical usage of static routes. The next series of steps will show an unconventional use of static routes. This task should be considered an academic exercise. It is only feasible because of the simplicity of the lab environment. Add more routers or subnets into the mix and this methodology would quickly become unwieldy.

Administrative distance is a property that is used to distinguish the trustworthiness of different routing protocols. Cisco IOS routers prefer routes with a lower administrative distance. By default static routes have an administrative distance of 1, which all but guarantees that they will be used in the routing table.

It is optional to specify a different administrative distance on static routes. In this next series of steps, you will define a set of backup static routes with an administrative distance of 2. The only way these routes will end up in the routing table is if one of the routes with an administrative distance of 1 becomes unavailable. You will also verify the behavior of when an interface fails in the new configuration.

Step 1 Access the console of R1 and add three static routes. The new routes will specify the same remote networks as the existing static routes, but they will specify a next hop on the alternate peer router and specify and administrative distance of 2.

On R1, enter the following commands:

```
R1# conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)# ip route 10.10.2.0 255.255.255.0 10.1.1.1 2
R1(config)# ip route 10.10.3.0 255.255.255.0 10.1.1.9 2
R1(config)# ip route 10.1.1.4 255.255.255.252 10.1.1.1 2
R1(config)# end
R1#
```

Step 2 Verify that there are now six static routes in the configuration. There are two to each of the remote networks. The second route to each remote network specifies an alternate next hop and an administrative distance of 2.

On R1, enter the following commands:

```
R1# show running-config | include route
ip route 10.1.1.4 255.255.255.252 10.1.1.9
ip route 10.1.1.4 255.255.255.252 10.1.1.1 2
ip route 10.10.2.0 255.255.255.0 10.1.1.9
ip route 10.10.2.0 255.255.255.0 10.1.1.1 2
ip route 10.10.3.0 255.255.255.0 10.1.1.1
ip route 10.10.3.0 255.255.255.0 10.1.1.9 2
```

Step 3 Verify that only three of the static routes appear in the routing table. Only the routes that have the default administrative distance of 1 are selected for the routing table.

On R1, enter the following command:

```
R1# show ip route static
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       + - replicated route, % - next hop override

Gateway of last resort is not set

      10.0.0.0/8 is variably subnetted, 9 subnets, 3 masks
S        10.1.1.4/30 [1/0] via 10.1.1.9
S        10.10.2.0/24 [1/0] via 10.1.1.9
S        10.10.3.0/24 [1/0] via 10.1.1.1
```

The values that you see within the brackets are [Administrative Distance / Metric]. These three routes all have an administrative distance of 1.

Step 4 Repeat the respective configuration of static routes on R2 and R3.

On R2, enter the following commands:

```
R2# conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)# ip route 10.10.1.0 255.255.255.0 10.1.1.5 2
R2(config)# ip route 10.10.3.0 255.255.255.0 10.1.1.10 2
R2(config)# ip route 10.1.1.0 255.255.255.252 10.1.1.5 2
R2(config)# end
R2#
```

On R3, enter the following commands:

```
R3# conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R3(config)# ip route 10.10.1.0 255.255.255.0 10.1.1.6 2
R3(config)# ip route 10.10.2.0 255.255.255.0 10.1.1.2 2
R3(config)# ip route 10.1.1.8 255.255.255.252 10.1.1.6 2
R3(config)# end
R3#
```

Step 5 Repeat the fault experiment that was performed earlier in the discovery by disabling interface Serial1/1 on R3.

On R3, enter the following commands:

```
R3# conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R3(config)# interface Serial 1/1
R3(config-if)# shutdown
R3(config-if)# end
R3#
*Oct 15 07:29:34.297: %SYS-5-CONFIG_I: Configured from console by console
*Oct 15 07:29:35.080: %LINK-5-CHANGED: Interface Serial1/1, changed state to
administratively down
R3#
*Oct 15 07:29:36.084: %LINEPROTO-5-UPDOWN: Line protocol on Interface
Serial1/1, changed state to down
```

Step 6 Access the console of R1 to verify that the display of the syslog message indicating that the Serial1/1 interface of R1 has "changed state to down."

```
R1#
*Oct 15 07:29:58.519: %LINEPROTO-5-UPDOWN: Line protocol on Interface
Serial1/1, changed state to down
```

Step 7 View the routing table on R1.

On R1, enter the following command:

```
R1# show ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       + - replicated route, % - next hop override

Gateway of last resort is not set

      10.0.0.0/8 is variably subnetted, 7 subnets, 3 masks
S        10.1.1.4/30 [1/0] via 10.1.1.9
C        10.1.1.8/30 is directly connected, Serial1/2
L        10.1.1.10/32 is directly connected, Serial1/2
C        10.10.1.0/24 is directly connected, Ethernet0/0
L        10.10.1.1/32 is directly connected, Ethernet0/0
S        10.10.2.0/24 [1/0] via 10.1.1.9
S        10.10.3.0/24 [2/0] via 10.1.1.9
```

There is a route to 10.10.3.0/24. The route through R2 (10.1.1.9) with an administrative distance of 2 replaced the route through R3 (10.1.1.1) with an administrative distance of 1 when the connection to the 10.1.1.0/30 network was lost.

Step 8 Access the console of PC1 and verify connectivity between PC1 and SRV1 using the **ping** and **traceroute** commands.

On PC1, enter the following commands:

```
PC1# ping 10.10.3.30
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.3.30, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 17/17/18 ms
PC1# traceroute 10.10.3.30
Type escape sequence to abort.
Tracing the route to 10.10.3.30
VRF info: (vrf in name/id, vrf out name/id)
  1 10.10.1.1 1 msec 0 msec 1 msec
  2 10.1.1.9 9 msec 9 msec 9 msec
  3 10.1.1.5 17 msec 18 msec 17 msec
  4 10.10.3.30 15 msec *  18 msec
```

Connectivity remains even with the loss of the link between R1 and R3. The path is now longer. The path from PC1 to SRV1 traverses R1, R2, and R3.

Step 9 Return to R3 to repair the interface fault.

On R3, enter the following commands:

```
R3# conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R3(config)# interface Serial 1/1
R3(config-if)# no shutdown
R3(config-if)# end
R3#
*Oct 15 07:34:30.570: %SYS-5-CONFIG_I: Configured from console by console
R3#
*Oct 15 07:34:30.968: %LINK-3-UPDOWN: Interface Serial1/1, changed state to up
*Oct 15 07:34:31.972: %LINEPROTO-5-UPDOWN: Line protocol on Interface
Serial1/1, changed state to up
```

Step 10 Access the console of R1 and verify the display of the syslog message indicating that its interface Serial1/1 returns to an "up" state.

```
R1#
*Oct 15 07:34:38.628: %LINEPROTO-5-UPDOWN: Line protocol on Interface
Serial1/1, changed state to up
```

Step 11 View the routing table on R1.

On R1, enter the following command:

```
R1# show ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       + - replicated route, % - next hop override

Gateway of last resort is not set

      10.0.0.0/8 is variably subnetted, 9 subnets, 3 masks
C        10.1.1.0/30 is directly connected, Serial1/1
L        10.1.1.2/32 is directly connected, Serial1/1
S        10.1.1.4/30 [1/0] via 10.1.1.9
C        10.1.1.8/30 is directly connected, Serial1/2
L        10.1.1.10/32 is directly connected, Serial1/2
C        10.10.1.0/24 is directly connected, Ethernet0/0
L        10.10.1.1/32 is directly connected, Ethernet0/0
S        10.10.2.0/24 [1/0] via 10.1.1.9
S        10.10.3.0/24 [1/0] via 10.1.1.1
```

The original route to 10.10.3.0 using R3 as the next hop and with an administrative distance of 1 has returned to the routing table.

Task 5: Configure and Verify the Default Route

Activity

A default route is a route to the network 0.0.0.0 with the subnet mask 0.0.0.0. Default routes can be defined statically. Default routes are most commonly used when there is a hierarchy in the network. For example, to get from a branch office network to the headquarters network (and the rest of the world), or to get from the corporate network to the Internet (and the rest of the world).

The lab environment is not hierarchical. In fact, it is perfectly symmetrical. So the use of a default route on R1, R2, or R3 is not very practical. But, even so, it can be enlightening to explore the behavior of a default route within the lab environment.

Step 1 Access R1 and remove all the static routes that are configured. Unfortunately, removing those routes is a tedious operation. Be sure to make good use of the Cisco IOS command history feature to ease the burden.

On R1, enter the following commands:

```
R1# conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)# no ip route 10.1.1.4 255.255.255.252 10.1.1.9
R1(config)# no ip route 10.1.1.4 255.255.255.252 10.1.1.1 2
R1(config)# no ip route 10.10.2.0 255.255.255.0 10.1.1.9
R1(config)# no ip route 10.10.2.0 255.255.255.0 10.1.1.1 2
R1(config)# no ip route 10.10.3.0 255.255.255.0 10.1.1.1
R1(config)# no ip route 10.10.3.0 255.255.255.0 10.1.1.9 2
R1(config)# end
R1#
```

Step 2 Verify that there are no route commands left in the configuration and that only local and connected routes appear in the routing table.

On R1, enter the following commands:

```
R1# show running-config | include route
R1# show ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       + - replicated route, % - next hop override

Gateway of last resort is not set

      10.0.0.0/8 is variably subnetted, 6 subnets, 3 masks
C        10.1.1.0/30 is directly connected, Serial1/1
L        10.1.1.2/32 is directly connected, Serial1/1
C        10.1.1.8/30 is directly connected, Serial1/2
L        10.1.1.10/32 is directly connected, Serial1/2
C        10.10.1.0/24 is directly connected, Ethernet0/0
L        10.10.1.1/32 is directly connected, Ethernet0/0
```

Step 3 Configure a default route using 10.1.1.1 (Serial1/1 on R3) on R1.

On R1, enter following commands:

```
R1# conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)# ip route 0.0.0.0 0.0.0.0 10.1.1.1
R1(config)# end
R1#
```

Step 4 Verify that this route is the only route in the running configuration and that there is a default route in the routing table.

On R1, enter the following commands:

```
R1# show running-config | include route
ip route 0.0.0.0 0.0.0.0 10.1.1.1
R1# show ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       + - replicated route, % - next hop override

Gateway of last resort is 10.1.1.1 to network 0.0.0.0

S*    0.0.0.0/0 [1/0] via 10.1.1.1
      10.0.0.0/8 is variably subnetted, 6 subnets, 3 masks
C        10.1.1.0/30 is directly connected, Serial1/1
L        10.1.1.2/32 is directly connected, Serial1/1
C        10.1.1.8/30 is directly connected, Serial1/2
L        10.1.1.10/32 is directly connected, Serial1/2
C        10.10.1.0/24 is directly connected, Ethernet0/0
L        10.10.1.1/32 is directly connected, Ethernet0/0
```

Step 5 Access the console of PC1 and use the **ping** command to verify connectivity with other IP addresses in the network.

On PC1, enter the following commands:

```
PC1# ping 10.10.3.30
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.3.30, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 10/10/10 ms
PC1# ping 10.10.2.20
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.2.20, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 12/13/15 ms
PC1# ping 10.1.1.9
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.1.9, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 8/9/10 ms
```

Replacing the specific routes in the configuration of R1 with a default route to R3 does provide connectivity throughout the network as long as there are no failed interfaces.

Step 6 Examine the path from PC1 to PC2 using the **traceroute** command.

On PC1, enter the following command:

```
PC1# traceroute 10.10.2.20
Type escape sequence to abort.
Tracing the route to 10.10.2.20
VRF info: (vrf in name/id, vrf out name/id)
  1 10.10.1.1 1 msec 0 msec 1 msec
  2 10.1.1.1 9 msec 5 msec 9 msec
  3 10.1.1.6 13 msec 13 msec 13 msec
  4 10.10.2.20 14 msec *  15 msec
```

R1 is the first hop in the path. R1 no longer has an explicit and efficient static route to 10.10.2.0/24, so it uses its default route and forwards this traffic to R3. R3 has a static route to the 10.10.2.0/24 network via R2. So, the path from PC1 to PC2 goes through R1, R3, and then R2.

Step 7 Examine the path from PC1 to 10.1.1.9 (Serial1/2 on R2) using **traceroute** command.

On PC1, enter the following command:

```
PC1# traceroute 10.1.1.9
Type escape sequence to abort.
Tracing the route to 10.1.1.9
VRF info: (vrf in name/id, vrf out name/id)
  1 10.10.1.1 0 msec 0 msec 0 msec
  2 10.1.1.9 11 msec *  9 msec
```

R1 did not need to use its default route to reach 10.1.1.9. R1 has a connected route to 10.1.1.8/30 in its routing table. This specific route is preferred over the default route and is used in this case.

Step 8 You have examined connectivity and the path from PC1 to PC2. Now access the console of PC2 and examine connectivity and the path from PC2 to PC1 using the **ping** and **traceroute** commands.

On PC2, enter the following command:

```
PC2# ping 10.10.1.10
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.1.10, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 12/13/14 ms
```

The explicit static routes to 10.10.1.0/24 that are defined on R2 and R3 will sustain connectivity to that subnet within the lab.

On PC2, enter the following command:

```
PC2# traceroute 10.10.1.10
Type escape sequence to abort.
Tracing the route to 10.10.1.10
VRF info: (vrf in name/id, vrf out name/id)
  1 10.10.2.1 1 msec 0 msec 0 msec
  2 10.1.1.10 13 msec 14 msec 13 msec
  3 10.10.1.10 15 msec *  13 msec
```

R2 is the first hop in this path, and R2 has an explicit and optimized static route to 10.10.1.0/24 that uses R1 as the next hop. Hence the path from PC2 to PC1 traverses R2 and then R1. R3 is not involved.

Contrast this situation to the path that was previously displayed for PC1 to PC2. That path required the default route on R1. The path traversed R1, R3, and then R2.

When the path from Host A to Host B is not simply the reverse of the path that is taken from Host B to Host A, it is called asymmetric routing. Asymmetric routing is generally an undesirable behavior.

At this point, you have experimented extensively with static routes. You have configured typical static routes and redundant static routes and a default static route. In each case, you have seen how to verify the status of the configuration and the routing table. You have also examined the packet forwarding behavior in each case including scenarios where there is an interface fault in place. Feel free to continue to independently explore the configuration and function of static routes in the lab environment.

This is the end of the discovery lab.

Challenge

1. Routers obtain information to determine what is entered into its routing table. What sources do the routers use to obtain this information? (Choose two)

   A. Administrators / Engineers
   B. The internet / cloud infrastructure
   C. Other Routers
   D. Servers

2. Static Routes are automatically updated when the network changes. True or False?

   A. True
   B. False

3. When does one use static routes?

   A. In large networks
   B. When the network is expected to scale
   C. In a small network

4. Which one is a disadvantage of Static Routing?

   A. Scalability
   B. Security
   C. Simplicity
   D. Conserving router resources

5. Which of the following approaches is tougher to maintain?

   A. Static Routing
   B. Dynamic Routing

6. When a dynamic routing protocol is used, routers discover networks by sharing routing table information. True or False?

   A. True
   B. False

7. Routers rely on which of the following to make decisions on forwarding packets at layer 3?

   A. MAC Address Table
   B. CDP Table
   C. Routing Table

Answer Key

Challenge

1. A, C
2. B
3. C
4. A
5. A
6. A
7. C

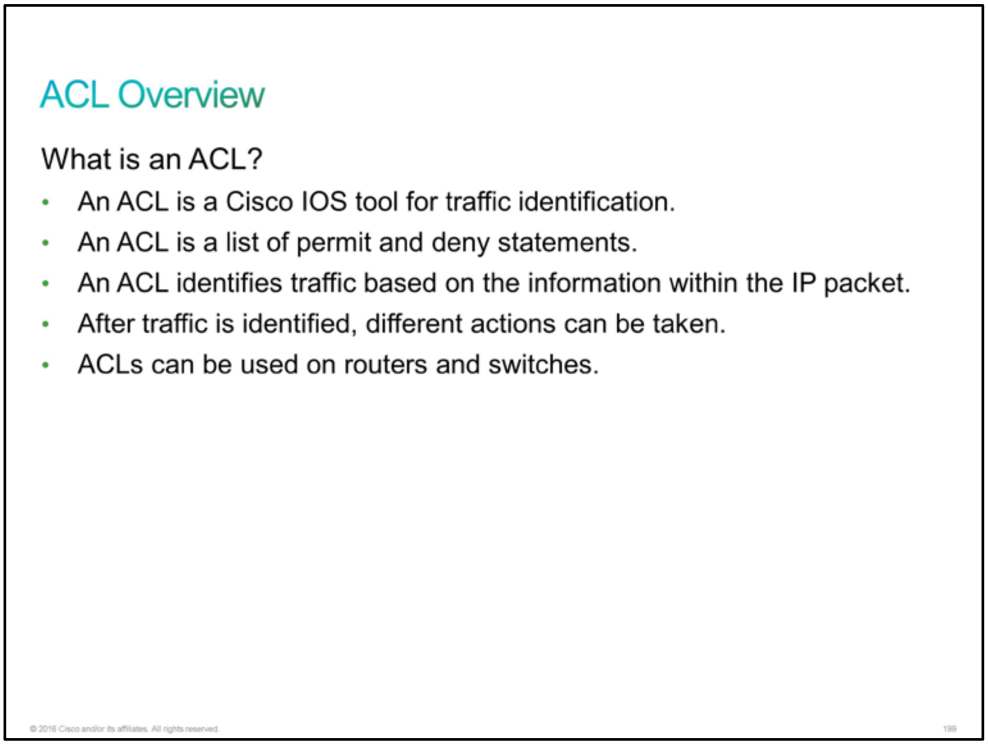
Lesson 8: Learning Basics of ACL

Introduction

Your boss sends you to your customer to update access control lists. You need to understand the general ACL operation. When updating access control list, you will need to compare wildcard masks with the subnet masks. You will also need to consider different types of ACLs. You will need to configure and verify standard numbered ACLs, which are needed for NAT and vty protection.

Introduction to ACL

An ACL is a Cisco IOS feature that is used for traffic identification. The ACL enables an administrator to create a set of rules in the form of permit and deny statements that describe which traffic should be identified.



ACL Overview

What is an ACL?

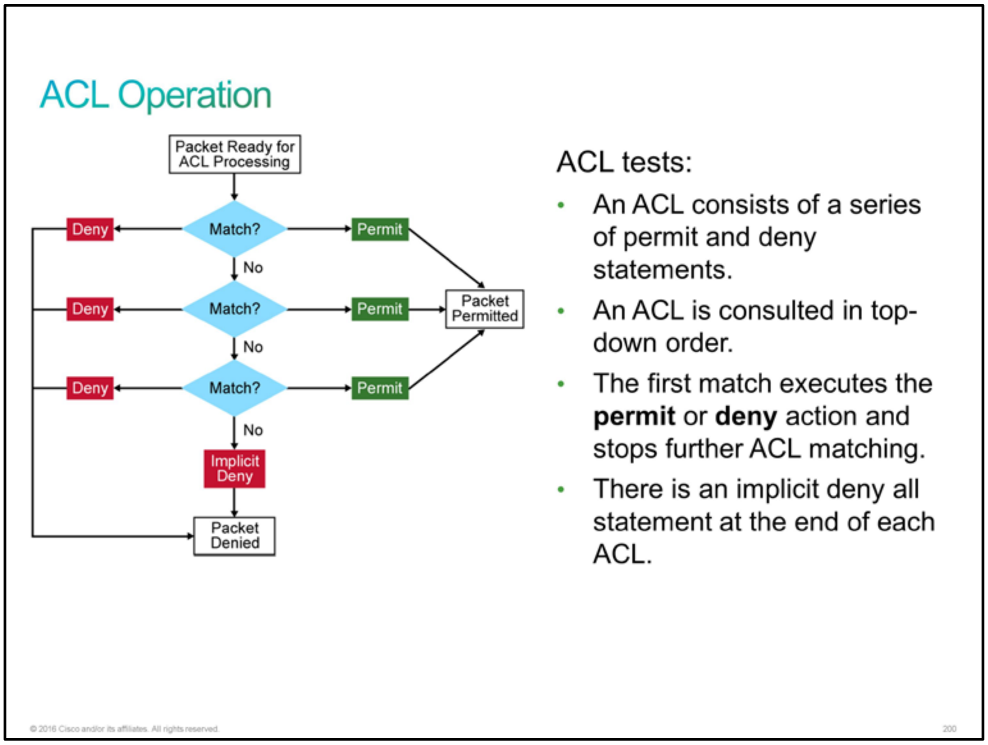
- An ACL is a Cisco IOS tool for traffic identification.
- An ACL is a list of permit and deny statements.
- An ACL identifies traffic based on the information within the IP packet.
- After traffic is identified, different actions can be taken.
- ACLs can be used on routers and switches.

Traffic identification is based on the header values in the IP packet. Identified traffic can receive different treatment, depending on which Cisco IOS function is using the ACLs.

ACLs are supported on a wide range of products, including routers and switches.

ACL Operation

ACL statements operate in a sequential, logical order. They evaluate packets from top down, one statement at a time. If a packet header and an ACL statement match, the rest of the statements in the list are skipped. The packet is then permitted or denied, as determined by the matched statement. If a packet header does not match an ACL statement, the packet is tested against the next statement in the list. This matching process continues until the end of the list is reached.



ACL Operation

ACL tests:

- An ACL consists of a series of permit and deny statements.
- An ACL is consulted in top-down order.
- The first match executes the **permit** or **deny** action and stops further ACL matching.
- There is an implicit deny all statement at the end of each ACL.

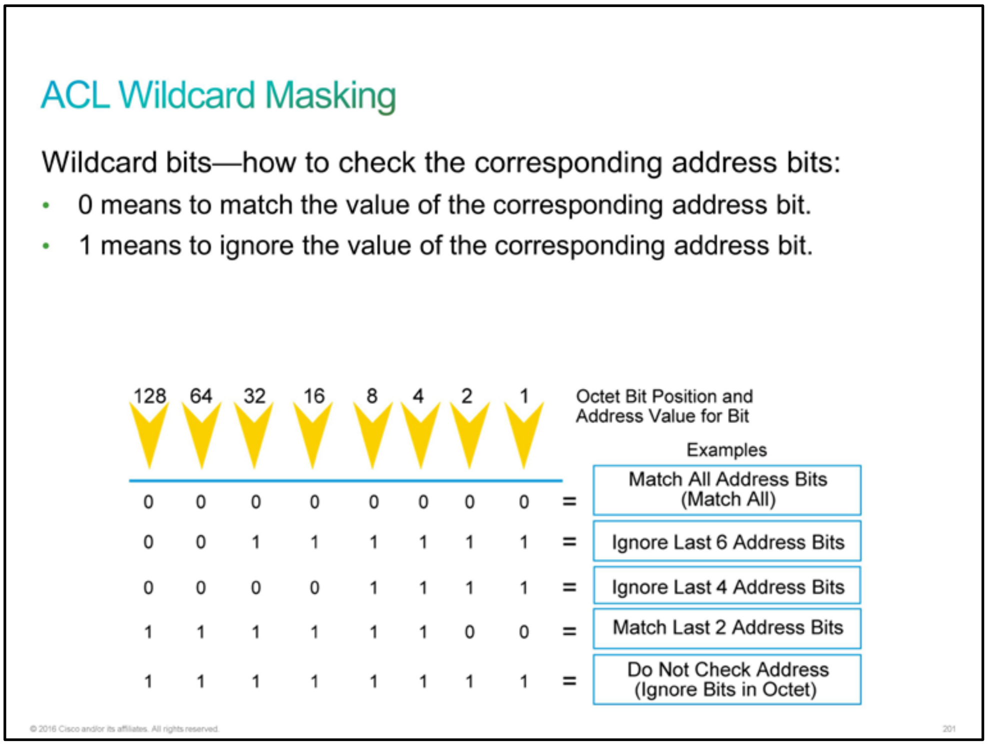
A final implied statement covers all packets for which conditions did not test true. This final test condition matches all other packets and results in a deny instruction. The router denies all these remaining packets. This final statement is often referred to as the "implicit deny any" statement. Because of this statement, an ACL should have at least one permit statement in it; otherwise, the ACL denies all packets.

ACL Wildcard Masking

When processing ACLs, a router needs a mechanism to determine which bits of an IP address must match. A wildcard mask describes which bits of an IP address must match in an IP packet to result in a match for a permit or deny statement.

ACL statements include masks, which are also called *wildcard masks*. A wildcard mask is a string of binary digits that tell the router which parts of the subnet number to look at. Although wildcard masks have no functional relationship with subnet masks, they do provide a similar function. The mask determines how much of an IP source or destination address to apply to the address match. The numbers 1 and 0 in the mask identify how to treat the corresponding IP address bits. However, they are used for different purposes and follow different rules.

Wildcard masks and subnet masks are both 32 bits long and use binary 1s and 0s. Subnet masks use binary 1s and 0s to identify the network, subnet, and host portion of an IP address. Wildcard masks use binary 1s and 0s to filter individual or groups of IP addresses to permit or deny access to resources based on an IP address. By carefully setting wildcard masks, you can permit or deny a single or several IP addresses.



ACL Wildcard Masking

Wildcard bits—how to check the corresponding address bits:

- 0 means to match the value of the corresponding address bit.
- 1 means to ignore the value of the corresponding address bit.

Octet Bit Position and Address Value for Bit

128	64	32	16	8	4	2	1		Examples
0	0	0	0	0	0	0	0	=	Match All Address Bits (Match All)
0	0	1	1	1	1	1	1	=	Ignore Last 6 Address Bits
0	0	0	0	1	1	1	1	=	Ignore Last 4 Address Bits
1	1	1	1	1	1	0	0	=	Match Last 2 Address Bits
1	1	1	1	1	1	1	1	=	Do Not Check Address (Ignore Bits in Octet)

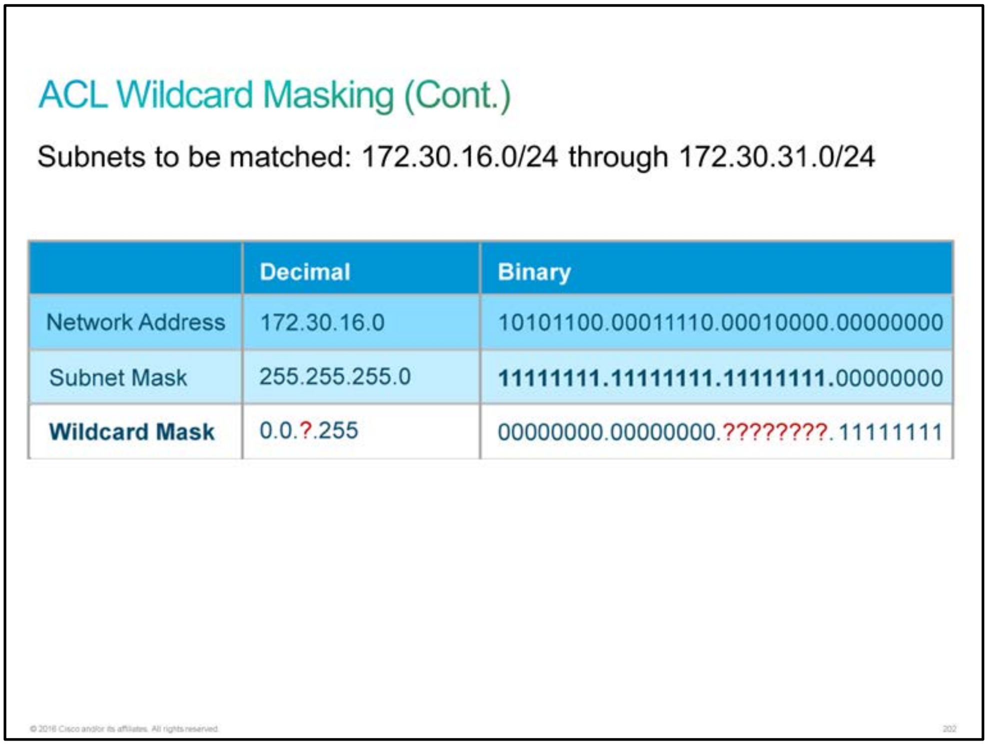
201

The figure shows how different wildcard masks filter IP addresses. As you look at the example, remember that binary 0 signifies a match and that binary 1 signifies ignore.

Note	A wildcard mask is sometimes referred to as an *inverse mask*. In a subnet mask, binary 1 is equal to a match and binary 0 is not a match. The reverse is true for wildcard masks. A 0 in a bit position of the wildcard mask indicates that the corresponding bit in the address must be matched. A 1 in a bit position of the wildcard mask indicates that the corresponding bit in the address is not interesting and can be ignored.

By carefully setting wildcard masks, you can permit or deny with one ACL statement. You can select a single IP address or many IP addresses.

Assume that you have subnetted your standard Class B address, and you want to create a wildcard mask that matches subnets 172.30.16.0/24 through 172.30.31.0/24. To use one ACL statement to match this range of subnets, use the IP address 172.30.16.0 (the first subnet to be matched) in the ACL, followed by the required wildcard mask. To better understand the process of creating the wildcard mask, look at the figure that follows. The wildcard mask must definitely match the first two octets because the numbers in those two octets are consistent throughout the subnets to be matched. Therefore, the wildcard mask must have all 0s in the first two octets. The wildcard mask must have all 1s in the last octet because it is used for host addresses, and there is no interest in individual hosts. With 1s in the last octet, the wildcard mask ignores the final octet.

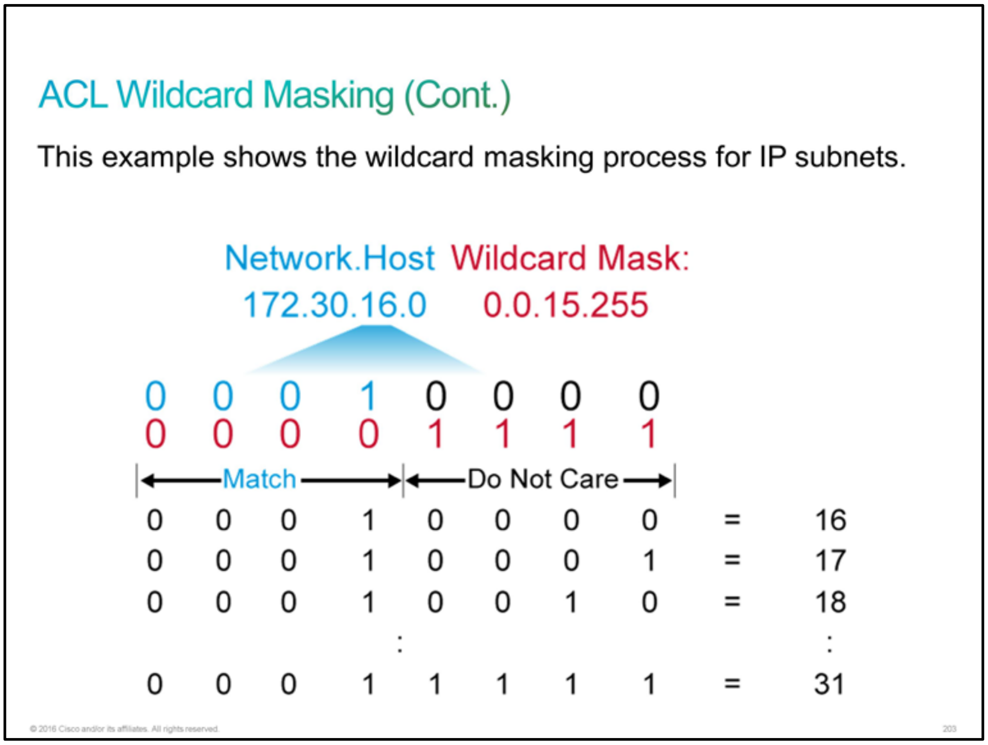


ACL Wildcard Masking (Cont.)

Subnets to be matched: 172.30.16.0/24 through 172.30.31.0/24

	Decimal	Binary
Network Address	172.30.16.0	10101100.00011110.00010000.00000000
Subnet Mask	255.255.255.0	**11111111.11111111.11111111.**00000000
Wildcard Mask	0.0.?.255	00000000.00000000.????????. 11111111

© 2016 Cisco and/or its affiliates. All rights reserved. 202



ACL Wildcard Masking (Cont.)

This example shows the wildcard masking process for IP subnets.

Network.Host Wildcard Mask:
172.30.16.0 0.0.15.255

0	0	0	1	0	0	0	0		
0	0	0	0	1	1	1	1		
Match				Do Not Care					
0	0	0	1	0	0	0	0	=	16
0	0	0	1	0	0	0	1	=	17
0	0	0	1	0	0	1	0	=	18
			:						:
0	0	0	1	1	1	1	1	=	31

203

In the figure, an administrator wants to test a range of IP subnets that will be either permitted or denied. Assume that the IP address is a Class B address (the first two octets are the network number), with 8 bits of subnetting (the third octet is for subnets). The administrator wants to use the IP wildcard masking bits to match subnets 172.30.16.0/24 to 172.30.31.0/24.

To use one ACL statement to match this range of subnets, use the IP address 172.30.16.0 in the ACL, which is the first subnet to be matched, followed by the required wildcard mask.

First, the wildcard mask matches the first two octets (172 and 30) of the IP address using corresponding 0 bits in the first two octets of the wildcard mask.

Because there is no interest in an individual host, the wildcard mask ignores the final octet by using the corresponding 1 bit in the wildcard mask. For example, the final octet of the wildcard mask is 255 in decimal.

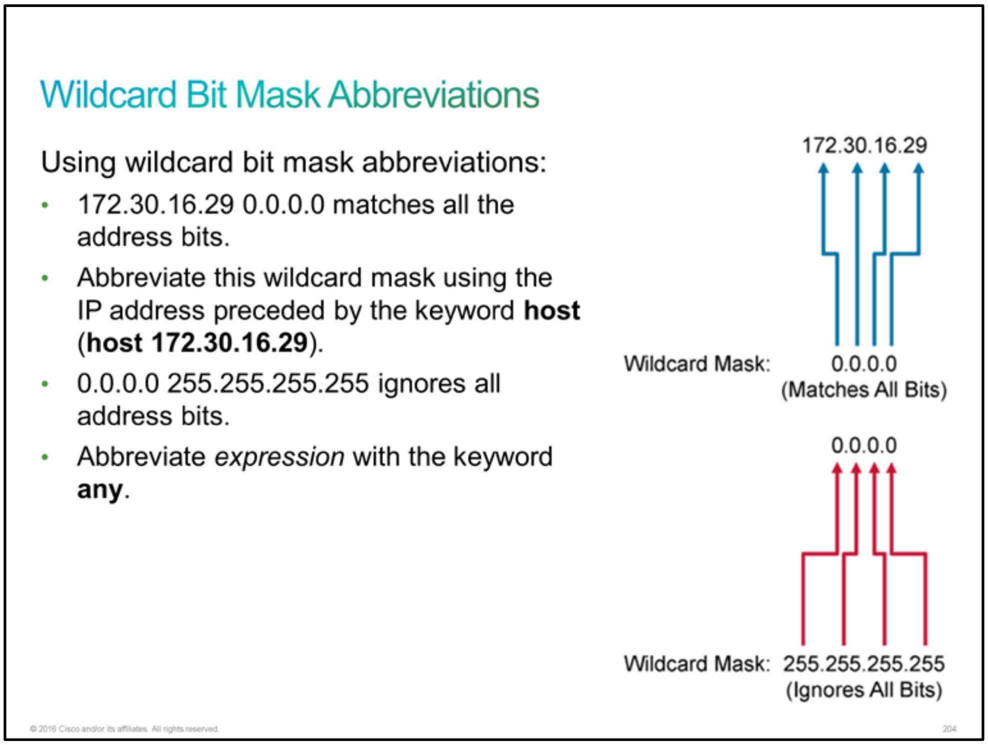
In the third octet, where the subnet address occurs, the wildcard mask of decimal 15, or binary 00001111, matches the high-order 4 bits of the IP address. In this case, the wildcard mask matches subnets starting with the 172.30.16.0/24 subnet. For the final (low-end) 4 bits in this octet, the wildcard mask indicates that the bits can be ignored. In these positions, the address value can be binary 0 or binary 1. Thus, the wildcard mask matches subnet 16, 17, 18, and so on, up to subnet 31. The wildcard mask does not match any other subnets.

In the example, the address 172.30.16.0 with the wildcard mask 0.0.15.255 matches subnets from 172.30.16.0/24 to 172.30.31.0/24.

Sometimes, you must use more than one ACL statement to match a range of subnets. For example, to match 10.1.4.0/24 to 10.1.8.0/24, use 10.1.4.0 0.0.3.255 and 10.1.8.0 0.0.0.255.

Wildcard Bit Mask Abbreviations

The 0 and 1 bit in an ACL wildcard mask cause the ACL to either match or ignore the corresponding bit in the IP address. Working with decimal representations of binary wildcard mask bits can be tedious. For the most common uses of wildcard masking, you can use abbreviations so that you do not have to enter as many numbers when configuring address test conditions.



Wildcard Bit Mask Abbreviations

Using wildcard bit mask abbreviations:

- 172.30.16.29 0.0.0.0 matches all the address bits.
- Abbreviate this wildcard mask using the IP address preceded by the keyword **host** (**host 172.30.16.29**).
- 0.0.0.0 255.255.255.255 ignores all address bits.
- Abbreviate *expression* with the keyword **any**.

172.30.16.29

Wildcard Mask: 0.0.0.0 (Matches All Bits)

0.0.0.0

Wildcard Mask: 255.255.255.255 (Ignores All Bits)

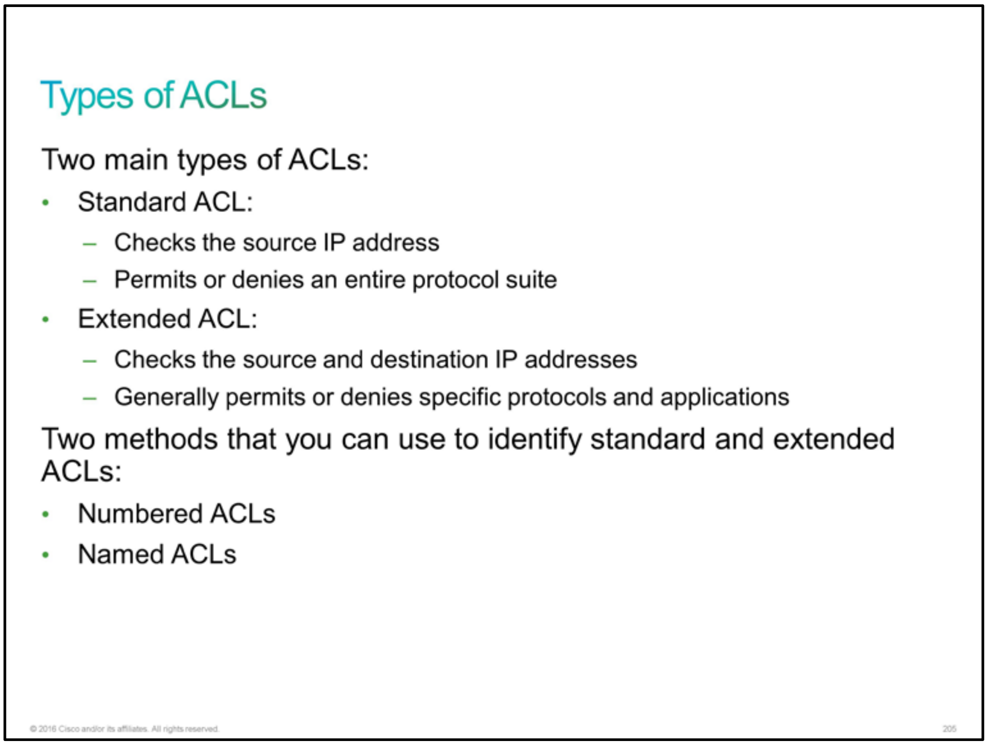
© 2016 Cisco and/or its affiliates. All rights reserved. 204

In the example, instead of entering 172.30.16.29 0.0.0.0, you can use the string **host 172.30.16.29**. Using the abbreviation **host** communicates the same test condition to Cisco IOS Software.

In the example, instead of entering 0.0.0.0 255.255.255.255, you can use the keyword **any** by itself. Using the abbreviation **any** communicates the same test condition to Cisco IOS Software.

Types of ACLs

There are two main types of ACLs, standard and extended, and two methods of identifying ACLs, numbering and naming.



Types of ACLs

Two main types of ACLs:

- Standard ACL:
  - Checks the source IP address
  - Permits or denies an entire protocol suite
- Extended ACL:
  - Checks the source and destination IP addresses
  - Generally permits or denies specific protocols and applications

Two methods that you can use to identify standard and extended ACLs:

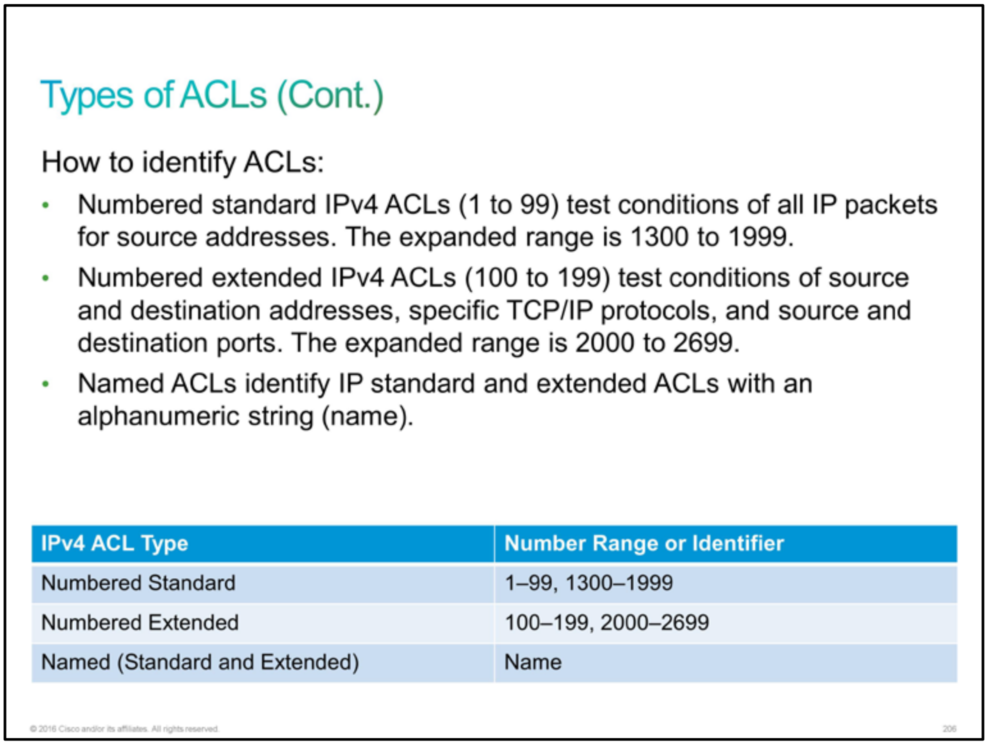
- Numbered ACLs
- Named ACLs

ACLs can be categorized into the following types:

- **Standard ACLs:** Standard IP ACLs check the source addresses of the packets that can be routed. The result either permits or denies the output for an entire protocol suite, which is based on the source network, subnet, or host IP address.
- **Extended ACLs:** Extended IP ACLs check both the source and destination packet addresses. They can also check for specific protocols, port numbers, and other parameters, which allows administrators more flexibility and control.

There are two methods that you can use to identify standard and extended ACLs:

- **Numbered ACLs:** Use a number for identification
- **Named ACLs:** Use a descriptive name or number for identification



Types of ACLs (Cont.)

How to identify ACLs:

- Numbered standard IPv4 ACLs (1 to 99) test conditions of all IP packets for source addresses. The expanded range is 1300 to 1999.
- Numbered extended IPv4 ACLs (100 to 199) test conditions of source and destination addresses, specific TCP/IP protocols, and source and destination ports. The expanded range is 2000 to 2699.
- Named ACLs identify IP standard and extended ACLs with an alphanumeric string (name).

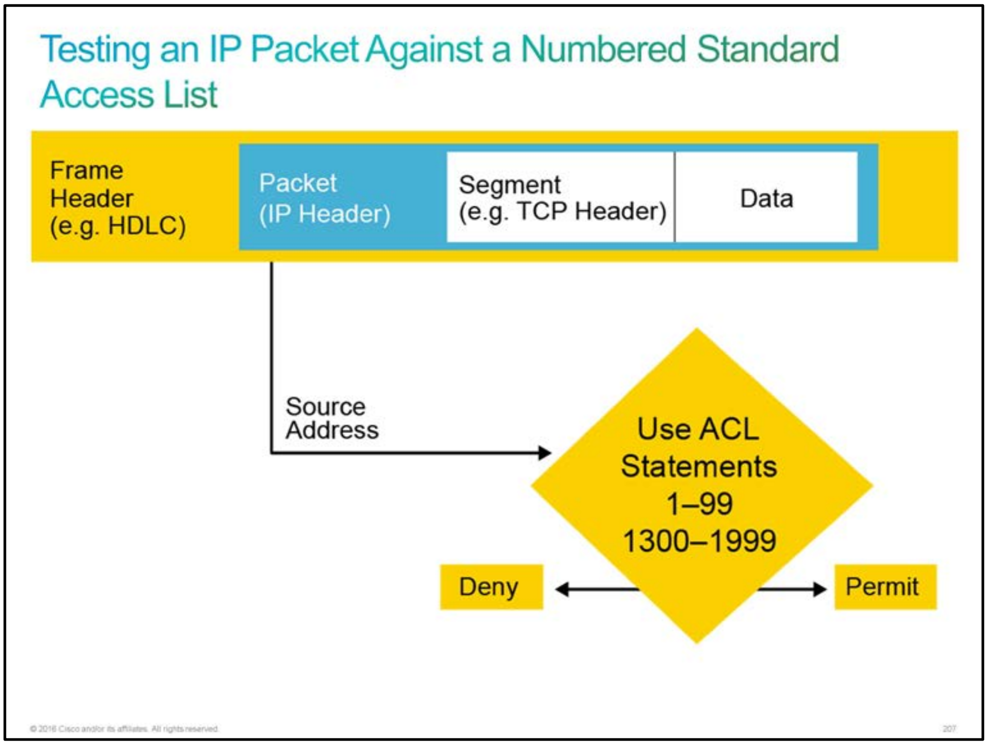
IPv4 ACL Type	Number Range or Identifier
Numbered Standard	1–99, 1300–1999
Numbered Extended	100–199, 2000–2699
Named (Standard and Extended)	Name

You can create many ACLs for a protocol. Select a different ACL number for each new ACL within a given protocol. However, on an interface, you can apply only one ACL per protocol, per direction.

Specifying an ACL number from 1 to 99 or 1300 to 1999 instructs the router to accept numbered standard IPv4 ACL statements. Specifying an ACL number from 100 to 199 or 2000 to 2699 instructs the router to accept numbered extended IPv4 ACL statements.

Testing an IP Packet Against a Numbered Standard Access List

Standard IPv4 ACLs, whether numbered (1 to 99 and 1300 to 1999) or named, filter packets that are based on a source address and mask, and they permit or deny the entire TCP/IP protocol suite.



Testing an IP Packet Against a Numbered Standard Access List

Frame Header (e.g. HDLC)

Packet (IP Header)

Segment (e.g. TCP Header)

Data

Source Address

Use ACL Statements 1–99 1300–1999

Deny

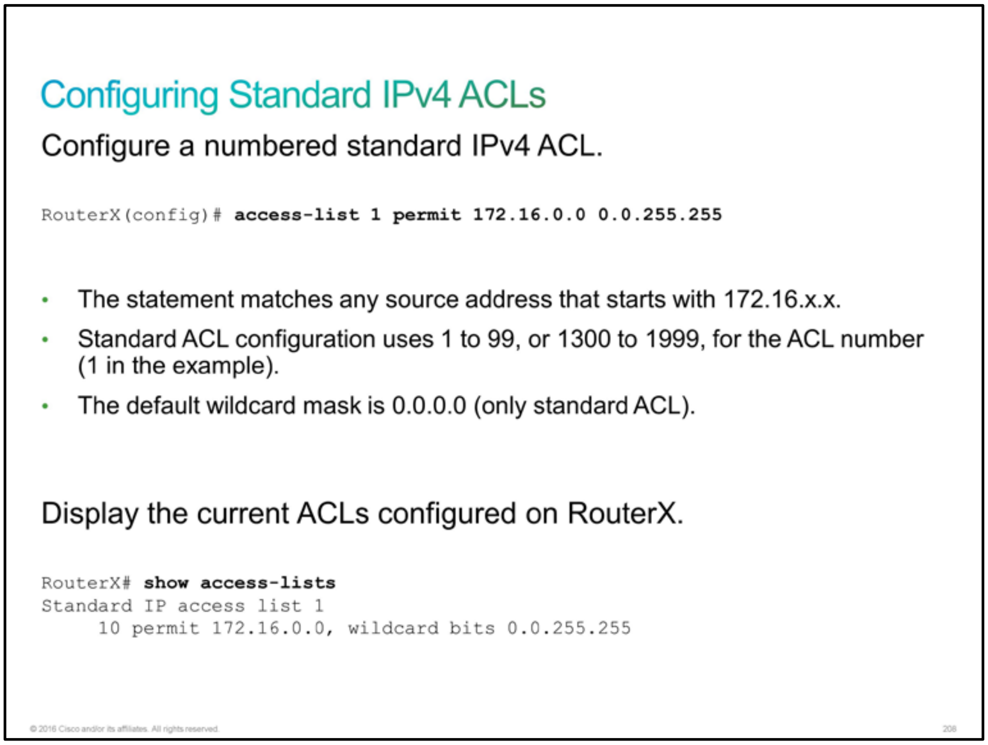
Permit

207

Note The standard ACL may not provide the required level of control that you require. You may need a more precise tool for selecting network traffic.

Configuring Standard IPv4 ACLs

You can configure numbered standard IPv4 ACLs on a Cisco router in the global configuration mode. The **access-list** command creates an entry in a standard IPv4 filter list. The following example shows the syntax of this command.



Configuring Standard IPv4 ACLs

Configure a numbered standard IPv4 ACL.

```
RouterX(config)# access-list 1 permit 172.16.0.0 0.0.255.255
```

- The statement matches any source address that starts with 172.16.x.x.
- Standard ACL configuration uses 1 to 99, or 1300 to 1999, for the ACL number (1 in the example).
- The default wildcard mask is 0.0.0.0 (only standard ACL).

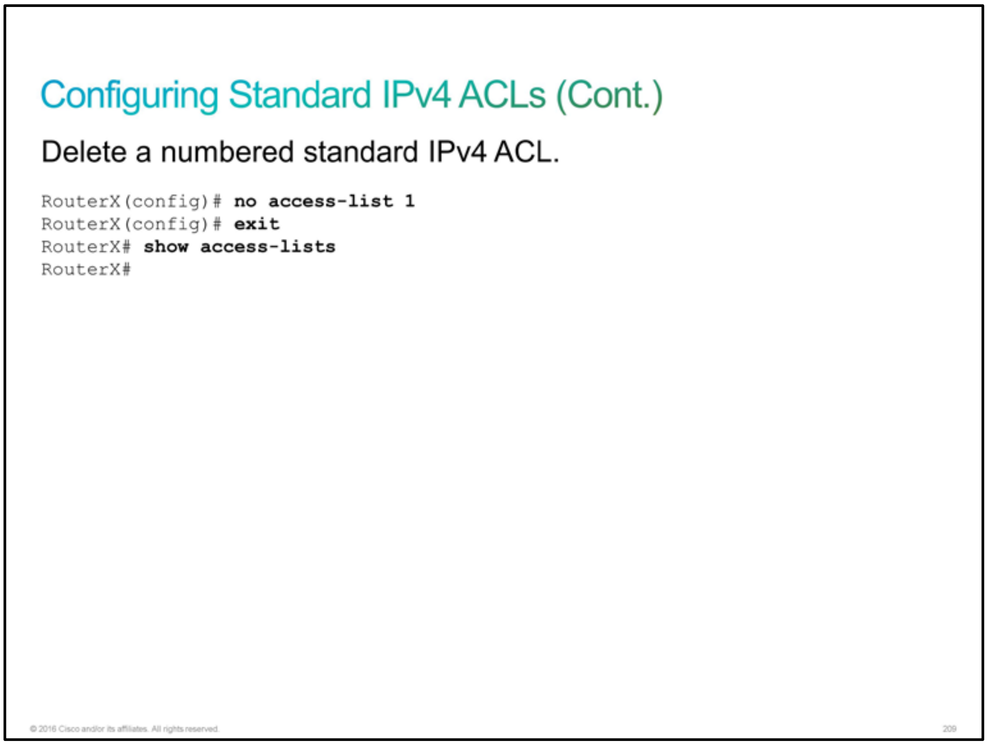
Display the current ACLs configured on RouterX.

```
RouterX# show access-lists
Standard IP access list 1
     10 permit 172.16.0.0, wildcard bits 0.0.255.255
```

208

The output of the **show access-list** command displays the current ACLs that are configured on router RouterX.

Use the **no access-list 1** command to remove the entire ACL 1.



Configuring Standard IPv4 ACLs (Cont.)

Delete a numbered standard IPv4 ACL.

```
RouterX(config)# no access-list 1
RouterX(config)# exit
RouterX# show access-lists
RouterX#
```

209

To remove the ACL, use the **no access-list** *number* command in the global configuration mode. Issue the **show access-list** command to confirm that ACL 1 has been removed. With numbered ACLs, you cannot remove individual entries with the **no access-list** command, because this command removes the entire ACL. The traditional way of removing or modifying a single numbered ACL entry would be to copy the whole ACL to a text editor, make the changes that are needed, and remove the entire ACL from the router using the **no access-list** command. You can then copy and paste the modified ACL from the text editor. Newer Cisco IOS Software releases allow easier editing by using sequence numbering.

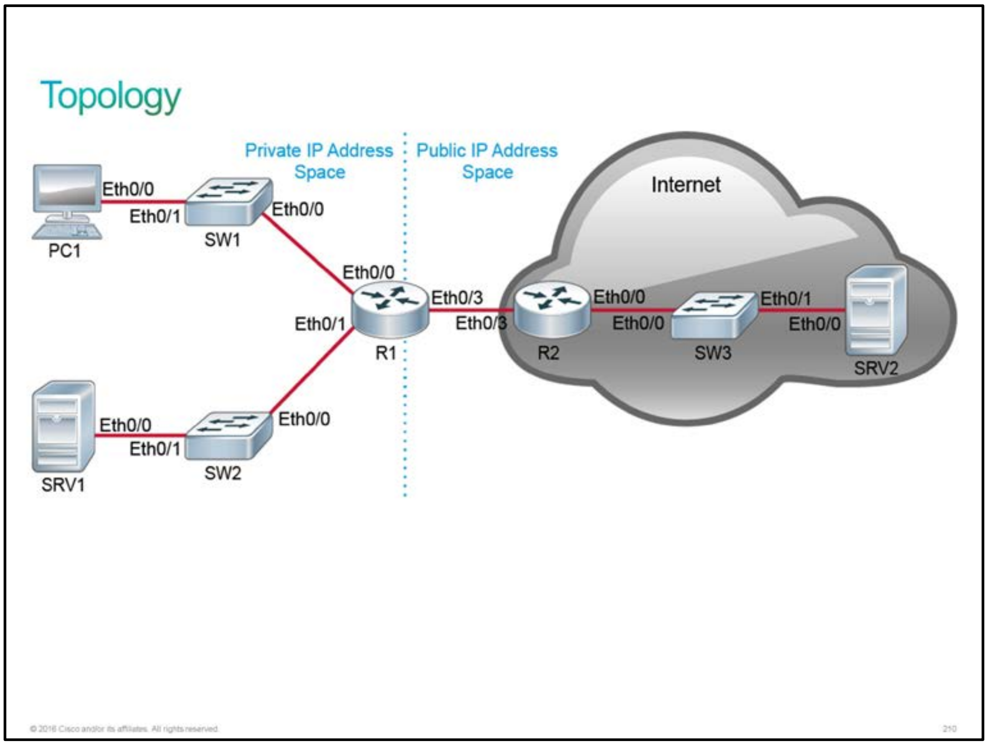
Discovery 11: Configure and Verify ACLs

Introduction

In this discovery lab, you will explore the basics of ACLs. ACLs are also commonly referred to simply as access lists. The lab is prepared with the devices represented in the topology diagram and the connectivity table. All devices have their basic configurations in place including hostnames and IP addresses. Note, neither routing nor NAT has yet been implemented, so there is no connectivity between the private IP address space and the public IP address space.

This discovery lab is broken into two main sections. The first section makes use of an access list that has already been prepared on R1 to demonstrate the importance of statement order in an access list and to demonstrate the effectiveness of using wildcard masks to specify ranges of IP addresses. In the second section of the discovery, you will create a new access list yourself. In both sections, you will demonstrate the function of the ACLs by applying them to the vty lines of R1 using the **access-class** command. When applied in this fashion, the ACL controls which IP addresses are allowed to initiate connections to the EXEC of the router. You will use other devices as the source of Telnet connection attempts to the EXEC of R1.

Topology



Job Aid

Device Information

Device Information Table

Device	Characteristic	Value
PC1	Hostname	PC1
PC1	IP address	10.10.1.10/24
PC1	Default gateway	10.10.1.1
SRV1	Hostname	SRV1
SRV1	IP address	10.10.2.20/24
SRV1	Default gateway	10.10.2.1
SRV2	Hostname	SRV2
SRV2	IP address	203.0.113.30/24
SRV2	Default gateway	203.0.113.1
SW1	Hostname	SW1
SW1	VLAN 1 IP address	10.10.1.4/24
SW1	Default gateway	10.10.1.1
SW1	Ethernet0/0 description	Link to R1
SW1	Ethernet0/1 description	Link to PC1
SW2	Hostname	SW2
SW2	VLAN 1 IP address	10.10.2.4/24
SW2	Default gateway	10.10.2.1
SW2	Ethernet0/0 description	Link to R1
SW2	Ethernet0/1 description	Link to SRV1
SW3	Hostname	SW3
SW3	VLAN 1 IP address	203.0.113.4/24
SW3	Default gateway	203.0.113.1

Device	Characteristic	Value
SW3	Ethernet0/0 description	Link to R2
SW3	Ethernet0/1 description	Link to SRV2
R1	Hostname	R1
R1	Ethernet0/0 description	Link to SW1
R1	Ethernet0/0 IP address	10.10.1.1/24
R1	Ethernet0/1 description	Link to SW2
R1	Ethernet0/1 IP address	10.10.2.1/24
R1	Ethernet0/3 description	Link to R2
R1	Ethernet0/3 IP address	198.51.100.2/30
R2	Hostname	R2
R2	Ethernet0/0 description	Link to SW3
R2	Ethernet0/0 IP address	203.0.113.1/24
R2	Ethernet0/3 description	Link to R1
R2	Ethernet0/3 IP address	198.51.100.1/30

PC and SRVs in the virtual lab environment are simulated as routers, so you should use Cisco IOS commands to configure them or make verifications.

Task 1: Configure Numbered Standard IPv4 ACLs

Activity

Step 1 On the R1 router, display access list 10 by passing the output of the **show run** command through the **include** filter, specifying the string **access-list 10**.

To specify a string that includes space characters to the include or section filters, enclose the string in a pair of parentheses.

```
R1# sh run | include access-list 10
access-list 10 permit 10.10.1.10
access-list 10 deny   10.10.1.0 0.0.0.255
access-list 10 permit 10.10.0.0 0.0.255.255
access-list 10 deny   10.0.0.0 0.255.255.255
access-list 10 permit any
```

This ACL demonstrates the use of progressively less restrictive wildcard masks.

Each line in the ACL becomes progressively less specific. Using the asterisk (*) character to represent any octet (0–255), the access list could be interpreted as follows:

- Permit 10.10.1.10
- Deny 10.10.1.*
- Permit 10.10.*.*
- Deny 10.*.*.*
- Permit *.*.*.*

Step 2 To demonstrate this access list in action, you will enable Telnet access on R1 and assign this access list to its vty lines. You will perform this task in the next several steps. First, enter the global configuration mode on R1, then enter the line configuration mode for the five vty lines.

Configuration of vty lines for remote access may not be familiar to you yet. Do not worry. You will be walked through the example.

```
R1# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)# line vty 0 4
R1(config-line)#
```

Step 3 The simplest way to enable Telnet on the vty lines is to define a line password. Set the vty line password to "Cisco123."

Cisco IOS passwords are case-sensitive.

```
R1(config-line)# password Cisco123
```

Step 4 Apply access list 10 to the vty lines in the inbound direction using the **access-class** command.

On R1, enter the following commands:

```
R1(config-line)# access-class 10 in
```

Defining an ACL on a Cisco IOS device does not change the behavior of the device until the ACL is assigned in some fashion. Different commands are used in different configuration modes to assign ACLs for different reasons. In this case, access list 10 will control the addresses that are permitted to initiate remote access connections to the EXEC of R1.

Step 5 Leave the configuration mode on R1.

On R1, enter the following commands:

```
R1(config-line)# end
R1#
```

Step 6 View access list 10, this time using the **show access-list** command.

On R1, enter the following command:

```
R1# show access-list 10
Standard IP access list 10
    10 permit 10.10.1.10
    20 deny   10.10.1.0, wildcard bits 0.0.0.255
    30 permit 10.10.0.0, wildcard bits 0.0.255.255
    40 deny   10.0.0.0, wildcard bits 0.255.255.255
    50 permit any
```

The **show access-list** command shows the numbered ACL in a different format than is used to configure it in the global configuration mode. You will soon see that there is another important aspect to the **show access-list** command.

Step 7 The first line in access list 10 permits the unique address 10.10.1.10. This address is the IP address of PC1. Access the console of PC1 and execute a Telnet to R1. Authenticate with the "Cisco123" password. Note the change in the system prompt, indicating that you are connected to R1.

On PC1, enter the following commands:

```
PC1# telnet 10.10.1.1
Trying 10.10.1.1 ... Open


User Access Verification

Password: Cisco123
R1>
```

Step 8 You are now using the console of PC1 to access the EXEC of R1 via a Telnet session. Execute the **show ip interface brief** command to provide further evidence that you are connected to R1 via PC1. You should see the status of the R1 interfaces, but not the PC1 interfaces.

```
R1> sh ip int brief
Interface                  IP-Address      OK? Method Status
Protocol
Ethernet0/0                10.10.1.1       YES NVRAM  up                    up
Ethernet0/1                10.10.2.1       YES NVRAM  up                    up
Ethernet0/2                unassigned      YES NVRAM  administratively down
down
Ethernet0/3                198.51.100.2    YES NVRAM  up                    up
Serial1/0                  unassigned      YES NVRAM  administratively down
down
Serial1/1                  unassigned      YES NVRAM  administratively down
down
Serial1/2                  unassigned      YES NVRAM  administratively down
down
Serial1/3                  unassigned      YES NVRAM  administratively down
down
```

Step 9 Use the **exit** command (or equivalently the **logout** command) to terminate the Telnet session and return to the CLI of PC1. Note that the system prompt returns to PC1>.

```
R1> exit

[Connection to 10.10.1.1 closed by foreign host]
PC1#
```

Step 10 Return to the console of R1 and once again show access list 10.

On R1, enter the following command:

```
R1# show access-list 10
Standard IP access list 10
    10 permit 10.10.1.10 (2 matches)
    20 deny   10.10.1.0, wildcard bits 0.0.0.255
    30 permit 10.10.0.0, wildcard bits 0.0.255.255
    40 deny   10.0.0.0, wildcard bits 0.255.255.255
    50 permit any
```

Routers maintain a hit counter for each line in an ACL. The **show access-list** command displays the number of times each line in an applied ACL has been matched.

When an ACL is applied to the vty lines with an access class, every permitted connection increments the appropriate line match counter by two, while every denied connection increments the appropriate line match counter by only one.

Understand that when an ACL is processed, the first line that matches is used. For example, note that the address 10.10.1.10 actually matches each line in access list 10, but in processing, it was only the first line that was activated.

Step 11 Assuming that the first line did not match, the second line of access list 10 denies access from 10.10.1.0 0.0.0.255. That is, it denies 10.10.1.*. The SW1 IP address is 10.10.1.4. That address should pass by the first line and match this second line. Access the console of SW1 and use Telnet to attempt to connect to R1. The attempt should fail.

On SW1, enter the following command:

```
SW1# telnet 10.10.1.1
Trying 10.10.1.1 ...
% Connection refused by remote host
```

The connection was refused.

Step 12 Return to the console of R1 and view access list 10 again.

On R1, enter the following command:

```
R1# show access-list 10
Standard IP access list 10
    10 permit 10.10.1.10 (2 matches)
    20 deny   10.10.1.0, wildcard bits 0.0.0.255 (1 match)
    30 permit 10.10.0.0, wildcard bits 0.0.255.255
    40 deny   10.0.0.0, wildcard bits 0.255.255.255
    50 permit any
```

There is a match on the second line due to the connection attempt from SW1. Again, permitted connections increment the hit counter by two while denied connections increment the hit counter only by one.

Step 13 Assuming that neither of the first two lines are matched, the third line of access list 10 permits access from 10.10.0.0 0.0.255.255. That is it permits 10.10.*.*. An example of an address that meets these conditions belongs to SRV1 (10.10.2.20). Access the console of SRV1, use Telnet to connect to R1 (10.10.1.1) and authenticate with the password "Cisco123." At the R1> system prompt, simply exit the session.

On SRV1, enter the following command:

```
SRV1# telnet 10.10.1.1
Trying 10.10.1.1 ... Open
User Access Verification

Password: Cisco123

R1> exit

[Connection to 10.10.1.1 closed by foreign host]
SRV1#
```

Step 14 Return to the console of R1 and show access list 10 one more time.

On R1, enter the following command:

```
R1# show access-list 10
Standard IP access list 10
    10 permit 10.10.1.10 (2 matches)
    20 deny   10.10.1.0, wildcard bits 0.0.0.255 (1 match)
    30 permit 10.10.0.0, wildcard bits 0.0.255.255 (2 matches)
    40 deny   10.0.0.0, wildcard bits 0.255.255.255
    50 permit any
```

The hit counter for the third line was incremented due to the successful connection from SRV1.

Step 15 The fourth line in access list 10 will deny connections from 10.0.0.0 0.255.255.255, assuming that there was not a match on any of the first three lines. That is, it denies 10.*.*.*. There are no appropriate addresses in the lab topology to test the fourth line. The fifth line permits connections from any address that did not match the first four lines. That is, any address that starts with something other than 10. R2, on the public address side of the topology, is an example. Access the console of R2 and execute a Telnet session to the R1 public IP address (198.51.100.2). Authenticate with the password "Cisco123." Exit the session when you reach the R1 command prompt.

On R2, enter the following commands:

```
R2# telnet 198.51.100.2
Trying 198.51.100.2 ... Open
User Access Verification

Password: Cisco123

R1> exit

[Connection to 198.51.100.2 closed by foreign host]
R2#
```

Step 16 Return to the console of R1 and show access list 10 one more time to verify the hit counter on the last line of the ACL.

On R1, enter the following command:

```
R1# show access-list 10
Standard IP access list 10
    10 permit 10.10.1.10 (2 matches)
    20 deny   10.10.1.0, wildcard bits 0.0.0.255 (1 match)
    30 permit 10.10.0.0, wildcard bits 0.0.255.255 (2 matches)
    40 deny   10.0.0.0, wildcard bits 0.255.255.255
    50 permit any (2 matches)
```

Task 2: Filter Traffic Using ACLs

Activity

Step 1 Now it is time to create an ACL yourself. Define a new, numbered standard IP access list. Use the number 20. The ACL should permit the IP addresses of PC1 (10.10.1.10) and SRV1 (10.10.2.20). The ACL should be applied to the vty lines with the **access-class** command. Return to the privileged EXEC after configuring this ACL.

On R1, enter the following commands:

```
R1# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)# access-list 20 permit 10.10.1.10
R1(config)# access-list 20 permit 10.10.2.20
R1(config)# line vty 0 4
R1(config-line)# access-class 20 in
R1(config-line)# end
```

When you applied access list 20 to the vty lines, it overrode the previous application of access list 10 on the vty lines. Only one ACL can be applied inbound on the vty lines at one time. Access list 20 is the only ACL that is applied to the vty lines now.

Step 2 View your new access list.

On R1, enter the following commands:

```
R1# show access-list 20
Standard IP access list 20
    10 permit 10.10.1.10
    20 permit 10.10.2.20
```

Step 3 The ACL explicitly permits the address 10.10.1.10. This address belongs to PC1. Access the console of PC1 and verify that you can use Telnet to connect from there to R1. Exit the Telnet session at the R1> prompt.

On PC1, enter the following commands:

```
PC1# telnet 10.10.1.1
Trying 10.10.1.1 ... Open
User Access Verification

Password: Cisco123

R1> exit

[Connection to 10.10.1.1 closed by foreign host]
PC1#
```

Step 4 The SW1 IP address (10.10.1.4) will not match any lines of access list 20. Understand that there is an implicit "deny everything else" at the bottom of every ACL. Access the console of SW1 and verify that you cannot connect from SW1 to R1 using Telnet.

On SW1, enter the following commands:

```
SW1# telnet 10.10.1.1
Trying 10.10.1.1 ...
% Connection refused by remote host

SW1#
```

Step 5 Return to R1 and show access list 20 to examine the line match counters.

On R1, enter the following commands:

```
R1# show access-list 20
Standard IP access list 20
    10 permit 10.10.1.10 (2 matches)
    20 permit 10.10.2.20
```

The connection from PC1 incremented the match counter on the first line.

There is no record of the failed connection attempt from SW1.

Step 6 Sometimes it is advantageous to add an explicit **deny any** to the end of an ACL. For example, it will allow you to track the number of matches that passed through the previous lines. You can also add the **log** argument to the line, which will generate a syslog message providing further detail about the deny event. Add an explicit **deny any** to the end of access list 20 and specify the **log** argument.

On R1, enter the following commands:

```
R1# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)# access-list 20 deny any log
R1(config)# end
```

Step 7 View the updated ACL.

On R1, enter the following command:

```
R1# show access-list 20
Standard IP access list 20
    10 permit 10.10.1.10 (2 matches)
    20 permit 10.10.2.20
    30 deny   any log
```

Step 8 Return to the console of SW1 and use Telnet to attempt to connect to R1 one more time. Again, the connection should be refused, but this time, it is not blocked implicitly but instead by the explicit deny.

On SW1, enter the following command:

```
SW1# telnet 10.10.1.1
Trying 10.10.1.1 ...
% Connection refused by remote host

SW1#
```

Step 9 Return to the console of R1. You should see that the syslog message has already been displayed to the console.

```
R1#
*Oct 19 12:48:15.906: %SEC-6-IPACCESSLOGNP: list 20 denied 0 10.10.1.4 ->
0.0.0.0, 1 packet
```

The syslog message shows the source IP address of the denied connection (10.10.1.4).

You can configure Cisco IOS devices to send syslog messages to a central event management server for real-time processing and long-term archiving of syslog events. Exploration of this concept is beyond the scope of this discovery.

Step 10 Show access list 20 to verify the incremented match counter on the explicit deny statement.

On R1, enter the following command:

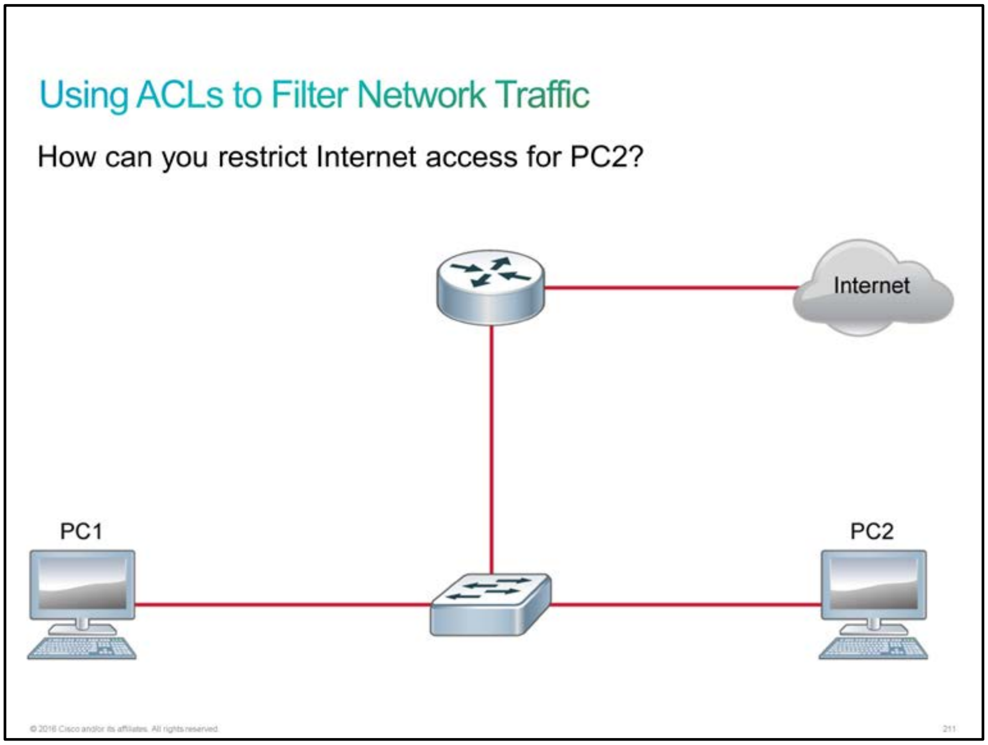
```
R1# show access-list 20
Standard IP access list 20
    10 permit 10.10.1.10 (2 matches)
    20 permit 10.10.2.20
    30 deny   any log (1 match)
```

You have now experimented with some basic ACL implementations. You worked with an existing ACL to test the order of precedence in a standard IP ACL. You also created your own numbered standard IP ACL and tested it. It also included a demonstration of the implicit **deny any** and the utility of adding an explicit **deny any**. Feel free to continue with independent exploration within the lab environment.

This is the end of the discovery lab.

Using ACLs to Filter Network Traffic

The figure introduces a common task for network administrators: a need to implement network traffic filtering to allow, limit, or restrict access to a network resource. A common mechanism that is used for traffic filtering is ACLs, which enable you to control access based on Layer 3 packet-header information.



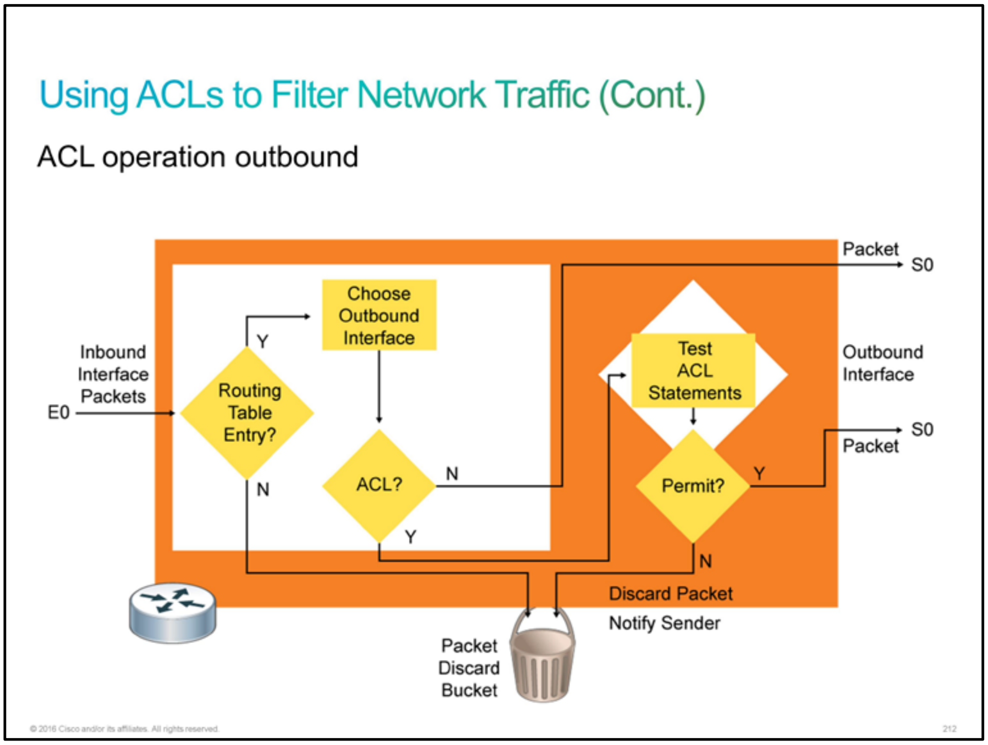
Using ACLs to Filter Network Traffic

How can you restrict Internet access for PC2?

In the scenario in the figure, you can implement traffic filtering on the router either inbound on an interface that is connected to the LAN or outbound on an interface that is connected to the Internet. By using a simple standard ACL, you can prevent packets from PC2 entering or leaving the interface. You should not forget to explicitly allow traffic for other devices in the LAN, such as PC1.

When you use ACLs for traffic filtering, they can operate inbound or outbound. The direction determines at which point packets are tested against the ACL as they pass through the router.

- **Outbound ACLs:** Incoming packets are routed to the outbound interface and then are processed through the outbound ACL. If packets match a permit statement, they are forwarded through the interface. If packets match a deny statement or if there is no match, they are discarded.
- **Inbound ACLs:** Incoming packets are processed by the ACL before they are routed to an outbound interface. An inbound ACL is efficient because it saves the overhead of routing lookups if the filtering tests deny the packet and it is discarded. If the tests permit the packet, it is processed for routing.



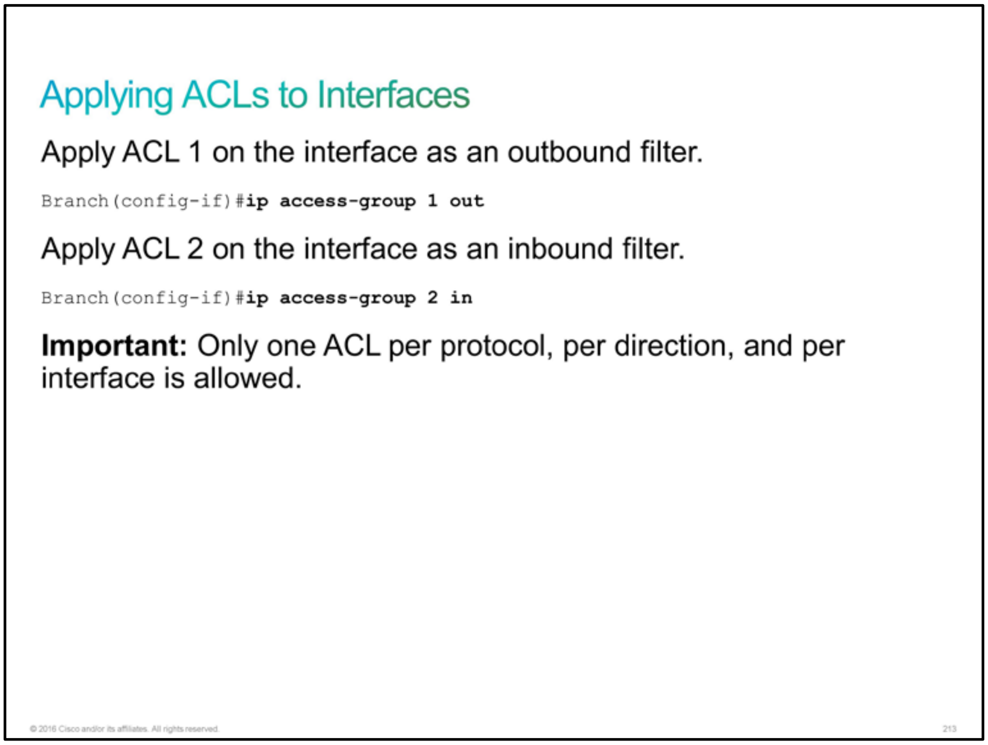
Using ACLs to Filter Network Traffic (Cont.)

ACL operation outbound

212

Applying ACLs to Interfaces

After you have configured an ACL, you link the ACL to an interface using the **ip access-group** command. Only one ACL per protocol, per direction, and per interface is allowed. The following figure shows examples of this command, showing how to apply the ACL as an inbound and outbound filter.



Applying ACLs to Interfaces

Apply ACL 1 on the interface as an outbound filter.

```
Branch(config-if)#ip access-group 1 out
```

Apply ACL 2 on the interface as an inbound filter.

```
Branch(config-if)#ip access-group 2 in
```

Important: Only one ACL per protocol, per direction, and per interface is allowed.

© 2016 Cisco and/or its affiliates. All rights reserved. 213

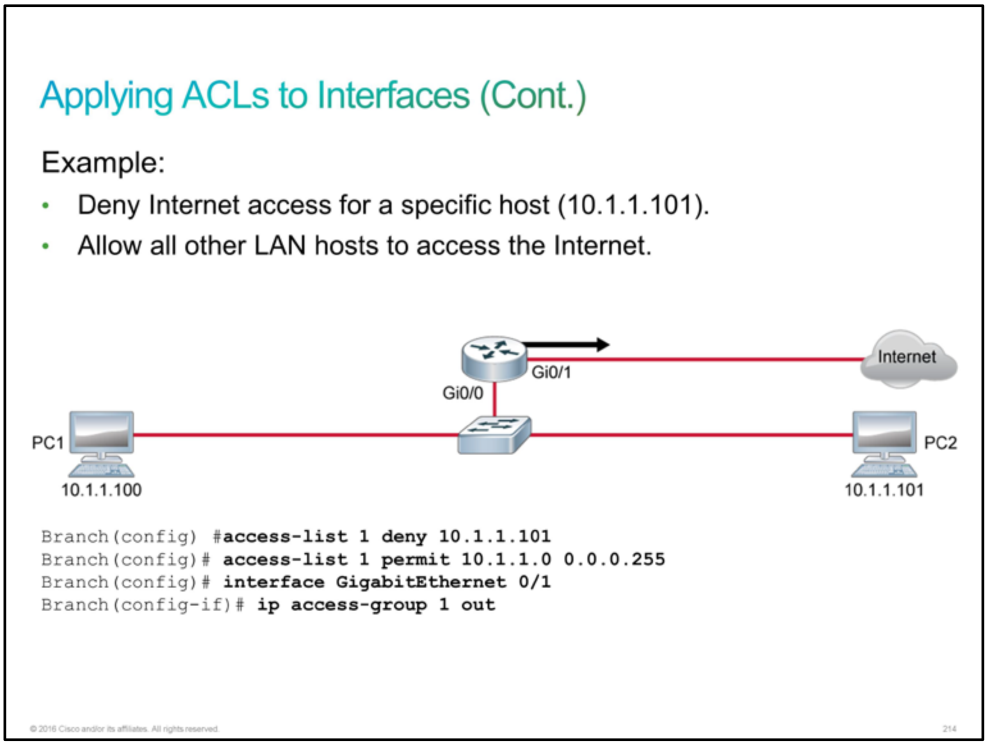
Note	To remove an ACL from an interface, enter the **no ip access-group** command on the interface, then enter the global **no access-list** command to remove the entire ACL if needed.

The table provides an example of the steps that are required to configure and apply a numbered standard ACL on a router.

Numbered Standard ACL Configuration Procedure

Step	Action	Notes
1	Use the **access-list** global configuration command to create an entry in a standard IPv4 ACL. `Branch(config)#` **access-list 1 permit 10.1.1.0 0.0.0.255**	The example ACL statement matches any address that starts with 10.1.1.x.

Step	Action	Notes
2	Use the **interface** configuration command to choose an interface in which to apply the ACL. `Branch(config)#` **interface GigabitEthernet 0/0**	After you enter the **interface** command, the CLI prompt changes from `(config)#` to `(config-if)#`.
3	Use the **ip access-group** interface configuration command to activate the existing ACL on the interface. `Branch(config-if)#` **ip access-group 1 in**	This example activates the standard IPv4 ACL 1 on the interface as an inbound filter.



Applying ACLs to Interfaces (Cont.)

Example:

- Deny Internet access for a specific host (10.1.1.101).
- Allow all other LAN hosts to access the Internet.

```
Branch(config) #access-list 1 deny 10.1.1.101
Branch(config)# access-list 1 permit 10.1.1.0 0.0.0.255
Branch(config)# interface GigabitEthernet 0/1
Branch(config-if)# ip access-group 1 out
```

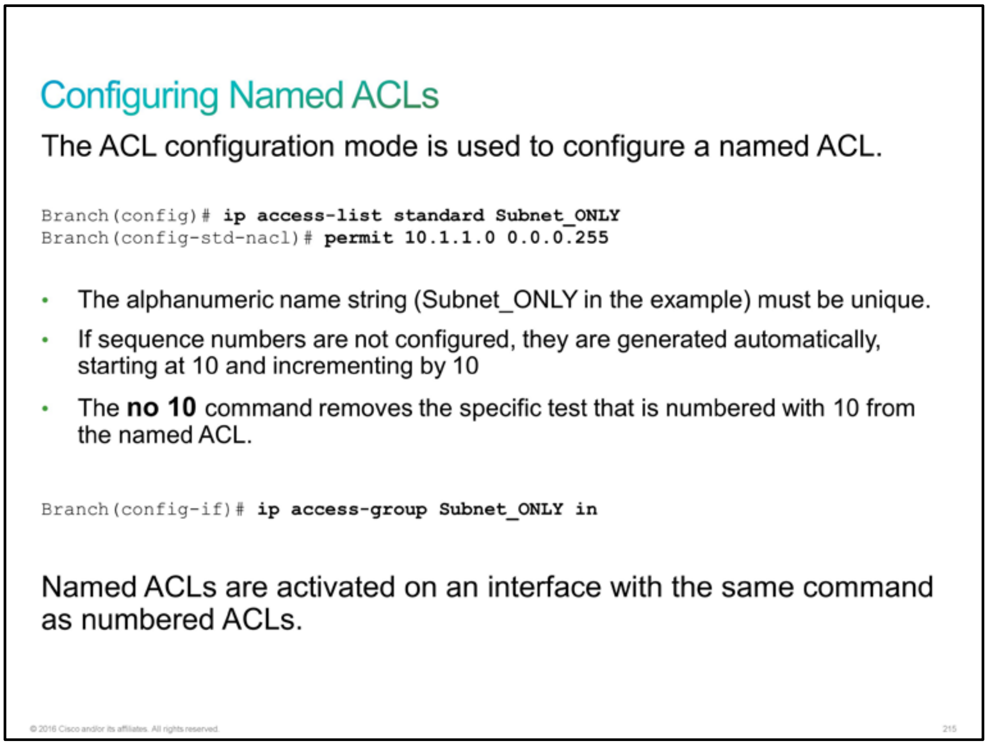
214

The figure shows a scenario in which ACL 1 is applied in the outbound direction on router Branch to provide traffic filtering. ACL 1 includes a deny statement that matches traffic from a specific host with the IP address 10.1.1.101. The second line in the ACL permits traffic from hosts within the network 10.1.1.0 /24. It is important to specify a permit statement because ACLs end with an implicit deny all statement.

Note	Alternatively, the ACL could be applied in the inbound direction on the interface GigabitEthernet0/0. This solution would not only prevent host PC2 from accessing the Internet but would also deny all communication between PC2 and the router.

Configuring Named ACLs

Designating ACLs by a descriptive name instead of a number provides a better way to describe the intention of the ACL.



Configuring Named ACLs

The ACL configuration mode is used to configure a named ACL.

```
Branch(config)# ip access-list standard Subnet_ONLY
Branch(config-std-nacl)# permit 10.1.1.0 0.0.0.255
```

- The alphanumeric name string (Subnet_ONLY in the example) must be unique.
- If sequence numbers are not configured, they are generated automatically, starting at 10 and incrementing by 10
- The **no 10** command removes the specific test that is numbered with 10 from the named ACL.

```
Branch(config-if)# ip access-group Subnet_ONLY in
```

Named ACLs are activated on an interface with the same command as numbered ACLs.

215

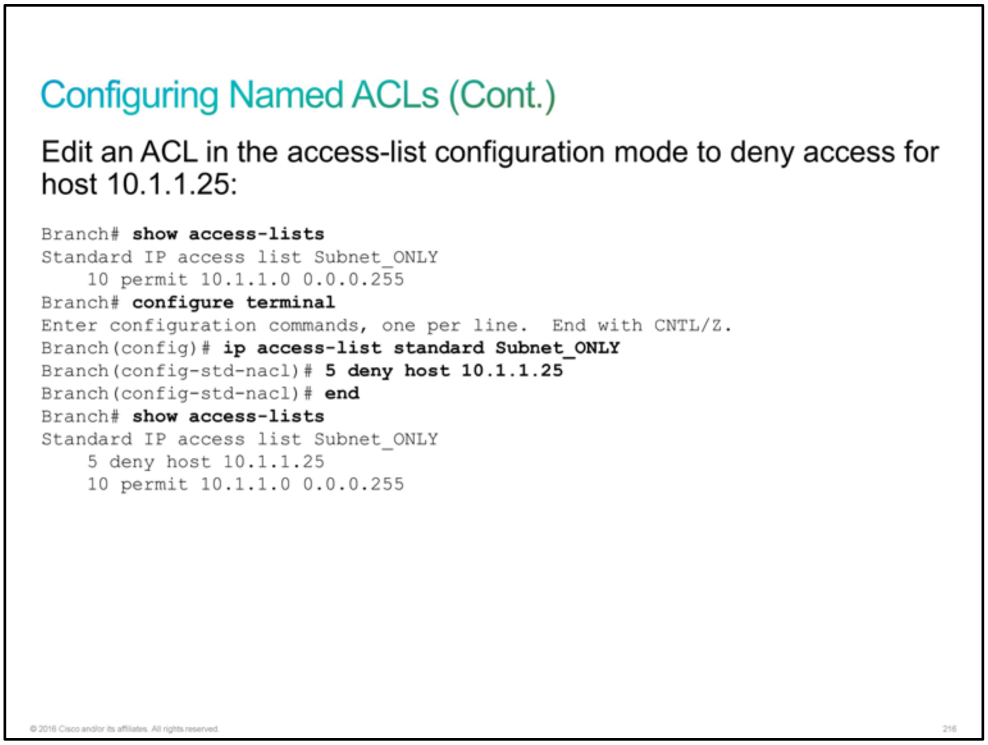
Naming an ACL makes it easier to understand its function. For example, an ACL to deny one subnet could be called "NO_Subnet." When you identify your ACL with a name instead of a number, the configuration mode and command syntax are slightly different.

You use the access-list configuration mode to define named ACLs. To enter this mode, use the **ip access-list** command.

Note	You can also define numbered ACLs using the access-list configuration mode. You simply specify an ACL number instead of a unique name.

In the figure, the command output shows the commands that are used to configure a standard ACL that is named Subnet_ONLY on the Branch router. The ACL permits traffic from hosts on the 10.1.1.0/24 subnet.

Capitalizing ACL names is not required, but it makes them stand out when you view the running configuration output.



Configuring Named ACLs (Cont.)

Edit an ACL in the access-list configuration mode to deny access for host 10.1.1.25:

```
Branch# show access-lists
Standard IP access list Subnet_ONLY
    10 permit 10.1.1.0 0.0.0.255
Branch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Branch(config)# ip access-list standard Subnet_ONLY
Branch(config-std-nacl)# 5 deny host 10.1.1.25
Branch(config-std-nacl)# end
Branch# show access-lists
Standard IP access list Subnet_ONLY
    5 deny host 10.1.1.25
    10 permit 10.1.1.0 0.0.0.255
```

Named IP ACLs allow you to add, modify, or delete individual entries in a specific ACL. You can use sequence numbers to insert statements anywhere in the named ACL.

When you add statements to the ACL, the default increment is 10. The figure shows an additional entry that is numbered 5 in the Subnet_ONLY ACL, which is inserted in front of line 10.

Note that a reload will change the sequence numbers in the ACL. The sequence numbers will be 10 and 20 instead of 5 and 10 after the reload. Use the **access-list resequence** command to renumber the ACL entries in an ACL without having to reload.

Challenge

1. Based on the information within the IP packet, an ACL identifies the traffic. True or False?

   A. True
   B. False

2. Which statement about ACLs is true?

   A. An ACL must have at least one permit action, else it just blocks all traffic
   B. ACLs go bottom-up through the entries looking for a match
   C. An ACL has an implicit permit at the end of the ACL.
   D. ACLs will check the packet against all entries looking for a match

3. Which of the following ACL statement will permit only 192.168.1.123 and nothing else?

   A. **access-list 10 permit 192.168.1.123**
   access-list 10 permit 192.168.1.0 0.0.0.255
   B. **access-list 10 permit 192.168.1.123**
   access-list 10 deny 192.168.1.0 0.0.0.255
   C. **access-list 10 deny 192.168.1.0 0.0.0.255**
   access-list 10 permit 192.168.1.123

4. Look at the following ACL Statements and choose the statement that is True.

```
access-list 20 permit 192.168.1.1
access-list 20 deny 192.168.1.0 0.0.0.255
access-list 20 permit 192.0.0.0 0.255.255.255
```

   A. Everything within 192.168.1.0 except 192.168.1.1 will be permitted. Hosts within 192.169.0.0/24 will be blocked.
   B. Everything within 192.168.1.0 except 192.168.1.1 will be permitted. Hosts within 192.169.0.0/24 will be permitted as well.
   C. Everything within 192.168.1.0 except 192.168.1.1 will be blocked. Hosts within 192.169.0.0/24 will be permitted.
   D. Everything except 192.168.1.1 will be blocked.

5. When using wildcard bits, which of the following is true?

   A. 0 means ignore the value of the corresponding address bit.
   B. 1 means match the value of the corresponding address bit.
   C. 1 means ignore the value of the corresponding address bit.
   D. 2 means match the value of the corresponding address bit.

6. In which type of ACL the packets are allowed pass through the router adding extra overhead of routing lookups when ACL filtering discards the packet?

   A. Inbound ACLs
   B. Outbound ACLs

7. Which of the following actions are possible in Named ACLs?

   A. Add a specific ACL entry
   B. Delete a specific ACL entry
   C. Modify a specific ACL entry
   D. All the above

Answer Key

Challenge

1. A
2. B
3. B
4. B
5. C
6. B
7. D

Glossary

10BASE-T

10-Mbps baseband Ethernet specification using two pairs of twisted-pair cabling (Categories 3, 4, or 5): one pair for transmitting data and the other for receiving data. 10BASE-T, which is part of the IEEE 802.3 specification, has a distance limit of approximately 328 feet (100 meters) per segment.

ACK

acknowledgment. Notification sent from one network device to another to acknowledge that some event occurred (for example, the receipt of a message).

ACL

access control list. A list kept by routers to control access to or from the router for a number of services (for example, to prevent packets with a certain IP address from leaving a particular interface on the router).

administrative distance

Rating of the trustworthiness of a routing information source. Administrative distance often is expressed as a numerical value between 0 and 255. The higher the value, the lower the trustworthiness rating.

AP

access point. A node on a wireless network that allows other wireless devices to connect to a wired network.

API

application programming interface. The means by which an application program talks to communications software. Standardized APIs allow application programs to be developed independently of the underlying method of communication. A set of standard software interrupts, calls, and data formats that computer application programs use to initiate contact with other devices (for example, network services, mainframe communications programs, or other program-to-program communications). Typically, APIs make it easier for software developers to create the links that an application needs to communicate with the operating system or with the network.

ARP

Address Resolution Protocol. Internet protocol that is used to map an IP address to a MAC address. Defined in RFC 826.

ASCII

American Standard Code for Information Interchange. An 8-bit code for character representation (7 bits plus parity).

ATM

Asynchronous Transfer Mode. The international standard for cell relay in which multiple service types (such as voice, video, or data) are conveyed in fixed-length (53-byte) cells. Fixed-length cells allow cell processing to occur in hardware, thereby reducing transit delays. ATM is designed to take advantage of high-speed transmission media, such as E3, SONET, and T3.

AUX

auxiliary.

BIA

burned-in address. Refers to the burned-in MAC address.

CAM

content-addressable memory.

CIDR

classless interdomain routing. Technique supported by BGP4 and based on route aggregation. CIDR allows routers to group routes to reduce the quantity of routing information carried by the core routers. With CIDR, several IP networks appear to networks outside the group as a single, larger entity. With CIDR, IP addresses and their subnet masks are written as four octets, separated by periods, followed by a forward slash and a two-digit number that represents the subnet mask.

CIFS

Common Internet File System.

CLI

Command Language Interpreter. The basic Cisco IOS configuration and management interface.

CoS

class of service. An indication of how an upper-layer protocol requires a lower-layer protocol to treat its messages. In SNA subarea routing, CoS definitions are used by subarea nodes to determine the optimal route to establish a given session. A CoS definition comprises a virtual route number and a transmission priority field. Also called ToS.

CPU

central processing unit. The hardware within a computer system or smartphone that carries out the instructions of a computer program by performing the basic arithmetical, logical, and input-output operations of the system.

CSMA/CD

Carrier Sense Multiple Access with Collision Detection. Media-access mechanism wherein devices ready to transmit data first check the channel for a carrier. If no carrier is sensed for a specific period of time, a device can transmit. If two devices transmit at once, a collision occurs and is detected by all colliding devices. This collision subsequently delays retransmissions from those devices for some random length of time. Ethernet and IEEE 802.3 use CSMA/CD access.

DCBXP

The Data Center Bridging Exchange Protocol (DCBXP) is an extension of LLDP. It is used to announce, exchange, and negotiate node parameters between peers.

DHCP

Dynamic Host Configuration Protocol (*common term*). Provides a mechanism for allocating IP addresses dynamically so that addresses can be reused when hosts no longer need them.

DNS

Domain Name System. System used on the Internet for translating names of network nodes into addresses.

DUAL

Diffusing Update Algorithm. Convergence algorithm used in EIGRP that provides loop-free operation at every instant throughout a route computation. Allows routers involved in a topology change to synchronize at the same time, while not involving routers that are unaffected by the change.

EIGRP

Enhanced Interior Gateway Routing Protocol. It's the advanced version of IGRP developed by Cisco. It provides superior convergence properties and operating efficiency, and it combines the advantages of link-state protocols with those of distance vector protocols.

EMI

electromagnetic interference. It's interference by electromagnetic signals that can cause reduced data integrity and increased error rates on transmission channels.

Ethernet

Baseband LAN specification invented by Xerox Corporation and developed jointly by Xerox, Intel, and Digital Equipment Corporation. Ethernet networks use CSMA/CD and run over a variety of cable types at 10 Mbps. Ethernet is similar to the IEEE 802.3 series of standards. It is the most commonly used LAN technology because its protocol is easy to understand, implement, manage, and maintain. It allows low-cost network implementations, provides extensive topological flexibility for network installation, and guarantees successful interconnection and operation of standards-compliant products, regardless of manufacturer.

Fast Ethernet

Any of a number of 100-Mbps Ethernet specifications. Fast Ethernet offers a speed increase 10 times that of the 10BaseT Ethernet specification while preserving such qualities as frame format, MAC mechanisms, and MTU. Such similarities allow the use of existing 10BaseT applications and network management tools on Fast Ethernet networks. Based on an extension to the IEEE 802.3 specification.

FCS

frame check sequence. Extra characters added to a frame for error control purposes. Used in HDLC, Frame Relay, and other data link layer protocols.

Frame Relay

Industry-standard, packet-switched data link layer protocol that handles multiple virtual circuits between connected devices.

FTP

File Transfer Protocol. Protocol for exchanging files over the Internet.

Gigabit Ethernet

Standard for a high-speed Ethernet, approved by the IEEE (Institute of Electrical and Electronics Engineers) 802.3z standards committee in 1996.

GUI

graphical user interface (*common term*). It is a user environment that uses pictorial as well as textual representations of the input and the output of applications and the hierarchical or other data structure in which information is stored. Such conventions as buttons, icons, and windows are typical, and many actions are performed using a pointing device (such as a mouse). Microsoft Windows and the Apple Macintosh are prominent examples of platforms using a GUI.

HDLC

High-Level Data Link Control. Bit-oriented synchronous data link layer protocol developed by ISO. Derived from SDLC, HDLC specifies a data encapsulation method on synchronous serial links using frame characters and checksums.

HTTP

Hypertext Transfer Protocol *(common term)*. The protocol that is used by web browsers and web servers to transfer files, such as text and graphic files.

HTTPS

Hypertext Transfer Protocol Secure.

IANA

Internet Assigned Numbers Authority. Organization operated under the auspices of the ISOC as a part of the IAB. IANA delegates authority for IP address-space allocation and domain-name assignment to the InterNIC and other organizations. IANA also maintains a database of assigned protocol identifiers that is used in the TCP/IP stack, including autonomous system numbers.

ICMP

Internet Control Message Protocol. Network layer Internet protocol that reports errors and provides other information that is relevant to IP packet processing. Documented in RFC 792.

IEEE

Institute of Electrical and Electronics Engineers. Professional organization whose activities include the development of communications and network standards. IEEE LAN standards are the predominant LAN standards today.

IEEE 802.2

IEEE LAN protocol that specifies an implementation of the LLC sublayer of the data link layer. IEEE 802.2 handles errors, framing, flow control, and the network layer (Layer 3) service interface. Used in IEEE 802.3 and IEEE 802.5 LANs.

IEEE 802.3

IEEE LAN protocol that specifies an implementation of the physical layer and the MAC sublayer of the data link layer. IEEE 802.3 uses CSMA/CD access at various speeds over a variety of physical media. Extensions to the IEEE 802.3 standard specify implementations for Fast Ethernet. Physical variations of the original IEEE 802.3 specification include 10BASE2, 10BASE5, 10BASEF, 10BASE-T, and 10BROAD36. Physical variations for Fast Ethernet include 100BASE-T, 100BASET4, and 100BASEX.

IETF

Internet Engineering Task Force. Task force consisting of over 80 working groups responsible for developing Internet standards. The IETF operates under the auspices of ISOC.

IP

Internet Protocol. Network layer protocol in the TCP/IP stack offering a connectionless internetwork service. IP provides features for addressing, type-of-service specification, fragmentation and reassembly, and security. Defined in RFC 791.

IP address

A 32-bit address assigned to hosts using TCP/IP. An IP address belongs to one of five classes (A, B, C, D, or E) and is written as 4 octets separated by periods (dotted decimal format). Each address consists of a network number, an optional subnetwork number, and a host number. The network and subnetwork numbers together are used for routing, and the host number is used to address an individual host within the network or subnetwork. A subnet mask is used to extract network and subnetwork information from the IP address. CIDR provides a new way of representing IP addresses and subnet masks. Also called an Internet address.

IPv4

IP version 4 *(common term)*. Internet Protocol version 4 is the fourth version in the development of IP and the first version of the protocol to be widely deployed. Along with IPv6, IPv4 is at the core of standards-based internetworking methods of the Internet. IPv4 is still used to route most traffic across the Internet. IPv4 is a connectionless protocol for use on packet-switched link layer networks (for example, Ethernet). It operates on a best-effort delivery model in that it does not guarantee delivery and does not assure proper sequencing or avoidance of duplicate delivery.

IPv6

IP version 6 *(common term)*. Replacement for the current version of IP (version 4). IPv6 includes support for flow ID in the packet header, which can be used to identify flows. Formerly called IPng (next generation).

IS-IS

Intermediate System-to-Intermediate System. OSI link-state hierarchical routing protocol based on DECnet Phase V routing, whereby ISs (routers) exchange routing information based on a single metric to determine network topology.

ISO

International Organization for Standardization. International organization that is responsible for a wide range of standards, including those relevant to networking. ISO developed the OSI reference model, a popular networking reference model.

ISP

Internet service provider. Company that provides Internet access to other companies and individuals.

ISR

integrated services router. An ISR specifies the elements to guarantee QoS on networks. For example, an ISR can be used to allow video and sound to reach the receiver without interruption. Every application that requires some kind of guarantee has to make an individual reservation.

LAN

local-area network. High-speed, low-error data network covering a relatively small geographic area (up to a few thousand meters). LANs connect workstations, peripherals, terminals, and other devices in a single building or other geographically limited area. LAN standards specify cabling and signaling at the physical and data link layers of the OSI model. Ethernet, FDDI, and Token Ring are widely used LAN technologies.

LED

light emitting diode. A semiconductor device that emits light produced by converting electrical energy. Status lights on hardware devices are typically LEDs.

LIR

local Internet registry.

LLDP

Link Layer Discovery Protocol.

MAC

Media Access Control. The lower of the two sublayers of the data link layer that is defined by the IEEE. The MAC sublayer handles access to shared media, such as whether token passing or contention will be used.

MAC address

a standardized data link layer address that is required for every port or device that connects to a LAN. Other devices in the network use these addresses to locate specific ports in the network and to create and update routing tables and data structures. A MAC address is 6 bytes long and is controlled by the IEEE. It is also known as a hardware address, MAC layer address, and physical address.

MIC

media interface connector. MIC is the FDDI de facto standard connector.

MMF

multimode fiber. An optical fiber supporting the propagation of multiple frequencies of light.

MTBF

mean time between failures.

MTU

maximum transmission unit. The maximum packet size, in bytes, that a particular interface can handle.

NAT

Network Address Translation. A mechanism for reducing the need for globally unique IP addresses. NAT allows an organization with addresses that are not globally unique to connect to the Internet by translating these addresses into globally routable address space. Also known as Network Address Translator.

network address

The network layer address referring to a logical, rather than a physical, network device. The network address is also called a protocol address.

NFS

Network File System. As commonly used, a distributed file system protocol suite developed by Sun Microsystems that allows remote file access across a network. In actuality, NFS is simply one protocol in the suite. NFS protocols include NFS, RPC, XDR, and others. These protocols are part of a larger architecture that Sun refers to as ONC.

NIC

network interface card. A board that provides network communication capabilities to and from a computer system. A NIC is also called an adapter.

NTP

Network Time Protocol. A protocol that is built on top of TCP that ensures accurate local timekeeping with reference to radio and atomic clocks that are located on the Internet. This protocol is capable of synchronizing distributed clocks within milliseconds over long time periods.

NVRAM

nonvolatile RAM. RAM that retains its contents when a unit is powered off.

OSI

Open Systems Interconnection. International standardization program created by ISO and ITU-T to develop standards for data networking that facilitate multivendor equipment interoperability.

OSI reference model

Open System Interconnection reference model. Network architectural model developed by ISO and ITU-T. The model consists of seven layers, each of which specifies particular network functions, such as addressing, flow control, error control, encapsulation, and reliable message transfer. The lowest layer (the physical layer) is closest to the media technology. The lower two layers are implemented in hardware and software whereas the upper five layers are implemented only in software. The highest layer (the application layer) is closest to the user. The OSI reference model is used universally as a method for teaching and understanding network functionality. Similar in some respects to SNA.

OSPF

Open Shortest Path First.
Link-state, hierarchical IGP routing algorithm proposed as a successor to RIP in the Internet community. OSPF features include least-cost routing, multipath routing, and load balancing. OSPF was derived from an early version of the IS-IS protocol.

OUI

Organizational Unique Identifier. Three octets that are assigned by the IEEE in a block of 48-bit LAN addresses.

PDU

protocol data unit. OSI term for packet.

POST

power-on self test. Set of hardware diagnostics that runs on a hardware device when this device is powered on.

PPP

Point-to-Point Protocol. Successor to SLIP that provides router-to-router and host-to-network connections over synchronous and asynchronous circuits. Whereas SLIP was designed to work with IP, PPP was designed to work with several network layer protocols, such as IP, IPX, and ARA. PPP also has built-in security mechanisms, such as CHAP and PAP. PPP relies on two protocols: LCP and NCP.

QoS

quality of service. Measure of performance for a transmission system that reflects its transmission quality and service availability.

RARP

Reverse Address Resolution Protocol. Protocol in the TCP/IP stack that provides a method for finding IP addresses based on MAC addresses.

RFC

Request for Comments. Document series that is used as the primary means for communicating information about the Internet. Some RFCs are designated by the IAB as Internet standards. Most RFCs document protocol specifications, such as Telnet and FTP, but some RFCs are humorous or historical. RFCs are available online from numerous sources.

RFI

radio frequency interference. Radio frequencies that create noise that interferes with information being transmitted across unshielded copper cable.

RIP

Routing Information Protocol. A distance-vector routing protocol that uses hop count as a routing metric.

RIP

Request in Progress.

RIPv2

Routing Information Protocol version 2.

RJ-45

"Registered Jack-45" is an eight-wire connector that is commonly used to connect computers onto a LAN.

ROM

read-only memory. Nonvolatile memory that can be read, but not written, by the microprocessor.

routing protocol

Protocol that accomplishes routing through the implementation of a specific routing algorithm. Examples of routing protocols include IGRP, OSPF, and RIP.

routing table

Table stored in a router or some other internetworking device that keeps track of routes to particular network destinations and, in some cases, metrics associated with those routes.

RTT

round-trip time. Time required for a network communication to travel from the source to the destination and back. RTT includes the time required for the destination to process the message from the source and to generate a reply. RTT is used by some routing algorithms to aid in calculating optimal routes.

SCP

Secure Copy Protocol. SCP provides a secure and authenticated method for transferring files.

SMF

single-mode fiber. Fiber-optic cabling with a narrow core that allows light to enter only at a single angle. Such cabling has higher bandwidth than multimode fiber, but requires a light source with a narrow spectral width (for example, a laser). Also called monomode fiber.

SNMP

Simple Network Management Protocol. Network management protocol used almost exclusively in TCP/IP networks. SNMP provides a means to monitor and control network devices, and to manage configurations, statistics collection, performance, and security.

SSH

Secure Shell Protocol. Protocol that provides a secure remote connection to a route through a TCP application.

SSHv1

Secure Shell Protocol version 1.

SSHv2

Secure Shell Protocol version 2.

SSL

Secure Socket Layer. Encryption technology for the Web used to provide secure transactions, such as the transmission of credit card numbers for e-commerce.

ST

straight tip.

static route

Route that is explicitly configured and entered into the routing table. Static routes take precedence over routes chosen by dynamic routing protocols.

subnet mask

A 32-bit binary number that is used to define which portion of an IP address refers to the subnet and which portion refers to the host. Sometimes referred to simply as *mask*.

SYN

synchronization.

SYN-ACK

synchronization-acknowledgment.

syslog

system logging.

TCP

Transmission Control Protocol. Connection-oriented transport layer protocol that provides reliable full-duplex data transmission. TCP is part of the TCP/IP protocol stack.

TCP/IP

Transmission Control Protocol/Internet Protocol. Common name for the suite of protocols developed by the U.S. DoD in the 1970s to support the construction of worldwide internetworks. TCP and IP are the two best-known protocols in the suite.

Telnet

standard terminal emulation protocol in the TCP/IP protocol stack. Telnet is used for remote terminal connection, enabling users to log into remote systems and use resources as if they were connected to a local system. Telnet is defined in RFC 854.

TFTP

Trivial File Transfer Protocol. Simplified version of FTP that allows files to be transferred from one computer to another over a network, usually without the use of client authentication (for example, username and password).

TIA

Telecommunications Industry Alliance. Organization that develops standards relating to telecommunications technologies. Together, the TIA and the EIA have formalized standards, such as EIA/TIA-232, for the electrical characteristics of data transmission.

TLS

Transport Layer Security. A future IETF protocol to replace SSL.

TLV

type, length, value.

Token Ring

Token-passing LAN that was developed and supported by IBM. Token Ring runs at 4 or 16 Mbps over a ring topology. Similar to IEEE 802.5.

TTL

Time to Live. A mechanism that limits the lifespan or lifetime of data in a computer or network.

UDP

User Datagram Protocol. Connectionless transport layer protocol in the TCP/IP protocol stack. UDP is a simple protocol that exchanges datagrams without acknowledgments or guaranteed delivery, requiring that error processing and retransmission be handled by other protocols. UDP is defined in RFC 768.

UTP

unshielded twisted-pair. Four-pair wire medium used in a variety of networks. UTP does not require the fixed spacing between connections that is necessary with coaxial-type connections. Five types of UTP cabling are commonly used: Category 1, Category 2, Category 3, Category 4, and Category 5.

VLAN

virtual LAN. Group of devices on one or more LANs that are configured (using management software) so that they can communicate as if they were attached to the same wire, when in fact they are located on a number of different LAN segments. Because VLANs are based on logical instead of physical connections, they are extremely flexible.

VLSM

variable-length subnet mask. Capability to specify a different subnet mask for the same network number on different subnets. VLSM can help optimize available address space.

VoIP

Voice over IP. The capability to carry normal telephony-style voice over an IP-based internet with POTS-like functionality, reliability, and voice quality. VoIP enables a router to carry voice traffic (for example, telephone calls and faxes) over an IP network. In VoIP, the DSP segments the voice signal into frames, which then are coupled in groups of two and stored in voice packets. These voice packets are transported using IP in compliance with ITU-T specification H.323. A primary attraction of VoIP is its ability to reduce expenses, because phone calls travel over the data network rather than over the phone company network.

vty

virtual type terminal. Commonly used as virtual terminal lines.

WAN

wide-area network. Data communications network that serves users across a broad geographic area and often uses transmission devices provided by common carriers. Frame Relay, SMDS, and X.25 are examples of WANs.

WINS

Windows Internet Name Service.

WLAN

wireless LAN. A LAN is a high-speed, low-error data network covering a relatively small geographic area (up to a few thousand meters). LANs connect workstations, peripherals, terminals, and other devices in a single building or other geographically limited area. LAN standards specify cabling and signaling at the physical and data link layers of the OSI model. Ethernet, FDDI, and Token Ring are widely used LAN technologies.