# 31 Days Before Your CCNA Exam (200-301)

## A Day-By-Day Review Guide for the CCNA 200-301 Certification Exam

ciscopress.com



Allan Johnson

# CCNA Countdown Calendar

The lines after the countdown number allow you to add the actual calendar days for reference.

| | | | |
|---|---|---|---|
| **31** ____ Networking Models, Devices, and Components | **30** ____ Ethernet Switching | **29** ____ Switch Configuration Basics | **28** ____ IPv4 Addressing |
| **24** ____ EtherChannel and HSRP | **23** ____ DHCP and DNS | **22** ____ Wireless Concepts | **21** ____ WLAN Configuration |
| **17** ____ The Routing Table | **16** ____ Inter-VLAN Routing | **15** ____ Static and Default Route Configuration | **14** ____ OSPF Operation |
| **10** ____ ACL Concepts | **9** ____ ACL Implementation | **8** ____ NAT | **7** ____ WAN, VPN, and IPsec |
| **3** ____ Cloud, Virtualization, and SDN | **2** ____ SDA and Cisco DNA Center | **1** ____ Network Automation | **EXAM** ____ Time ____ Location ____ |

**27**

IPv6 Addressing

**26**

VLAN and Trunking Concepts and Configuration

**25**

STP

**20**

LAN Security and Device Hardening

**19**

Basic Routing Concepts

**18**

Basic Router Configuration

**13**

Single-Area OSPF Implementation

**12**

Fine-Tuning and Troubleshooting OSPF

**11**

Network Security Concepts

**6**

QoS

**5**

CDP and LLDP

**4**

Device Monitoring, Management, and Maintenance

**DAY**

**27**

IPv6 Addressing

**26**

VLAN and Trunking Concepts and Configuration

**25**

STP

**20**

LAN Security and Device Hardening

**19**

Basic Routing Concepts

**18**

Basic Router Configuration

**13**

Single-Area OSPF Implementation

**12**

Fine-Tuning and Troubleshooting OSPF

**11**

Network Security Concepts

**6**

QoS

**5**

CDP and LLDP

**4**

Device Monitoring, Management, and Maintenance

# DAY

# 31 Days Before Your
# CCNA Exam

## A Day-By-Day Review Guide for the CCNA 200-301 Certification Exam

Allan Johnson

**Cisco Press** • 221 River Street • Hoboken, NJ 07030 USA

# 31 Days Before Your CCNA Exam

Allan Johnson



Published by:
Cisco Press
221 River Street
Hoboken, NJ 07030 USA



ScoutAutomatedPrintCode

Library of Congress Control Number: 2019919835

ISBN-13: 978-0-13-596408-8

ISBN-10: 0-13-596408-3

## Warning and Disclaimer

This book is designed to provide information about exam topics for the Cisco Certified Networking Associate (CCNA) certification. Every effort has been made to make this book as complete and as accurate as possible, but no warranty or fitness is implied.

The information is provided on an "as is" basis. The authors, Cisco Press, and Cisco Systems, Inc. shall have neither liability nor responsibility to any person or entity with respect to any loss or damages arising from the information contained in this book or from the use of the discs or programs that may accompany it.

The opinions expressed in this book belong to the author and are not necessarily those of Cisco Systems, Inc.

Microsoft and/or its respective suppliers make no representations about the suitability of the information contained in the documents and related graphics published as part of the services for any purpose. All such documents and related graphics are provided "as is" without warranty of any kind. Microsoft and/or its respective suppliers hereby disclaim all warranties and conditions with regard to this information, including all warranties and conditions of merchantability, whether express, implied or statutory, fitness for a particular purpose, title and non-infringement. In no event shall Microsoft and/or its respective suppliers be liable for any special, indirect or consequential damages or any damages whatsoever resulting from loss of use, data or profits, whether in an action of contract, negligence or other tortious action, arising out of or in connection with the use or performance of information available from the services.

The documents and related graphics contained herein could include technical inaccuracies or typographical errors. Changes are periodically added to the information herein. Microsoft and/or its respective suppliers may make improvements and/or changes in the product(s) and/or the program(s) described herein at any time. Partial screen shots may be viewed in full within the software version specified.

## Trademark Acknowledgments

All terms mentioned in this book that are known to be trademarks or service marks have been appropriately capitalized. Cisco Press or Cisco Systems, Inc., cannot attest to the accuracy of this information. Use of a term in this book should not be regarded as affecting the validity of any trademark or service mark.

## Special Sales

For information about buying this title in bulk quantities, or for special sales opportunities (which may include electronic versions; custom cover designs; and content particular to your business, training goals, marketing focus, or branding interests), please contact our corporate sales department at corpsales@pearsoned.com or (800) 382-3419.

For government sales inquiries, please contact governmentsales@pearsoned.com.

For questions about sales outside the U.S., please contact intlcs@pearson.com.

## Feedback Information

At Cisco Press, our goal is to create in-depth technical books of the highest quality and value. Each book is crafted with care and precision, undergoing rigorous development that involves the unique expertise of members from the professional technical community.

Readers' feedback is a natural continuation of this process. If you have any comments regarding how we could improve the quality of this book, or otherwise alter it to better suit your needs, you can contact us through email at feedback@ciscopress.com. Please make sure to include the book title and ISBN in your message.

We greatly appreciate your assistance.


| | |
|---|---|
| **Editor-in-Chief** | Mark Taub |
| **Production Line Manager** | Brett Bartow |
| **Alliances Manager, Cisco Press** | Arezou Gol |
| **Senior Editor** | James Manly |
| **Managing Editor** | Sandra Schroeder |
| **Development Editor** | Chris Cleveland |
| **Project Editor** | Mandie Frank |
| **Copy Editor** | Kitty Wilson |
| **Technical Editor** | Steve Stiles |
| **Editorial Assistant** | Cindy Teeters |
| **Designer** | Chuti Prasertsith |
| **Composition** | codeMantra |
| **Indexer** | Cheryl Ann Lenser |
| **Proofreader** | Charlotte Kughen |

CISCO

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.**

# About the Author

**Allan Johnson** entered the academic world in 1999, after 10 years as a business owner/operator to dedicate his efforts to his passion for teaching. He holds both an MBA and an MEd in occupational training and development. He taught a variety of technology courses to high school students and is an adjunct instructor at Del Mar College in Corpus Christi, Texas. Since 2006, Allan has worked full time for Cisco Networking Academy in several roles. He is currently engaged as curriculum lead.

## About the Technical Reviewer

**Steve Stiles** is a 20-year Cisco Network Academy Instructor for Rhodes State College and a Cisco Certified Instructor Trainer, having earned Cisco CCNA Security, CCNA CyberOps, and CCNP-level certifications, as well as numerous CompTIA certifications. He was the recipient of the 2012 Outstanding Teacher of the Year by the Ohio Association of Two Year Colleges and co-recipient for the Outstanding Faculty of the Year at Rhodes State College. Steve has a Bachelor's Degree from Western Governors in Information Technology–Security.

# Dedications

For my wife, Becky. Thank you for all your support during this crazy whirlwind of a year. You are the stabilizing force that keeps me grounded.

# Acknowledgments

As a technical author, I rely heavily on my technical editor; Steve Stiles had my back for this work. Thankfully, when James Manly contacted him, he was willing and able to do the arduous review work necessary to make sure that you get a book that is both technically accurate and unambiguous.

Wendell Odom's *CCNA 200-301 Official Cert Guide, Volume1* and *Volume 2* were two of my main sources. These two books have the breadth and depth needed to master the CCNA exam topics.

The Cisco Networking Academy authors for the online curriculum and series of Companion Guides take the reader deeper, past the CCNA exam topics, with the ultimate goal of preparing the student not only for CCNA certification but for more advanced college-level technology courses and degrees as well. Thank you especially to Rick Graziani, Bob Vachon, John Pickard, Dave Holzinger, Jane Gibbons, Martin Benson, Suk-Yi Pennock, Allan Reid, Jane Brooke, Anna Bolen, Telethia Willis, and the rest of the ACE team. Their excellent treatment of the material is reflected throughout this book.

James Manly, senior editor, has effectively assumed the mantle of juggling multiple projects simultaneously, steering each from beginning to end. This is my third project with James, and he is competently filling big shoes. Thank you, James, for shepherding this project for me.

Thank you to the professional and thorough review of this work by development editor Christopher Cleveland, project editor Mandie Frank, and copy editor Kitty Wilson. I've worked with the stellar Chris and Mandie on many previous projects. Their combined efforts ensure that what I authored is ready for publication. This is my first time working with Kitty. Her attention to the clarity of what you read plays no small part in making this the best edition to date.

And to the rest of the Pearson family who contributes in countless ways to bring a book to the reader, thank you for all your hard work.

# Credits

| | |
|---|---|
| Figure 23-10 | Screenshot of Windows 10 © Microsft 2019 |
| Figure 23-11 | Screenshot reprinted with permission from Apple Inc. |
| Figure 23-12 | Screenshot of Linux Ubuntu © Ubuntu 2019 |
| PostExam | Matthew Moran, *Building Your I.T. Career: A Complete Toolkit for a Dynamic Career in Any Economy*, 2nd Edition (Pearson IT Certification, 2013, ISBN: 9780789749437) |

# Contents at a Glance

# Contents

# Command Syntax Conventions

The conventions used to present command syntax in this book are the same conventions used in the IOS Command Reference. The Command Reference describes these conventions as follows:

- **Boldface** indicates commands and keywords that are entered literally as shown. In actual configuration examples and output (not general command syntax), boldface indicates commands that are manually input by the user (such as a **show** command).
- *Italic* indicates arguments for which you supply actual values.
- Vertical bars (|) separate alternative, mutually exclusive elements.
- Square brackets ([ ]) indicate an optional element.
- Braces ({ }) indicate a required choice.
- Braces within brackets ([{ }]) indicate a required choice within an optional element.

# Reader Services

**Register your copy** at www.ciscopress.com/title/9780135964088 for convenient access to downloads, updates, and corrections as they become available. To start the registration process, go to www.ciscopress.com/register and log in or create an account. (Be sure to check the box indicating that you would like to hear from us to receive exclusive discounts on future editions of this product.) Enter the product ISBN **9780135964088** and click **Submit**. When the process is complete, you will find any available bonus content under Registered Products.

# Introduction

If you're reading this introduction, you've probably already spent a considerable amount of time and energy pursuing your CCNA 200-301 certification. Regardless of how you got to this point in your travels through your CCNA studies, *31 Days Before Your CCNA Exam* most likely represents the last leg of your journey on your way to the destination: to become a Cisco Certified Network Associate. However, if you are like me, you might be reading this book at the *beginning* of your studies. If so, this book provides an excellent overview of the material you must now spend a great deal of time studying and practicing. But I must warn you: Unless you are extremely well versed in networking technologies and have considerable experience configuring and troubleshooting Cisco routers and switches, this book will *not* serve you well as the sole resource for your exam preparations. Therefore, let me spend some time discussing my recommendations for study resources.

## Study Resources

Cisco Press and Pearson IT Certification offer an abundance of CCNA-related books to serve as your primary source for learning how to install, configure, operate, and troubleshoot small to medium-size routed and switched networks.

### Primary Resources

First on the list of important resources is Wendell Odom's *CCNA 200-301 Official Cert Guide Library* (ISBN: 9781587147142). If you do not buy any other books, buy these. Wendell's method of teaching, combined with his technical expertise and down-to-earth style, is unsurpassed in our industry. As you read through his books, you sense that he is sitting right there next to you, walking you through the material. With your purchase, you get access to practice exams and study materials and other online resources that are worth the price of the book. There is no better resource on the market for a CCNA candidate.

If you are a Cisco Networking Academy student, you are blessed with access to the online version of the CCNA version 7 curriculum and the wildly popular Packet Tracer network simulator. The Cisco Network Academy curriculum has three courses. To learn more about CCNAv7 courses and to find an Academy near you, visit http://www.netacad.com.

However, if you are not an Academy student but want to benefit from the extensive authoring done for these courses, you can buy any or all of CCNAv7 Companion Guides (CGs) and Labs & Study Guides (LSGs) of the Academy's popular online curriculum. Although you will not have access to the Packet Tracer files, you will have access to the tireless work of an outstanding team of Cisco Academy instructors dedicated to providing students with comprehensive and engaging CCNA preparation course material. The titles and ISBNs for the CCNAv7 CGs and LSGs follow:

- *Introduction to Networks v7 Companion Guide* (ISBN: 9780136633662)
- *Introduction to Networks v7 Labs & Study Guide* (ISBN: 9780136634454)
- *Switching, Routing, and Wireless Essentials v7 Companion Guide* (ISBN: 9780136729358)
- *Switching, Routing, and Wireless Essentials v7 Labs & Study Guide* (ISBN: 9780136634386)

- *Enterprise Networking, Security, and Automation v7 Companion Guide* (ISBN: 9780136634324)
- *Enterprise Networking, Security, and Automation v7 Labs & Study Guide* (ISBN: 9780136634690)

You can find these books at http://www.ciscopress.com by clicking the Cisco Networking Academy link.

## Supplemental Resources

In addition to the book you hold in your hands, I recommend three supplemental resources to augment your final 31 days of review and preparation.

First is Scott Empson's very popular *CCNA 200-301 Portable Command Guide* (ISBN: 9780135937822). This guide is much more than just a listing of commands and what they do. Yes, it summarizes all the CCNA certification-level IOS commands, keywords, command arguments, and associated prompts. It also provides you with tips and examples of how to apply the commands to real-world scenarios. Configuration examples throughout the book provide you with a better understanding of how these commands are used in simple network designs.

Second, Kevin Wallace's *CCNA 200-301 Complete Video Course and Practice Test* (ISBN: 9780136582755) is a comprehensive training course that brings Cisco CCNA exam topics to life through the use of real-world demonstrations, animations, live instruction, and configurations, making learning these foundational networking topics easy and fun. Kevin's engaging style and love for the technology are infectious. The course also includes excellent practice tests.

Third, Wendell Odom's *IP Subnetting LiveLessons* (ISBN: 9780135497777) and *IP Subnetting Practice Questions Kit* (ISBN: 9780135647288) will help you master this crucial skill. Subnetting is not only an IPv4 address design skill, it is also crucial skill for troubleshooting situations where IPv4 addressing has been misconfigured. You are likely to have both types of questions on the CCNA exam.

## The Cisco Learning Network

Finally, if you have not done so already, you should register with The Cisco Learning Network at https://learningnetwork.cisco.com. Sponsored by Cisco, The Cisco Learning Network is a free social learning network where IT professionals can engage in the common pursuit of enhancing and advancing their IT careers. Here you can find many resources to help you prepare for your CCNA exam, in addition to a community of like-minded people ready to answer your questions, help you with your struggles, and share in your triumphs.

So which resources should you buy? The answer to that question depends largely on how deep your pockets are and how much you like books. If you're like me, you must have it all! I admit it; my bookcase is a testament to my Cisco "geekness." But if you are on a budget, choose one of the primary study resources and one of the supplemental resources (such as Wendell Odom's certification library and Scott Empson's command guide). Whatever you choose, you will be in good hands. Any or all of these authors will serve you well.

## Goals and Methods

The main goal of this book is to provide you with a clear and succinct review of the CCNA objectives. Each day's exam topics are grouped into a common conceptual framework and use the following format:

- A title for the day that concisely states the overall topic
- A list of one or more CCNA 200-301 exam topics to be reviewed
- A "Key Topics" section that introduces the review material and quickly orients you to the day's focus
- An extensive review section consisting of short paragraphs, lists, tables, examples, and graphics
- A "Study Resources" section to give you a quick reference for locating more in-depth treatment of the day's topics

The book counts down starting with Day 31 and continues through exam day to provide post-test information. Inside this book is also a calendar and checklist that you can tear out and use during your exam preparation.

Use the calendar to enter each actual date beside the countdown day and the exact day, time, and location of your CCNA exam. The calendar provides a visual for the time you can dedicate to each CCNA exam topic.

The checklist highlights important tasks and deadlines leading up to your exam. Use it to help you map out your studies.

## Who Should Read This Book?

The audience for this book is anyone finishing preparation for taking the CCNA 200-301 exam. A secondary audience is anyone needing a refresher review of CCNA exam topics—possibly before attempting to recertify or sit for another certification for which the CCNA is a prerequisite.

## Getting to Know the CCNA 200-301 Exam

For the current certification announced in June 2019, Cisco created the CCNA 200-301 exam. This book focuses on the entire list of topics published for the CCNA 200-301 exam.

The CCNA 200-301 exam is a 120-minute exam associated with the CCNA certification. This exam tests a candidate's knowledge and skills related to network fundamentals, network access, IP connectivity, IP services, security fundamentals, and automation and programmability. Use the following steps to access a tutorial at home that demonstrates the exam environment before you go to take the exam:

**Step 1.** Visit http://learningnetwork.cisco.com.

**Step 2.** Search for "cisco certification exam tutorial".

**Step 3.** Look through the top results to find the page with videos that walk you through each exam question type.

When you get to the testing center and check in, the proctor verifies your identity, gives you some general instructions, and takes you into a quiet room containing a PC. When you're at the PC, you have a few things to do before the timer starts on your exam. For instance, you can take the tutorial to get accustomed to the PC and the testing engine. Every time I sit for an exam, I go through the tutorial even though I know how the test engine works. It helps me settle my nerves and get focused. Anyone who has user-level skills in getting around a PC should have no problem with the testing environment.

When you start the exam, you are asked a series of questions. The questions are presented one at a time and must be answered before moving on to the next question. The exam engine does not let you go back and change any answers. Each exam question is in one of the following formats:

- Multiple choice
- Fill in the blank
- Drag and drop
- Testlet
- Simlet
- Simulation

The multiple-choice format simply requires that you point and click a circle or check box next to the correct answer(s). Cisco traditionally tells you how many answers you need to choose, and the testing software prevents you from choosing too many or too few.

Fill-in-the-blank questions usually require you only to type numbers. However, if words are requested, the case does not matter unless the answer is a command that is case sensitive (such as passwords and device names, when configuring authentication).

Drag-and-drop questions require you to click and hold, move a button or an icon to another area, and release the mouse button to place the object somewhere else—usually in a list. For some questions, to get the question correct, you might need to put a list of five things in the proper order.

A testlet contains one general scenario and several multiple-choice questions about the scenario. Testlets are ideal if you are confident in your knowledge of the scenario's content because you can leverage your strength over multiple questions.

A simlet is similar to a testlet, in that you are given a scenario with several multiple-choice questions. However, a simlet uses a network simulator to allow you access to a simulation of the command line of Cisco IOS Software. You can use **show** commands to examine a network's current behavior and answer the question.

A simulation also involves a network simulator, but you are given a task to accomplish, such as implementing a network solution or troubleshooting an existing network implementation. You do this by configuring one or more routers and switches. The exam grades the question based on the configuration you changed or added. A newer form of the simulation question is the GUI-based simulation, which simulates a graphical interface such as that found on a Linksys router or the Cisco Security Device Manager.

## Topics Covered on the CCNA Exam

Table I-1 summarizes the seven domains of the CCNA 200-301 exam:

**Table I-1 CCNA 200-301 Exam Domains and Weightings**

| Domain | Percentage of Exam |
|---|---|
| 1.0 Network Fundamentals | 20% |
| 2.0 Network Access | 20% |
| 3.0 IP Connectivity | 25% |
| 4.0 IP Services | 10% |
| 5.0 Security Fundamentals | 15% |
| 6.0 Automation and Programmability | 10% |

Although Cisco outlines general exam topics, not all topics might appear on the CCNA exam; likewise, topics that are not specifically listed might appear on the exam. The exam topics that Cisco provides and that this book covers provide a general framework for exam preparation. Be sure to check Cisco's website for the latest exam topics.

## Registering for the CCNA 200-301 Exam

If you are starting this book 31 days before you take the CCNA 200-301 exam, register for the exam right now. In my testing experience, there is no better motivator than a scheduled test date staring me in the face. I'm willing to bet the same holds true for you. Don't worry about unforeseen circumstances. You can cancel your exam registration for a full refund up to 24 hours before taking the exam. So if you're ready, gather the following information and register right now!

- Legal name
- Social Security or passport number
- Company name
- Valid email address
- Method of payment

You can schedule your exam at any time by visiting www.pearsonvue.com/cisco/. I recommend that you schedule it for 31 days from now. The process and available test times vary based on the local testing center you choose.

Remember, there is no better motivation for study than an actual test date. *Sign up today*.

# Networking Models, Devices, and Components

## CCNA 200-301 Exam Topics

- Explain the role and function of network components
- Describe characteristics of network topology architectures
- Compare physical interface and cabling types
- Identify interface and cable issues (collisions, errors, mismatch duplex and/or speed
- Compare TCP to UDP

## Key Points

Both the Open Systems Interconnection (OSI) and Transmission Control Protocol/Internet Protocol (TCP/IP) networking models are important conceptual frameworks for understanding networks. Today we review the layers and functions of each model, along with the process of data flow from source to destination. We also spend some time on Transmission Control Protocol (TCP) and the User Datagram Protocol (UDP). Then we wrap up the day with a look at devices used in today's networks, the media used to interconnect those devices, and the different types of network topologies.

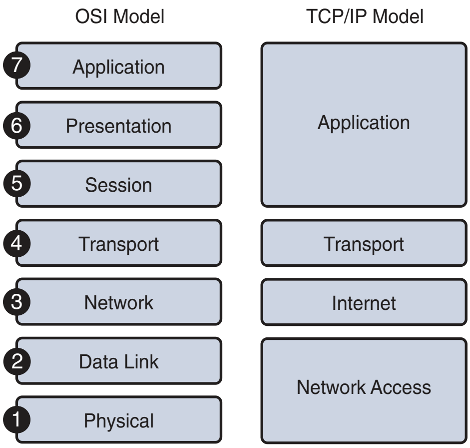
**NOTE:** This day might seem a bit long. However, you need to be very familiar with all of this content. Scan the day, focusing on areas where you feel least confident in your knowledge.

## The OSI and TCP/IP Models

To understand how communication occurs across the network, you can use layered models as a framework for representing and explaining networking concepts and technologies. Layered models, such as the TCP/IP and OSI models, support interoperability between competing vendor product lines.

The OSI model principally serves as a tool for explaining networking concepts and troubleshooting. However, the protocols of the TCP/IP suite are the rules by which networks now operate. Because both models are important, you should be well versed in each model's layers and know how the models map to each other. Figure 31-1 summarizes the two models.

**Figure 31-1 OSI and TCP/IP Models**




Using two models can be confusing; however, these simple guidelines might help:

- When discussing layers of a model, we are usually referring to the OSI model.
- When discussing protocols, we are usually referring to the TCP/IP model.

The next sections quickly review the OSI layers and the TCP/IP protocols.

## OSI Layers

Table 31-1 summarizes the layers of the OSI model and provides a brief functional description.

**Table 31-1 OSI Model Layers and Functions**

| Layer | Functional Description |
|---|---|
| Application (7) | Refers to interfaces between network and application software. Also includes authentication services. |
| Presentation (6) | Defines the format and organization of data. Includes encryption. |
| Session (5) | Establishes and maintains end-to-end bidirectional flows between endpoints. Includes managing transaction flows. |
| Transport (4) | Provides a variety of services between two host computers, including connection establishment and termination, flow control, error recovery, and segmentation of large data blocks into smaller parts for transmission. |
| Network (3) | Refers to logical addressing, routing, and path determination. |
| Data link (2) | Formats data into frames appropriate for transmission onto some physical medium. Defines rules for when the medium can be used. Defines the means by which to recognize transmission errors. |
| Physical (1) | Defines the electrical, optical, cabling, connectors, and procedural details required for transmitting bits, represented as some form of energy passing over a physical medium. |

The following mnemonic phrase, in which the first letter represents the layer (*A* stands for *application*), can help in memorizing the name and order of the layers from top to bottom:

```
All People Seem To Need Data Processing
```

## TCP/IP Layers and Protocols

The TCP/IP model defines four categories of functions that must occur for communications to succeed. Most protocol models describe vendor-specific protocol stacks. However, because the TCP/IP model is an open standard, one company does not control the definition of the model.

Table 31-2 summarizes the TCP/IP layers, their functions, and the most common protocols.

**Table 31-2 TCP/IP Layer Functions**

| TCP/IP Layer | Function | Example Protocols |
|---|---|---|
| Application | Represents data to the user and controls dialogue | DNS, Telnet, SMTP, POP3, IMAP, DHCP, HTTP, FTP, SNMP |
| Transport | Supports communication between diverse devices across diverse networks | TCP, UDP |
| Internet | Determines the best path through the network | IP, ARP, ICMP |
| Network access | Controls the hardware devices and media that make up the network | Ethernet, Wireless |

In the coming days, we review these protocols in more detail. For now, a brief description of the main TCP/IP protocols follows:

- **Domain Name System (DNS):** Provides the IP address of a website or domain name so that a host can connect to it
- **Telnet:** Enables administrators to log in to a host from a remote location
- **Simple Mail Transfer Protocol (SMTP), Post Office Protocol (POP3), and Internet Message Access Protocol (IMAP):** Facilitate the sending of email messages between clients and servers
- **Dynamic Host Configuration Protocol (DHCP):** Assigns IP addressing to requesting clients
- **Hypertext Transfer Protocol (HTTP):** Transfers information between web clients and web servers
- **File Transfer Protocol (FTP):** Facilitates the download and upload of files between an FTP client and an FTP server
- **Simple Network Management Protocol (SNMP):** Enables network management systems to monitor devices attached to the network
- **Transmission Control Protocol (TCP):** Supports virtual connections between hosts on the network to provide reliable delivery of data

- **User Datagram Protocol (UDP):** Supports faster, unreliable delivery of lightweight or time-sensitive data
- **Internet Protocol (IP):** Provides a unique global address to computers for communicating over the network
- **Address Resolution Protocol (ARP):** Finds a host's hardware address when only the IP address is known
- **Internet Control Message Protocol (ICMP):** Sends error and control messages, including reachability of another host and availability of services
- **Ethernet:** Serves as the most popular LAN standard for framing and preparing data for transmission onto the media
- **Wireless**: Includes both IEEE 802.11 standards for wireless local-area networks (WLANs) and cellular access options.

## Protocol Data Units and Encapsulation

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 is commonly known as the *encapsulation process*. The data structure at any given layer is called a *protocol data unit* (*PDU*). Table 31-3 lists the PDUs at each layer of the OSI model.

**Table 31-3 PDUs at Each Layer of the OSI Model**

| OSI Layer | PDU |
|---|---|
| Application | Data |
| Presentation | Data |
| Session | Data |
| Transport | Segment |
| Network | Packet |
| Data link | Frame |
| Physical | Bits |

The following steps summarize the communication process from any source to any destination:

**Step 1.** Data is created at the application layer of the originating source device.

**Step 2.** As the data passes down the protocol stack in the source device, it is segmented and encapsulated.

**Step 3.** The data is generated onto the media at the network access layer of the stack.

**Step 4.** The data is transported through the internetwork, which consists of media and any intermediary devices.

**Step 5.** The destination device receives the data at the network access layer.

**Step 6.** As the data passes up the stack in the destination device, it is decapsulated and reassembled.

**Step 7.** The data is passed to the destination application at the application layer of the destination device.

# The TCP/IP Application Layer

The application layer of the TCP/IP model provides an interface between software such as a web browser and the network itself. The process of requesting and receiving a web page works like this:

**Step 1.** An HTTP request is sent, including an instruction to "get" a file (which is often a website's home page).

**Step 2.** An HTTP response is sent from the web server with a code in the header, usually either 200 (request succeeded, and information is returned in response) or 404 (page not found).

The HTTP request and the HTTP response are encapsulated in headers. The content of the headers allows the application layers on each end device to communicate. Regardless of the application layer protocol (HTTP, FTP, DNS, and so on), all headers use the same general process for communicating between application layers on the end devices.

# The TCP/IP Transport Layer

The transport layer, through TCP, provides a mechanism to guarantee delivery of data across the network. TCP supports error recovery to the application layer through the use of basic acknowledgment logic. Adding to the process for requesting a web page, TCP operation works like this:

**Step 1.** The web client sends an HTTP request for a specific web server down to the transport layer.

**Step 2.** TCP encapsulates the HTTP request with a TCP header and includes the destination port number for HTTP.

**Step 3.** Lower layers process and send the request to the web server.

**Step 4.** The web server receives HTTP requests and sends a TCP acknowledgment back to the requesting web client.

**Step 5.** The web server sends the HTTP response down to the transport layer.

**Step 6.** TCP encapsulates the HTTP data with a TCP header.

**Step 7.** Lower layers process and send the response to the requesting web client.

**Step 8.** The requesting web client sends an acknowledgment back to the web server.

If data is lost at any point during this process, TCP must recover the data. HTTP at the application layer does not get involved in error recovery.

In addition to providing TCP, the transport layer provides UDP, a connectionless, unreliable protocol for sending data that does not require or need error recovery. Table 31-4 lists the main features that the transport protocols support. Both TCP and UDP support the first function; only TCP supports the rest.

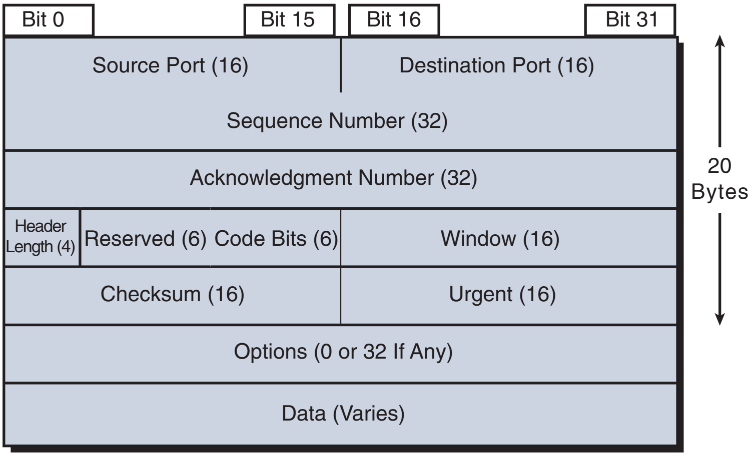
**Table 31-4 TCP/IP Transport Layer Features**

| Function | Description |
|---|---|
| Multiplexing using ports | Function that enables receiving hosts to choose the correct application for which the data is destined, based on the destination port number. |
| Error recovery (reliability) | Process of numbering and acknowledging data with Sequence and Acknowledgment header fields. |
| Flow control using windowing | Process that involves a sliding window size that the two end devices dynamically agree upon at various points during the virtual connection. The window size, represented in bytes, is the maximum amount of data the source will send before receiving an acknowledgment from the destination. |
| Connection establishment and termination | Process used to initialize port numbers and Sequence and Acknowledgment fields. |
| Ordered data transfer and data segmentation | A continuous stream of bytes from an upper-layer process that is "segmented" for transmission and delivered to upper-layer processes at the receiving device, with the bytes in the same order. |

## TCP Header

TCP provides error recovery, but to do so, it consumes more bandwidth and uses more processing cycles than UDP. TCP and UDP rely on IP for end-to-end delivery. TCP is concerned with providing services to the applications of the sending and receiving computers. To provide all these services, TCP uses a variety of fields in its header (see Figure 31-2).

**Figure 31-2 TCP Header**



## Port Numbers

The first two fields of the TCP header—the source and destination ports—are also part of the UDP header (shown later, in Figure 31-7). Port numbers provide TCP (and UDP) with a way to multiplex multiple applications on the same computer. Web browsers now support multiple tabs or pages. Each time you open a new tab and request another web page, TCP assigns a different source port number and sometimes multiple port numbers. For example, you might have five web pages open. TCP almost always assigns destination port 80 for all five sessions. However, the source port for each is different. This is how TCP (and UDP) multiplexes the conversation so that the web browser knows in which tab to display the data.

TCP and UDP usually dynamically assign the source ports, starting at 1024 up to a maximum of 65535. Port numbers below 1024 are reserved for well-known applications. Table 31-5 lists several popular applications and their well-known port numbers.

**Table 31-5 Popular Applications and Their Well-Known Port Numbers**

| Port Number | Protocol | Application |
|---|---|---|
| 20 | TCP | FTP data |
| 21 | TCP | FTP control |
| 22 | TCP | SSH |
| 23 | TCP | Telnet |
| 25 | TCP | SMTP |
| 53 | UDP, TCP | DNS |
| 67, 68 | UDP | DHCP |
| 69 | UDP | TFTP |
| 80 | TCP | HTTP (WWW) |
| 110 | TCP | POP3 |
| 161 | UDP | SNMP |
| 443 | TCP | HTTPS (SSL) |
| 16384–32767 | UDP | RTP-based voice (VoIP) and video |

## Error Recovery

TCP provides error recovery, also known as *reliability*, during data transfer sessions between two end devices that have established a connection. The Sequence and Acknowledgment fields in the TCP header track every byte of data transfer and ensure that missing bytes are retransmitted.

In Figure 31-3, the Acknowledgment field sent by the web client (4000) implies the next byte to be received; this is called *positive acknowledgment*.

**Figure 31-3 TCP Acknowledgment Without Errors**

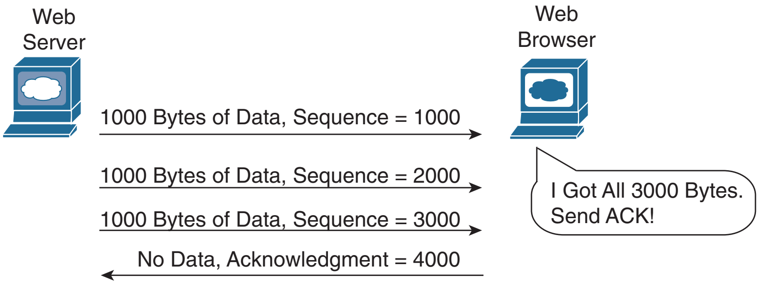
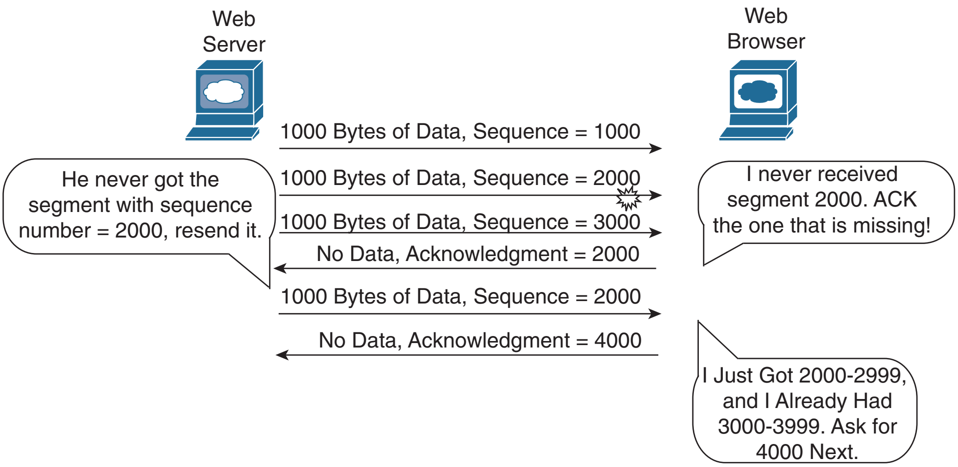



Figure 31-4 shows the same scenario, except now with some errors. The second TCP segment was lost in transmission. Therefore, the web client replies with an ACK field set to 2000. This is called a *positive acknowledgment with retransmission* (*PAR*) because the web client is requesting that some of the data be retransmitted. The web server now re-sends data starting at segment 2000. In this way, lost data is recovered.

**Figure 31-4 TCP Acknowledgment with Errors**




Although not shown, the web server also sets a retransmission timer and awaits acknowledgment, just in case the acknowledgment is lost or all transmitted segments are lost. If that timer expires, the web server sends all segments again.

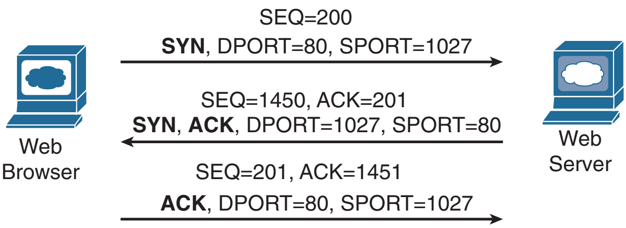
# Flow Control

TCP handles flow control through a process called *windowing*. The two end devices negotiate the window size when initially establishing the connection; then they dynamically renegotiate window size during the life of the connection, increasing its size until it reaches the maximum window size of 65,535 bytes or until errors occur. Window size is specified in the Window field of the TCP header. After sending the amount of data specified in the window size, the source must receive an acknowledgment before sending the next window size of data.

## Connection Establishment and Termination

Connection establishment is the process of initializing Sequence and Acknowledgment fields and agreeing on port numbers and window size. The three-way connection establishment phase shown in Figure 31-5 must occur before data transfer can proceed.

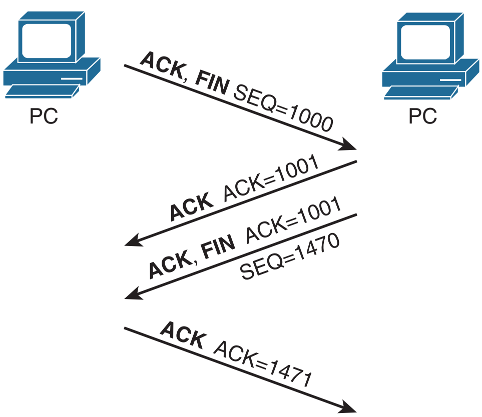
**Figure 31-5 TCP Connection Establishment**




In the figure, DPORT and SPORT are the destination and source ports. SEQ is the sequence number. In bold are SYN and ACK, each representing a 1-bit flag in the TCP header used to signal connection establishment. TCP initializes the Sequence Number and Acknowledgment Number fields to any number that fits into the 4-byte fields. The initial Sequence Number is a random 32-bit number generated with each new transmission. The Acknowledgment Number is received back and increments the sender's sequence number by 1.

When data transfer is complete, a four-way termination sequence occurs. This sequence uses an additional flag, called the FIN bit (see Figure 31-6).

**Figure 31-6 TCP Connection Termination**




## UDP

TCP establishes and terminates connections between endpoints, whereas UDP does not. Therefore, UDP is called a *connectionless protocol*. It provides no reliability, no windowing, and no reordering of the data. However, UDP does provide data transfer and multiplexing using port numbers, and it does so with fewer bytes of overhead and less processing than TCP. Applications that use UDP, such as VoIP, trade the possibility of some data loss for less delay. Figure 31-7 compares the two headers.

**Figure 31-7 TCP and UDP Headers**

| 2 | 2 | 4 | 4 | 4 bits | 6 bits | 6 bits | 2 | 2 | 2 | 3 | 1 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Source Port | Dest. Port | Sequence Number | Ack. Number | Offset | Reserved | Flags | Window Size | Checksum | Urgent | Options | PAD |

TCP Header

| 2 | 2 | 2 | 2 |
|---|---|---|---|
| Source Port | Dest. Port | Length | Checksum |

UDP Header

* Unless Specified, Lengths Shown Are the Numbers of Bytes

# The TCP/IP Internet Layer

The Internet layer of the TCP/IP model and its Internet Protocol (IP) define addresses so that each host computer can have a different IP address. In addition, the Internet layer defines the process of routing so that routers can determine the best path for sending packets to the destination. Continuing with the web page example, IP addresses the data as it passes from the transport layer to the Internet layer:

**Step 1.** The web client sends an HTTP request.

**Step 2.** TCP encapsulates the HTTP request.

**Step 3.** IP encapsulates the transport segment into a packet, adding source and destination addresses.

**Step 4.** Lower layers process and send the request to the web server.

**Step 5.** The web server receives HTTP requests and sends a TCP acknowledgment back to the requesting web client.

**Step 6.** The web server sends the HTTP response down to the transport layer.

**Step 7.** TCP encapsulates the HTTP data.

**Step 8.** IP encapsulates the transport segment into a packet, adding source and destination addresses.

**Step 9.** Lower layers process and send the response to the requesting web client.

**Step 10.** The requesting web client sends an acknowledgment back to the web server.

The operation of IP includes not only addressing but also the process of routing the data from source to destination. IP is further discussed and reviewed in the upcoming days.

# The TCP/IP Network Access Layer

IP depends on the network access layer to deliver IP packets across a physical network. Therefore, the network access layer defines the protocols and hardware required to deliver data across some physical network by specifying exactly how to physically connect a networked device to the physical media over which data can be transmitted.

The network access layer includes many protocols to deal with the different types of media that data can cross on its way from source device to destination device. For example, data might need to travel first on an Ethernet link and then cross a Point-to-Point (PPP) link, then a Frame Relay link, then a Multiprotocol Label Switching (MPLS) link, and then finally an Ethernet link to reach the destination. At each transition from one media type to another, the network access layer provides the protocols, cabling standards, headers, and trailers to send data across the physical network.

Many times, a local link address is needed to transfer data from one hop to the next. For example, in an Ethernet LAN, Media Access Control (MAC) addresses are used between the sending device and its local gateway router. At the gateway router (depending on the needs of the outbound interface), the Ethernet header might be replaced with an MPLS label. The label serves the same purpose as MAC addresses in Ethernet: to get the data across the link from one hop to the next so that the data can continue its journey to the destination. Some protocols, such as PPP, do not need a link address because only one other device on the link can receive the data.

With the network access layer, we can now finalize our web page example. The following greatly simplifies and summarizes the process of requesting and sending a web page:

**Step 1.** The web client sends an HTTP request.

**Step 2.** TCP encapsulates the HTTP request.

**Step 3.** IP encapsulates the transport segment into a packet, adding source and destination addresses.

**Step 4.** The network access layer encapsulates the packet in a frame, addressing it for the local link.

**Step 5.** The network access layer sends the frame as bits on the media.

**Step 6.** Intermediary devices process the bits at the network access and Internet layers and then forward the data toward the destination.

**Step 7.** The web server receives the bits on the physical interface and sends them up through the network access and Internet layers.

**Step 8.** The web server sends a TCP acknowledgment back to the requesting web client.

**Step 9.** The web server sends the HTTP response down to the transport layer.

**Step 10.** TCP encapsulates the HTTP data.

**Step 11.** IP encapsulates the transport segment into a packet, adding source and destination addresses.

**Step 12.** The network access layer encapsulates the packet in a frame, addressing it for the local link.

**Step 13.** The network access layer sends the frame as bits on the media.

**Step 14.** Lower layers process and send the response to the requesting web client.

**Step 15.** The response travels back to the source over multiple data links.

**Step 16.** The requesting web client receives the response on the physical interface and sends the data up through the network access and Internet layers.

**Step 17.** The requesting web client sends a TCP acknowledgment back to the web server.

**Step 18.** The web page is displayed in the requesting device's browser.

# Data Encapsulation Summary

Each layer of the TCP/IP model adds its own header information. As the data travels down through the layers, it is encapsulated with a new header. At the network access layer, a trailer is also added. This encapsulation process is described in five steps:

**Step 1.** Create and encapsulate the application data with any required application layer headers. For example, the HTTP OK message can be returned in an HTTP header, followed by part of the contents of a web page.

**Step 2.** Encapsulate the data supplied by the application layer inside a transport layer header. For end-user applications, a TCP or UDP header is typically used.

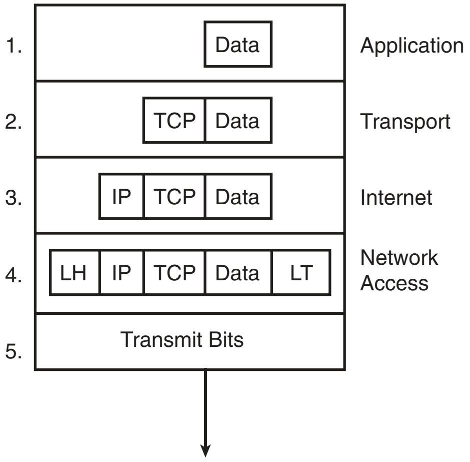
**Step 3.** Encapsulate the data supplied by the transport layer inside an Internet layer (IP) header. IP is the only protocol available in the TCP/IP network model at the Internet layer.

**Step 4.** Encapsulate the data supplied by the Internet layer inside a network access layer header and trailer. This is the only layer that uses both a header and a trailer.

**Step 5.** Transmit the bits. The physical layer encodes a signal onto the medium to transmit the frame.

The numbers in Figure 31-8 correspond to the five steps in the list, graphically showing the same encapsulation process.

**Figure 31-8 Five Steps of Data Encapsulation**




---

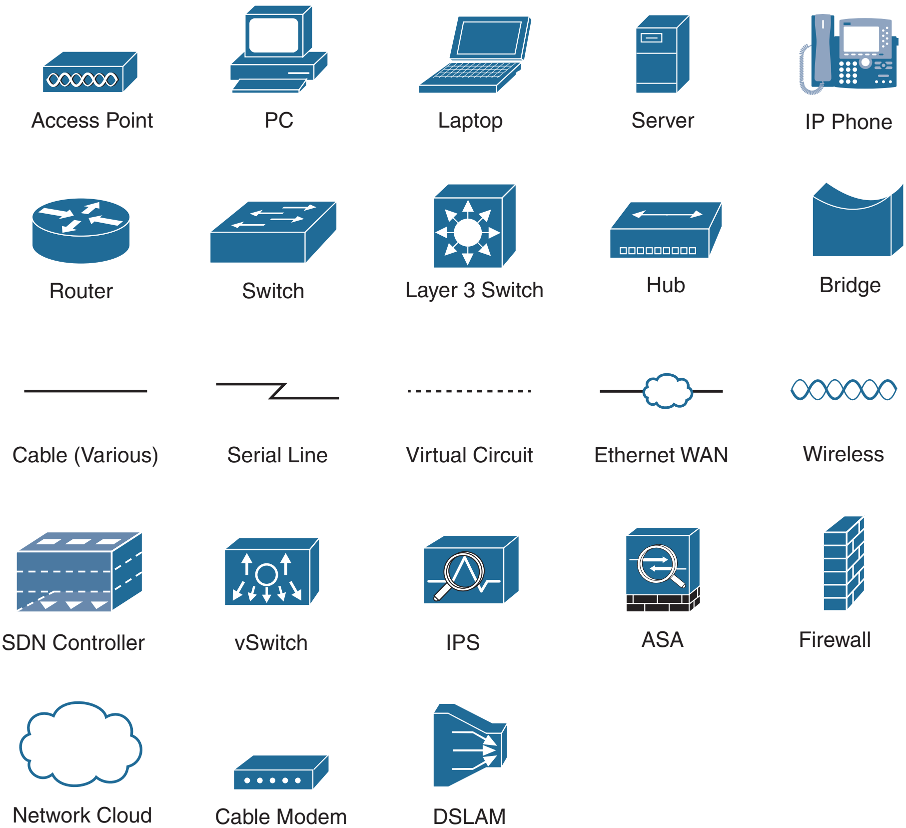
**NOTE:** The letters LH and LT stand for link header and link trailer, respectively, and refer to the data link layer header and trailer.

---

# Networking Icons

In order to interpret networking diagrams or topologies, you must understand the symbols or icons used to represent different networking devices and media. The icons in Figure 31-9 are the most common networking symbols for CCNA studies.

**Figure 31-9 Networking Icons**




# Devices

In today's wired networks, switches are almost exclusively used to connect end devices to a single LAN. Occasionally, you might see a hub connecting end devices, but hubs are really legacy devices. The following are the differences between a hub and a switch:

- Hubs were typically chosen as intermediary devices within very small LANs, in which bandwidth usage was not an issue or cost limitations were a factor. In today's networks, switches have replaced hubs.
- Switches replaced hubs as local-area network (LAN) intermediary devices because a switch can segment collision domains and provide enhanced security.

## Switches

When choosing a switch, these are the main factors to consider:

- **Cost:** The cost is determined by the number and type of ports, network management capabilities, embedded security technologies, and optional advanced switching technologies.
- **Interface characteristics:** The number of ports must be sufficient both for now and for future expansion. Other characteristics include uplink speeds, a mixture of UTP and fiber, and modularity.
- **Hierarchical network layer:** Switches at the access layer have different requirements than switches at the distribution or core layers.

### Access Layer Switches

Access layer switches facilitate the connection of end devices to the network. Features of access layer switches include the following:

- Port security
- VLANs
- Fast Ethernet/Gigabit Ethernet
- Power over Ethernet (PoE)
- Link aggregation
- Quality of service (QoS)

### Distribution Layer Switches

Distribution layer switches receive the data from the access layer switches and forward it to the core layer switches. Features of distribution layer switches include the following:

- Layer 3 support
- High forwarding rate
- Gigabit Ethernet/10 Gigabit Ethernet
- Redundant components
- Security policies/access control lists
- Link aggregation
- QoS

### Core Layer Switches

Core layer switches make up the backbone and are responsible for handling the majority of data on a switched LAN. Features of core layer switches include the following:

- Layer 3 support
- Very high forwarding rate
- Gigabit Ethernet/10 Gigabit Ethernet

- Redundant components
- Link aggregation
- QoS

## Routers

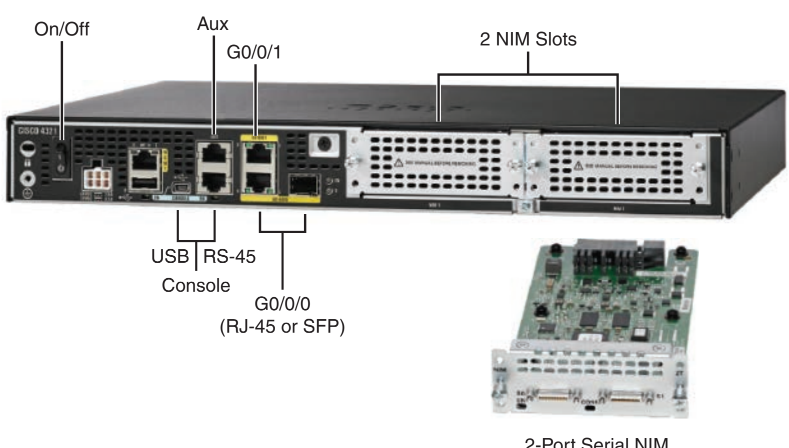
Routers are the primary devices used to interconnect networks—LANs, WANs, and WLANs. When choosing a router, the main factors to consider are as follows:

- **Expandability:** Provides flexibility to add new modules as needs change.
- **Media:** Determines the type of interfaces the router needs to support for the various network connections.
- **Operating system features:** Determines the version of IOS loaded on the router. Different IOS versions support different feature sets. Features to consider include security, QoS, VoIP, and routing complexity, among others.

Figure 31-10 shows a Cisco 4321 router, which provides the following connections:

- **Console ports:** Two console ports for the initial configuration, using a regular RJ-45 port and a USB Type-B (mini-B USB) connector.
- **AUX port:** An RJ-45 port for remote management access.
- **LAN interfaces:** Two Gigabit Ethernet interfaces for LAN access (G0/0/0 and G0/0/1). If the RJ-45 G0/0/0 port is used, then the small form-factor pluggable (SFP) port cannot be used. WAN services would then be provided through an expansion card in the network interface module (NIM) slots.
- **Ethernet WAN:** The other G0/0/0 physical port, an SFP port that would support various Ethernet WAN connections, typically fiber. If it is used, the Gi0/0 RJ-45 port is disabled.
- **NIM slots:** Two slots that support different types of interface modules, including serial (shown in Figure 31-10), digital subscriber line (DSL), switch port, and wireless.

**Figure 31-10 Backplane of the Cisco 4321 Integrated Services Router (ISR)**



## Specialty Devices

Switches and routers make up the backbone of a network. In addition, many networks integrate various specialized network devices.

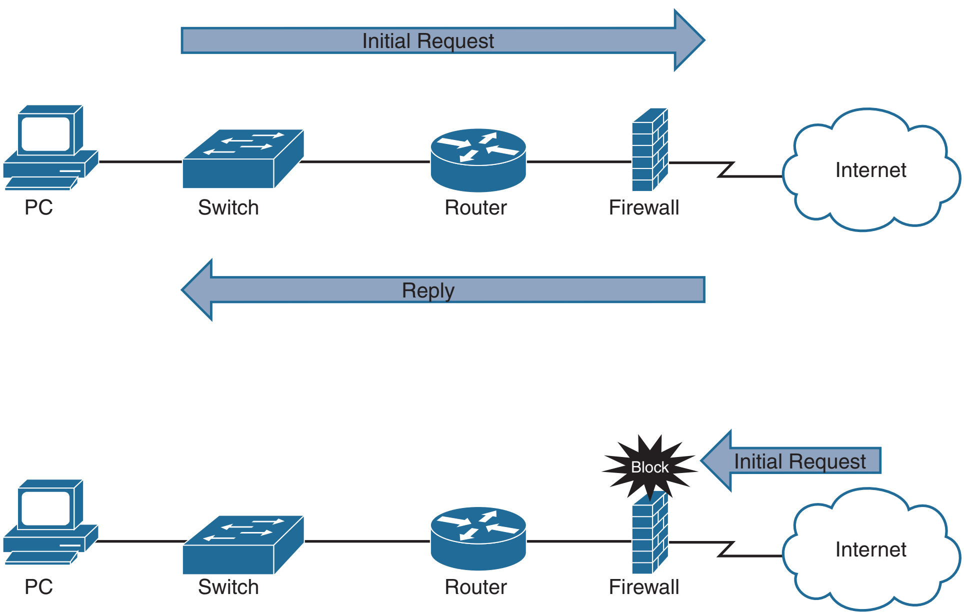
### Firewalls

A firewall is a networking device, either hardware or software based, that controls access to the organization's network. This controlled access is designed to protect data and resources from outside threats.

Organizations implement software firewalls through a network operating system (NOS) such as Linux/UNIX, Windows servers, and macOS servers. The firewall is configured on the server to allow or block certain types of network traffic. Hardware firewalls are often dedicated network devices that can be implemented with little configuration.

Figure 31-11 shows a basic stateful firewall.

**Figure 31-11 The Function of a Firewall**

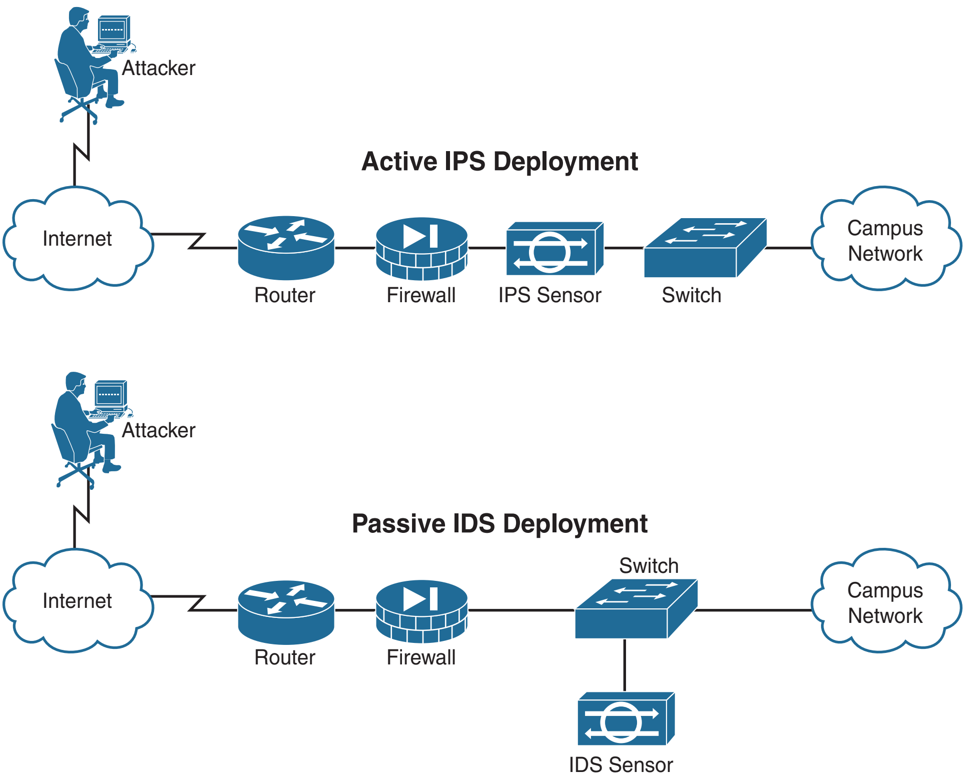



A stateful firewall allows traffic to originate from an inside, trusted network and go out to an untrusted network, such as the Internet. The firewall allows return traffic that comes back from the untrusted network to the trusted network. However, the firewall blocks traffic that originates from an untrusted network.

## IDS and IPS

Both intrusion detection systems (IDS) and intrusion prevention systems (IPS) can recognize network attacks; they differ primarily in their network placement. An IDS device receives a copy of traffic to be analyzed. An IPS device is placed inline with the traffic, as Figure 31-12 shows.

**Figure 31-12 IPS and IDS Comparison**




An IDS is a passive detection system. It can detect the presence of an attack, log the information, and send an alert.

An IPS has the same functionality as an IDS, but in addition, an IPS is an active device that continually scans the network, looking for inappropriate activity. It can shut down any potential threats. The IPS looks for any known signatures of common attacks and automatically tries to prevent those attacks.

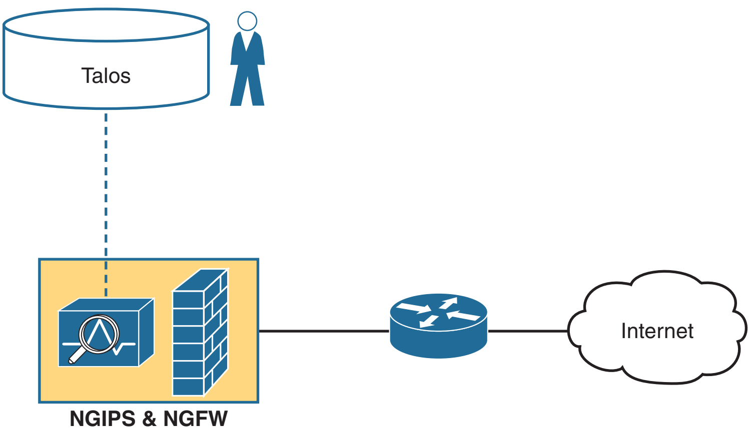
## Next-Generation Firewalls

Although the term *next-generation* in relation to firewalls has been around at least since the earlier 2010s, it can be misleading. Next-generation firewalls (NGFWs) or next-generation IPSs (NGIPSs) are actually what Cisco currently sells as its Cisco Adaptative Security Appliance (ASA) and Firepower product lines. Be sure to visit www.cisco.com/go/firewalls for more information on Cisco's current firewall offerings.

An NGFW typically has the following features:

- **Traditional firewall:** An NGFW performs traditional firewall functions, such as stateful firewall filtering, NAT/PAT, and VPN termination.
- **Application Visibility and Control (AVC):** AVC makes it possible to look deeply into the application layer data to identify the application to defend against attacks that use random port numbers.
- **Advanced Malware Protection (AMP):** AMP can block file transfers that would install malware and save copies of files for later analysis.
- **Uniform resource locator (URL) filtering:** URL filtering examines the URLs in each web request, categorizes the URLs, and either filters or rate limits the traffic based on rules. The Cisco Talos security group monitors and creates reputation scores for each domain known in the Internet, and URL filtering can use those scores in its decisions to categorize, filter, or rate limit.
- **NGIPS:** Cisco's NGFW products can also run their NGIPS feature along with the firewall, as shown in Figure 31-13.

**Figure 31-13 NGFW with NPIPS Module**

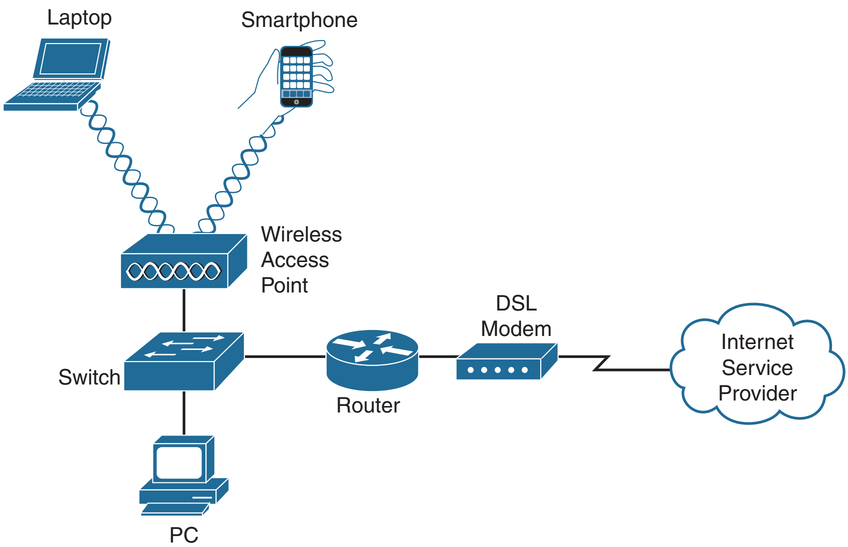



## Access Points and Wireless LAN Controllers

Wireless LANs (WLANs) are commonly used in networks. Users expect to be able to connect seamlessly as they move from location to location within a home, a small business, or an enterprise campus network. To enable this connectivity, network administrators manage a collection of wireless access points (APs) and wireless LAN controllers (WLCs).

In small networks, APs are typically used when a router is already providing Layer 3 services, as in Figure 31-14.

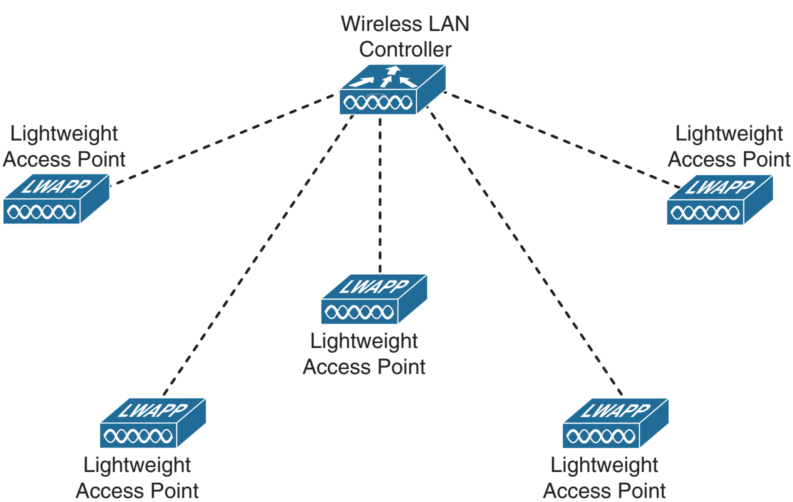
**Figure 31-14 Small Network with an AP**




An AP has an Ethernet port that enables it to be connected to a switch port. In a home or small office network, an AP can be another wireless router with all the Layer 3 services turned off: You simply connect one of the AP's switch ports to one of the switch ports on the wireless router.

APs are also used when the coverage area of an existing WLAN needs to be extended. In larger networks, a wireless LAN controller (WLC) is typically used to manage multiple APs, as in Figure 31-15.

**Figure 31-15 Example of a Wireless LAN Controller Implementation**



WLCs can use the older Lightweight Access Point Protocol (LWAPP) or the more current Control and Provisioning of Wireless Access Points (CAPWAP). With a WLC, VLAN pooling can be used to assign IP addresses to wireless clients from a pool of IP subnets and their associated VLANs.

# Physical Layer

Before any network communications can occur, a wired or wireless physical connection must be established. The type of physical connection depends on the network setup. In larger networks, switches and APs are often two separate dedicated devices. In a very small business (three or four employees) or home network, wireless and wired connections are combined into one device and include a broadband method of connecting to the Internet. These wireless broadband routers offer a switching component with multiple ports and an AP, which allows wireless devices to connect as well. Figure 31-16 shows the backplane of a Cisco WRP500 Wireless Broadband Router.

**Figure 31-16 Cisco RV160W Wireless-AC VPN Router**


## Network Media Forms and Standards

Three basic forms of network media exist:

- **Copper cable:** The signals are patterns of electrical pulses.
- **Fiber-optic cable:** The signals are patterns of light.
- **Wireless:** The signals are patterns of microwave transmissions.

Messages are encoded and then placed onto the media. Encoding is the process of converting data into patterns of electrical, light, or electromagnetic energy so that it can be carried on the media.

Table 31-6 summarizes the three most common networking media in use today.

**Table 31-6 Networking Media**

| Media | Physical Components | Frame Encoding Technique | Signaling Methods |
|---|---|---|---|
| Copper cable | UTP<br>Coaxial<br>Connectors<br>NICs<br>Ports<br>Interfaces | Manchester encoding<br>Nonreturn to zero (NRZ) techniques<br>4B/5B codes used with Multi-Level Transition Level 3 (MLT-3) signaling<br>8B/10B<br>PAM5 | Changes in the electromagnetic field.<br>Intensity of the electromagnetic field.<br>Phase of the electromagnetic wave. |
| Fiber-optic cable | Single-mode fiber<br>Multimode fiber<br>Connectors<br>NICs<br>Interfaces<br>Lasers and LEDs<br>Photoreceptors | Pulses of light<br>Wavelength multiplexing using different colors | A pulse equals 1.<br>No pulse is 0. |
| Wireless | Access points<br>NICs<br>Radio<br>Antennas | Direct Sequence Spread Spectrum (DSSS)<br>Orthogonal Frequency Division Multiplexing (OFDM) | Radio waves. |

Each media type has advantages and disadvantages. When choosing the media, consider each of the following:

- **Cable length:** Does the cable need to span a room or run from building to building?
- **Cost:** Does the budget allow for using a more expensive media type?
- **Bandwidth:** Does the technology used with the media provide adequate bandwidth?
- **Ease of installation:** Does the implementation team have the capability to install the cable, or is a vendor required?
- **Susceptible to EMI/RFI:** Will the local environment interfere with the signal?

Table 31-7 summarizes the media standards for LAN cabling.

**Table 31-7 Media Standard, Cable Length, and Bandwidth**

| Ethernet Type | Bandwidth | Cable Type | Maximum Distance |
|---|---|---|---|
| 10BASE-T | 10 Mbps | Cat3/Cat5 UTP | 100 m |
| 100BASE-TX | 100 Mbps | Cat5 UTP | 100 m |
| 100BASE-TX | 200 Mbps | Cat5 UTP | 100 m |
| 100BASE-FX | 100 Mbps | Multimode fiber | 400 m |
| 100BASE-FX | 200 Mbps | Multimode fiber | 2 km |
| 1000BASE-T | 1 Gbps | Cat5e UTP | 100 m |
| 1000BASE-TX | 1 Gbps | Cat6 UTP | 100 m |
| 1000BASE-SX | 1 Gbps | Multimode fiber | 550 m |
| 1000BASE-LX | 1 Gbps | Single-mode fiber | 2 km |
| 10GBASE-T | 10 Gbps | Cat6a/Cat7 UTP | 100 m |
| 10GBASE-SX4 | 10 Gbps | Multimode fiber | 550 m |
| 10GBASE-LX4 | 10 Gbps | Single-mode fiber | 2 km |

# LAN Device Connection Guidelines

End devices are pieces of equipment that are either the original source or the final destination of a message. Intermediary devices connect end devices to the network to assist in getting a message from the source end device to the destination end device.

Connecting devices in a LAN is usually done with unshielded twisted pair (UTP) cabling. Although many newer devices have an automatic crossover feature that enables you to connect either a straight-through or a crossover cable, you still need to know the following basic rules:

Use straight-through cables for the following connections:

- Switch to router Ethernet port
- Computer to switch
- Computer to hub

Use crossover cables for the following connections:

- Switch to switch
- Switch to hub
- Hub to hub
- Router to router (Ethernet ports)
- Computer to computer
- Computer to router Ethernet port

# LANs and WANs

A local-area network (LAN) is a network of computers and other components located relatively close together in a limited area. LANs can vary widely in size, from one computer connected to a router in a home office, to hundreds of computers in a corporate office. However, in general, a LAN spans a limited geographic area. The fundamental components of a LAN include the following:

- Computers
- Interconnections (NICs and the media)
- Networking devices (hubs, switches, and routers)
- Protocols (Ethernet, IP, ARP, DHCP, DNS, and so on)

A wide-area network (WAN) generally connects LANs that are geographically separated. A collection of LANs connected by one or more WANs is called an *internetwork*; thus, we have the Internet. The term *intranet* is often used to refer to a privately owned connection of LANs and WANs.

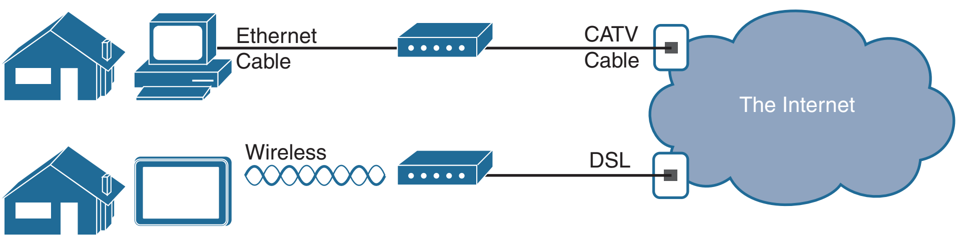
Depending on the type of service, connecting to the WAN normally works in one of the following ways:

- 60-pin serial connection to a CSU/DSU (legacy)
- RJ-45 T1 controller connection to a CSU/DSU (legacy)
- RJ-11 connection to a dialup or DSL modem
- Cable coaxial connection to a cable modem
- Fiber Ethernet connection to service provider's switch

# Small Office/Home Office (SOHO)

With the growing number of remote workers, enterprises have an increasing need for secure, reliable, and cost-effective ways to connect people working in small offices or home offices (SOHO) or other remote locations to resources on corporate sites. For SOHO workers, this is typically done through a cable or DSL connection, as shown in Figure 31-17.

**Figure 31-17 SOHO Connections to the Internet**



Remote connection technologies to support teleworkers include the following:

- Traditional private WAN technologies, including Frame Relay, ATM, and leased lines, although these technologies are now considered legacy
- Remote secure virtual private network (VPN) access through a broadband connection over the public Internet

Components needed for teleworker connectivity include the following:

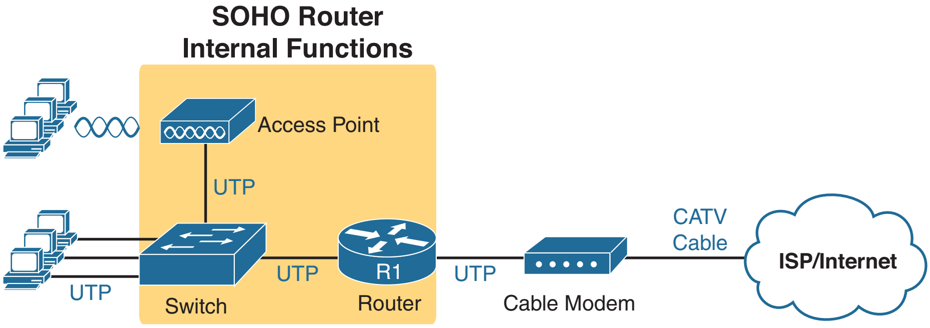
- **Home office components:** Computer, broadband access (cable or DSL), and a VPN router or VPN client software installed on the computer
- **Corporate components:** VPN-capable routers, VPN concentrators, multifunction security appliances, authentication, and central management devices for resilient aggregation and termination of the VPN connections

## SOHO Routers

The gateway to the Internet for a SOHO is typically an integrated multifunction routers. SOHO routers have the following features:

- Use the Internet and VPN technology for their WAN connections to send data back and forth to the rest of the enterprise.
- Use a multifunction device that handles routing, LAN switching, VPN, wireless, and other features, as shown in Figure 31-18.

**Figure 31-18 Internal Functions of a SOHO Router**



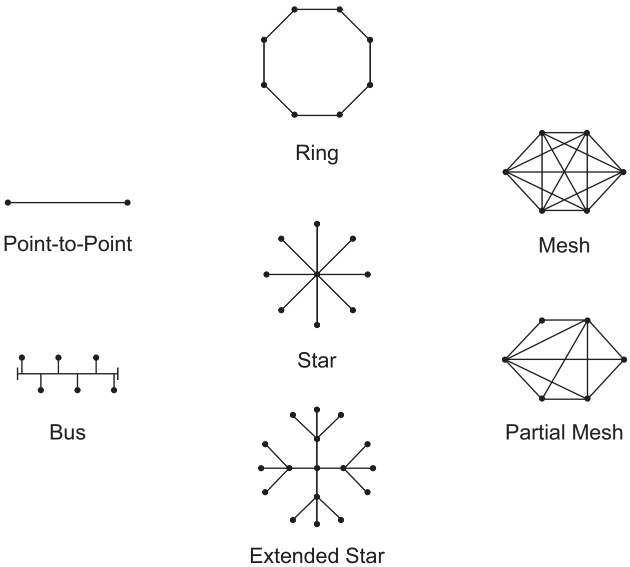

In reality, the access point and switch are integrated into the router.

# Physical and Logical Topologies

Network diagrams are usually referred to as *topologies*. A topology graphically displays the interconnection methods used between devices.

Physical topologies refer to the physical layout of devices and how they are cabled. Seven basic physical topologies exist (see Figure 31-19).

**Figure 31-19 Physical Topologies**




Logical topologies refer to the way that a signal travels from one point on the network to another and are largely determined by the access method—deterministic or nondeterministic. Ethernet is a nondeterministic access method. Logically, Ethernet operates as a bus topology. However, Ethernet networks are almost always physically designed as star or extended star topologies.

Other access methods use a deterministic access method. Token Ring and Fiber Distributed Data Interface (FDDI) both logically operate as ring topologies, passing data from one station to the next. Although these networks can be designed as physical rings, like Ethernet, they are often designed as star or extended star topologies. Logically, however, they operate like ring topologies.

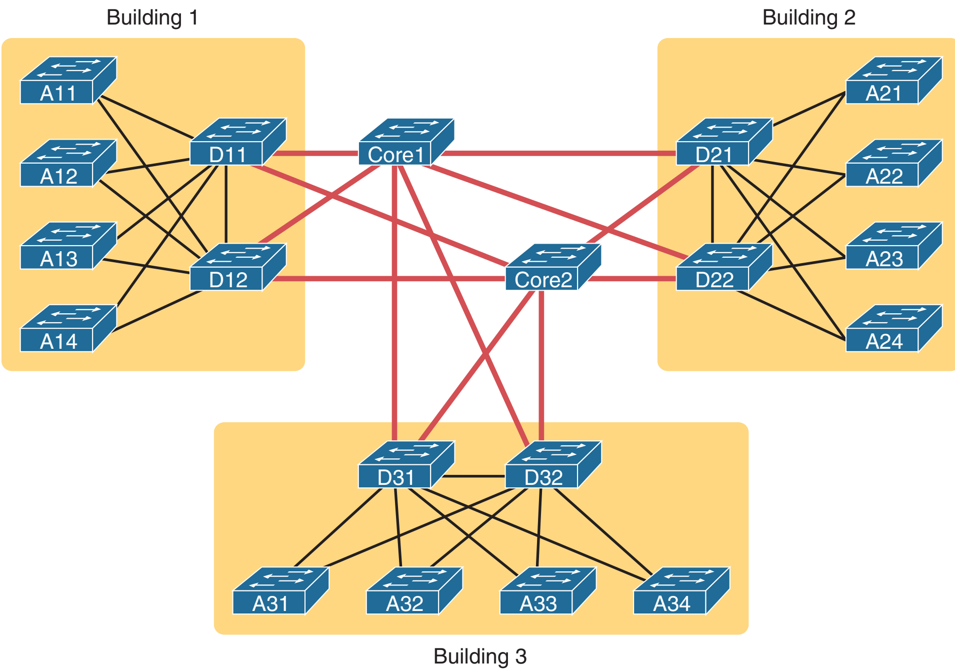
# Hierarchical Campus Designs

Hierarchical campus design involves dividing the network into discrete layers. Each layer provides specific functions that define its role within the overall network. By separating the various functions that exist on a network, the network design becomes modular, which facilitates scalability and performance. The hierarchical design model is divided into three layers:

- **Access layer:** Provides local and remote user access
- **Distribution layer:** Controls the flow of data between the access and core layers
- **Core layer:** Acts as the high-speed redundant backbone

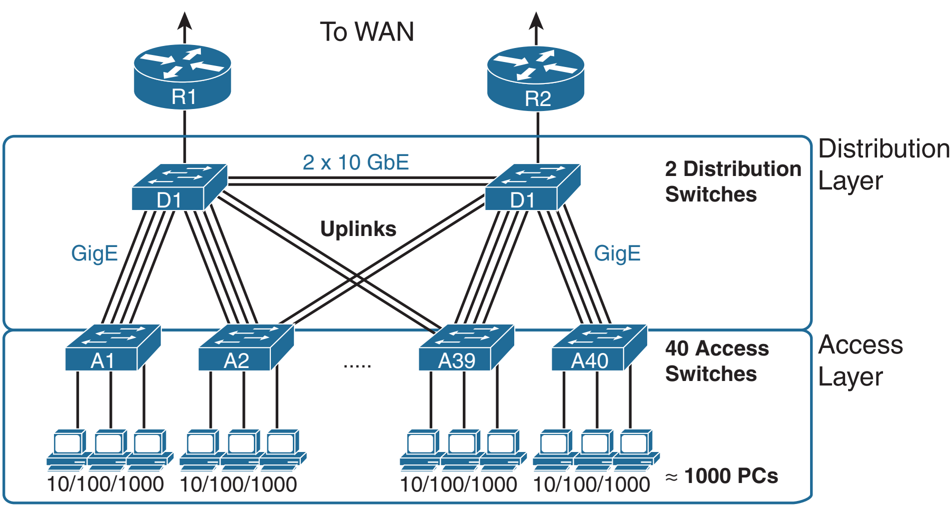
Figure 31-20 shows an example of the three-tier hierarchical campus network design.

**Figure 31-20 Three-Tier Campus Design**




For smaller networks, the core is often collapsed into the distribution layer for a two-tier design, as in Figure 31-21.

**Figure 31-21 Two-Tier Campus Design**



A two-tier design solves two major design needs:

- Provides a place to connect end-user devices (the access layer, with access switches)
- Connects the switches with a reasonable number of cables and switch ports by connecting all 40 access switches to two distribution switches

For very small networks and home networks, all three tiers can be seen in one device, such as the wireless broadband router shown earlier in Figure 31-16.

# Study Resources

For today's exam topics, refer to the following resources for more study.

| Resource | Module or Chapter |
|---|---|
| Introduction to Networks v7 | 1 |
| | 3 |
| | 4 |
| CCNA 200-301 Official Cert Guide, Volume 1 | 1 |
| | 2 |
| | 3 |
| | 5 |
| | 15 |
| | 26 |
| CCNA 200-301 Official Cert Guide, Volume 2 | 5 |
| | 13 |
| Portable Command Guide | 6 |

*This page intentionally left blank*

# Day 30

# Ethernet Switching

## CCNA 200-301 Exam Topics

- Explain the role and function of network components
- Describe switching concepts

## Key Topics

Today we review the concepts behind Ethernet switching, including the history of switching development, how switching actually works, and the variety of switch features. We also review the details of Ethernet operation.

## Evolution to Switching

Today's LANs almost exclusively use switches to interconnect end devices; however, this was not always the case. Initially, devices were connected to a physical bus, a long run of coaxial backbone cabling. With the introduction of 10BASE-T and UTP cabling, the hub gained popularity as a cheaper, easier way to connect devices. But even 10BASE-T with hubs had the following limitations:

- A frame sent from one device can collide with a frame sent by another device attached to that LAN segment. Devices were in the same collision domain sharing the bandwidth.
- Broadcasts sent by one device were heard and processed by all other devices on the LAN. Devices were in the same broadcast domain. Much like to hubs, switches forward broadcast frames out all ports except for the incoming port.

Ethernet bridges were soon developed to solve some of the inherent problems in a shared LAN. A bridge basically segmented a LAN into two collision domains, which reduced the number of collisions in a LAN segment. This increased the performance of the network by decreasing unnecessary traffic from another segment.

When switches arrived on the scene, these devices provided the same benefits of bridges, as well as the following:

- A larger number of interfaces to break up the collision domain into more segments
- Hardware-based switching instead of using software to make the decision

In a LAN where all nodes are connected directly to the switch, the throughput of the network increases dramatically. With each computer connected to a separate port on the switch, each is in a separate collision domain and has its own dedicated segment. There are three primary reasons for this increase:

- Dedicated bandwidth to each port
- Collision-free environment
- Full-duplex operation

# Switching Logic

Ethernet switches selectively forward individual frames from a receiving port to the port where the destination node is connected. During this instant, the switch creates a full-bandwidth, logical, point-to-point connection between the two nodes.

Switches create this logical connection based on the source and destination Media Access Control (MAC) addresses in the Ethernet header. Specifically, the primary job of a LAN switch is to receive Ethernet frames and then make a decision to either forward the frame or ignore the frame. To accomplish this, the switch performs three actions:

**Step 1.** Decides when to forward a frame or when to filter (not forward) a frame, based on the destination MAC address

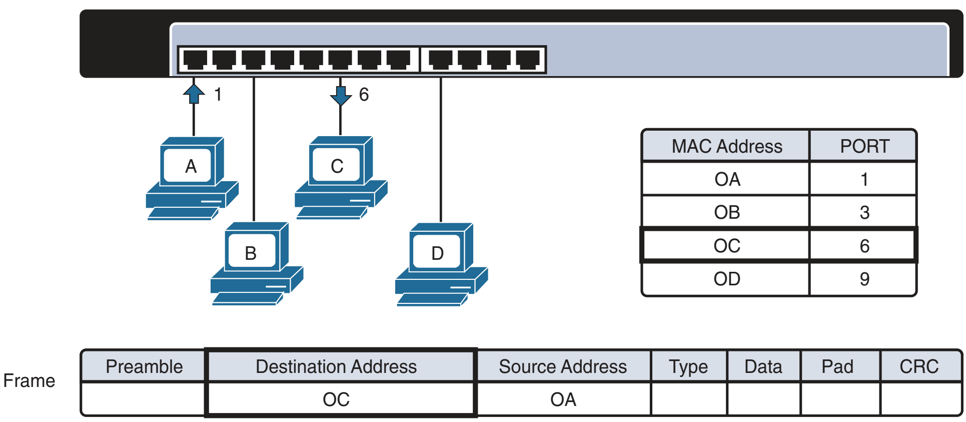
**Step 2.** Learns MAC addresses by examining the source MAC address of each frame the switch receives

**Step 3.** Creates a (Layer 2) loop-free environment with other switches by using Spanning Tree Protocol (STP)

To make the decision to forward or filter, the switch uses a dynamically built MAC address table stored in RAM. By comparing the frame's destination MAC address with the fields in the table, the switch decides how to forward and/or filter the frame.

For example, in Figure 30-1, the switch receives a frame from Host A with the destination MAC address OC. The switch looks in its MAC table, finds an entry for the MAC address, and forwards the frame out port 6. The switch also filters the frame by not forwarding it out any other port, including the port on which the frame was received.

**Figure 30-1 Switch Forwarding Based on MAC Address**




In addition to forwarding and filtering frames, the switch refreshes the timestamp for the source MAC address of the frame. In Figure 30-1, the MAC address for Host A, OA, is already in the MAC table, so the switch refreshes the entry. Entries that are not refreshed eventually are removed (after the default 300 seconds in Cisco IOS).

Continuing the example in Figure 30-1, assume that another device, Host E, is attached to port 10. Host B then sends a frame to the new Host E. The switch does not yet know where Host E is located, so it forwards the frame out all active ports (in a process known as flooding) except for the port on which the frame was received. The new Host E receives the frame. When it replies to Host B, the switch learns Host E's MAC address and port for the first time and stores it in the MAC address table. Subsequent frames destined for Host E then are sent out only port 10.

Finally, LAN switches must have a method for creating a loop-free path for frames to take within the LAN. STP provides loop prevention in Ethernet networks where redundant physical links exist.

# Collision and Broadcast Domains

A collision domain is the set of LAN interfaces whose frames could collide with each other. All shared media environments, such as those created by using hubs, are collision domains. When one host is attached to a switch port, the switch creates a dedicated connection, thereby eliminating the potential for a collision. Switches reduce collisions and improve bandwidth use on network segments because they provide full-duplex, dedicated bandwidth to each network segment.

Out of the box, however, a switch cannot provide relief from broadcast traffic. A collection of connected switches forms one large broadcast domain. If a frame with the destination address FFFF.FFFF.FFFF crosses a switch port, that switch must flood the frame out all other active ports. Each attached device must then process the broadcast frame at least up to the network layer. Routers and VLANs are used to segment broadcast domains. Day 26, "VLAN and Trunking Concepts and Configurations," reviews the use of VLANs to segment broadcast domains.

# Frame Forwarding

Switches operate in several ways to forward frames. They can differ in forwarding methods, port speeds, memory buffering, and the OSI layers used to make the forwarding decision. The following sections discuss these concepts in greater detail.

## Switch Forwarding Methods

Switches use one of the following forwarding methods to switch data between network ports:

- **Store-and-forward switching:** The switch stores received frames in its buffers, analyzes each frame for information about the destination, and evaluates the data integrity using the cyclic redundancy check (CRC) in the frame trailer. The entire frame is stored, and the CRC is calculated before any of the frame is forwarded. If the CRC passes, the frame is forwarded to the destination.

- **Cut-through switching:** The switch buffers just enough of the frame to read the destination MAC address so that it can determine which port to forward the data to. When the switch determines a match between the destination MAC address and an entry in the MAC address table, the frame is forwarded out the appropriate port(s). This happens as the rest of the initial frame is still being received. The switch does not perform any error checking on the frame.

- **Fragment-free mode:** The switch waits for the collision window (64 bytes) to pass before forwarding the frame. This means that each frame is checked into the data field to make sure that no fragmentation has occurred. Fragment-free mode provides better error checking than cut-through, with practically no increase in latency.

## Symmetric and Asymmetric Switching

Symmetric switching provides switched connections between ports with the same bandwidth, such as all 100-Mbps ports or all 1000-Mbps ports. An asymmetric LAN switch provides switched connections between ports of unlike bandwidth, such as a combination of 10-Mbps, 100-Mbps, and 1000-Mbps ports.

## Memory Buffering

Switches store frames for a brief time in a memory buffer. Two methods of memory buffering exist:

- **Port-based memory:** Frames are stored in queues that are linked to specific incoming ports.
- **Shared memory:** Frames are deposited into a common memory buffer that all ports on the switch share.

## Layer 2 and Layer 3 Switching

A Layer 2 LAN switch performs switching and filtering based only on MAC addresses. A Layer 2 switch is completely transparent to network protocols and user applications. A Layer 3 switch functions similarly to a Layer 2 switch. But instead of using only the Layer 2 MAC address information for forwarding decisions, a Layer 3 switch can also use IP address information. Layer 3 switches are also capable of performing Layer 3 routing functions, reducing the need for dedicated routers on a LAN. Because Layer 3 switches have specialized switching hardware, they can typically route data as quickly as they can switch data.

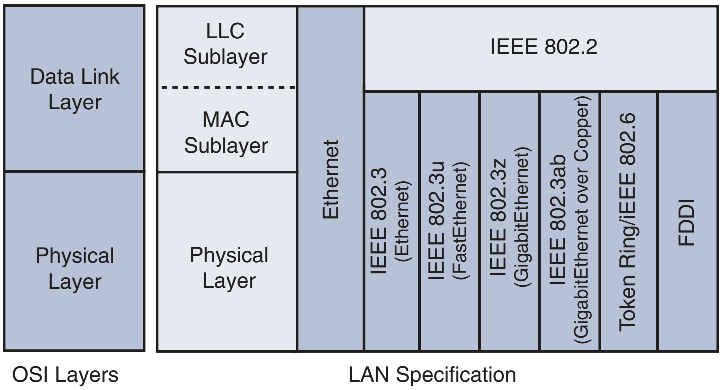
# Ethernet Overview

802.3 is the IEEE standard for Ethernet, and the two terms are commonly used interchangeably. The terms *Ethernet* and *802.3* both refer to a family of standards that together define the physical and data link layers of the definitive LAN technology. Figure 30-2 shows a comparison of Ethernet standards to the OSI model.

Ethernet separates the functions of the data link layer into two distinct sublayers:

- **Logical Link Control (LLC) sublayer:** Defined in the 802.2 standard
- **Media Access Control (MAC) sublayer:** Defined in the 802.3 standard

**Figure 30-2 Ethernet Standards and the OSI Model**




The LLC sublayer handles communication between the network layer and the MAC sublayer. In general, LLC provides a way to identify the protocol that is passed from the data link layer to the network layer. In this way, the fields of the MAC sublayer are not populated with protocol type information, as was the case in earlier Ethernet implementations.

The MAC sublayer has two primary responsibilities:

- **Data encapsulation:** Included here is frame assembly before transmission, frame parsing upon reception of a frame, data link layer MAC addressing, and error detection.
- **Media Access Control:** Because Ethernet is a shared medium and all devices can transmit at any time, media access is controlled by a method called Carrier Sense Multiple Access/Collision Detect (CSMA/CD) when operating in half-duplex mode.

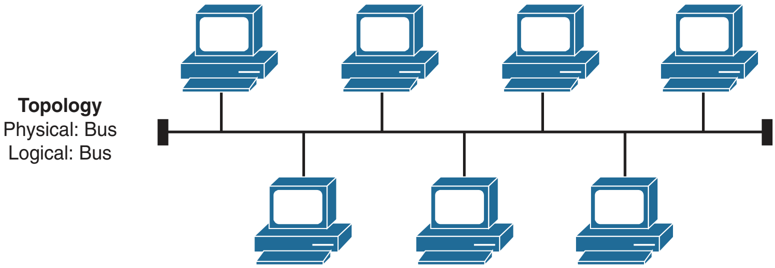
At the physical layer, Ethernet specifies and implements encoding and decoding schemes that enable frame bits to be carried as signals across both unshielded twisted pair (UTP) copper cables and optical fiber cables. In early implementations, Ethernet used coaxial cabling.

# Legacy Ethernet Technologies

Ethernet is best understood by first considering the two early Ethernet specifications, 10BASE-5 and 10BASE-2. With these two specifications, the network engineer installs a series of coaxial cables connecting each device on the Ethernet network, as in Figure 30-3.

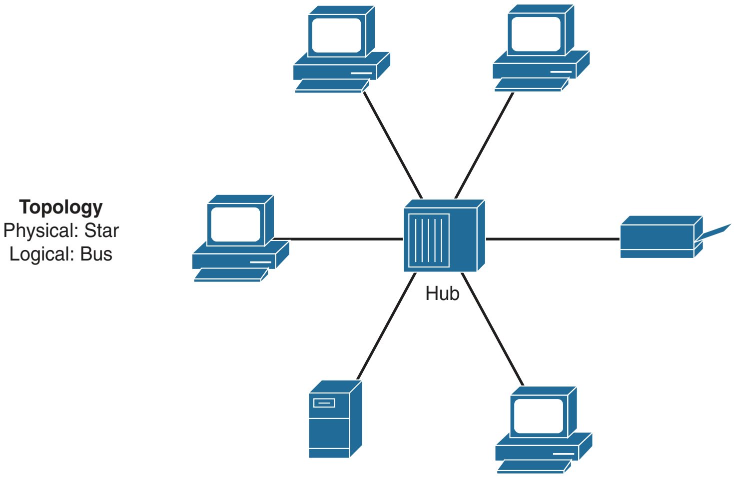
The series of cables creates an electrical circuit, called a *bus*, that is shared among all devices on the Ethernet. When a computer wants to send some bits to another computer on the bus, it sends an electrical signal, and the electricity propagates to all devices on the Ethernet.

**Figure 30-3 Ethernet Physical and Logical Bus Topology**




With the change of media to UTP and the introduction of the first hubs, Ethernet physical topologies migrated to a star, as shown in Figure 30-4.

**Figure 30-4 Ethernet Physical Star and Logical Bus Topology**




Regardless of the change in the physical topology from a bus to a star, hubs logically operate similarly to a traditional bus topology and require the use of CSMA/CD.

## CSMA/CD

Because Ethernet is a shared medium in which every device has the right to send at any time, it also defines a specification to ensure that only one device sends traffic at a time. The CSMA/CD algorithm defines how the Ethernet logical bus is accessed.

CSMA/CD logic helps prevent collisions and also defines how to act when a collision does occur. The CSMA/CD algorithm works like this:

**Step 1.** A device with a frame to send listens until the Ethernet is not busy.

**Step 2.** When the Ethernet is not busy, the sender(s) begin(s) sending the frame.

**Step 3.** The sender(s) listen(s) to make sure that no collision occurs.

**Step 4.** If a collision occurs, the devices that were sending a frame each send a jamming signal to ensure that all stations recognize the collision.

**Step 5.** When the jamming is complete, each sender randomizes a timer and waits until the timer expires before trying to resend the collided frame.

**Step 6.** When each random timer expires, the process starts again from the beginning.

When CSMA/CD is in effect, a device's network interface card (NIC) operates in half-duplex mode, either sending or receiving frames. CSMA/CD is disabled when a NIC autodetects that it can operate in—or is manually configured to operate in—full-duplex mode. In full-duplex mode, a NIC can send and receive simultaneously.

## Legacy Ethernet Summary

LAN hubs occasionally appear, but switches generally are used instead of hubs. Keep in mind the following key points about the history of Ethernet:

- The original Ethernet LANs created an electrical bus to which all devices connected.
- 10BASE-2 and 10BASE-5 repeaters extended the length of LANs by cleaning up the electrical signal and repeating it (a Layer 1 function) but without interpreting the meaning of the electrical signal.
- Hubs are repeaters that provide a centralized connection point for UTP cabling—but they still create a single electrical bus that the various devices share, just as with 10BASE-5 and 10BASE-2.
- Because collisions can occur in any of these cases, Ethernet defines the CSMA/CD algorithm, which tells devices how to both avoid collisions and take action when collisions do occur.

# Current Ethernet Technologies

Refer to Table 30-1 and notice the different 802.3 standards. Each new physical layer standard from the IEEE requires many differences at the physical layer. However, each of these physical layer standards uses the same 802.3 header, and each uses the upper LLC sublayer as well. Table 30-1 lists today's most commonly used IEEE Ethernet physical layer standards.

**Table 30-1 Today's Most Common Types of Ethernet**

| Common Name | Speed | Alternative Name | Name of IEEE Standard | Cable Type, Maximum Length |
|---|---|---|---|---|
| Ethernet | 10 Mbps | 10BASE-T | 802.3 | Copper, 100 m |
| Fast Ethernet | 100 Mbps | 100BASE-TX | 802.3u | Copper, 100 m |
| Gigabit Ethernet | 1000 Mbps | 1000BASE-LX | 802.3z | Fiber, 550 m |
| Gigabit Ethernet | 1000 Mbps | 1000BASE-T | 802.3ab | Copper, 100 m |
| 10GigE (Gigabit Ethernet) | 10 Gbps | 10GBASE-T | 802.3an | Copper, 100 m |
| 10GigE (Gigabit Ethernet) | 10 Gbps | 10GBASE-S | 802.3ae | Fiber, 400 m |

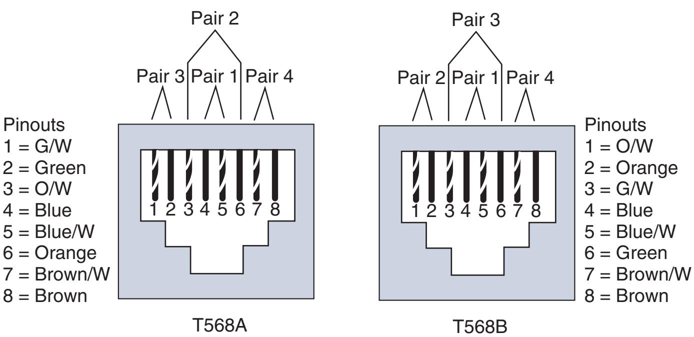
# UTP Cabling

The three most common Ethernet standards used today—10BASE-T (Ethernet), 100BASE-TX (Fast Ethernet, or FE), and 1000BASE-T (Gigabit Ethernet, or GE)—use UTP cabling. Some key differences exist, particularly with the number of wire pairs needed in each case and the type (category) of cabling.

The UTP cabling in popular Ethernet standards includes either two or four pairs of wires. The cable ends typically use an RJ-45 connector. The RJ-45 connector has eight specific physical locations into which the eight wires in the cable can be inserted; these are called pin positions or, simply, pins.

The Telecommunications Industry Association (TIA) and the Electronics Industry Alliance (EIA) define standards for UTP cabling, with color coding for wires and standard pinouts on the cables. Figure 30-5 shows two TIA/EIA pinout standards, with the color coding and pair numbers listed.

**Figure 30-5 TIA/EIA Standard Ethernet Cabling Pinouts**




For the exam, you should be well prepared to choose which type of cable (straight-through or crossover) is needed in each part of the network. In short, devices on opposite ends of a cable that use the same pair of pins to transmit need a crossover cable. Devices that use an opposite pair of pins to transmit need a straight-through cable. Table 30-2 lists typical devices and the pin pairs they use, assuming that they use 10BASE-T and 100BASE-TX.

**Table 30-2 10BASE-T and 100BASE-TX Pin Pairs Used**

| Devices That Transmit on 1,2 and Receive on 3,6 | Devices That Transmit on 3,6 and Receive on 1,2 |
|---|---|
| PC NICs | Hubs |
| Routers | Switches |
| Wireless access points (Ethernet interfaces) | — |
| Networked printers (printers that connect directly to the LAN) | — |

1000BASE-T requires four wire pairs because Gigabit Ethernet transmits and receives on each of the four wire pairs simultaneously.

However, Gigabit Ethernet does have a concept of straight-through and crossover cables, with a minor difference in the crossover cables. The pinouts for a straight-through cable are the same—pin 1 to pin 1, pin 2 to pin 2, and so on.

A crossover cable has the 568A standard on one end and the 568B standard on the other end. This crosses the pairs at pins 1,2 and 3,6.

# Benefits of Using Switches

A collision domain is a set of devices whose frames may collide. All devices on a 10BASE-2, 10BASE-5, or other network using a hub risk collisions between the frames that they send. Thus, devices on one of these types of Ethernet networks are in the same collision domain and use CSMA/CD to detect and resolve collisions.

LAN switches significantly reduce, or even eliminate, the number of collisions on a LAN. Unlike a hub, a switch does not create a single shared bus. Instead, a switch does the following:

- It interprets the bits in the received frame so that it can typically send the frame out the one required port instead of out all other ports.
- If a switch needs to forward multiple frames out the same port, the switch buffers the frames in memory, sending one at a time and thereby avoiding collisions.

In addition, switches with only one device cabled to each port of the switch allow the use of full-duplex operation. Full-duplex operation means that the NIC can send and receive concurrently, effectively doubling the bandwidth of a 100-Mbps link to 200 Mbps—100 Mbps for sending and 100 Mbps for receiving.

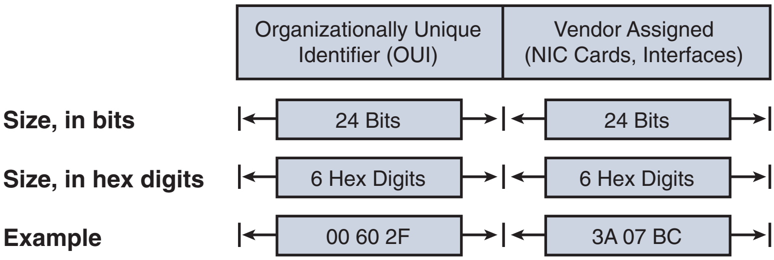
These seemingly simple switch features provide significant performance improvements compared with using hubs. In particular, consider these points:

- If only one device is cabled to each port of a switch, no collisions can occur.
- Devices connected to one switch port do not share their bandwidth with devices connected to another switch port. Each has its own separate bandwidth, meaning that a switch with 100-Mbps ports has 100 Mbps of bandwidth per port.

# Ethernet Addressing

The IEEE defines the format and assignment of LAN addresses. To ensure a unique MAC address, the first half of the address identifies the manufacturer of the card. This code is called the organizationally unique identifier (OUI). Each manufacturer assigns a MAC address with its own OUI as the first half of the address. The second half of the address is assigned by the manufacturer and is never used on another card or network interface with the same OUI. Figure 30-6 shows the structure of a unicast Ethernet address.

**Figure 30-6 Structure of a Unicast Ethernet Address**




Ethernet also has group addresses, which identify more than one NIC or network interface. The IEEE defines two general categories of group addresses for Ethernet:

- **Broadcast addresses:** A broadcast address implies that all devices on the LAN should process the frame and has the value FFFF.FFFF.FFFF.
- **Multicast addresses:** A multicast address allows a subset of devices on a LAN to communicate. When IP multicasts over an Ethernet network, the multicast MAC addresses that IP uses follow this format: 0100.5exx.xxxx. The xx.xxxx portion is divided between IPv4 multicast (00:0000–7F:FFFF) and MPLS multicast (80:0000–8F:FFFF). Multiprotocol Label Switching (MPLS) is a CCNP topic.

# Ethernet Framing

The physical layer helps you get a string of bits from one device to another. The framing of the bits allows the receiving device to interpret the bits. The term *framing* refers to the definition of the fields assumed to be in the data that is received. Framing defines the meaning of the bits transmitted and received over a network.

The framing used for Ethernet has changed a couple times over the years. Figure 30-7 shows each iteration of Ethernet, with the current version shown at the bottom.

**Figure 30-7 Ethernet Frame Formats**

**DIX**

| **Preamble** | Destination | Source | **Type** | Data and Pad | FCS |
|---|---|---|---|---|---|
| **8** | 6 | 6 | **2** | 46 – 1500 | 4 |

**IEEE 802.3 (Original)**

| **Preamble** | **SFD** | Destination | Source | **Length** | Data and Pad | FCS |
|---|---|---|---|---|---|---|
| **7** | **1** | 6 | 6 | **2** | 46 – 1500 | 4 |

**IEEE 802.3 (Revised 1997)**

| | **Preamble** | **SFD** | Destination | Source | **Length/** | Data and Pad | FCS |
|---|---|---|---|---|---|---|---|
| Bytes | **7** | **1** | 6 | 6 | **Type 2** | 46 – 1500 | 4 |

Table 30-3 further explains the fields in the last version shown in Figure 30-7.

**Table 30-3 IEEE 802.3 Ethernet Field Descriptions**

| Field | Field Length, in Bytes | Description |
|---|---|---|
| Preamble | 7 | Synchronization |
| Start Frame Delimiter (SFD) | 1 | Signifies that the next byte begins the Destination MAC field |
| Destination MAC Address | 6 | Identifies the intended recipient of this frame |
| Source MAC Address | 6 | Identifies the sender of this frame |
| Length | 2 | Defines the length of the data field of the frame (either length or type is present, but not both) |
| Type | 2 | Defines the type of protocol listed inside the frame (either length or type is present, but not both) |
| Data and Pad | 46–1500 | Holds data from a higher layer, typically a Layer 3 PDU (generic) and often an IP packet |
| Frame Check Sequence (FCS) | 4 | Provides a method for the receiving NIC to determine whether the frame experienced transmission errors |

# The Role of the Physical Layer

We have already discussed the most popular cabling used in LANs: UTP. To fully understand the operation of the network, you should know some additional basic concepts of the physical layer.

The OSI physical layer accepts a complete frame from the data link layer and encodes it as a series of signals that are transmitted onto the local media.

The delivery of frames across the local media requires the following physical layer elements:

- The physical media and associated connectors
- A representation of bits on the media
- Encoding of data and control information
- Transmitter and receiver circuitry on the network devices

Data is represented on three basic forms of network media:

- Copper cable
- Fiber
- Wireless (IEEE 802.11)

Bits are represented on the medium by changing one or more of the following characteristics of a signal:

- Amplitude
- Frequency
- Phase

The nature of the actual signals representing the bits on the media depends on the signaling method in use. Some methods use one attribute of a signal to represent a single 0 and use another attribute of a signal to represent a single 1. The actual signaling method and its detailed operation are not important to your CCNA exam preparation.

## Study Resources

For today's exam topics, refer to the following resources for more study.

| Resource | Module or Chapter |
|---|---|
| Introduction to Networks v7 | 4 |
| | 6 |
| | 7 |
| CCNA 200-301 Official Cert Guide, Volume 1 | 5 |
| | 8 |

# Day 29

# Switch Configuration Basics

## CCNA 200-301 Exam Topics

- Identify interface and cable issues (collisions, errors, mismatched duplex and/or speed)
- Configure and verify IPv4 addressing and subnetting

## Key Topics

Today we review Cisco IOS basics and the commands necessary to perform a basic initial configuration of a switch. Although not explicitly called out in the exam topics, you can expect to see questions that assume you have this skill. We review verification techniques such as the **ping**, **traceroute**, and **show** commands. And we review interface and cable issues.

## Accessing and Navigating the Cisco IOS

By now, you are very familiar with connecting to Cisco devices and configuring them using the command-line interface (CLI). Here we quickly review methods for accessing and navigating the CLI.

### Connecting to Cisco Devices

You can access a device directly or from a remote location. Figure 29-1 shows the many ways you can connect to Cisco devices.

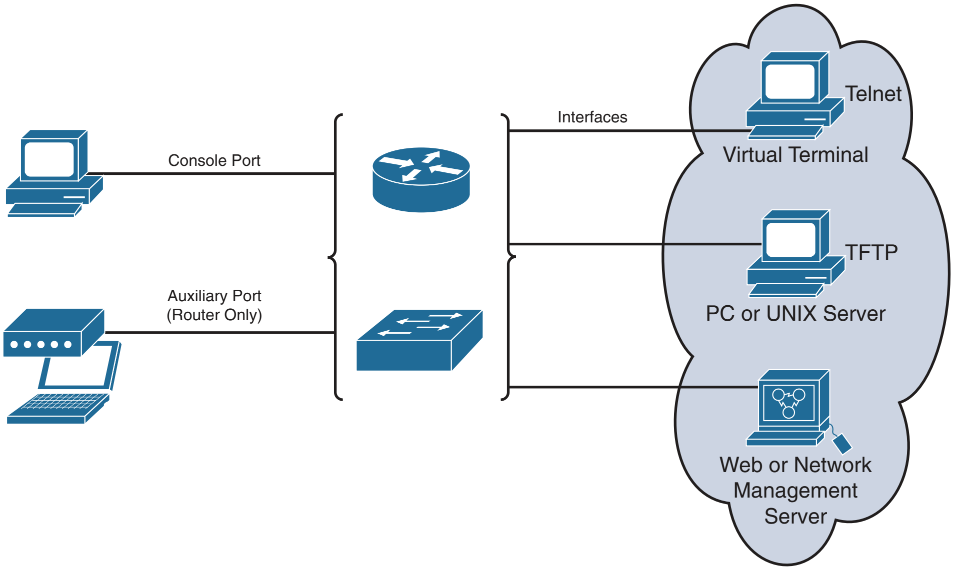
The two ways to configure Cisco devices are as follows:

- **Console terminal:** Use an RJ-45–to–RJ-45 rollover cable and a computer with the terminal communications software (such as HyperTerminal or Tera Term) to establish a direct connection. Optionally, you can connect a mini-USB cable to the mini-USB console port, if available.
- **Remote terminal:** Use an external modem connected to the auxiliary port (routers only) to remotely configure the device.

After a device is configured, you can access it using three additional methods:

- Establish a terminal (vty) session using Telnet.
- Configure the device through the current connection (console or auxiliary) or download a previously written startup config file from a Trivial File Transfer Protocol (TFTP) server on the network.
- Download a configuration file using a network management software application.

**Figure 29-1 Sources for Cisco Device Configuration**




## CLI EXEC Sessions

Cisco IOS separates the EXEC session into two basic access levels:

- **User EXEC mode:** Access to only a limited number of basic monitoring and troubleshooting commands, such as **show** and **ping**
- **Privileged EXEC mode:** Full access to all device commands, including configuration and management

## Using the Help Facility

Cisco IOS has extensive command-line input help facilities, including context-sensitive help. Two types of help are available:

- **Word help:** Enter a character sequence of an incomplete command immediately followed by a question mark (for example, **sh?**) to get a list of available commands that start with the character sequence.
- **Command syntax help:** Enter the **?** command to get command syntax help to see all the available arguments to complete a command (for example, **show ?**). Cisco IOS then displays a list of available arguments.

As part of the help facility, Cisco IOS displays console error messages when incorrect command syntax is entered. Table 29-1 shows sample error messages, what they mean, and how to get help.

**Table 29-1 Console Error Messages**

| Example Error Message | Meaning | How to Get Help |
|---|---|---|
| `switch# `**`cl`**` % Ambiguous command: "cl"` | You did not enter enough characters for your device to recognize the command. | Reenter the command, followed by a question mark (?), without a space between the command and the question mark. The possible keywords that you can enter with the command appear. |
| `switch# `**`clock`**` % Incomplete command.` | You did not enter all the keywords or values required by this command. | Reenter the command, followed by a question mark (?), with a space between the command and the question mark. |
| `switch# `**`clock ste`**` ^ % Invalid input detected at '^' marker.` | You entered the command incorrectly. The caret (^) marks the point of the error. | Enter a question mark (?) to display all the available commands or parameters. |

## CLI Navigation and Editing Shortcuts

Table 29-2 summarizes the shortcuts for navigating and editing commands in the CLI. Although not specifically tested on the CCNA exam, these shortcuts can save you time when using the simulator during the exam.

**Table 29-2 Hot Keys and Shortcuts**

| Keyboard Command | What Happens |
|---|---|
| **Navigation Key Sequences** | |
| Up arrow or Ctrl+P | Displays the most recently used command. If you press the sequence again, the next most recent command appears, until the history buffer is exhausted. (The P stands for *previous*.) |
| Down arrow or Ctrl+N | Moves forward to the more recently entered commands, in case you have gone too far back into the history buffer. (The N stands for *next*.) |
| Left arrow or Ctrl+B | Moves the cursor backward in the currently displayed command without deleting characters. (The B stands for *back*.) |
| Right arrow or Ctrl+F | Moves the cursor forward in the currently displayed command without deleting characters. (The F stands for *forward*.) |
| Tab | Completes a partial command name entry. |
| Backspace | Moves the cursor backward in the currently displayed command, deleting characters. |
| Ctrl+A | Moves the cursor directly to the first character of the currently displayed command. |
| Ctrl+E | Moves the cursor directly to the end of the currently displayed command. |
| Ctrl+R | Redisplays the command line with all characters. This command is useful when messages clutter the screen. |
| Ctrl+D | Deletes a single character. |
| Esc+B | Moves back one word. |
| Esc+F | Moves forward one word. |

| Keyboard Command | What Happens |
|---|---|
| **At the --More Prompt** | |
| Enter key | Displays the next line. |
| Spacebar | Displays the next screen. |
| Any other alphanumeric key | Returns to the EXEC prompt. |
| **Break Keys** | |
| Ctrl+C | When in any configuration mode, ends the configuration mode and returns to privileged EXEC mode. When in setup mode, reverts to the command prompt. |
| Ctrl+Z | When in any configuration mode, ends the configuration mode and returns to privileged EXEC mode. When in user or privileged EXEC mode, logs you out of the router. |
| Ctrl+Shift+6 | Acts as an all-purpose break sequence. Use to abort DNS lookups, traceroutes, and pings. |

## Command History

By default, the Cisco IOS stores in a history buffer the 10 commands you have most recently entered. This gives you a quick way to move backward and forward in the history of commands, choose one, and then edit it before reissuing the command. To view or configure the command history buffer, use the commands in Table 29-3. Although this table shows the switch prompt, these commands are also appropriate for a router.

**Table 29-3 Command History Buffer Commands**

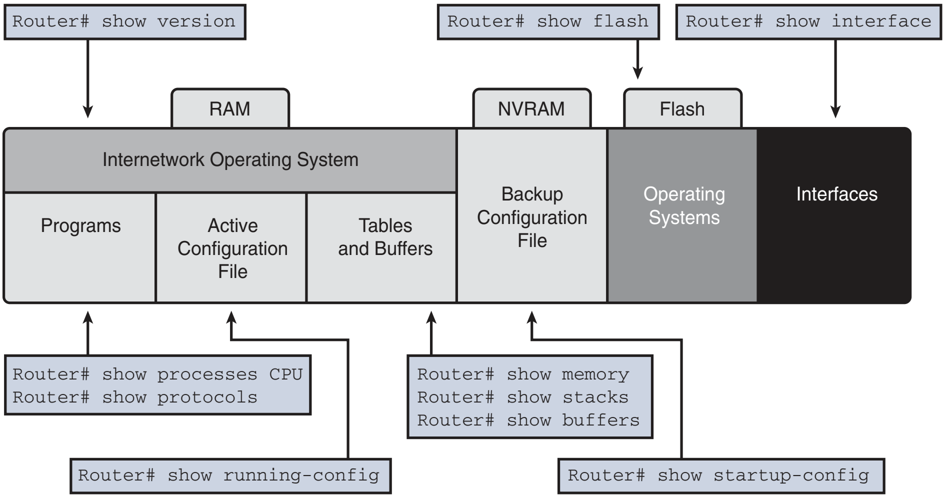
| Command Syntax | Description |
|---|---|
| `switch#` **`show history`** | Displays the commands currently stored in the history buffer. |
| `switch#` **`terminal history`** | Enables terminal history. This command can be run from either user or privileged EXEC mode. |
| `switch#` **`terminal history size 50`** | Configures the terminal history size. The terminal history can maintain 0–256 command lines. |
| `switch#` **`terminal no history size`** | Resets the terminal history size to the default value of 20 command lines in Cisco IOS 15. |
| `switch#` **`terminal no history`** | Disables terminal history. |

## IOS Examination Commands

To verify and troubleshoot network operation, you use **show** commands. Figure 29-2 delineates the different **show** commands, as follows:

- Commands applicable to Cisco IOS (stored in RAM)
- Commands that apply to the backup configuration file stored in NVRAM
- Commands that apply to Flash or specific interfaces

**Figure 29-2 Typical *show* Commands and the Information Provided**




## Subconfiguration Modes

To enter global configuration mode, enter the **configure terminal** command. From global configuration mode, Cisco IOS provides a multitude of subconfiguration modes. Table 29-4 summarizes the most common subconfiguration modes pertinent to the CCNA exam.

**Table 29-4 Cisco Device Subconfiguration Modes**

| Prompt | Name of Mode | Examples of Commands Used to Reach This Mode |
|---|---|---|
| `hostname(config)#` | Global | configure terminal |
| `hostname(config-line)#` | Line | line console 0 line vty 0 15 |
| `hostname(config-if)#` | Interface | interface fastethernet 0/0 |
| `hostname(config-router)#` | Router | router rip router eigrp 100 |

# Basic Switch Configuration Commands

Table 29-5 reviews basic switch configuration commands.

**Table 29-5 Basic Switch Configuration Commands**

| Command Description | Command Syntax |
|---|---|
| Enter global configuration mode. | Switch# **configure terminal** |
| Configure a name for the device. | Switch(config)# **hostname S1** |
| Enter the interface configuration mode for the VLAN 1 interface. | S1(config)# **interface vlan 1** |
| Configure the interface IP address. | S1(config-if)# **ip address 172.17.99.11 255.255.255.0** |

| Command Description | Command Syntax |
|---|---|
| Enable the interface. | `S1(config-if)#` **`no shutdown`** |
| Return to global configuration mode. | `S1(config-if)#` **`exit`** |
| Enter the interface to assign the VLAN. | `S1(config)#` **`interface fastethernet 0/6`** |
| Define the VLAN membership mode for the port. | `S1(config-if)#` **`switchport mode access`** |
| Assign the port to a VLAN. | `S1(config-if)#` **`switchport access vlan 123`** |
| Configure the interface duplex mode to enable AUTO duplex configuration. | `S1(config-if)#` **`duplex auto`** |
| Configure the interface speed and enable AUTO speed configuration. | `S1(config-if)#` **`speed auto`** |
| Enable auto-MDIX on the interface. | `S1(config-if)#` **`mdix auto`** |
| Return to global configuration mode. | `S1(config-if)#` **`exit`** |
| Configure the default gateway on the switch. | `S1(config)#` **`ip default-gateway 172.17.50.1`** |
| Configure the HTTP server for authentication using the **enable** password, which is the default method of HTTP server user authentication. | `S1(config)#` **`ip http authentication enable`** |
| Enable the HTTP server. | `S1(config)#` **`ip http server`** |
| Switch from global configuration mode to line configuration mode for console 0. | `S1(config)#` **`line console 0`** |
| Set **cisco** as the password for the console 0 line on the switch. | `S1(config-line)#` **`password cisco`** |
| Set the console line to require the password to be entered before access is granted. | `S1(config-line)#` **`login`** |
| Return to global configuration mode. | `S1(config-if)#` **`exit`** |
| Switch from global configuration mode to line configuration mode for vty terminals 0–15. | `S1(config)#` **`line vty 0 15`** |
| Set **cisco** as the password for the vty lines on the switch. | `S1(config-line)#` **`password cisco`** |
| Set the vty line to require the password to be entered before access is granted. | `S1(config-line)#` **`login`** |
| Return to global configuration mode. | `S1(config-line)#` **`exit`** |
| Configure **cisco** as the enable password to enter privileged EXEC mode. | `S1(config)#` **`enable password cisco`** |
| Configure **class** as the enable secret password to enter privileged EXEC mode. This password overrides **enable password**. | `S1(config)#` **`enable secret class`** |
| Encrypt all the system passwords that are stored in plaintext. | `S1(config)#` **`service password-encryption`** |
| Configure a login banner. The **#** character delimits the beginning and end of the banner. | `S1 (config)#` **`banner login #Authorized Personnel Only!#`** |

| Command Description | Command Syntax |
|---|---|
| Configure a message of the day (MOTD) login banner. The # character delimits the beginning and end of the banner. | `S1(config)#` **`banner motd #Device maintenance will be occurring on Friday!#`** |
| Return to privileged EXEC mode. | `S1(config)#` **`end`** |
| Save the running configuration to the switch startup configuration. | `S1#` **`copy running-config startup-config`** |

To configure multiple ports with the same command, use the **interface range** command. For example, to configure ports 6–10 as access ports belonging to VLAN 10, you enter the following:

```
Switch(config)# interface range FastEthernet 0/6 - 10
Switch(config-if-range)# switchport mode access
Switch(config-if-range)# switchport access vlan 10
```

# Half Duplex, Full Duplex, and Port Speed

Half-duplex communication is unidirectional data flow in which a device can either send or receive on an Ethernet LAN—but not both at the same time. Today's LAN networking devices and end device network interface cards (NICs) operate at full duplex as long as the device is connected to another device capable of full-duplex communication. Full-duplex communication increases the effective bandwidth by allowing both ends of a connection to transmit and receive data simultaneously; this is known as *bidirectional*. This microsegmented LAN is collision free. Gigabit Ethernet and 10-Gbps NICs require full-duplex connections to operate. Port speed is simply the bandwidth rating of the port. The most common speeds today are 100 Mbps, 1 Gbps, and 10 Gbps.

Although the default duplex and speed setting for Cisco Catalyst 2960 and 3560 switches is **auto**, you can manually configure speed with the **speed** and **duplex** commands.

**NOTE:** Setting the duplex mode and speed of switch ports can cause issues if one end is mismatched or set to autonegotiation. In addition, all fiber-optic ports, such as 100BASE-FX ports, operate only at one preset speed and are always full duplex.

## Automatic Medium-Dependent Interface Crossover (auto-MDIX)

In the past, switch-to-switch or switch-to-router connections required using different Ethernet cables (crossover or straight-through). Using the automatic medium-dependent interface crossover (auto-MDIX) feature on an interface eliminates this problem. When auto-MDIX is enabled, the interface automatically detects the required cable connection type (straight-through or crossover) and configures the connection appropriately.

The auto-MDIX feature is enabled by default on Catalyst 2960 and Catalyst 3560 switches. The Gigabit Ethernet standard requires auto-MDIX, so any 1000-Mbps port has this capability. When using auto-MDIX on an interface, the interface speed and duplex must be set to **auto** so that the feature operates correctly.

# Verifying Network Connectivity

Using and interpreting the output of various testing tools is often the first step in isolating the cause of a network connectivity issue. The **ping** command can systematically test connectivity by looking for answers to the following questions, in this order:

**Step 1.** Can an end device ping itself?

**Step 2.** Can an end device ping its default gateway?

**Step 3.** Can an end device ping the destination?

By using the **ping** command in this ordered sequence, you can isolate problems more quickly. If local connectivity is not an issue—in other words, if the end device can successfully ping its default gateway—using the **traceroute** utility can help isolate the point in the path from source to destination where the traffic stops.

As a first step in the testing sequence, verify the operation of the TCP/IP stack on the local host by pinging the loopback address, 127.0.0.1, as Example 29-1 demonstrates.

**Example 29-1 Testing the TCP/IP Stack on a Windows PC**

```
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=64
Reply from 127.0.0.1: bytes=32 time<1ms TTL=64
Reply from 127.0.0.1: bytes=32 time<1ms TTL=64
Reply from 127.0.0.1: bytes=32 time<1ms TTL=64

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
```

This test should succeed regardless of whether the host is connected to the network, so a failure indicates a software or hardware problem on the host itself. Either the network interface is not operating properly or support for the TCP/IP stack has been inadvertently removed from the operating system.

Next, verify connectivity to the default gateway. Determine the default gateway address by using **ipconfig** and then attempt to ping it, as in Example 29-2.

**Example 29-2 Testing Connectivity to the Default Gateway on a Windows PC**

```
C:\> ipconfig

Windows IP Configuration

```

```
Ethernet adapter Local Area Connection:

         Connection-specific DNS Suffix . : cisco.com
         IP Address. . . . . . . . . . . .: 192.168.1.25
         Subnet Mask . . . . . . . . . . .: 255.255.255.0
         Default Gateway . . . . . . . . . : 192.168.1.1

C:\> ping 192.168.1.1

Pinging 192.168.1.1 with 32 bytes of data:

Reply from 192.168.1.1: bytes=32 time=162ms TTL=255
Reply from 192.168.1.1: bytes=32 time=69ms TTL=255
Reply from 192.168.1.1: bytes=32 time=82ms TTL=255
Reply from 192.168.1.1: bytes=32 time=72ms TTL=255

Ping statistics for 192.168.1.1:
        Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
        Minimum = 69ms, Maximum = 162ms, Average = 96ms
```

Failure here can indicate several problems, which must be checked in a systematic sequence. One possible order might be the following:

**Step 1.** Is the cabling from the PC to the switch correct? Are link lights lit?

**Step 2.** Is the configuration on the PC correct according to the logical map of the network?

**Step 3.** Are the affected interfaces on the switch the cause of the problem? Is there a duplex, speed, or auto-MDIX mismatch? Are there VLAN misconfigurations?

**Step 4.** Is the cabling from the switch to the router correct? Are link lights lit?

**Step 5.** Is the configuration on the router interface correct according to the logical map of the network? Is the interface active?

Finally, verify connectivity to the destination by pinging it. Assume that you are trying to reach a server at 192.168.3.100. Example 29-3 shows a successful **ping** test to the destination.

**Example 29-3 Testing Connectivity to the Destination on a Windows PC**

```
PC> ping 192.168.3.100

Pinging 192.168.3.100 with 32 bytes of data:

Reply from 192.168.3.100: bytes=32 time=200ms TTL=126
Reply from 192.168.3.100: bytes=32 time=185ms TTL=126
Reply from 192.168.3.100: bytes=32 time=186ms TTL=126
```

```
Reply from 192.168.3.100: bytes=32 time=200ms TTL=126

Ping statistics for 192.168.3.100:
   Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
   Minimum = 185ms, Maximum = 200ms, Average = 192ms
```

Failure here indicates a failure in the path beyond the default gateway interface because you already successfully tested connectivity to the default gateway. From a Windows PC, the best tool to use to find the break in the path is the **tracert** command (see Example 29-4).

**NOTE:** Both macOS and Linux use the **traceroute** command rather than **tracert**.

**Example 29-4 Tracing the Route from a Windows PC**

```
C:\> tracert 192.168.3.100

Tracing route to 192.168.3.100 over a maximum of 30 hops:

  1   97 ms    75 ms    72 ms  192.168.1.1
  2   104 ms  119 ms   117 ms  192.168.2.2
  3   *         *        *      Request timed out.
  4   *         *        *      Request timed out.
  5   *         *        *      Request timed out.
  6   ^C
C:\>
```

**NOTE:** Failure at hops 3, 4, and 5 in Example 29-4 could indicate that these routers are configured to not send ICMP messages back to the source.

As shown in in Example 29-4, the last successful hop on the way to the destination was 192.168.2.2. If you have administrator rights to 192.168.2.2, you can continue your research by remotely accessing the command line on 192.168.2.2 and investigating why traffic will not go any further. In addition, other devices between 192.168.2.2 and 192.168.3.100 could be the source of the problem. The point is, you want to use your **ping** and **tracert** tests, as well as your network documentation, to proceed in a logical sequence from source to destination.

Regardless of how simple or complex your network is, using **ping** and **tracert** from the source to the destination is a simple yet powerful way to systematically verify end-to-end connectivity, as well as locate breaks in a path from one source to one destination.

# Troubleshoot Interface and Cable Issues

The physical layer is often the reason a network issue exists—power outage, disconnected cable, power-cycled devices, hardware failures, and so on. This section looks at some troubleshooting tools, in addition to the approach of actually walking over to the wiring closet or network device and "physically" checking Layer 1.

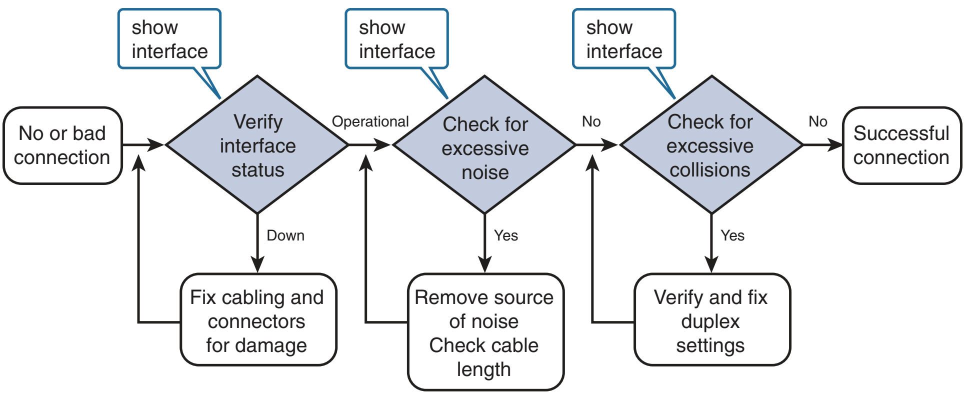
## Media Issues

Besides failing hardware, common physical layer issues occur with media. Consider a few examples:

- New equipment is installed that introduces electromagnetic interference (EMI) sources into the environment.
- Cable runs too close to powerful motors, such as an elevator.
- Poor cable management puts a strain on some RJ-45 connectors, causing one or more wires to break.
- New applications change traffic patterns.
- When new equipment is connected to a switch, the connection operates in half-duplex mode or a duplex mismatch occurs, which can lead to an excessive number of collisions.

Figure 29-3 shows an excellent troubleshooting flowchart that you can use in troubleshooting switch media issues.

**Figure 29-3 Troubleshooting Switch Media Issues**




Next, examine the output from the **show interface** and **show interface status** commands, as described in the next section.

## Interface Status and Switch Configuration

Because today we are focusing on switch troubleshooting, we look at the **show** commands that help in troubleshooting the basic configuration.

### Interface Status Codes

In general, interfaces are either "up" or "down." However, when an interface is "down" and you don't know why, the code in the **show interfaces** command provides more information to help you determine the reason. Table 29-6 lists the code combinations and some possible causes for each status.

**Table 29-6 LAN Switch Interface Status Codes**

| Line Status | Protocol Status | Interface Status | Typical Root Cause |
|---|---|---|---|
| Administratively down | Down | disabled | The interface is configured with the **shutdown** command. |
| Down | Down | notconnect | No cable exists, the cable is bad, incorrect cable pinouts are used, the two connected devices have mismatched speeds, or the device on the other end of the cable is powered off or the other interface is shut down. |
| Up | Down | notconnect | An interface up/down state is not expected on LAN switch interfaces. This indicates a Layer 2 problem on Layer 3 devices. |
| Down | Down (err-disabled) | err-disabled | Port security has disabled the interface. The network administrator must manually reenable the interface. |
| Up | Up | connect | The interface is working. |

## Duplex and Speed Mismatches

One of the most common problems is issues with speed and/or duplex mismatches. On switches and routers, the **speed** {**10** | **100** | **1000**} interface subcommand and the **duplex** {**half** | **full**} interface subcommand set these values. Note that configuring both speed and duplex on a switch interface disables the IEEE-standard autonegotiation process on that interface.

The **show interfaces status** and **show interfaces** commands list both the speed and duplex settings on an interface, as Example 29-5 shows.

**Example 29-5 Commands to Verify Speed and Duplex Settings**

```
S1# show interface status

Port  Name  Status       Vlan    Duplex  Speed  Type
Fa0/1       connected    trunk   full    100    10/100BaseTX
Fa0/2       connected    1       half    100    10/100BaseTX
Fa0/3       connected    1       a-full  a-100  10/100BaseTX
Fa0/4       disabled     1       auto    auto   10/100BaseTX
Fa0/5       disabled     1       auto    auto   10/100BaseTX
Fa0/6       notconnect   1       auto    auto   10/100BaseTX

!Remaining output omitted
S1# show interface fa0/3
FastEthernet0/1 is up, line protocol is up (connected)
   Hardware is Fast Ethernet, address is 001b.5302.4e81 (bia 001b.5302.4e81)
   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
ARP type: ARPA, ARP Timeout 04:00:00
Last input never, 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 1000 bits/sec, 1 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
   2745 packets input, 330885 bytes, 0 no buffer
   Received 1386 broadcasts (0 multicast)
   0 runts, 0 giants, 0 throttles
   0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
   0 watchdog, 425 multicast, 0 pause input
   0 input packets with dribble condition detected
   56989 packets output, 4125809 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
```

Notice that both commands show the duplex and speed settings of the interface. However, the **show interface status** command is preferred for troubleshooting duplex or speed mismatches because it shows exactly how the switch determined the duplex and speed of the interface. In the duplex column, **a-full** means the switch autonegotiated full duplex. The setting **full** or **half** means that the switch was configured at that duplex setting. Autonegotiation has been disabled. In the speed column, **a-100** means the switch autonegotiated 100 Mbps as the speed. The setting **10** or **100** means that the switch was configured at that speed setting.

Finding a duplex mismatch can be much more difficult than finding a speed mismatch because if the duplex settings do not match on the ends of an Ethernet segment, the switch interface will still be in a connect (up/up) state. In this case, the interface works, but the network might work poorly, with hosts experiencing poor performance and intermittent communication problems. To identify duplex mismatch problems, check the duplex setting on each end of the link and watch for incrementing collision and late collision counters, as highlighted in the output at the end of Example 29-5.

## Common Layer 1 Problems On "Up" Interfaces

When a switch interface is "up," it does not necessarily mean that the interface is operating in an optimal state. For this reason, Cisco IOS tracks certain counters to help identify problems that can occur even though the interface is in a connect state. The output in Example 29-5 highlights these counters. Table 29-7 summarizes three general types of Layer 1 interface problems that can occur while an interface is in the "up," connected, state.

**Table 29-7 Common LAN Layer 1 Problem Indicators**

| Type of Problem | Counter Values Indicating This Problem | Common Root Causes |
|---|---|---|
| Excessive noise | Many input errors, few collisions | Wrong cable category (Cat5, Cat5E, Cat6), damaged cables, EMI |
| Collisions | More than roughly 0.1% of all frames are collisions | Duplex mismatch (seen on the half-duplex side), jabber, DoS attack |
| Late collisions | Increasing late collisions | Collision domain or single cable too long, duplex mismatch |

# Study Resources

For today's exam topics, refer to the following resources for more study.

| Resource | Module or Chapter |
|---|---|
| Introduction to Networks v7 | 2 |
| | 17 |
| CCNA 200-301 Official Cert Guide, Volume 1 | 2 |
| | 4 |
| | 6 |
| | 7 |
| Portable Command Guide | 8 |

# IPv4 Addressing

## CCNA 200-301 Exam Topics

- Configure and verify IPv4 addressing and subnetting
- Describe the need for private IPv4 addressing

## Key Topics

Today we focus on reviewing the structure of an IPv4 address, the classes, and private and public IPv4 addresses. Then we turn our focus to IPv4 subnetting.

By now, you should be able to subnet quickly. For example, you should be able to quickly answer a question such as the following: If you are given a /16 network, what subnet mask would you use to maximize the total number of subnets while still providing enough addresses for the largest subnet with 500 hosts? The answer is 255.255.254.0, or /23. This gives you 128 subnets with 510 usable hosts per subnet. You should be able to quickly calculate this information.

The CCNA exam promises to contain many subnetting and subnetting-related questions. Therefore, we devote some time to this necessary skill and also look at designing addressing schemes using variable-length subnet masking (VLSM).

## IPv4 Addressing

Although IPv6 is rapidly permeating the networks of the world, most networks still have large IPv4 implementations. Especially on private networks, migration away from IPv4 will take years to complete. Clearly, IPv4 and your skill in its use are still in demand.

### Header Format

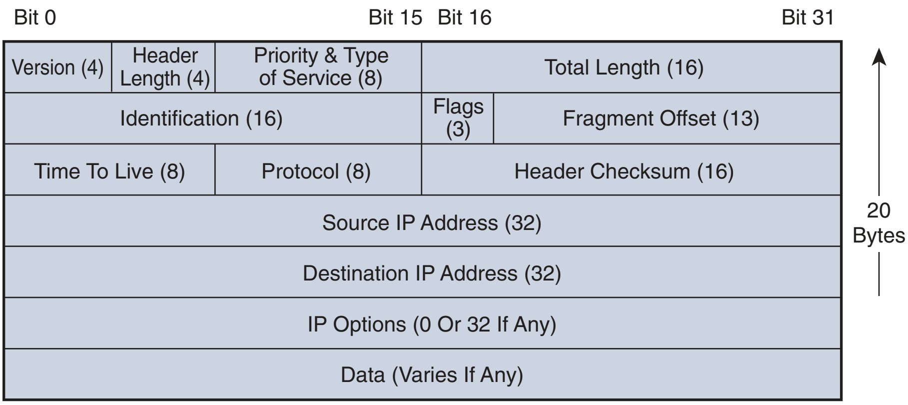
Figure 28-1 shows the layout of the IPv4 header.

Note that each IP packet carries this header, which includes a source IP address and destination IP address.

An IP address consists of two parts:

- **Network ID:** The high-order, or leftmost, bits specify the network address component of the address.
- **Host ID:** The low-order, or rightmost, bits specify the host address component of the address.

**Figure 28-1 IPv4 Header Format**




## Classes of Addresses

From the beginning, IPv4 was designed with class structure: Classes A, B, C, D, and E. Class D is used for multicasting addresses, and Class E is reserved for experimentation. Classes A, B, and C are assigned to network hosts. To provide a hierarchical structure, these classes are divided into network and host portions, as Figure 28-2 shows. The high-order bits specify the network ID, and the low-order bits specify the host ID.

**Figure 28-2 Network/Host Boundary for Each Class of IPv4 Address**

| | 8 Bits | 8 Bits | 8 Bits | 8 Bits |
|---|---|---|---|---|
| **Class A:** | **Network** | Host | Host | Host |
| **Class B:** | **Network** | **Network** | Host | Host |
| **Class C:** | **Network** | **Network** | **Network** | Host |
| **Class D:** | Multicast | | | |
| **Class E:** | Research | | | |

In a classful addressing scheme, devices that operate at Layer 3 can determine the address class of an IP address from the format of the first few bits in the first octet. Initially, this was important so that a networking device could apply the default subnet mask for the address and determine the host address. Table 28-1 summarizes how addresses are divided into classes, the default subnet mask, the number of networks per class, and the number of hosts per classful network address.

**Table 28-1 IPv4 Address Classes**

| Address Class | First Octet Range (Decimal) | First Octet Bits (Highlighted Bits Do Not Change) | Network (N) and Host (H) Portions of Addresses | Default Subnet Mask (Decimal and Binary) | Number of Possible Networks and Hosts per Network |
|---|---|---|---|---|---|
| A | 1–127 | 00000000–01111111 | N.H.H.H | 255.0.0.0 | $2^7$, or 128, networks |
| | | | | 11111111.00000000.00000000.00000000 | $2^{24}$–2, or 16,777,214, hosts per network |
| B | 128–191 | 10000000–10111111 | N.N.H.H | 255.255.0.0 | $2^{14}$, or 16,384, networks |
| | | | | 11111111.11111111.00000000.00000000 | $2^{16}$–2, or 65,534, hosts per network |
| C | 192–223 | 11000000–11011111 | N.N.N.H | 255.255.255.0 | $2^{21}$, or 2,097,152, networks |
| | | | | 11111111.11111111.11111111.00000000 | $2^8$–2, or 254, hosts per network |
| D | 224–239 | 11100000–11101111 | Not used for host addressing | | |
| E | 240–255 | 11110000–11111111 | Not used for host addressing | | |

In the last column of Table 28-1, the –2 for hosts per network accounts for the reserved network and broadcast addresses for each network. These two addresses cannot be assigned to hosts.

**NOTE:** We do not review the process of converting between binary and decimal. At this point in your studies, you should be comfortable moving between the two numbering systems. If not, take some time to practice this necessary skill. You can search the Internet for binary conversion tricks, tips, and games to practice. The Cisco Learning Network has a fun game you can play, at https://learningnetwork.cisco.com/docs/DOC-1803.

## Purpose of the Subnet Mask

Subnet masks are always a series of 1 bits followed by a series of 0 bits. The boundary where the series changes from 1s to 0s is the boundary between the network and the host. This is how a device that operates at Layer 3 determines the network address for a packet: by finding the bit boundary where the series of 1 bits ends and the series of 0 bits begins. The bit boundary for default subnet masks breaks on the octet boundary. Determining the network address for an IP address that uses a default mask is easy.

Say that a router receives a packet destined for 192.168.1.51. By ANDing the IP address and the subnet mask, the router determines the network address for the packet. By the ANDing rules, a 1 AND a 1 equals 1. All other possibilities equal 0. Table 28-2 shows the results of the ANDing operation. Notice that the host bits in the last octet are ignored.

**Table 28-2 ANDing an IP Address and Subnet Mask to Find the Network Address**

| | | |
|---|---|---|
| Destination address | 192.168.1.51 | 11000000.10101000.00000001.00110011 |
| Subnet mask | 255.255.255.0 | 11111111.11111111.11111111.00000000 |
| Network address | 192.168.1.0 | 11000000.10101000.00000001.00000000 |

The bit boundary can now occur in just about any place in the 32 bits. Table 28-3 summarizes the values for the last nonzero octet in a subnet mask.

**Table 28-3 Subnet Mask Binary Values**

| Mask (Decimal) | Mask (Binary) | Network Bits | Host Bits |
|---|---|---|---|
| 0 | 00000000 | 0 | 8 |
| 128 | 10000000 | 1 | 7 |
| 192 | 11000000 | 2 | 6 |
| 224 | 11100000 | 3 | 5 |
| 240 | 11110000 | 4 | 4 |
| 248 | 11111000 | 5 | 3 |
| 252 | 11111100 | 6 | 2 |
| 254 | 11111110 | 7 | 1 |
| 255 | 11111111 | 8 | 0 |

# Private and Public IP Addressing

RFC 1918, "Address Allocation for Private Internets," eased the demand for IP addresses by reserving the following addresses for use in private internetworks:

- **Class A:** 10.0.0.0/8 (10.0.0.0–10.255.255.255)
- **Class B:** 172.16.0.0/12 (172.16.0.0–172.31.255.255)
- **Class C:** 192.168.0.0/16 (192.168.0.0–192.168.255.255)

If you are addressing a nonpublic intranet, these private addresses are normally used instead of globally unique public addresses. This provides flexibility in your addressing design. Any organization can take full advantage of an entire Class A address (10.0.0.0/8). Forwarding traffic to the public Internet requires translation to a public address using Network Address Translation (NAT). But by overloading an Internet-routable address with many private addresses, a company needs only a handful of public addresses. Day 8, "NAT," reviews NAT operation and configuration in greater detail.

# Subnetting in Four Steps

Everyone has a preferred method of subnetting. Each teacher uses a slightly different strategy to help students master this crucial skill, and each of the suggested study resources has a slightly different way of approaching this subject.

The method I prefer consists of four steps:

**Step 1.** Determine how many bits to borrow, based on the host requirements.

**Step 2.** Determine the new subnet mask.

**Step 3.** Determine the subnet multiplier.

**Step 4.** List the subnets, including the subnetwork address, host range, and broadcast address.

The best way to demonstrate this method is to use an example. Assume that you are given the network address 192.168.1.0 with the default subnet mask 255.255.255.0. The network address and subnet mask can be written as 192.168.1.0/24. The /24 represents the subnet mask in a shorter notation and means that the first 24 bits are network bits.

Now further assume that you need 30 hosts per network and want to create as many subnets for the given address space as possible. With these network requirements, you can now subnet the address space.

## Determine How Many Bits to Borrow

To determine the number of bits you can borrow, you first must know how many host bits you have to start with. Because the first 24 bits are network bits in this example, the remaining 8 bits are host bits.

Because our requirement specifies 30 host addresses per subnet, we need to first determine the minimum number of host bits to leave. The remaining bits can be borrowed:

Host Bits = Bits Borrowed + Bits Left

To provide enough address space for 30 hosts, we need to leave 5 bits. Use the following formula:

$2^{BL} - 2$ = number of host addresses

The exponent BL is bits left in the host portion.

Remember, the –2 accounts for the network and broadcast addresses that cannot be assigned to hosts.

In this example, leaving 5 bits in the host portion provides the right number of host addresses:

$2^5 - 2 = 30$

Because we have 3 bits remaining in the original host portion, we borrow all these bits to satisfy the requirement to "create as many subnets as possible." To determine how many subnets we can create, use the following formula:

$2^{BB}$ = number of subnets

The exponent BB is bits borrowed from the host portion.

In this example, borrowing 3 bits from the host portion creates eight subnets: $2^3 = 8$.

As Table 28-4 shows, the 3 bits are borrowed from the leftmost bits in the host portion. The highlighted bits in the table show all possible combinations of manipulating the 8 bits borrowed to create the subnets.

**Table 28-4 Binary and Decimal Value of the Subnetted Octet**

| Subnet Number | Last Octet Binary Value | Last Octet Decimal Value |
|---|---|---|
| 0 | 00000000 | .0 |
| 1 | 00100000 | .32 |
| 2 | 01000000 | .64 |
| 3 | 01100000 | .96 |
| 4 | 10000000 | .128 |
| 5 | 10100000 | .160 |
| 6 | 11000000 | .192 |
| 7 | 11100000 | .224 |

## Determine the New Subnet Mask

Notice in Table 28-4 that the network bits now include the 3 borrowed host bits in the last octet. Add these 3 bits to the 24 bits in the original subnet mask, and you have a new subnet mask, /27. In decimal format, you turn on the 128, 64, and 32 bits in the last octet, for a value of 224. The new subnet mask is thus 255.255.255.224.

## Determine the Subnet Multiplier

Notice in Table 28-4 that the last octet decimal value increments by 32 with each subnet number. The number 32 is the subnet multiplier. You can quickly find the subnet multiplier by using one of two methods:

- **Method 1:** Subtract the last nonzero octet of the subnet mask from 256. In this example, the last nonzero octet is 224. The subnet multiplier is therefore 256 – 224 = 32.
- **Method 2:** The decimal value of the last bit borrowed is the subnet multiplier. In this example, we borrowed the 128 bit, the 64 bit, and the 32 bit. The 32 bit is the last bit we borrowed and is, therefore, the subnet multiplier.

By using the subnet multiplier, you no longer have to convert binary subnet bits to decimal.

## List the Subnets, Host Ranges, and Broadcast Addresses

Listing the subnets, host ranges, and broadcast addresses helps you see the flow of addresses within one address space. Table 28-5 documents our subnet addressing scheme for the 192.168.1.0/24 address space.

**Table 28-5 Subnet Addressing Scheme for 192.168.1.0/24: 30 Hosts per Subnet**

| Subnet Number | Subnet Address | Host Range | Broadcast Address |
|---|---|---|---|
| 0 | 192.168.1.0 | 192.168.1.1–192.168.1.30 | 192.168.1.31 |
| 1 | 192.168.1.32 | 192.168.1.33–192.168.1.62 | 192.168.1.63 |
| 2 | 192.168.1.64 | 192.168.1.65–192.168.1.94 | 192.168.1.95 |

| Subnet Number | Subnet Address | Host Range | Broadcast Address |
|---|---|---|---|
| 3 | 192.168.1.96 | 192.168.1.97–192.168.1.126 | 192.168.1.127 |
| 4 | 192.168.1.128 | 192.168.1.129–192.168.1.158 | 192.168.1.159 |
| 5 | 192.168.1.160 | 192.168.1.161–192.168.1.190 | 192.168.1.191 |
| 6 | 192.168.1.192 | 192.168.1.193–192.168.1.222 | 192.168.1.223 |
| 7 | 192.168.1.224 | 192.168.1.225–192.168.1.254 | 192.168.1.255 |

Following are three examples using the four subnetting steps. For brevity, step 4 lists only the first three subnets.

## Subnetting Example 1

Subnet the address space 172.16.0.0/16 to provide at least 80 host addresses per subnet while creating as many subnets as possible.

**Step 1.** There are 16 host bits. Leave 7 bits for host addresses ($2^7 - 2 = 126$ host addresses per subnet). Borrow the first 9 host bits to create as many subnets as possible ($2^9 = 512$ subnets).

**Step 2.** The original subnet mask is /16, or 255.255.0.0. Turn on the next 9 bits starting in the second octet, for a new subnet mask of /25 or 255.255.255.128.

**Step 3.** The subnet multiplier is 128, which can be found as 256 – 128 = 128, or because the 128 bit is the last bit borrowed.

For step 4, Table 28-6 lists the first three subnets, host ranges, and broadcast addresses.

**Table 28-6 Subnet Addressing Scheme for Example 1**

| Subnet Number | Subnet Address | Host Range | Broadcast Address |
|---|---|---|---|
| 0 | 172.16.0.0 | 172.16.0.1–172.16.0.126 | 172.16.0.127 |
| 1 | 172.16.0.128 | 172.16.0.129–172.16.0.254 | 172.16.0.255 |
| 2 | 172.16.1.0 | 172.16.1.1–172.16.1.126 | 172.16.1.127 |

## Subnetting Example 2

Subnet the address space 172.16.0.0/16 to provide at least 80 subnet addresses.

**Step 1.** There are 16 host bits. Borrow the first 7 host bits to create at least 80 subnets ($2^7 = 128$ subnets). That leaves 9 bits for host addresses, or $2^9 - 2 = 510$ host addresses per subnet.

**Step 2.** The original subnet mask is /16, or 255.255.0.0. Turn on the next 7 bits starting in the second octet, for a new subnet mask of /23, or 255.255.254.0.

**Step 3.** The subnet multiplier is 2, which can be found as 256 – 254 = 2, or because the 2 bit is the last bit borrowed.

For step 4, Table 28-7 lists the first three subnets, host ranges, and broadcast addresses.

**Table 28-7 Subnet Addressing Scheme for Example 2**

| Subnet Number | Subnet Address | Host Range | Broadcast Address |
|---|---|---|---|
| 0 | 172.16.0.0 | 172.16.0.1–172.16.1.254 | 172.16.1.255 |
| 1 | 172.16.2.0 | 172.16.2.1–172.16.3.254 | 172.16.3.255 |
| 2 | 172.16.4.0 | 172.16.4.1–172.16.5.254 | 172.16.5.255 |

## Subnetting Example 3

Subnet the address space 172.16.10.0/23 to provide at least 60 host addresses per subnet while creating as many subnets as possible.

**Step 1.** There are 9 host bits. Leave 6 bits for host addresses ($2^6 - 2 = 62$ host addresses per subnet). Borrow the first 3 host bits to create as many subnets as possible ($2^3 = 8$ subnets).

**Step 2.** The original subnet mask is /23, or 255.255.254.0. Turn on the next 3 bits starting with the last bit in the second octet, for a new subnet mask of /26, or 255.255.255.192.

**Step 3.** The subnet multiplier is 64, which can be found as 256 – 192 = 64, or because the 64 bit is the last bit borrowed.

For step 4, Table 28-8 lists the first three subnets, host ranges, and broadcast addresses.

**Table 28-8 Subnet Addressing Scheme for Example 3**

| Subnet Number | Subnet Address | Host Range | Broadcast Address |
|---|---|---|---|
| 0 | 172.16.10.0 | 172.16.10.1–172.16.10.62 | 172.16.10.63 |
| 1 | 172.16.10.64 | 172.16.10.65–172.16.10.126 | 172.16.10.127 |
| 2 | 172.16.10.128 | 172.16.10.129–172.16.10.190 | 172.16.10.191 |

# VLSM

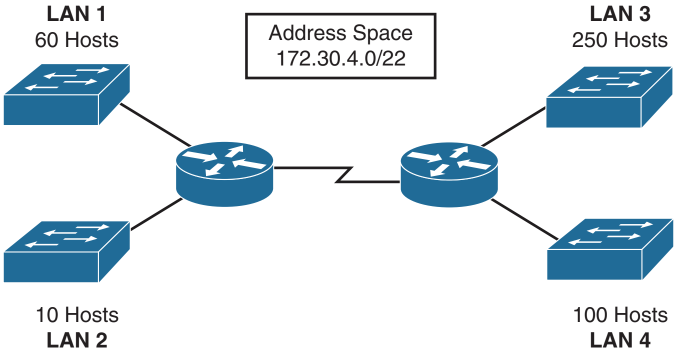
You probably noticed that the starting address space in Subnetting Example 3 is not an entire classful address. In fact, it is subnet 5 from Subnetting Example 2. In Subnetting Example 3, therefore, we "subnetted a subnet." In a nutshell, VLSM is subnetting a subnet.

With VLSM, you can customize subnets to fit your network. Subnetting works the same way. You just have to do it more than once to complete your addressing scheme. To avoid overlapping address spaces, start with your largest host requirement, create a subnet for it, and then continue with the next-largest host requirement.

Consider a small example. Given the address space 172.30.4.0/22 and the network requirements in Figure 28-3, apply an addressing scheme that conserves the most addresses for future growth.

We need five subnets: four LAN subnets and one WAN subnet. Starting with the largest host requirement on LAN 3, begin subnetting the address space.

**Figure 28-3 VLSM Example Topology**




To satisfy the 250-host requirement, we leave 8 host bits ($2^8 - 2 = 254$ hosts per subnet). Because we have 10 host bits total, we borrow 2 bits to create the first round of subnets ($2^2 = 4$ subnets). The starting subnet mask is /22, or 255.255.252.0. We turn on the next 2 bits in the subnet mask to get /24, or 255.255.255.0. The multiplier is 1. The four subnets are as follows:

- **Subnet 0:** 172.30.4.0/24
- **Subnet 1:** 172.30.5.0/24
- **Subnet 2:** 172.30.6.0/24
- **Subnet 3:** 172.30.7.0/24

Assigning Subnet 0 to LAN 3, we are left with three /24 subnets. Continuing on to the next-largest host requirement on LAN 4, we further subnet Subnet 1, 172.30.5.0/24.

To satisfy the 100-host requirement, we leave 7 bits ($2^7 - 2 = 128$ hosts per subnet). Because we have 8 host bits total, we can borrow only 1 bit to create the subnets ($2^1 = 2$ subnets). The starting subnet mask is /24, or 255.255.255.0. We turn on the next bit in the subnet mask to get /25, or 255.255.255.128. The multiplier is 128. The two subnets are as follows:

- **Subnet 0:** 172.30.5.0/25
- **Subnet 1:** 172.30.5.128/25

Assigning Subnet 0 to LAN 4, we are left with one /25 subnet and two /24 subnets. Continuing on to the next-largest host requirement on LAN 1, we further subnet Subnet 1, 172.30.5.128/25.

To satisfy the 60-host requirement, we leave 6 bits ($2^6 - 2 = 62$ hosts per subnet). Because we have 7 host bits total, we borrow 1 bit to create the subnets ($2^1 = 2$ subnets). The starting subnet mask is /25, or 255.255.255.128. We turn on the next bit in the subnet mask to get /26, or 255.255.255.192. The multiplier is 64. The two subnets are as follows:

- **Subnet 0:** 172.30.5.128/26
- **Subnet 1:** 172.30.5.192/26

Assigning Subnet 0 to LAN 1, we are left with one /26 subnet and two /24 subnets. Finishing our LAN subnetting with LAN 2, we further subnet Subnet 1, 172.30.5.192/26.

To satisfy the 10-host requirement, we leave 4 bits ($2^4 - 2 = 14$ hosts per subnet). Because we have 6 host bits total, we borrow 2 bits to create the subnets ($2^2 = 4$ subnets). The starting subnet mask is /26, or 255.255.255.192. We turn on the next 2 bits in the subnet mask to get /28, or 255.255.255.240. The multiplier is 16. The four subnets are as follows:

- **Subnet 0:** 172.30.5.192/28
- **Subnet 1:** 172.30.5.208/28
- **Subnet 2:** 172.30.5.224/28
- **Subnet 3:** 172.30.5.240/28

Assigning Subnet 0 to LAN 2, we are left with three /28 subnets and two /24 subnets. To finalize our addressing scheme, we need to create a subnet for the WAN link, which needs only two host addresses. We further subnet Subnet 1, 172.30.5.208/28.

To satisfy the two-host requirement, we leave 2 bits ($2^2 - 2 = 2$ hosts per subnet). Because we have 4 host bits total, we borrow 2 bits to create the subnets ($2^2 = 4$ subnets). The starting subnet mask is /28, or 255.255.255.240. We turn on the next 2 bits in the subnet mask to get /30, or 255.255.255.252. The multiplier is 4. The four subnets are as follows:

- **Subnet 0:** 172.30.5.208/30
- **Subnet 1:** 172.30.5.212/30
- **Subnet 2:** 172.30.5.216/30
- **Subnet 3:** 172.30.5.220/30

We assign Subnet 0 to the WAN link. We are left with three /30 subnets, two /28 subnets, and two /24 subnets.

# Study Resources

For today's exam topics, refer to the following resources for more study.

| Resource | Module or Chapter |
|---|---|
| Cisco Network Academy: CCNA1 | 11 |
| CCNA 200-301 Official Cert Guide, Volume 1 | 11 |
| | 12 |
| | 13 |
| | 14 |
| CCNA 200-301 Official Cert Guide, Volume 2 | 7 |
| Portable Command Guide | 1 |
| | 2 |
| | 3 |
| | 4 |

# IPv6 Addressing

## CCNA 200-301 Exam Topics

- Configure and verify IPv6 addressing and prefix
- Compare IPv6 address types

## Key Topics

In the early 1990s, the Internet Engineering Task Force (IETF) grew concerned about the exhaustion of IPv4 network addresses and began to look for a replacement for this protocol. This activity led to the development of what is now known as IPv6. Today's review focuses on the IPv6 protocol and IPv6 address types. We also review the various ways to implement IPv6 addressing, including subnetting, autoconfiguring hosts, and running IPv6 and IPv4 in a dual-stack configuration. IPv6 configuration on routers will be reviewed on Day 18, "Basic Router Configuration."

**NOTE:** If you have not yet purchased a copy of Rick Graziani's *IPv6 Fundamentals* to add to your library of study tools, now is the time to do so. His book is my definitive source for everything IPv6.

## Overview and Benefits of IPv6

Scaling networks today requires a limitless supply of IP addresses and improved mobility that private addressing and NAT alone cannot meet. IPv6 satisfies the increasingly complex requirements of hierarchical addressing that IPv4 does not provide. The main benefits and features of IPv6 include the following:

- **Extended address space:** A 128-bit address space represents about 340 trillion trillion trillion addresses.
- **Stateless address autoconfiguration:** IPv6 provides host devices with a method for generating their own routable IPv6 addresses. IPv6 also supports stateful configuration using DHCPv6.
- **Eliminates the need for NAT/PAT:** NAT/PAT was conceived as part of the solution to IPv4 address depletion. With IPv6, address depletion is no longer an issue. NAT64, however, does play an important role in providing backward compatibility with IPv4.
- **Simpler header:** A simpler header offers several advantages over IPv4:
  - Better routing efficiency for performance and forwarding-rate scalability
  - No broadcasts and, thus, no potential threat of broadcast storms

  - No requirement for processing checksums
  - Simpler and more efficient extension header mechanisms
- **Mobility and security:** Mobility and security help ensure compliance with mobile IP and IPsec standards:
  - IPv4 does not automatically enable mobile devices to move without breaks in established network connections.
  - In IPv6, mobility is built in, which means that any IPv6 node can use mobility when necessary.
  - IPsec is enabled on every IPv6 node and is available for use, making the IPv6 Internet more secure.
- **Transition strategies:** You can incorporate existing IPv4 capabilities with the added features of IPv6 in several ways:
  - You can implement a dual-stack method, with both IPv4 and IPv6 configured on the interface of a network device.
  - You can use tunneling, which will become more prominent as the adoption of IPv6 grows.

# The IPv6 Protocol

Table 27-1 compares the binary and alphanumeric representations of IPv4 and IPv6 addresses.

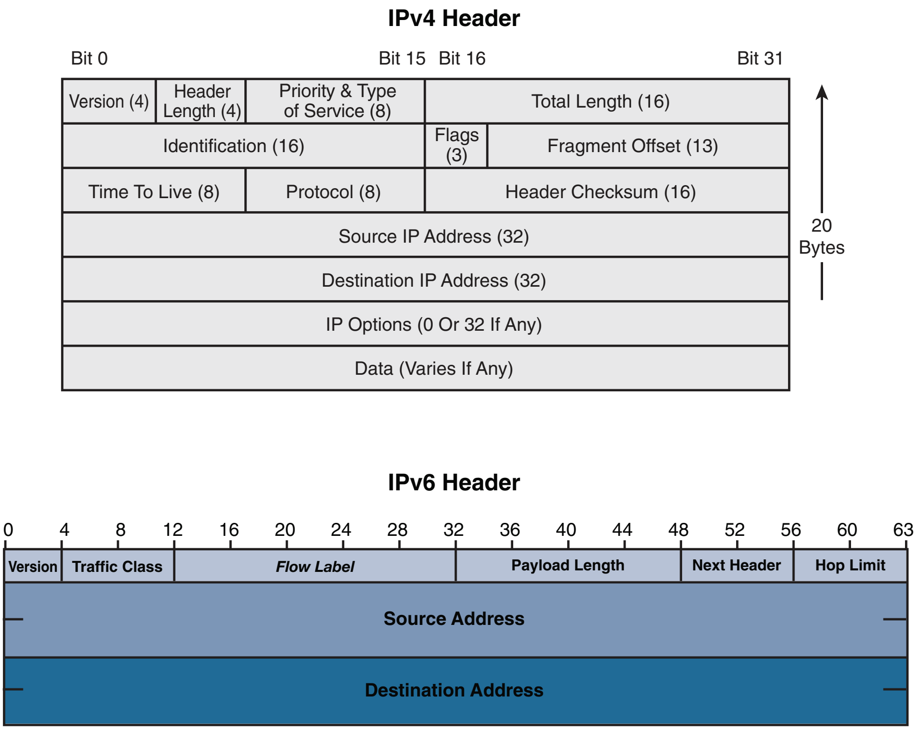
**Table 27-1 IPv4 and IPv6 Address Comparison**

| | IPv4 (4 Octets) | IPv6 (16 Octets) |
|---|---|---|
| Binary representation | 11000000.10101000.00001010. 01100101 | 10100101.00100100.01110010.11010011.00101100.10000000.11011101.00000010.00000000.00101001.11101100.01111010.00000000.00101011.11101010.01110011 |
| Alphanumeric representation | 192.168.10.101 | 2001:0DB8:2C80:DD02:0029:EC7A:002B:EA73 |
| Total IP addresses | 4,294,967,296, or $2^{32}$ | $3.4 \times 10^{38}$, or $2^{128}$ |

Figure 27-1 compares the IPv4 header with the main IPv6 header. Notice that the IPv6 header is represented in 64-bit words instead of the 32-bit words used by IPv4.

**NOTE:** Refer to RFC 2460 and the "Study Resources" section for the full specification of IPv6.

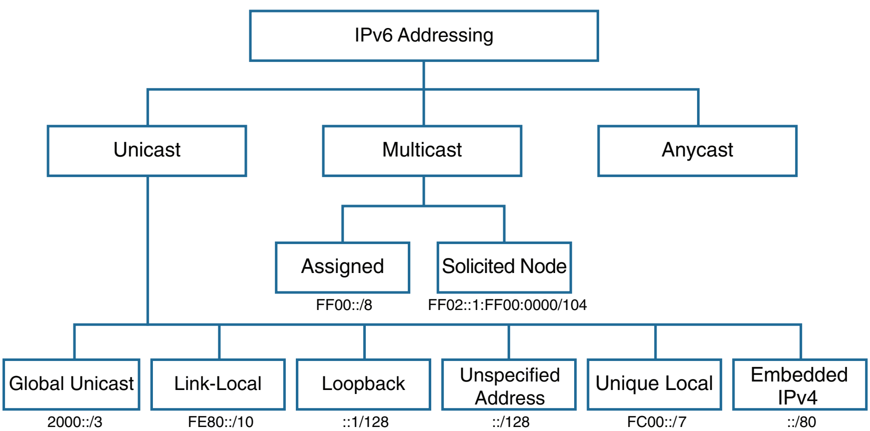
**Figure 27-1 IPv6 Header Format**




# IPv6 Address Types

IPv4 has three address types: unicast, multicast, and broadcast. IPv6 does not use broadcast addresses. Instead, IPv6 uses unicast, multicast, and anycast addresses. Figure 27-2 illustrates these three types of IPv6 addresses.

**Figure 27-2 IPv6 Address Types**



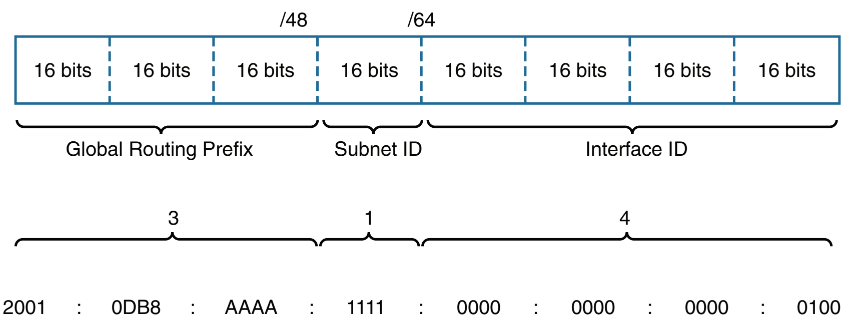
# Unicast

The first classification of IPv6 address types shown in Figure 27-2 is the unicast address. A unicast address uniquely identifies an interface on an IPv6 device. A packet sent to a unicast address is received by the interface that is assigned to that address. Much as with IPv4, source IPv6 addresses must be unicast addresses. Because unicast addressing—as opposed to multicast and anycast addressing—is the major focus for a CCNA candidate, we spend some time reviewing the Unicast branch in Figure 27-2.

## Global Unicast Address

IPv6 has an address format that enables aggregation upward, eventually to the ISP. An IPv6 global unicast address is globally unique. Like a public IPv4 address, it can be routed in the Internet without modification. An IPv6 global unicast address consists of a 48-bit global routing prefix, a 16-bit subnet ID, and a 64-bit interface ID. Use Rick Graziani's method of breaking down the IPv6 address with the 3-1-4 rule (also known as the *pi rule*, for 3.14), shown in Figure 27-3.

**Figure 27-3 Graziani's 3-1-4 Rule for Remembering the Global Unicast Address Structure**

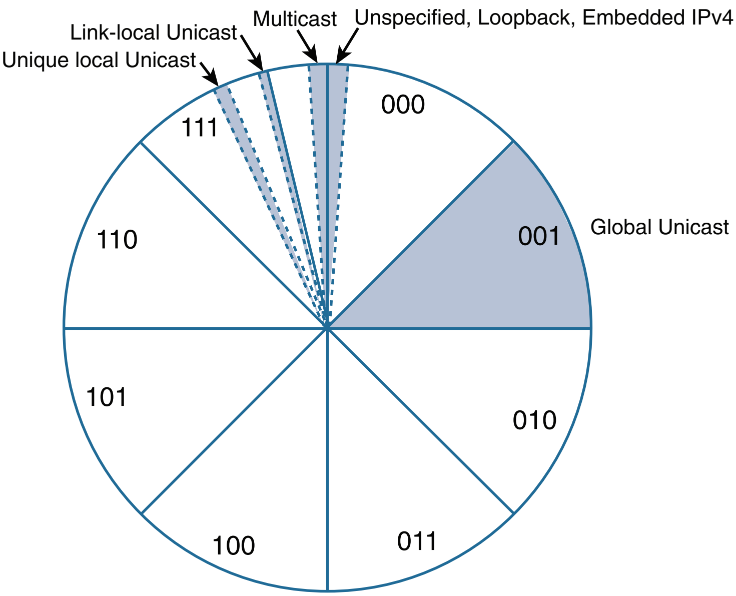



Each number refers to the number of hextets, or 16-bit segments, of that portion of the address:

- **3:** Three hextets for the global routing prefix
- **1:** One hextet for the subnet ID
- **4:** Four hextets for the interface ID

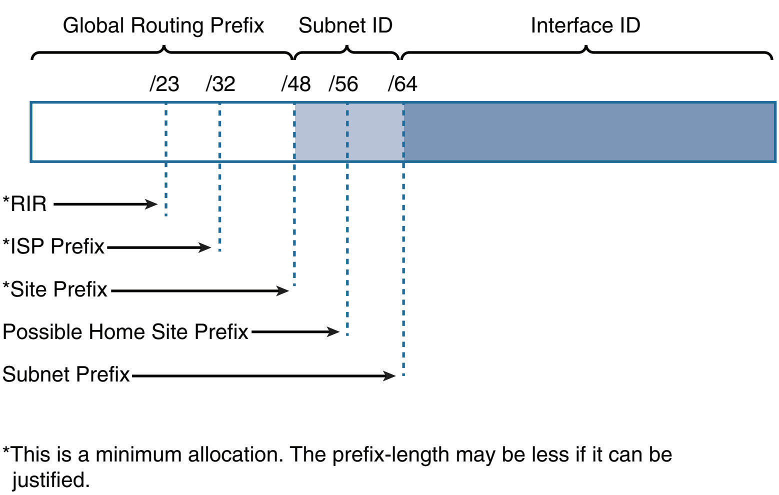
Global unicast addresses that are currently assigned by the Internet Assigned Numbers Authority (IANA) use the range of addresses that start with binary value 001 (2000::/3). This range represents one-eighth of the total IPv6 address space and is the largest block of assigned addresses. Figure 27-4 shows how the IPv6 address space is divided into an eight-piece pie based on the value of the first 3 bits.

**Figure 27-4 Allocation of IPv6 Address Space**




Using the 2000::/3 pie piece, the IANA assigns /23 or shorter address blocks to the five Regional Internet Registries (RIRs). From there, ISPs are assigned /32 or shorter address blocks. ISPs then assign each site—that is, each customer—a /48 or shorter address block. Figure 27-5 shows the breakdown of global routing prefixes.

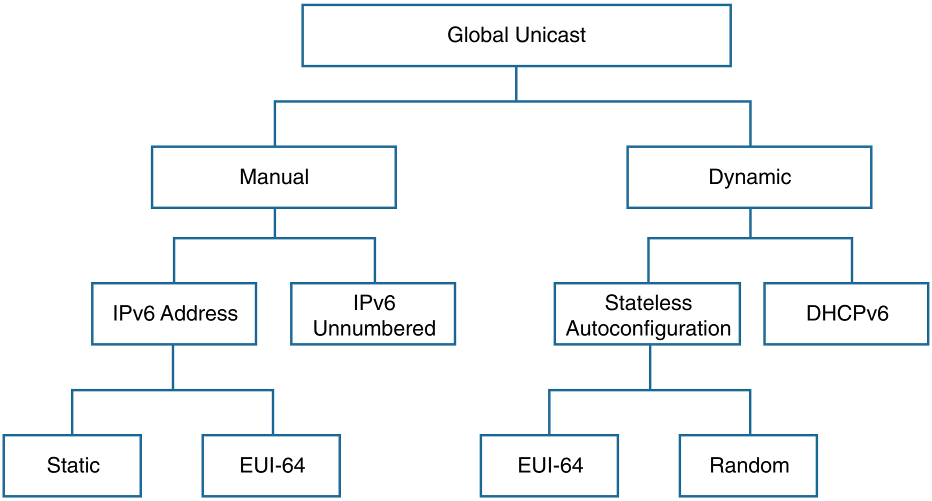
**Figure 27-5 Classification of Global Routing Prefix Sizes**



In IPv6, an interface can be configured with multiple global unicast addresses, which can be on the same or different subnets. In addition, an interface does not have to be configured with a global unicast address, but it must at least have a link-local address.

A global unicast address can be further classified into the various configuration options available, as Figure 27-6 shows.

**Figure 27-6 Global Unicast Address Configuration Options**




We review EUI-64 and stateless address autoconfiguration in more detail later in this day. In upcoming days, we review the rest of the configuration options in Figure 27-6 in more detail. For now, Table 27-2 summarizes them.

**Table 27-2 Summary of Global Unicast Configuration Options**

| Global Unicast | Configuration Option | Description |
|---|---|---|
| Manual | Static | Much as with IPv4, the IPv6 address and prefix are statically configured on the interface. |
| | EUI-64 | The prefix is configured manually. The EUI-64 process uses the MAC address to generate the 64-bit interface ID. |
| | IPv6 unnumbered | Much as with IPv4, an interface can be configured to use the IPv6 address of another interface on the same device. |
| Dynamic | Stateless address autoconfiguration | SLAAC determines the prefix and prefix length from neighbor discovery router advertisement messages and then creates the interface ID using the EUI-64 method. |
| | DHCPv6 | Much as with IPv4, a device can receive some or all of its addressing from a DHCPv6 server. |

## Link-Local Address

As Figure 27-2 shows, link-local addresses are a type of unicast address. Link-local addresses are confined to a single link. They need to be unique only to that link because packets with a link-local source or destination address are not routable off the link.

Link-local addresses are configured in one of three ways:

- Dynamically, using EUI-64
- Using a randomly generated interface ID
- Statically, entering the link-local address manually

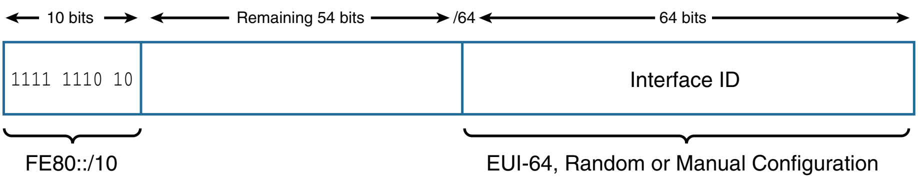
Link-local addresses provide a unique benefit in IPv6. A device can create its link-local address completely on its own. Link-local unicast addresses are in the range FE80::/10 to FEBF::/10, as Table 27-3 shows.

**Table 27-3 Range of Link-Local Unicast Addresses**

| Link-Local Unicast Address | Range of First Hextet | Range of First Hextet in Binary |
|---|---|---|
| FE80::/10 | FE80 | 1111 1110 10 00 0000 |
| | FEBF | 1111 1110 10 11 1111 |

Figure 27-7 shows the format of a link-local unicast address.

**Figure 27-7 Link-Local Unicast Address**




## Loopback Address

The loopback address for IPv6 is an all-0s address except for the last bit, which is set to 1. As in IPv4, an end device uses the IPv6 loopback address to send an IPv6 packet to itself to test the TCP/IP stack. The loopback address cannot be assigned to an interface and is not routable outside the device.

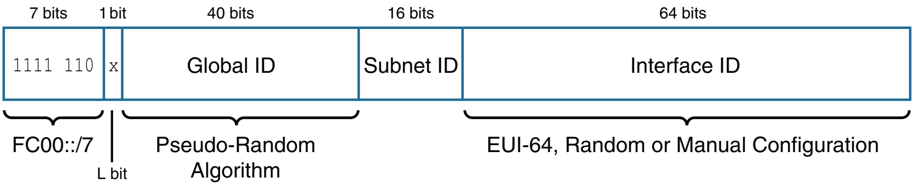
## Unspecified Address

The unspecified unicast address is the all-0s address, represented as ::. It cannot be assigned to an interface but is reserved for communications when the sending device does not have a valid IPv6 address yet. For example, a device uses :: as the source address when using the duplicate address detection (DAD) process. The DAD process ensures a unique link-local address. Before a device can begin using its newly created link-local address, it sends out an all-nodes multicast to all devices on the link, with its new address as the destination. If the device receives a response, it knows that link-local address is in use and, therefore, needs to create another link-local address.

## Unique Local Address

Unique local addresses (ULA) are defined by RFC 4193, "Unique Local IPv6 Unicast Addresses." Figure 27-8 shows the format for ULAs.

**Figure 27-8 Unique Local Address**




These are private addresses. However, unlike in IPv4, IPv6 ULAs are globally unique. This is possible because of the relatively large amount of address space in the Global ID portion shown in Figure 27-8: 40 bits, or more than 1 trillion unique global IDs. As long as a site uses the pseudo-random global ID algorithm, it will have a very high probability of generating a unique global ID.

Unique local addresses have the following characteristics:

- Possess a globally unique prefix or at least have a very high probability of being unique
- Allow sites to be combined or privately interconnected without address conflicts or addressing renumbering
- Remain independent of any Internet service provider and can be used within a site without having Internet connectivity
- If accidentally leaked outside a site by either routing or the Domain Name System (DNS), don't cause a conflict with any other addresses
- Can be used just like a global unicast address

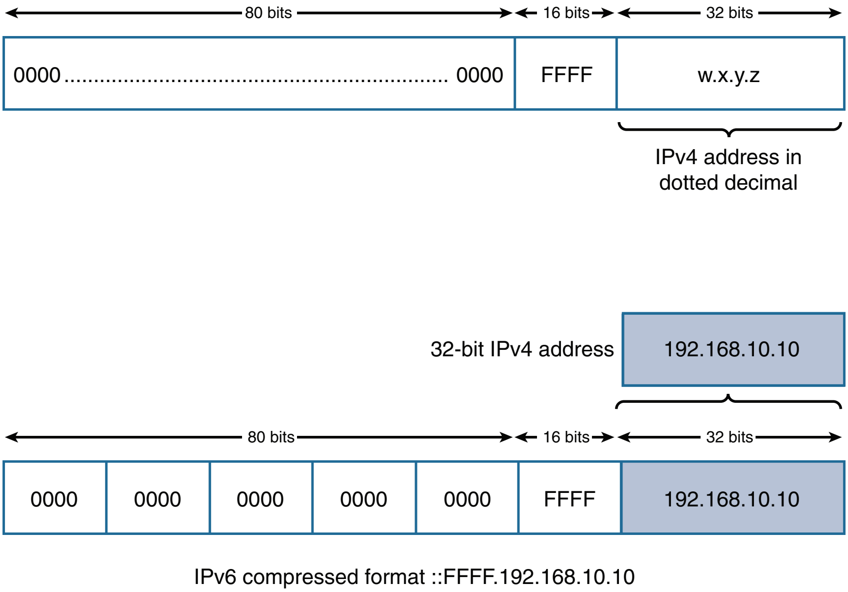
## IPv4 Embedded Address

IPv4 and IPv6 packets are not compatible. Features such as NAT-PT (now deprecated) and NAT64 are required to translate between the two address families. IPv4-mapped IPv6 addresses are used by transition mechanisms on hosts and routers to create IPv4 tunnels that deliver IPv6 packets over IPv4 networks.

**NOTE:** NAT64 is beyond the scope of the CCNA exam topics.

To create an IPv4-mapped IPv6 address, the IPv4 address is embedded within the low-order 32 bits of IPv6. Basically, IPv6 just puts an IPv4 address at the end, adds 16 all-1 bits, and pads the rest of the address. The address does not have to be globally unique. Figure 27-9 illustrates this IPv4-mapped IPv6 address structure.

**Figure 27-9 IPv4-Mapped IPv6 Address**




## Multicast

The second major classification of IPv6 address types in Figure 27-2 is multicast. Multicast is a technique by which a device sends a single packet to multiple destinations simultaneously. An IPv6 multicast address defines a group of devices known as a *multicast group* and is equivalent to IPv4 224.0.0.0/4. IPv6 multicast addresses have the prefix FF00::/8.

Two types of IPv6 multicast addresses are used:

- Assigned multicast
- Solicited-node multicast

### Assigned Multicast

Assigned multicast addresses are used in context with specific protocols.

Two common IPv6 assigned multicast groups include the following:

- **FF02::1 All-nodes multicast group:** This is a multicast group that all IPv6-enabled devices join. As with a broadcast in IPv4, all IPv6 interfaces on the link process packets sent to this address. For example, a router sending an ICMPv6 Router Advertisement (RA) uses the all-nodes FF02::1 address. IPv6-enabled devices can then use the RA information to learn the link's address information, such as prefix, prefix length, and default gateway.
- **FF02::2 All-routers multicast group:** This is a multicast group that all IPv6 routers join. A router becomes a member of this group when it is enabled as an IPv6 router with the **ipv6 unicast-routing** global configuration command. A packet sent to this group is received and

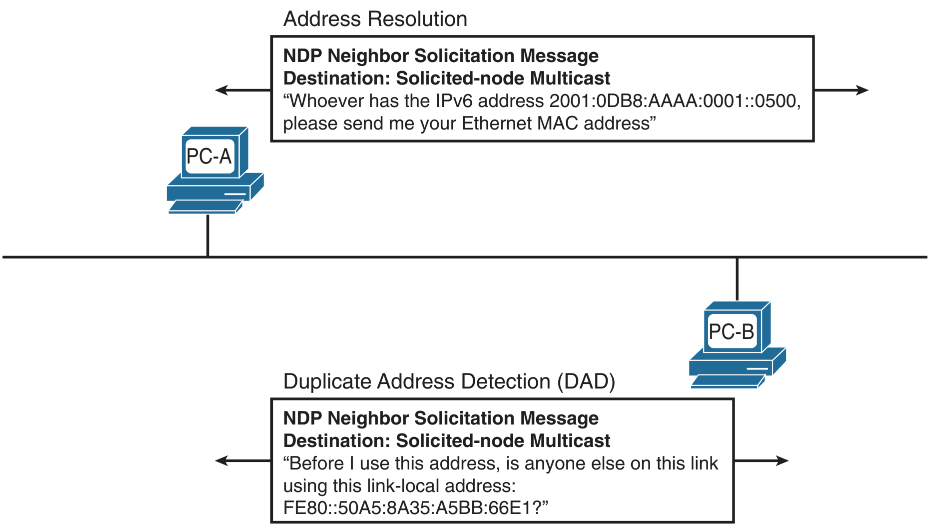
processed by all IPv6 routers on the link or network. For example, IPv6-enabled devices send ICMPv6 Router Solicitation (RS) messages to the all-routers multicast address requesting an RA message.

## Solicited-Node Multicast

In addition to every unicast address assigned to an interface, a device has a special multicast address known as a solicited-node multicast address (refer to Figure 27-2). These multicast addresses are automatically created using a special mapping of the device's unicast address with the solicited-node multicast prefix FF02:0:0:0:0:1:FF00::/104.

As Figure 27-10 shows, solicited-node multicast addresses are used for two essential IPv6 mechanisms, both part of Neighbor Discovery Protocol (NDP):

**Figure 27-10 Uses of Solicited-Node Multicasts**

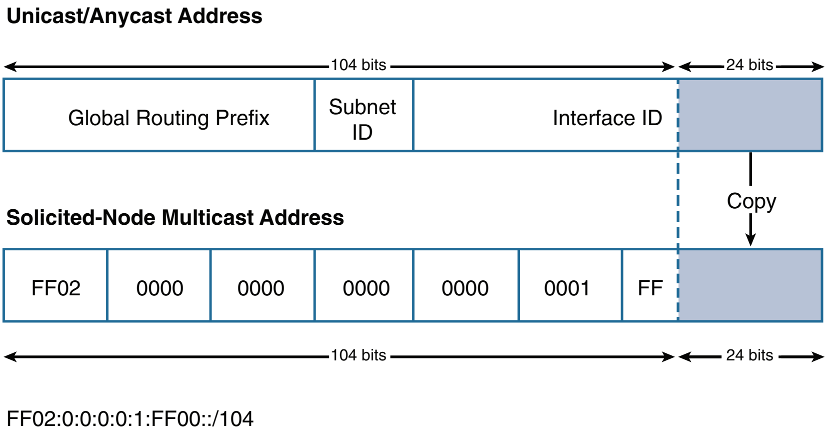



- **Address resolution:** In this mechanism, which is equivalent to ARP in IPv4, an IPv6 device sends an NS message to a solicited-node multicast address to learn the link layer address of a device on the same link. The device recognizes the IPv6 address of the destination on that link but needs to know its data link address.

- **Duplicate address detection (DAD):** As mentioned earlier, DAD allows a device to verify that its unicast address is unique on the link. An NS message is sent to the device's own solicited-node multicast address to determine whether anyone else has this same address.

As Figure 27-11 shows, the solicited-node multicast address consists of two parts:

**Figure 27-11 Solicited-Node Multicast Address Structure**

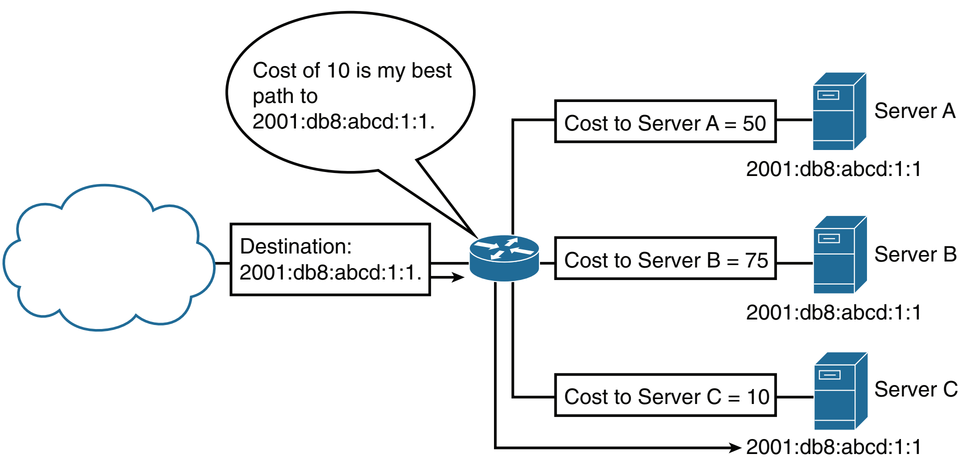



- **FF02:0:0:0:0:FF00::/104 multicast prefix:** This is the first 104 bits of the all solicited-node multicast address.
- **Least significant 24 bits:** These bits are copied from the far-right 24 bits of the global unicast or link-local unicast address of the device.

## Anycast

The last major classification of IPv6 address types in Figure 27-2 is the anycast address. An anycast address can be assigned to more than one device or interface. A packet sent to an anycast address is routed to the "nearest" device that is configured with the anycast address, as Figure 27-12 shows.

**Figure 27-12 Example of Anycast Addressing**



# Representing the IPv6 Address

An IPv6 address can look rather intimidating to someone who is used to IPv4 addressing. However, an IPv6 address can be easier to read and is much simpler to subnet than IPv4.

## Conventions for Writing IPv6 Addresses

IPv6 conventions use 32 hexadecimal numbers, organized into eight hextets of four hex digits separated by colons, to represent a 128-bit IPv6 address. For example:

2340:1111:AAAA:0001:1234:5678:9ABC

To make things a little easier, two rules allow you to shorten what must be configured for an IPv6 address:

- **Rule 1:** Omit the leading 0s in any given hextet.
- **Rule 2:** Omit the all-0s hextets. Represent one or more consecutive hextets of all hex 0s with a double colon (::), but only for one such occurrence in a given address.

For example, in the following address, the highlighted hex digits represent the portion of the address that can be abbreviated:

FE00:0000:0000:0001:0000:0000:0000:0056

This address has two locations in which one or more hextets have four hex 0s, so two main options work for abbreviating this address with the :: abbreviation in one of the locations. The following two options show the two briefest valid abbreviations:

- FE00::1:0:0:0:56
- FE00:0:0:1::56

In the first example, the second and third hextets preceding 0001 were replaced with ::. In the second example, the fifth, sixth, and seventh hextets were replaced with ::. In particular, note that the :: abbreviation, meaning "one or more hextets of all 0s," cannot be used twice because that would be ambiguous. Therefore, the abbreviation FE00::1::56 would not be valid.

## Conventions for Writing IPv6 Prefixes

An IPv6 prefix represents a range or block of consecutive IPv6 addresses. The number that represents the range of addresses, called a *prefix*, is usually seen in IP routing tables, just as you see IP subnet numbers in IPv4 routing tables.

As with IPv4, when writing or typing a prefix in IPv6, the bits past the end of the prefix length are all binary 0s. The following IPv6 address is an example of an address assigned to a host:

2000:1234:5678:9ABC:1234:5678:9ABC:1111/64

The prefix in which this address resides is as follows:

2000:1234:5678:9ABC:**0000:0000:0000:0000**/64

When abbreviated, this is:

2000:1234:5678:9ABC::/64

If the prefix length does not fall on a hextet boundary (that is, is not a multiple of 16), the prefix value should list all the values in the last hextet. For example, assume that the prefix length in the previous example is /56. By convention, the rest of the fourth hextet is written, after being set to binary 0s, as follows:

2000:1234:5678:9A**00**::/56

The following list summarizes some key points about how to write IPv6 prefixes:

- The prefix has the same value as the IP addresses in the group for the first number of bits, as defined by the prefix length.
- Any bits after the prefix length number of bits are binary 0s.
- The prefix can be abbreviated with the same rules as for IPv6 addresses.
- If the prefix length is not on a hextet boundary, write down the value for the entire hextet.

Table 27-4 shows several sample prefixes, their formats, and a brief explanation.

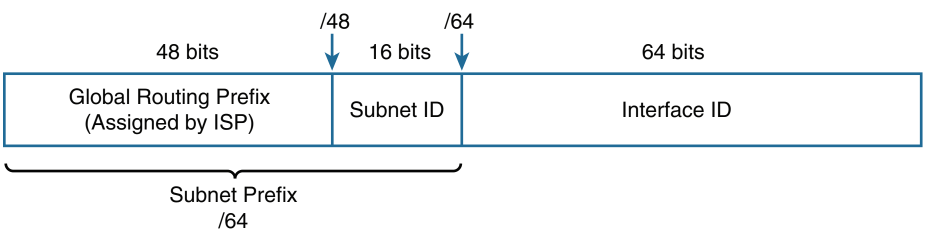
**Table 27-4 Example IPv6 Prefixes and Their Meanings**

| Prefix | Explanation | Incorrect Alternative |
|---|---|---|
| 2000::/3 | All addresses whose first 3 bits are equal to the first 3 bits of hex number 2000 (bits are 001) | 2000/3 (omits ::) 2::/3 (omits the rest of the first hextet) |
| 2340:1140::/26 | All addresses whose first 26 bits match the listed hex number | 2340:114::/26 (omits the last digit in the second hextet) |
| 2340:1111::/32 | All addresses whose first 32 bits match the listed hex number | 2340:1111/32 (omits ::) |

# IPv6 Subnetting

In many ways, subnetting IPv6 addresses is much simpler than subnetting IPv4 addresses. A typical site is assigned an IPv6 address space with a /48 prefix length. Because the least significant bits are used for the interface ID, that leaves 16 bits for the subnet ID and a /64 subnet prefix length, as Figure 27-13 shows.

**Figure 27-13 /64 Subnet Prefix**




For our subnetting examples, we use 2001:0DB8:000A::/48, or simply 2001:DB8:A::/48, which includes subnets 2001:DB8:A::/64 through 2001:DB8:A:FFFF::/64. That's $2^{16}$, or 65,536 subnets, each with $2^{64}$, or 18 quintillion, interface addresses.

## Subnetting the Subnet ID

To subnet in a small to medium-size business, simply increment the least significant bits of the subnet ID (as in Example 27-1) and assign /64 subnets to your networks.

**Example 27-1 Subnetting the Subnet ID**

```
2001:DB8:A:0001::/64
2001:DB8:A:0002::/64
2001:DB8:A:0003::/64
2001:DB8:A:0004::/64
2001:DB8:A:0005::/64
```

Of course, if you are administering a larger implementation, you can use the four hexadecimal digits of the subnet ID to design a quick and simple four-level hierarchy. Most large enterprise networks have plenty of room to design a logical address scheme that aggregates addresses for an optimal routing configuration. In addition, applying for and receiving another /48 address is not difficult.

## Subnetting into the Interface ID

If you extend your subnetting into the interface ID portion of the address, it is a best practice to subnet on the nibble boundary. A nibble is 4 bits, or one hexadecimal digit. For example, let's borrow the first 4 bits from the interface ID portion of the network address 2001:DB8:A:1::/64. That means the network 2001:DB8:A:1::/64 would now have $2^4$, or 16, subnets from 2001:DB8:A:1:0000::/68 to 2001:DB8:A:1:F000::/68. Listing the subnets is easy, as Example 27-2 shows.

**Example 27-2 Subnetting into the Interface ID**

```
2001:DB8:A:1:0000::/68
2001:DB8:A:1:1000::/68
2001:DB8:A:1:2000::/68
2001:DB8:A:1:3000::/68
      thru
2001:DB8:A:1:F000::/68
```

# EUI-64 Concept

Day 18 reviews static IPv6 addressing, including how to configure a router to use EUI-64 addressing (EUI stands for Extended Unique Identifier). Today we are reviewing the concept behind the EUI-64 configuration.

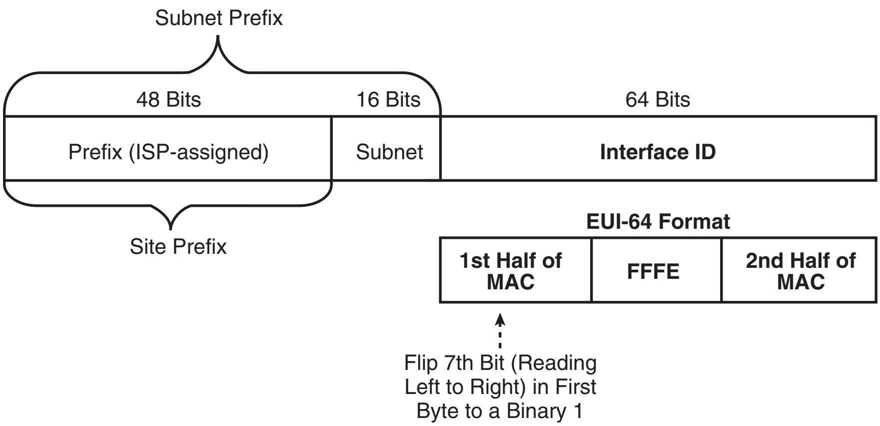
Recall from Figure 27-13 that the second half of the IPv6 address is called the *interface ID*. The value of the interface ID portion of a global unicast address can be set to any value, as long as no other host in the same subnet attempts to use the same value. However, the size of the interface ID was chosen to allow easy autoconfiguration of IP addresses by plugging the MAC address of a network card into the interface ID field in an IPv6 address.

MAC addresses are 6 bytes (48 bits) in length. To complete the 64-bit interface ID, IPv6 fills in 2 more bytes by separating the MAC address into two 3-byte halves. It then inserts hex FFFE between the halves and sets the seventh bit in the first byte to binary 1 to form the interface ID field. Figure 27-14 shows this format, called the EUI-64 format.

For example, the following two lines list a host's MAC address and corresponding EUI-64 format interface ID, assuming the use of an address configuration option that uses the EUI-64 format:

- **MAC address:** 0034:5678:9ABC
- **EUI-64 interface ID:** 0234:56FF:FE78:9ABC

**Figure 27-14 IPv6 Address Format with Interface ID and EUI-64**




**NOTE:** To change the seventh bit (reading left to right) in the example, convert hex 00 to binary 00000000, change the seventh bit to 1 (00000010), and then convert back to hex, for hex 02 as the first two digits.

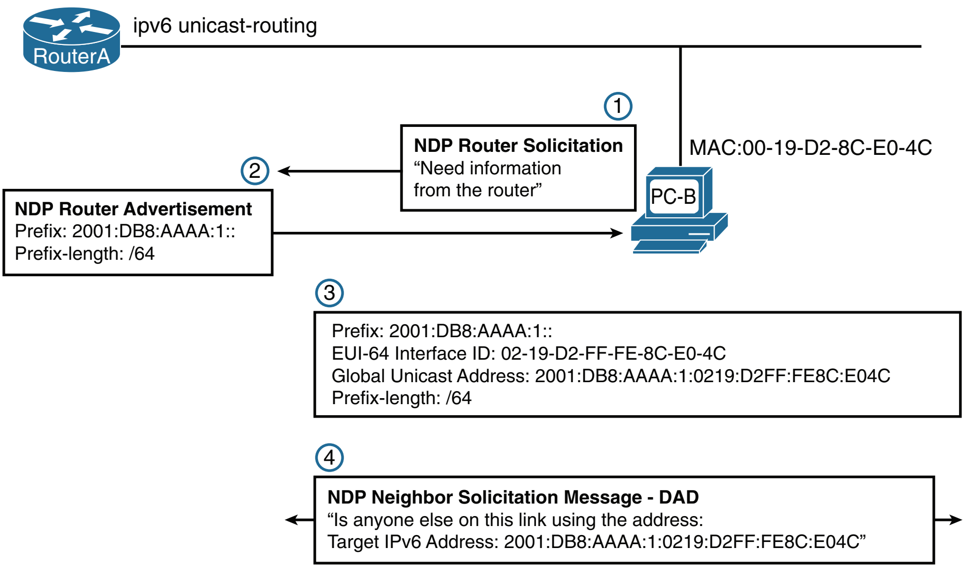
# Stateless Address Autoconfiguration

IPv6 supports two methods of dynamic configuration of IPv6 addresses:

- **Stateless address autoconfiguration (SLAAC):** A host dynamically learns the /64 prefix through the IPv6 Neighbor Discovery Protocol (NDP) and then calculates the rest of its address by using the EUI-64 method.
- **DHCPv6:** This works the same conceptually as DHCP in IPv4. We review DHCPv6 on Day 23, "DHCP and DNS."

By using the EUI-64 process and Neighbor Discovery Protocol (NDP), SLAAC allows a device to determine its entire global unicast address without any manual configuration and without a DHCPv6 server. Figure 27-15 illustrates the SLAAC process between a host and a router configured with the **ipv6 unicast-routing** command, which means it will send and receive NDP messages.

**Figure 27-15 Neighbor Discovery and the SLAAC Process**




# Migration to IPv6

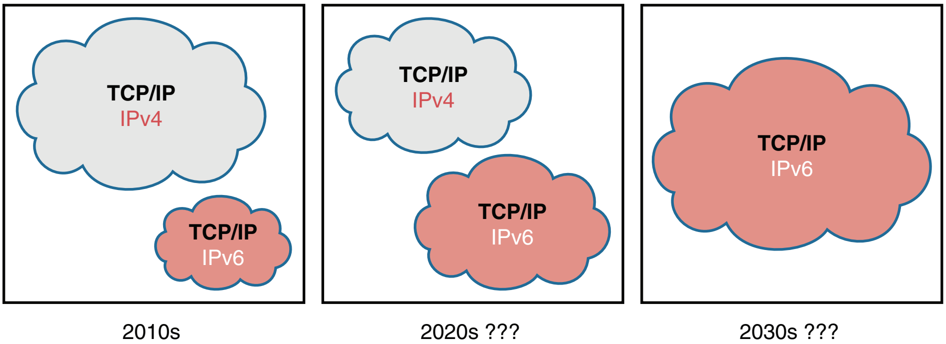
Two major transition strategies are currently used to migrate to IPv6:

- **Dual-stacking:** In this integration method, a node has implementation and connectivity to both an IPv4 network and an IPv6 network. This is the recommended option and involves running IPv4 and IPv6 at the same time.
- **Tunneling:** Tunneling is a method for transporting IPv6 packets over IPv4-only networks by encapsulating the IPv6 packet inside IPv4. Several tunneling techniques are available.

Because of the simplicity of running dual-stacking, it will most likely be the preferred strategy as IPv4-only networks begin to disappear. But it will probably still be decades before we see enterprise networks running exclusively IPv6. Figure 27-16 illustrates one way Wendell Odom thinks about the transition to IPv6: "But who knows how long it will take?"

Remember this advice: "Dual-stack where you can; tunnel where you must." These two methods are the most common techniques to transition from IPv4 to IPv6. Dual-stacking is easy enough: Just configure all your devices to use both IPv4 and IPv6 addressing. Tunneling is more complex and beyond the scope of the CCNA exam topics.

**Figure 27-16 Transition to IPv6 Using Dual-Stacking**




# Study Resources

For today's exam topics, refer to the following resources for more study.

| Resource | Module or Chapter |
|---|---|
| Cisco Network Academy: CCNA 1 | 4 |
| | 6 |
| | 7 |
| CCNA 200-301 Official Cert Guide, Volume 1 | 5 |
| | 8 |
| Portable Command Guide | 5 |

*This page intentionally left blank*

# Day 26

# VLAN and Trunking Concepts and Configurations

## CCNA 200-301 Exam Topics

- Configure and verify VLANs (normal range) spanning multiple switches
- Configure and verify interswitch connectivity

## Key Points

Most large networks today implement virtual local-area networks (VLANs). Without VLANs, a switch considers every port to be in the same broadcast domain. With VLANs, switch ports can be grouped into different VLANs, essentially segmenting the broadcast domain. Today we review VLAN concepts, consider traffic types, discuss VLAN types, and review the concept of trunking, including Dynamic Trunking Protocol (DTP). Then we review the commands to configure and verify VLANs, trunking, and inter-VLAN routing.

## VLAN Concepts

Although a switch comes out of the box with only one VLAN, normally a switch is configured to have two or more VLANs. With such a switch, you can create multiple broadcast domains by putting some interfaces into one VLAN and other interfaces into other VLANs.

Consider these reasons for using VLANs:

- Grouping users by department instead of by physical location
- Segmenting devices into smaller LANs to reduce processing overhead for all devices on the LAN
- Reducing the workload of STP by limiting a VLAN to a single access switch
- Enforcing better security by isolating sensitive data to separate VLANs
- Separating IP voice traffic from data traffic
- Assisting troubleshooting by reducing the size of the failure domain (that is, the number of devices that can cause a failure or that can be affected by one)

Benefits of using VLANs include the following:

- **Security:** Sensitive data can be isolated to one VLAN, separated from the rest of the network.
- **Cost reduction:** Reduced need for expensive network upgrades and more efficient use of existing bandwidth and uplinks lead to cost savings.
- **Higher performance:** Dividing flat Layer 2 networks into multiple logical broadcast domains reduces unnecessary traffic on the network and boosts performance.

- **Broadcast storm mitigation:** VLAN segmentation prevents broadcast storms from propagating throughout the entire network.
- **Ease of management and troubleshooting:** A hierarchical addressing scheme groups network addresses contiguously. Because a hierarchical IP addressing scheme makes problematic components easier to locate, network management and troubleshooting are more efficient.

## Traffic Types

The key to successful VLAN deployment is understanding the traffic patterns and the various traffic types in the organization. Table 26-1 lists the common types of network traffic to evaluate before placing devices and configuring VLANs.

**Table 26-1 Traffic Types**

| Traffic Type | Description |
|---|---|
| Network management | Many types of network management traffic can be present on the network. To make network troubleshooting easier, some designers assign a separate VLAN to carry certain types of network management traffic. |
| IP telephony | Two types of IP telephony traffic exist: signaling information between end devices and the data packets of the voice conversation. Designers often configure the data to and from the IP phones on a separate VLAN designated for voice traffic so that they can apply quality-of-service measures to give high priority to voice traffic. |
| IP multicast | Multicast traffic can produce a large amount of data streaming across the network. Switches must be configured to keep this traffic from flooding to devices that have not requested it, and routers must be configured to ensure that multicast traffic is forwarded to the network areas where it is requested. |
| Normal data | Normal data traffic is typical application traffic that is related to file and print services, email, Internet browsing, database access, and other shared network applications. |
| Scavenger class | Scavenger class includes all traffic with protocols or patterns that exceed their normal data flows. Applications assigned to this class have little or no contribution to the organizational objectives of the enterprise and are typically entertainment oriented. |

## Types of VLANs

Some VLAN types are defined by the type of traffic they support; others are defined by the specific functions they perform. The principal VLAN types and their descriptions follow:

- **Data VLAN:** Configured to carry only user-generated traffic, ensuring that voice and management traffic is separated from data traffic.
- **Default VLAN:** All the ports on a switch are members of the default VLAN when the switch is reset to factory defaults. The default VLAN for Cisco switches is VLAN 1. VLAN 1 has all the features of any VLAN, except that you cannot rename it, and you cannot delete it. It is a security best practice to restrict VLAN 1 to serve as a conduit only for Layer 2 control traffic (for example, CDP) and support no other traffic.
- **Black hole VLAN:** A security best practice is to define a black hole VLAN to be a dummy VLAN distinct from all other VLANs defined in the switched LAN. All unused switch ports

are assigned to the black hole VLAN so that any unauthorized device connecting to an unused switch port is prevented from communicating beyond the switch to which it is connected.

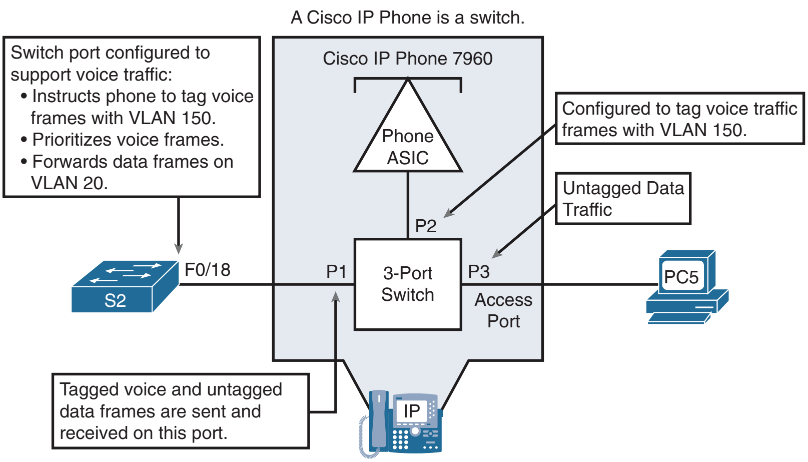
- **Native VLAN:** This VLAN type serves as a common identifier on opposing ends of a trunk link. A security best practice is to define a native VLAN to be a dummy VLAN distinct from all other VLANs defined in the switched LAN. The native VLAN is not used for any traffic in the switched network unless legacy bridging devices happen to be present in the network or a multiaccess interconnection exists between switches joined by a hub.
- **Management VLAN:** The network administrator defines this VLAN as a means to access the management capabilities of a switch. By default, VLAN 1 is the management VLAN. It is a security best practice to define the management VLAN to be a VLAN distinct from all other VLANs defined in the switched LAN. You do this by configuring and activating a new VLAN interface.
- **Voice VLANs:** A voice VLAN enables switch ports to carry IP voice traffic from an IP phone. The network administrator configures a voice VLAN and assigns it to access ports. Then when an IP phone is connected to the switch port, the switch sends CDP messages that instruct the attached IP phone to send voice traffic tagged with the voice VLAN ID.

## Voice VLAN Example

Figure 26-1 shows an example of using one port on a switch to connect a user's IP phone and PC. The switch port is configured to carry data traffic on VLAN 20 and voice traffic on VLAN 150. The Cisco IP Phone contains an integrated three-port 10/100 switch to provide the following dedicated connections:

- Port 1 connects to the switch or other VoIP device.
- Port 2 is an internal 10/100 interface that carries the IP Phone traffic.
- Port 3 (access port) connects to a PC or other device.

**Figure 26-1 Cisco IP Phone Switching Voice and Data Traffic**

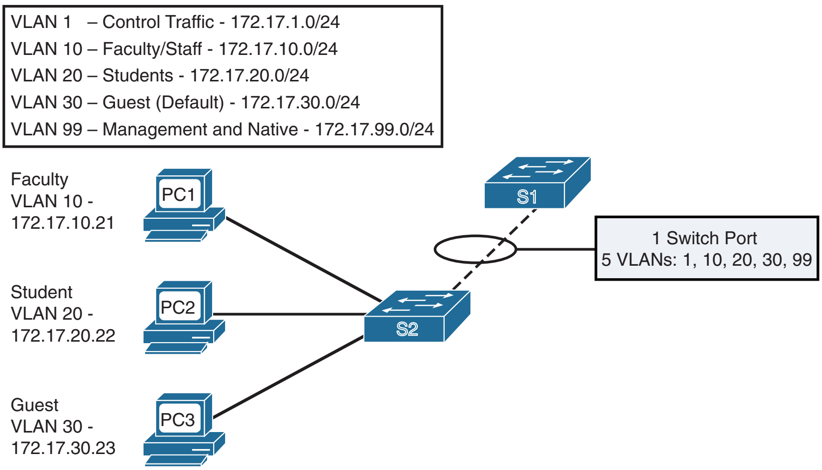



The traffic from PC5 attached to the IP Phone passes through the IP Phone untagged. The link between S2 and the IP Phone acts as a modified trunk to carry both the tagged voice traffic and the untagged data traffic.

# Trunking VLANs

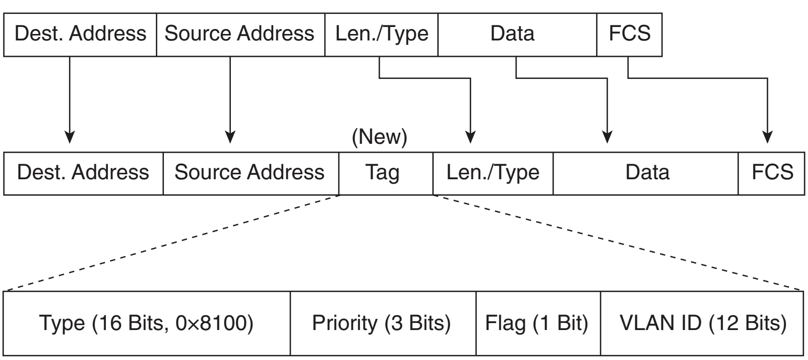
A VLAN trunk is an Ethernet point-to-point link between an Ethernet switch interface and an Ethernet interface on another networking device, such as a router or a switch, carrying the traffic of multiple VLANs over the singular link. A VLAN trunk enables you to extend the VLANs across an entire network. A VLAN trunk does not belong to a specific VLAN; instead, it serves as a conduit for VLANs between switches. Figure 26-2 shows a small switched network with a trunk link between S1 and S2 carrying multiple VLAN traffic.

**Figure 26-2 Example of a VLAN Trunk**




When a frame is placed on a trunk link, information about the VLAN it belongs to must be added to the frame. This is accomplished by using IEEE 802.1Q frame tagging. When a switch receives a frame on a port configured in access mode and destined for a remote device through a trunk link, the switch takes apart the frame and inserts a VLAN tag, recalculates the frame check sequence (FCS), and sends the tagged frame out the trunk port. Figure 26-3 shows the 802.1Q tag inserted in an Ethernet frame.

**Figure 26-3 Fields of the 802.1Q Tag Inside an Ethernet Frame**



The VLAN tag field consists of a 16-bit Type field called the EtherType field and a Tag control information field. The EtherType field is set to the hexadecimal value 0x8100. This value is called the tag protocol ID (TPID) value. With the EtherType field set to the TPID value, the switch receiving the frame knows to look for information in the Tag control information field. The Tag control information field contains the following:

- **3 bits of user priority:** Provides expedited transmission of Layer 2 frames, such as voice traffic
- **1 bit of Canonical Format Identifier (CFI):** Enables Token Ring frames to be easily carried across Ethernet links
- **12 bits of VLAN ID (VID):** Provides VLAN identification numbers

**NOTE:** Although 802.1Q is the recommended method for tagging frames, you should be aware of the Cisco proprietary legacy trunking protocol called Inter-Switch Link (ISL).

## Dynamic Trunking Protocol

Dynamic Trunking Protocol (DTP) is a Cisco-proprietary protocol that negotiates both the status of trunk ports and the trunk encapsulation of trunk ports. DTP manages trunk negotiation only if the port on the other switch is configured in a trunk mode that supports DTP. A switch port on a Cisco Catalyst switch supports a number of trunking modes. The trunking mode defines how the port negotiates using DTP to set up a trunk link with its peer port. The following is a brief description of each trunking mode:

- If the switch is configured with the **switchport mode trunk** command, the switch port periodically sends DTP messages to the remote port, advertising that it is in an unconditional trunking state.
- If the switch is configured with the **switchport mode trunk dynamic auto** command, the local switch port advertises to the remote switch port that it is able to trunk but does not request to go to the trunking state. After a DTP negotiation, the local port ends up in the trunking state only if the remote port trunk mode has been configured so that the status is **on** or **desirable**. If both ports on the switches are set to **auto**, they do not negotiate to be in a trunking state. They negotiate to be in the access mode state.
- If the switch is configured with the **switchport mode dynamic desirable** command, the local switch port advertises to the remote switch port that it is able to trunk and asks the remote switch port to go to the trunking state. If the local port detects that the remote port has been configured as **on**, **desirable**, or **auto** mode, the local port ends up in the trunking state. If the remote switch port is in the **nonegotiate** mode, the local switch port remains as a nontrunking port.
- If the switch is configured with the **switchport nonegotiate** command, the local port is considered to be in an unconditional trunking state. Use this feature when you need to configure a trunk with a switch from another switch vendor.

Table 26-2 summarizes the results of DTP negotiations based on the different DTP configuration commands on local and remote ports.

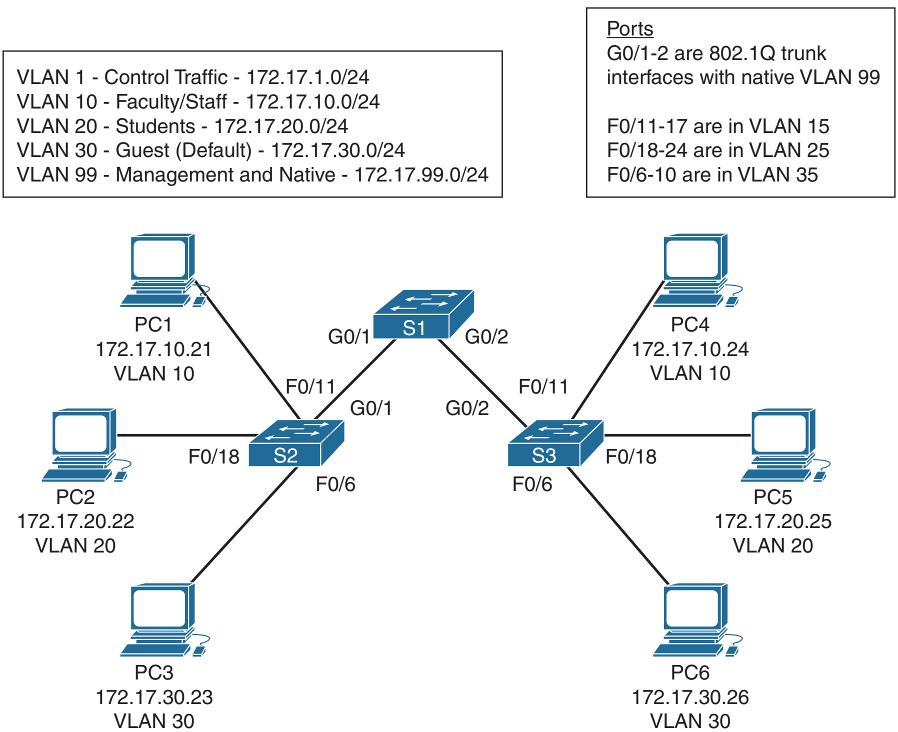
**Table 26-2 Trunk Negotiation Results Between a Local Port and a Remote Port**

| | Dynamic Auto | Dynamic Desirable | Trunk | Access |
|---|---|---|---|---|
| Dynamic Auto | Access | Trunk | Trunk | Access |
| Dynamic Desirable | Trunk | Trunk | Trunk | Access |
| Trunk | Trunk | Trunk | Trunk | Not recommended |
| Access | Access | Access | Not recommended | Access |

# VLAN Configuration and Verification

Refer to the topology in Figure 26-4 as you review the commands in this section for configuring, verifying, and troubleshooting VLAN and trunking. The packet tracer activity later in the day uses this same topology.

**Figure 26-4 Day 26 Sample Topology**




The default configuration of a Cisco switch is to put all interfaces in VLAN 1. You can verify this with the **show vlan brief** command, as demonstrated for S2 in Example 26-1.

**Example 26-1 Default VLAN Configuration**

```
S2# show vlan brief

VLAN Name Status Ports
---- ------------------------------- --------- ------------------------------
1    default              active      Fa0/1, Fa0/2, Fa0/3, Fa0/4
                                      Fa0/5, Fa0/6, Fa0/7, Fa0/8
                                      Fa0/9, Fa0/10, Fa0/11, Fa0/12
                                      Fa0/13, Fa0/14, Fa0/15, Fa0/16
                                      Fa0/17, Fa0/18, Fa0/19, Fa0/20
                                      Fa0/21, Fa0/22, Fa0/23, Fa0/24
                                      Gig0/1, Gig0/2
1002 fddi-default         active
1003 token-ring-default   active
1004 fddinet-default      active
1005 trnet-default        active
S2#
```

A VLAN is created in one of two ways: either in global configuration mode or directly under the interface. The advantage to configuring in global configuration mode is that you can then assign a name with the **name** *vlan-name* command. The advantage to configuring the VLAN in interface configuration mode is that you assign the VLAN to the interface and create the VLAN with just one command. However, to name the VLAN, you still have to go back to the global configuration method. Example 26-2 shows the creation of VLANs 10 and 20 using these two methods. VLAN 20 is then named, and the remaining VLANs are created in global configuration mode.

**Example 26-2 Creating VLANs**

```
S2# config t
Enter configuration commands, one per line. End with CNTL/Z.
S2(config)# vlan 10
S2(config-vlan)# name Faculty/Staff
S2(config-vlan)# interface fa0/18
S2(config-if)# switchport access vlan 20
% Access VLAN does not exist. Creating vlan 20
S2(config-if)# vlan 20
S2(config-vlan)# name Students
S2(config-vlan)# vlan 30
S2(config-vlan)# name Guest(Default)
S2(config-vlan)# vlan 99
S2(config-vlan)# name Management&Native
S2(config-vlan)# end
%SYS-5-CONFIG_I: Configured from console by console
S2#
```

Notice in Example 26-3 that all the VLANs are created, but only VLAN 20 is assigned to an interface.

**Example 26-3 Verifying VLAN Creation**

```
S2# show vlan brief

VLAN Name                        Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                     active    Fa0/1, Fa0/2, Fa0/3, Fa0/4
                                           Fa0/5, Fa0/6, Fa0/7, Fa0/8
                                           Fa0/9, Fa0/10, Fa0/11, Fa0/12
                                           Fa0/13, Fa0/14, Fa0/15, Fa0/16
                                           Fa0/17, Fa0/19, Fa0/20, Fa0/21
                                           Fa0/22, Fa0/23, Fa0/24, Gig1/1
                                           Gig1/2
10   Faculty/Staff               active
20   Students                    active    Fa0/18
30   Guest(Default)              active
99   Management&Native           active
1002 fddi-default                active
1003 token-ring-default          active
1004 fddinet-default             active
1005 trnet-default               active
S2#
```

To assign the remaining interfaces to the VLANs specified in Figure 26-4, either you can configure one interface at a time or you can use the **range** command to configure all the interfaces that belong to a VLAN with one command, as shown in Example 26-4.

**Example 26-4 Assigning VLANs to Interfaces**

```
S2# config t
Enter configuration commands, one per line. End with CNTL/Z.
S2(config)# interface range fa 0/11 - 17
S2(config-if-range)# switchport access vlan 10
S2(config-if-range)# interface range fa 0/18 - 24
S2(config-if-range)# switchport access vlan 20
S2(config-if-range)# interface range fa 0/6 - 10
S2(config-if-range)# switchport access vlan 30
S2(config-if-range)# end
%SYS-5-CONFIG_I: Configured from console by console
S2#
```

The **show vlan brief** command in Example 26-5 verifies that all interfaces specified in Figure 26-4 have been assigned to the appropriate VLAN. Notice that unassigned interfaces still belong to the default VLAN 1.

**Example 26-5 Verifying VLAN Assignments to Interfaces**

```
S2# show vlan brief

VLAN Name Status Ports
---- -------------------------------- --------- -------------------------------
1 default                    active Fa0/1, Fa0/2, Fa0/3, Fa0/4
 Fa0/5, Gig0/1, Gig0/2
10 Faculty/Staff             active Fa0/11, Fa0/12, Fa0/13, Fa0/14
 Fa0/15, Fa0/16, Fa0/17
20 Students                  active Fa0/18, Fa0/19, Fa0/20, Fa0/21
                                        Fa0/22, Fa0/23, Fa0/24
30 Guest(Default)            active Fa0/6, Fa0/7, Fa0/8, Fa0/9
                                        Fa0/10
99 Management&Native         active
1002 fddi-default            active
1003 token-ring-default      active
1004 fddinet-default         active
1005 trnet-default           active
S2#
```

You can also verify a specific interface's VLAN assignment with the **show interfaces** *type number* **switchport** command, as shown for FastEthernet 0/11 in Example 26-6.

**Example 26-6 Verifying an Interface's VLAN Assignment**

```
S2# show interfaces fastethernet 0/11 switchport
Name: Fa0/11
Switchport: Enabled
Administrative Mode: dynamic auto
Operational Mode: static access
Administrative Trunking Encapsulation: dot1q
Operational Trunking Encapsulation: native
Negotiation of Trunking: On
Access Mode VLAN: 10 (Faculty/Staff)
Trunking Native Mode VLAN: 1 (default)
Voice VLAN: none
Administrative private-vlan host-association: none
Administrative private-vlan mapping: none
```

```
Administrative private-vlan trunk native VLAN: none
Administrative private-vlan trunk encapsulation: dot1q
Administrative private-vlan trunk normal VLANs: none
Administrative private-vlan trunk private VLANs: none
Operational private-vlan: none
Trunking VLANs Enabled: ALL
Pruning VLANs Enabled: 2-1001
Capture Mode Disabled
Capture VLANs Allowed: ALL
Protected: false
Appliance trust: none
S2#
```

For the sample topology shown in Figure 26-4, you would configure the VLANs on S1 and S3 as well, but only S3 needs VLANs assigned to interfaces.

# Trunking Configuration and Verification

Following security best practices, we are configuring a different VLAN for the management and default VLAN. In a production network, you would want to use a different one for each: one for the management VLAN and one for the native VLAN. For expediency here, we are using VLAN 99 for both.

We first define a new management interface for VLAN 99, as in Example 26-7.

**Example 26-7 Defining a New Management Interface**

```
S1# config t
Enter configuration commands, one per line. End with CNTL/Z.
S1(config)# interface vlan 99
%LINK-5-CHANGED: Interface Vlan99, changed state to up
S1(config-if)# ip address 172.17.99.31 255.255.255.0
S1(config-if)# end
%SYS-5-CONFIG_I: Configured from console by console
S1#
```

Then we repeat the configuration on S2 and S3. The IP address is used to test connectivity to the switch, as is the IP address the network administrator uses for remote access (Telnet, SSH, SDM, HTTP, and so on).

Depending on the switch model and Cisco IOS version, DTP might have already established trunking between two switches that are directly connected. For example, the default trunk configuration for 2950 switches is **dynamic desirable**. Therefore, a 2950 initiates trunk negotiations. For our purposes, assume that the switches are all 2960s. The 2960 default trunk configuration is **dynamic auto**, and in this configuration, the interface does not initiate trunk negotiations.

In Example 26-8, the first five interfaces on S1 are configured for trunking. Also notice that the native VLAN is changed to VLAN 99.

**Example 26-8 Trunk Configuration and Native VLAN Assignment**

```
S1# config t
Enter configuration commands, one per line. End with CNTL/Z.
S1(config)# interface range g0/1 - 2
S1(config-if-range)# switchport mode trunk
S1(config-if-range)# switchport trunk native vlan 99
S1(config-if-range)# end
%SYS-5-CONFIG_I: Configured from console by console
S1#
%CDP-4-NATIVE_VLAN_MISMATCH: Native VLAN mismatch discovered on FastEthernet0/1
   (99), with S2 FastEthernet0/1 (1).
%CDP-4-NATIVE_VLAN_MISMATCH: Native VLAN mismatch discovered on FastEthernet0/3
   (99), with S3 FastEthernet0/3 (1).
```

If you wait for the next round of CDP messages, you should get the error message shown in Example 26-8. Although the trunk is working between S1 and S2 and between S1 and S3, the switches do not agree on the native VLAN. Repeat the trunking commands on S2 and S3 to correct the native VLAN mismatch.

**NOTE:** The encapsulation type—dot1q or isl—might need to be configured, depending on the switch model. The syntax for configuring the encapsulation type is as follows:

```
Switch(config-if)# switchport trunk encapsulation { dot1q | isl | negotiate }
```

The 2960 Series supports only 802.1Q, so this command is not available.

To verify that trunking is operational, use the commands in Example 26-9.

**Example 26-9 Verifying Trunk Configuration**

```
S1# show interfaces trunk
Port     Mode    Encapsulation  Status     Native vlan
Gig0/1   on      802.1q         trunking   99
Gig0/2   on      802.1q         trunking   99

Port     Vlans allowed on trunk
Gig0/1   1-1005
Gig0/2   1-1005
Port     Vlans allowed and active in management domain
Gig0/1   1,10,20,30,99
Gig0/2   1,10,20,30,99
```

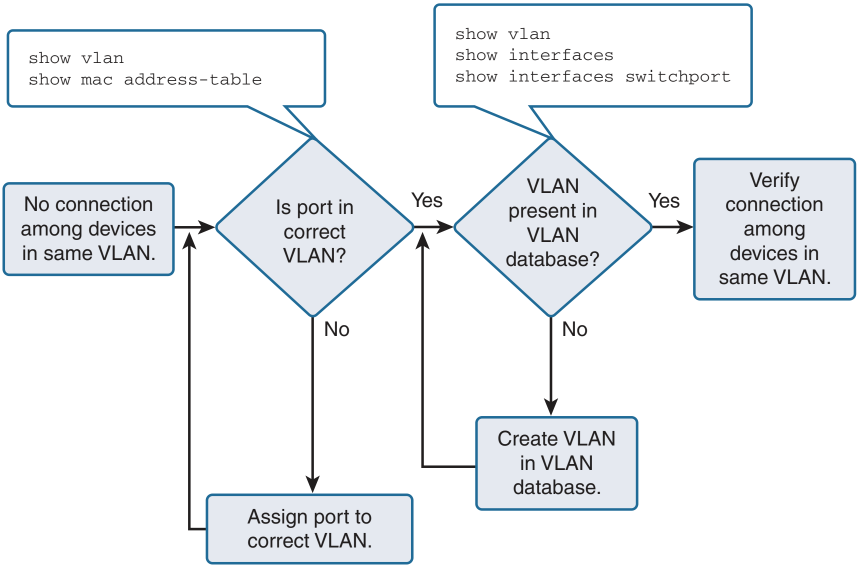
```
Port     Vlans in spanning tree forwarding state and not pruned
Gig0/1  1,10,20,30,99
Gig0/2  1,10,20,30,99

S1# show interface g0/1 switchport
Name: Gig0/1
Switchport: Enabled
Administrative Mode: trunk
Operational Mode: trunk
Administrative Trunking Encapsulation: dot1q
Operational Trunking Encapsulation: dot1q
Negotiation of Trunking: On
Access Mode VLAN: 1 (default)
Trunking Native Mode VLAN: 99 (Management&Native)
Voice VLAN: none
Administrative private-vlan host-association: none
Administrative private-vlan mapping: none
Administrative private-vlan trunk native VLAN: none
Administrative private-vlan trunk encapsulation: dot1q
Administrative private-vlan trunk normal VLANs: none
Administrative private-vlan trunk private VLANs: none
Operational private-vlan: none
Trunking VLANs Enabled: ALL
Pruning VLANs Enabled: 2-1001
Capture Mode Disabled
Capture VLANs Allowed: ALL
Protected: false
Appliance trust: none
S1#
```

Remember that hosts on the same VLAN must be configured with an IP address and subnet mask on the same subnet. The ultimate test of your configuration, then, is to verify that end devices on the same VLAN can now ping each other. If they can't, use the verification commands to systematically track down the problem with your configuration.

# VLAN Troubleshooting

If connectivity issues arise between VLANs and you have already resolved potential IP addressing issues, you can use the flowchart in Figure 26-5 to methodically track down any issues related to VLAN configuration errors.

**Figure 26-5 VLAN Troubleshooting Flowchart**




The flowchart in Figure 26-5 works in this way:

**Step 1.** Use the **show vlan** command to check whether the port belongs to the expected VLAN. If the port is assigned to the wrong VLAN, use the **switchport access vlan** command to correct the VLAN membership. Use the **show mac address-table** command to check which addresses were learned on a particular port of the switch and see the VLAN to which that port is assigned.

**Step 2.** If the VLAN to which the port is assigned is deleted, the port becomes inactive. Use the **show vlan** or **show interfaces switchport** command to discover issues with deleted VLANs. If the port is inactive, it is not functional until the missing VLAN is created using the **vlan** *vlan_id* command.

Table 26-3 summarizes these commands, which can be particularly helpful in troubleshooting VLAN issues.

**Table 26-3 VLAN Troubleshooting Commands**

| EXEC Command | Description |
|---|---|
| **show vlan**<br>**show vlan brief** | Lists each VLAN and all interfaces assigned to that VLAN (but does not include operational trunks) |
| **show vlan id** *num* | Lists both access and trunk ports in the VLAN |
| **show interfaces switchport**<br>**show interfaces** *type number* **switchport** | Identifies the interface's access VLAN and voice VLAN, the configured and operational mode (access or trunk), and the state of the port (up or down) |
| **show mac address-table** | Lists MAC table entries, including the associated VLAN |
| **show interface status** | Summarizes the status listing for all interfaces (connected, notconnect, err-disabled), the VLAN, duplex, speed, and type of port |

## Disabled VLANs

VLANs can be manually disabled. You can verify that VLANs are active by using the **show vlan** command. As Example 26-10 shows, VLANs can be in one of two states: either *active* or *act/lshut*. The second of these states means that the VLAN is shut down.

**Example 26-10 Enabling and Disabling VLANs on a Switch**

```
S1# show vlan brief
VLAN Name                             Status        Ports
---- -------------------------------- --------- -------------------------------
1 default                        active          Fa0/1, Fa0/2, Fa0/3, Fa0/4
                                                  Fa0/5, Fa0/6, Fa0/7, Fa0/8
                                                  Fa0/9, Fa0/10, Fa0/11, Fa0/12
                                                  Fa0/14, Fa0/15, Fa0/16, Fa0/17
                                                  Fa0/18, Fa0/19, Fa0/20, Fa0/21
                                                  Fa0/22, Fa0/23, Fa0/24, Gi0/1
10 VLAN0010                   act/lshutFa0/13
20 VLAN0020                   active
30 VLAN0030                   act/lshut
40 VLAN0040                   active
S1# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
S1(config)# no shutdown vlan 10
S1(config)# vlan 30
S1(config-vlan)# no shutdown
S1(config-vlan)#
```

The highlighted commands in Example 26-10 show the two configuration methods you can use to enable a VLAN that had been shut down.

# Trunking Troubleshooting

To summarize issues with VLANs and trunking, you need to check for four potential issues, in this order:

**Step 1.** Identify all access interfaces and their assigned access VLANs and reassign them into the correct VLANs, as needed.

**Step 2.** Determine whether the VLANs exist and are active on each switch. If needed, configure and activate the VLANs to resolve problems.

**Step 3.** Check the allowed VLAN lists on the switches on both ends of the trunk and ensure that the lists of allowed VLANs are the same.

**Step 4.** Ensure that, for any links that should use trunking, one switch does not think it is trunking, while the other switch does not think it is trunking.

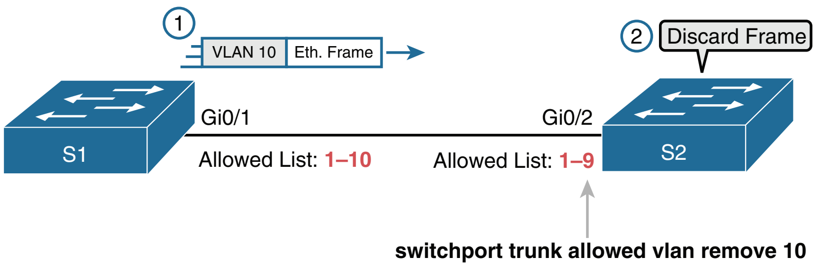
The previous section reviewed steps 1 and 2. Next, we review steps 3 and 4.

## Check Both Ends of a Trunk

For the CCNA exam, you should be ready to notice a couple oddities that happen with some unfortunate configuration choices on trunks.

It is possible to configure a different allowed VLAN list on the opposite ends of a VLAN trunk. As Figure 26-6 shows, when the VLAN lists do not match, the trunk cannot pass traffic for that VLAN.

**Figure 26-6 Mismatched VLAN-Allowed Lists on a Trunk**




You can isolate this problem only by comparing the allowed lists on both ends of the trunk. Example 26-9 displays the output of the **show interfaces trunk** command on S2.

To compare the allowed VLANs on each switch, you need to look at the second of three lists of VLANs listed by the **show interfaces trunk** command. See the output in Example 26-11.

**Example 26-11 Verifying the Allowed VLANs on S2**

```
S2# show interfaces trunk
Port     Mode       Encapsulation Status    Native vlan
Gi0/2    desirable 802.1q         trunking 1

Port     Vlans allowed on trunk
Gi0/2    1-4094

Port      Vlans allowed and active in management domain
Gi0/2     1-9

Port      Vlans in spanning tree forwarding state and not pruned
Gi0/2      1-9
```

To add VLAN 10 to S2's trunk, enter the following commands:

```
S2(config)# interface g0/2
S2(config-if)# switchport trunk allowed vlan add 10
```

The keyword **add** provides the capability to add one or more VLANs to the trunk without having to specify again all the existing VLANs that are already allowed.

## Check Trunking Operational States

Trunks can be misconfigured. In some cases, both switches conclude that their interfaces do not trunk. In other cases, one switch believes that its interface is correctly trunking, while the other switch does not.

The most common incorrect configuration—which results in both switches not trunking—is a configuration that uses the **switchport mode dynamic auto** command on both switches on the link. The keyword **auto** does not mean that trunking happens automatically. Instead, both switches passively wait on the other device on the link to begin negotiations.

With this particular incorrect configuration, the **show interfaces switchport** command on both switches confirms both the administrative state (auto) and the fact that both switches operate as static access ports. Example 26-12 highlights those parts of the output for S2.

**Example 26-12 Verifying the Trunking State for a Specific Interface**

```
SW2# show interfaces gigabit0/2 switchport
Name: Gi0/2
Switchport: Enabled
Administrative Mode: dynamic auto
Operational Mode: static access
Administrative Trunking Encapsulation: dot1q
Operational Trunking Encapsulation: native
! lines omitted for brevity
```

Always check the trunk's operational state on both sides of the trunk. The best commands for checking trunking-related facts are **show interfaces trunk** and **show interfaces switchport**.

# Study Resources

For today's exam topics, refer to the following resources for more study.

| Resource | Module or Chapter |
|---|---|
| Cisco Network Academy: CCNA 2 | 3 |
| CCNA 200-301 Official Cert Guide, Volume 1 | 8 |
| Portable Command Guide | 9 |
| | 10 |

# Day 25

# STP

## CCNA 200-125 Exam Topics

- Configure, verify, and troubleshoot STP protocols
- Configure, verify, and troubleshoot STP-related optional features
- Describe the benefits of switch stacking and chassis aggregation

## Key Topics

Today's review covers the operation and configuration of Spanning Tree Protocol (STP). The original STP IEEE 802.1D standard allowed for only one instance of STP to run for an entire switched network. Today's network administrators can implement Per-VLAN Spanning Tree (PVST) and Rapid STP (RSTP), both of which improve the original standard.

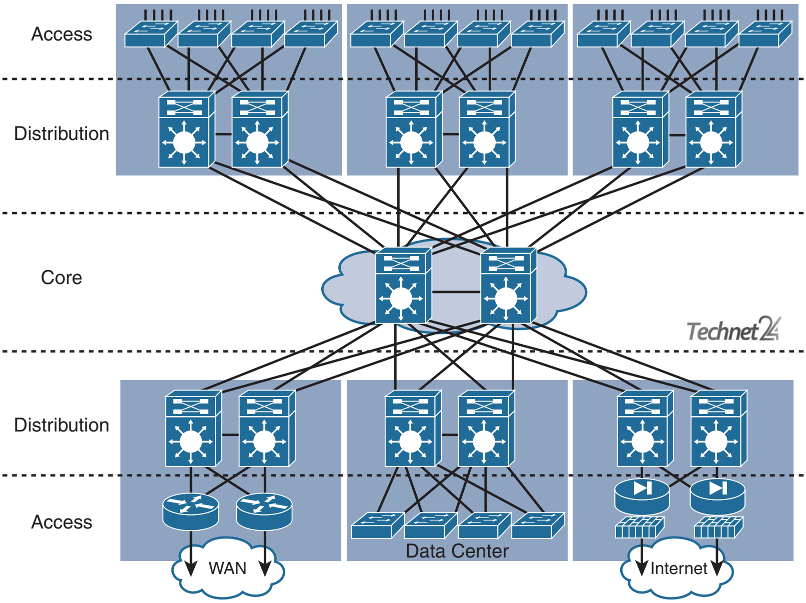
## STP Concepts and Operation

A key characteristic of a well-built communications network is its resiliency. A resilient network is capable of handling a device or link failure through redundancy. A redundant topology can eliminate a single point of failure by using multiple links, multiple devices, or both. STP helps prevent loops in a redundant switched network. Figure 25-1 shows an example of a three-layer topology (core, distribution, access) with redundant links.

Without STP, redundancy in a switched network can introduce the following issues:

- **Broadcast storms:** Each switch floods broadcasts endlessly.
- **Multiple-frame transmission:** Multiple copies of unicast frames are delivered to the destination, causing unrecoverable errors.
- **MAC database instability:** Instability in the content of the MAC address table results from different ports of the switch receiving copies of the same frame.

**Figure 25-1 Redundant Switched Topology**




# STP Algorithm

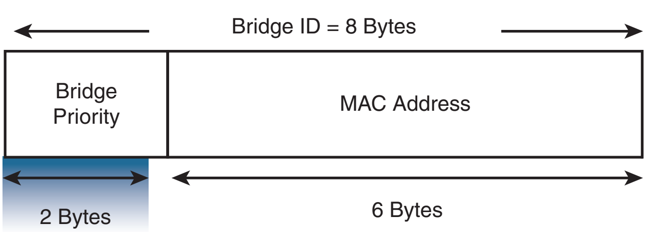
STP is an IEEE Committee standard defined as 802.1D. STP places certain ports in the blocking state so that they do not listen to, forward, or flood data frames. STP creates a tree that ensures that only one path exists for each network segment at any one time. If any segment experiences a disruption in connectivity, STP rebuilds a new tree by activating the previously inactive but redundant path.

The algorithm STP uses chooses the interfaces that should be placed into a forwarding state. For any interfaces not chosen to be in a forwarding state, STP places the interfaces in blocking state.

Switches exchange STP configuration messages every 2 seconds, by default, using a multicast frame called the bridge protocol data unit (BPDU). Blocked ports listen for these BPDUs to detect whether the other side of the link is down, thus requiring an STP recalculation. One piece of information included in the BPDU is the bridge ID (BID).

As Figure 25-2 shows, the BID is unique to each switch. It consists of a priority value (2 bytes) and the bridge MAC address (6 bytes).

**Figure 25-2 Bridge ID**



The default priority is 32,768. The root bridge is the bridge with the lowest BID. Therefore, if the default priority value is not changed, the switch with the lowest MAC address becomes the root.

# STP Convergence

STP convergence is the process by which switches collectively realize that something has changed in the LAN topology. The switches determine whether they need to change which ports block and which ports forward. The following steps summarize the STP algorithm used to achieve convergence:

**Step 1.** Elect a root bridge (that is, the switch with the lowest BID). Only one root bridge can exist per network. All ports on the root bridge are forwarding ports.

**Step 2.** Elect a root port for each nonroot switch, based on the lowest root path cost. Each nonroot switch has one root port. The root port is the port through which the nonroot bridge has its best path to the root bridge.

**Step 3.** Elect a designated port for each segment, based on the lowest root path cost. Each link has one designated port.

**Step 4.** The root ports and designated ports transition to the forwarding state, and the other ports stay in the blocking state.

Table 25-1 summarizes the reasons STP places a port in forwarding or blocking state.

**Table 25-1 STP: Reasons for Forwarding or Blocking**

| Characterization of Port | STP State | Description |
|---|---|---|
| All the root switch's ports | Forwarding | The root switch is always the designated switch on all connected segments. |
| Each nonroot switch's root port | Forwarding | This is the port through which the switch has the least cost to reach the root switch. |
| Each LAN's designated port | Forwarding | The switch forwarding the lowest-cost BPDU onto the segment is the designated switch for that segment. |
| All other working ports | Blocking | The port is not used for forwarding frames, nor are any frames received on these interfaces considered for forwarding. BPDUs are still received. |

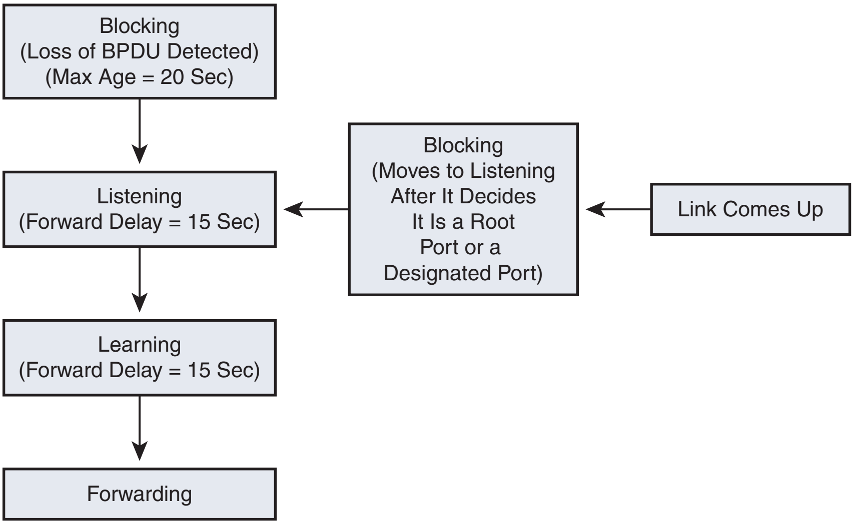
Port bandwidth is used to determine the cost to reach the root bridge. Table 25-2 lists the default port costs defined by the IEEE; these had to be revised with the advent of 10-Gbps ports.

**Table 25-2 Default IEEE Port Costs**

| Ethernet Speed | Original IEEE Cost | Revised IEEE Cost |
|---|---|---|
| 10 Mbps | 100 | 100 |
| 100 Mbps | 10 | 19 |
| 1 Gbps | 1 | 4 |
| 10 Gbps | 1 | 2 |

STP uses the four states in Figure 25-3 as port transitions from blocking to forwarding.

**Figure 25-3 Spanning Tree Port States**




A fifth state, disabled, occurs either when a network administrator manually disables the port or when a security violation disables the port.

# STP Varieties

Several varieties of STP emerged after the original IEEE 802.1D:

- **STP:** The original specification of STP, defined in 802.1D, provides a loop-free topology in a network with redundant links. STP is sometimes referred to as Common Spanning Tree (CST) because it assumes one spanning tree instance for the entire bridged network, regardless of the number of VLANs.
- **PVST+:** Per-VLAN Spanning Tree Plus (PVST+) is a Cisco enhancement of STP that provides a separate 802.1D spanning tree instance for each VLAN configured in the network.
- **RSTP:** Rapid STP (RSTP), or IEEE 802.1w, is an evolution of STP that provides faster convergence than STP. However, RSTP still provides for only a single instance of STP.
- **Rapid PVST+:** Rapid PVST+ is a Cisco enhancement of RSTP that uses PVST+. Rapid PVST+ provides a separate instance of 802.1w per VLAN.
- **MSTP and MST:** Multiple Spanning Tree Protocol (MSTP) is an IEEE standard inspired by the earlier Cisco-proprietary Multiple Instance STP (MISTP) implementation. MSTP maps multiple VLANs into the same spanning tree instance. The Cisco implementation of MSTP is Multiple Spanning Tree (MST), which provides up to 16 instances of RSTP and combines many VLANs with the same physical and logical topology into a common RSTP instance.

Part of your switch administration skill set is the ability to decide which type of STP to implement. Table 25-3 summarizes the features of the various STP flavors.

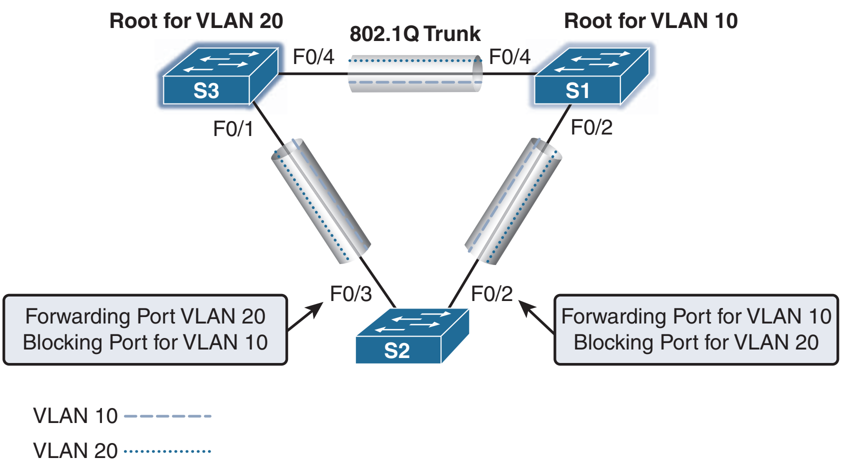
**Table 25-3 Features of STP Varieties**

| Protocol | Standard | Resources Needed | Convergence | Tree Calculation |
|---|---|---|---|---|
| STP | 802.1D | Low | Slow | All VLANs |
| PVST+ | Cisco | High | Slow | Per VLAN |
| RSTP | 802.1w | Medium | Fast | All VLANs |
| Rapid PVST+ | Cisco | Very high | Fast | Per VLAN |
| MSTP | 802.1s, Cisco | Medium or high | Fast | Per instance |

# PVST Operation

PVST Plus (PVST+) is the default setting on all Cisco Catalyst switches. In a PVST+ environment, you can tune the spanning-tree parameters so that half the VLANs forward on each uplink trunk. You do this by configuring one switch to be elected the root bridge for half of the VLANs in the network and a second switch to be elected the root bridge for the other half of the VLANs. In the example in Figure 25-4, S1 is the root bridge for VLAN 10, and S3 is the root bridge for VLAN 20.

**Figure 25-4 PVST+ Topology Example**




From the perspective of S2, a port is forwarding or blocking depending on the VLAN instance. After convergence, port F0/2 will be forwarding VLAN 10 frames and blocking VLAN 20 frames. Port F0/3 will be forwarding VLAN 20 frames and blocking VLAN 10 frames.

Switched networks running PVST+ have the following characteristics:

- Configured PVST per VLAN allows redundant links to be fully utilized.
- Each additional spanning tree instance for a VLAN adds more CPU cycles to all switches in the network.

## Port States

The spanning tree is determined immediately after a switch is finished booting. If a switch port transitions directly from the blocking state to the forwarding state without information about the full topology during the transition, the port can temporarily create a data loop. For this reason, STP introduces the five port states. Table 25-4 describes the port states that ensure that no loops are created during the creation of the logical spanning tree.

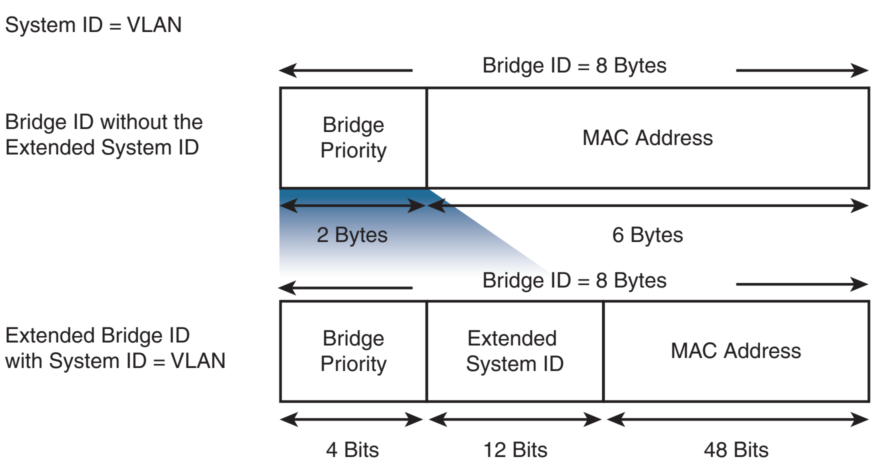
**Table 25-4 PVST Port States**

| Operation Allowed | Blocking | Listening | Learning | Forwarding | Disabled |
|---|---|---|---|---|---|
| Can receive and process BPDUs | Yes | Yes | Yes | Yes | No |
| Can forward data frames received on the interface | No | No | No | Yes | No |
| Can forward data frames switched from another interface | No | No | No | Yes | No |
| Can learn MAC addresses | No | No | Yes | Yes | No |

## Extended System ID

PVST+ requires a separate instance of spanning tree for each VLAN. The BID field in the BPDU must carry VLAN ID (VID) information, as Figure 25-5 shows.

**Figure 25-5 Bridge ID for PVST+ with Extended System ID**



The BID includes the following fields:

- **Bridge Priority:** A 4-bit field is still used to carry bridge priority. However, the priority is conveyed in discrete values in increments of 4096 instead of discrete values in increments of 1 because only the first 4 most-significant bits are available from the 16-bit field.
- **Extended System ID:** A 12-bit field carrying the VID for PVST+.
- **MAC Address:** A 6-byte field with the MAC address of a single switch.

# Rapid PVST+ Operation

In Rapid PVST+, a single instance of RSTP runs for each VLAN. This is why Rapid PVST+ has a very high demand for switch resources (CPU cycles and RAM).

**NOTE:** Rapid PVST+ is simply the Cisco implementation of RSTP on a per-VLAN basis. The rest of this review uses the terms *RSTP* and *Rapid PVST+* interchangeably.

With RSTP, the IEEE improved the convergence performance of STP from 50 seconds to less than 10 seconds with its definition of Rapid STP (RSTP) in the standard 802.1w. RSTP is identical to STP in the following ways:

- It elects the root switch by using the same parameters and tiebreakers.
- It elects the root port on nonroot switches by using the same rules.
- It elects designated ports on each LAN segment by using the same rules.
- It places each port in either forwarding or discarding state, although RSTP calls the blocking state the discarding state.

## RSTP Interface Behavior

The main changes with RSTP can be seen when changes occur in the network. RSTP acts differently on some interfaces based on what is connected to the interface:

- **Edge-type behavior and PortFast:** RSTP improves convergence for edge-type connections by immediately placing the port in forwarding state when the link is physically active.
- **Link-type shared:** RSTP does not do anything differently from STP on link-type shared links. However, because most links between switches today are full duplex, point-to-point, and not shared, this does not matter.
- **Link-type point-to-point:** RSTP improves convergence over full-duplex links between switches. RSTP recognizes the loss of the path to the root bridge through the root port in 6 seconds (based on three times the hello timer value of 2 seconds). RSTP thus recognizes a lost path to the root much more quickly.

RSTP uses different terminology to describe port states. Table 25-5 lists the port states for RSTP and STP.

**Table 25-5 RSTP and STP Port States**

| Operational State | STP State (802.1D) | RSTP State (802.1w) | Forwards Data Frames in This State? |
|---|---|---|---|
| Enabled | Blocking | Discarding | No |
| Enabled | Listening | Discarding | No |
| Enabled | Learning | Learning | No |
| Enabled | Forwarding | Forwarding | Yes |
| Disabled | Disabled | Discarding | No |

RSTP removes the need for listening state and reduces the time required for learning state by actively discovering the network's new state. STP passively waits on new BPDUs and reacts to them during the listening and learning states. With RSTP, the switches negotiate with neighboring switches by sending RSTP messages. The messages enable the switches to quickly determine whether an interface can be immediately transitioned to a forwarding state. In many cases, the process takes only a second or two for the entire RSTP domain.

## RSTP Port Roles

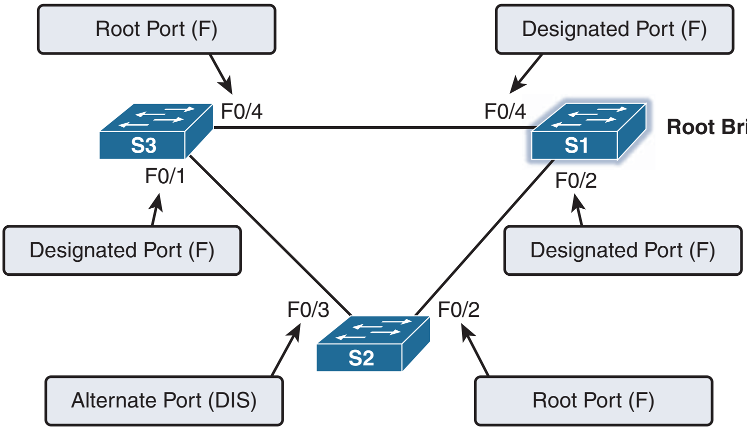
RSTP adds three more port roles in addition to the root port and designated port roles defined in STP. Table 25-6 lists and defines the port roles.

**Table 25-6 RSTP and STP Port Roles**

| RSTP Role | STP Role | Definition |
|---|---|---|
| Root port | Root port | A single port on each nonroot switch in which the switch hears the best BPDU out of all the received BPDUs |
| Designated port | Designated port | Of all switch ports on all switches attached to the same segment/collision domain, the port that advertises the "best" BPDU |
| Alternate port | — | A port on a switch that receives a suboptimal BPDU |
| Backup port | — | A nondesignated port on a switch that is attached to the same segment/collision domain as another port on the same switch |
| Disabled | — | A port that is administratively disabled or that is not capable of working for other reasons |

Figure 25-6 shows an example of these RSTP port roles.

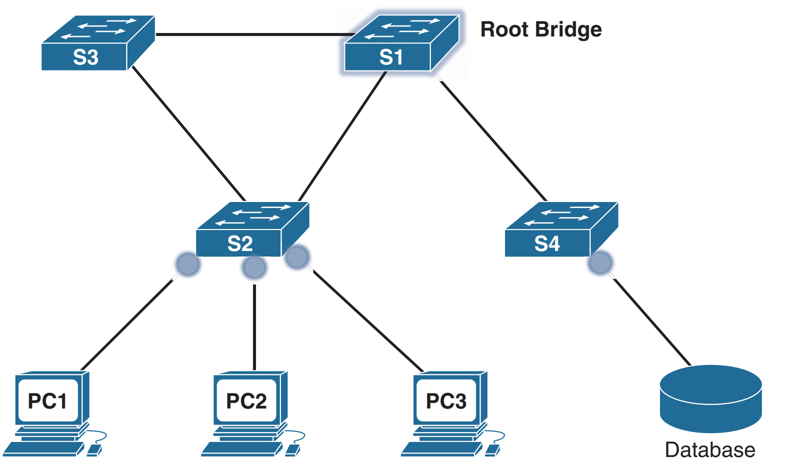
**Figure 25-6 RSTP Port Roles**




## Edge Ports

In addition to the port roles just described, RSTP uses an edge port concept that corresponds to the PVST+ PortFast feature. An edge port connects directly to an end device. Therefore, the switch assumes that no other switch is connected to it. RSTP edge ports should immediately transition to the forwarding state, thereby skipping the time-consuming original 802.1D listening and learning port states. The only caveat is that the port must be a point-to-point link. If it is a shared link, the port is not an edge port, and PortFast should not be configured. Why? Another switch could be added to a shared link—on purpose or inadvertently. Figure 25-7 shows examples of edge ports.

**Figure 25-7 Edge Ports in RSTP**



# Configuring and Verifying Varieties of STP

By default, all Cisco switches use STP without any configuration by the network administrator. However, because STP runs on a per-VLAN basis, you can take advantage of several options to load balance traffic across redundant links.

## STP Configuration Overview

Before you configure or alter the behavior of STP, it is important to know the current default settings listed in Table 25-7.

**Table 25-7 Default STP Configuration on the Cisco Catalyst 2960**

| Feature | Default Setting |
|---|---|
| Enable state | Enables STP on VLAN 1 |
| Spanning tree mode | PVST+ (Rapid PVST+ and MSTP disabled) |
| Switch priority | 32768 |
| Spanning tree port priority (configurable on a per-interface basis) | 128 |
| Spanning tree port cost (configurable on a per-interface basis) | 1000 Mbps: 4<br>100 Mbps: 19<br>10 Mbps: 100 |
| Spanning tree VLAN port priority (configurable on a per-VLAN basis) | 128 |
| Spanning tree VLAN port cost (configurable on a per-VLAN basis) | 1000 Mbps: 4<br>100 Mbps: 19<br>10 Mbps: 100 |
| Spanning tree timers | Hello time: 2 seconds<br>Forward-delay time: 15 seconds<br>Maximum-aging time: 20 seconds<br>Transmit hold count: 6 BPDUs |

## Configuring and Verifying the BID

Regardless of which PVST you use, two main configuration options can help you achieve load balancing: the bridge ID and the port cost manipulation. The bridge ID influences the choice of root switch and can be configured per VLAN. Each interface's (per-VLAN) STP cost to reach the root influences the choice of designated port on each LAN segment. Because PVST requires that a separate instance of spanning tree run for each VLAN, the BID field is required to carry VLAN ID (VID) information. This is accomplished by reusing a portion of the Priority field as the extended system ID to carry a VID.

To change the bridge ID, use one of the following commands:

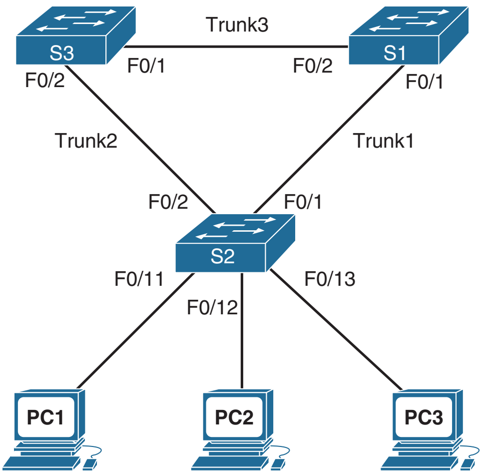
```
Switch(config)# spanning-tree vlan vlan-id root {primary | secondary}
Switch(config)# spanning-tree vlan vlan-id priority priority
```

To change the interface cost, use the following command:

```
Switch(config-if)# spanning-tree vlan vlan-id cost cost
```

Figure 25-8 shows a simple three-switch STP topology without redundant links.

**Figure 25-8 STP Topology**




The network administrator wants to ensure that S1 is always the root bridge and S2 is the backup root bridge. The following commands achieve this objective:

```
S1(config)# spanning-tree vlan 1 root primary
!---------
S2(config)# spanning-tree vlan 1 root secondary
```

The **primary** keyword automatically sets the priority to 24576 or to the next 4096 increment value below the lowest bridge priority detected on the network.

The **secondary** keyword automatically sets the priority to 28672, assuming that the rest of the network is set to the default priority of 32768.

Alternatively, the network administrator can explicitly configure the priority value in increments of 4096 between 0 and 65536 using the following command:

```
S1(config)# spanning-tree vlan 1 priority 24576
!---------
S2(config)# spanning-tree vlan 1 priority 28672
```

**NOTE:** In this example, these commands changed the priority values only for VLAN 1. Additional commands must be entered for each VLAN to take advantage of load balancing.

To verify the current spanning tree instances and root bridges, use the **show spanning-tree** command (see Example 25-1).

**Example 25-1 Verifying Spanning Tree Configurations**

```
S1# show spanning-tree

VLAN0001
  Spanning tree enabled protocol ieee
  Root ID    Priority    24577
             Address      001b.5302.4e80
             This bridge is the root
             Hello Time  2 sec Max Age 20 sec Forward Delay 15 sec

  Bridge ID Priority   24577 (priority 24576 sys-id-ext 1)
             Address     001b.5302.4e80
             Hello Time  2 sec Max Age 20 sec Forward Delay 15 sec
             Aging Time  300

Interface         Role Sts Cost      Prio.NbrType
---------------- ---- --- --------- -------- --------------------------------
Fa0/1            Desg  FWD 19        128.1  P2p
Fa0/2            Desg  FWD 19        128.2  P2p
```

Because an extended system ID is used in the BID, the value of the priority includes the addition of the VLAN ID. Therefore, a priority of 24576 plus a VLAN of 1 results in a priority output of 24577.

## Configuring PortFast and BPDU Guard

To speed convergence for access ports when they become active, you can use Cisco's proprietary PortFast technology. After PortFast is configured and a port is activated, the port immediately transitions from the blocking state to the forwarding state.

In a valid PortFast configuration, BPDUs should never be received because receipt of a BPDU indicates that another bridge or switch is connected to the port, potentially causing a spanning tree loop. When it is enabled, BPDU Guard puts the port in an errdisabled (error-disabled) state upon receipt of a BPDU. This effectively shuts down the port. The BPDU Guard feature provides a secure response to invalid configurations because you must manually put the interface back into service.

Example 25-2 shows the interface commands to configure PortFast and BPDU Guard on S2 in Figure 25-8.

**Example 25-2 Configuring PortFast and BPDU Guard**

```
S2# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
S2(config)# interface range f0/11 - f0/13
S2(config-if-range)# switchport mode access
S2(config-if-range)# spanning-tree portfast
S2(config-if-range)# spanning-tree bpduguard enable
```

Alternatively, you can configure the global commands **spanning-tree portfast default** and **spanning-tree bpduguard default**, which enable PortFast and BPDU Guard on all access ports.

# Configuring Rapid PVST+

Remember that PVST+ is the default operation of Cisco switches. To change to Rapid PVST+, use a single global command on all switches: **spanning-tree mode rapid-pvst**.

Table 25-8 summarizes all the commands related to Rapid PVST+.

**Table 25-8 Commands for Rapid PVST+**

| Description | Command |
|---|---|
| Configure Rapid PVST+ and the spanning tree mode | `Switch(config)#` **`spanning-tree mode rapid-pvst`** |
| Specify a link type as point-to-point (not normally necessary because the shared link type is unusual) | `Switch(config-if)#` **`spanning-tree link-type point-to-point`** |
| Force the renegotiation with neighboring switches on all interfaces or the specified interface | `Switch#` **`clear spanning-tree detected protocols`** [**`interface`** *`interface-id`*] |

# Verifying STP

Several commands enable you to verify the state of the current STP implementation. Table 25-9 summarizes commands most likely to appear on the CCNA exam.

**Table 25-9 STP Verification Commands**

| Description | Command |
|---|---|
| Displays STP information | `Switch#` **`show spanning-tree`** |
| Displays STP information for active interfaces only | `Switch#` **`show spanning-tree active`** |
| Displays abbreviated information for all STP instances | `Switch#` **`show spanning-tree bridge`** |
| Displays detailed information for all STP instances | `Switch#` **`show spanning-tree detail`** |
| Displays STP information for the specified interface | `Switch#` **`show spanning-tree interface`** *`interface-id`* |
| Displays STP information for the specified VLAN | `Switch#` **`show spanning-tree vlan`** *`vlan-id`* |
| Displays a summary of STP port states | `Switch#` **`show spanning-tree summary`** |

**NOTE:** Ideally, you should review the output of these commands today on lab equipment or a simulator. At the very least, refer to the examples in your study resources.

# Study Resources

For today's exam topics, refer to the following resources for more study.

| Resource | Module or Chapter |
|---|---|
| Cisco Network Academy: CCNA 1 | 4 |
| | 6 |
| | 7 |
| CCNA 200-301 Official Cert Guide, Volume 1 | 5 |
| | 8 |
| Portable Command Guide | 11 |

# Day 24

# EtherChannel and HSRP

## CCNA 200-301 Exam Topics

- Configure and verify (Layer 2/Layer 3) EtherChannel (LACP)
- Describe the purpose of first hop redundancy protocol

## Key Topics

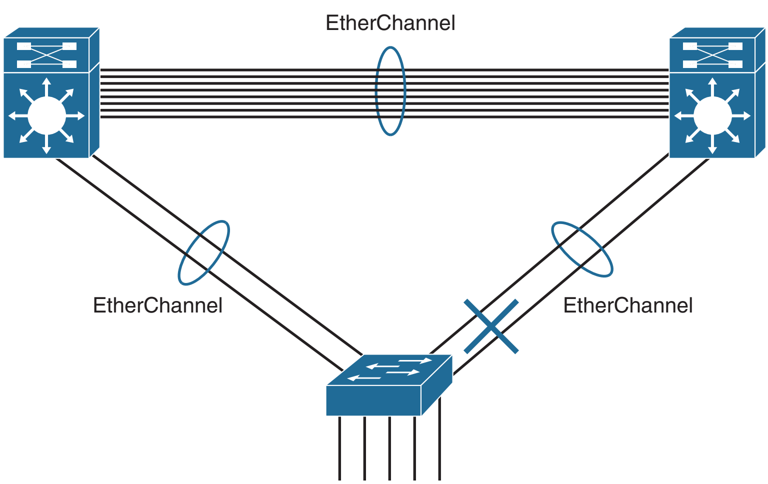
EtherChannel technology enables you to bundle multiple physical interfaces into one logical channel to increase the bandwidth on point-to-point links. In addition, EtherChannel provides a way to prevent the need for Spanning Tree Protocol (STP) convergence when only a single port or cable failure occurs.

Most end devices do not store routes to reach remote networks. Instead, an end device is typically configured with a default gateway that handles routing for the device. But what if that default gateway fails? To ensure that a device will still have access to remote networks, you should implement some type of default gateway redundancy in the network. That is the role of first-hop redundancy protocols (FHRPs).

## EtherChannel Operation

EtherChannel, a technology that Cisco developed, can bundle up to eight equal-speed links between two switches, as you can see between the two distribution layer switches in Figure 24-1.

**Figure 24-1 Sample EtherChannel Topology**



STP sees the bundle of links as a single interface. As a result, if at least one of the links is up, STP convergence does not have to occur. This makes much better use of available bandwidth while reducing the number of times STP must converge. Without the use of EtherChannel or modification of the STP configuration, STP would block all the links except one.

## Benefits of EtherChannel

When EtherChannel is configured, the resulting virtual interface is called a *port channel*. The physical interfaces are bundled together into a port channel interface. EtherChannel has the following benefits:

- Most configuration tasks can be done on the EtherChannel interface instead of on each individual port, thus ensuring configuration consistency throughout the links.
- EtherChannel relies on the existing switch ports to increase bandwidth. No hardware upgrades are needed.
- Load balancing is possible between links that are part of the same EtherChannel. (Load balancing configuration is beyond the scope of the CCNA exam.)
- EtherChannel creates an aggregation that STP recognizes as one logical link.
- EtherChannel provides redundancy. The loss of one physical link does not create a change in the topology.

## Implementation Restrictions

Keep in mind a few limitations when implementing EtherChannel on Cisco 2960 Catalyst switches:

- Interface types, such as Fast Ethernet and Gigabit Ethernet, cannot be mixed within the same EtherChannel.
- Each EtherChannel can consist of up to eight compatibly configured Ethernet ports.
- Cisco IOS Software currently supports up to six EtherChannels.
- Some servers also support EtherChannel to the switch to increase bandwidth; however, the server then needs at least two EtherChannels to provide redundancy because it can send traffic to only one switch through the EtherChannel.
- The EtherChannel configuration must be consistent on the two switches. The trunking configuration (native VLAN, allowed VLANs, and so on) must be the same. All ports also must be Layer 2 ports.
- All ports in the EtherChannel must be Layer 2 ports, or all ports within the EtherChannel must be Layer 3 ports.

**NOTE:** You can configure Layer 3 EtherChannels on multilayer switches; however, that is beyond the scope of the CCNA exam.

# EtherChannel Protocols

You can configure EtherChannel as static or unconditional; however, you also can use two protocols to configure the negotiation process: Port Aggregation Protocol (PAgP, which is Cisco proprietary) and Link Aggregation Control Protocol (LACP, which is IEEE 802.3ad). These two protocols ensure that the two sides of the link have compatible configurations—same speed, duplex setting, and VLAN information. The modes for each differ slightly.

## Port Aggregation Protocol

PAgP is a Cisco-proprietary protocol that aids in the automatic creation of EtherChannel links. PAgP checks for configuration consistency and manages link additions and failures between two switches. It ensures that when an EtherChannel is created, all ports have the same type of configuration. PAgP uses the following modes:

- **On:** This mode forces the interface to channel without PAgP.
- **Desirable:** The interface initiates negotiations with other interfaces by sending PAgP packets.
- **Auto:** The interface responds to the PAgP packets that it receives but does not initiate PAgP negotiation.

The modes must be compatible on the two sides of the EtherChannel. For example, Sw1 and Sw2 in Figure 24-2 must be configured with a particular combination of settings, as shown in Table 24-1.

**Figure 24-2 Two-Switch EtherChannel Topology**

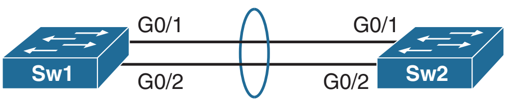



**Table 24-1 PAgP Mode Settings**

| Sw1 | Sw2 | Channel Established? |
|---|---|---|
| On | On | Yes |
| Auto/Desirable | Desirable | Yes |
| On/Auto/Desirable | Not configured | No |
| On | Desirable | No |
| Auto/On | Auto | No |

## Link Aggregation Control Protocol

The Link Aggregation Control Protocol (LACP) is part of an IEEE specification (802.3ad) that allows a switch to negotiate an automatic bundle by sending LACP packets to the peer. It performs a function similar to PAgP with Cisco EtherChannel. Cisco devices support both PAgP and LACP. LACP uses the following modes:

- **On:** This mode forces the interface to channel without LACP.
- **Active:** The interface initiates negotiations with other interfaces by sending LACP packets.

- **Passive:** The interface responds to the LACP packets that it receives but does not initiate LACP negotiation.

As with PAgP, the LACP modes must be compatible on the two sides of the EtherChannel. For example, Sw1 and Sw2 in Figure 24-2 must be configured with a particular combination of settings, as shown in Table 24-2.

**Table 24-2 LACP Mode Settings**

| Sw1 | Sw2 | Channel Established? |
|---|---|---|
| On | On | Yes |
| Active/Passive | Active | Yes |
| On/Active/Passive | Not configured | No |
| On | Active | No |
| Passive/On | Passive | No |

**NOTE:** For both the PAgP and LACP protocols, the on mode creates the EtherChannel configuration unconditionally, without PAgP or LACP dynamic negotiation. You should probably memorize the mode settings for both PAgP and LACP in preparation for the CCNA exam.

# Configuring EtherChannel

To implement EtherChannel, follow these steps:

**Step 1.** Specify the interfaces that you want to bundle together in one link by using the **interface range** *interfaces* command.

**Step 2.** Create a port channel by using the **channel-group** *identifier* **mode** *mode* command. *identifier* can be any number between 1 and 6, inclusive, and does not have to match the other switch. The mode is either **on** or one of the PAgP or LACP modes.

**Step 3.** Enter interface configuration mode for the new port channel with the **interface port-channel** *identifier* command. *identifier* is the same number used with the **channel-group** command.

**Step 4.** Configure the trunking and VLAN settings.

Using the topology in Figure 24-2, assume that Sw1 is already configured for EtherChannel with G0/1 and G0/2 trunking. The native VLAN is 86. The allowed VLANs are 1, 10, 20, and 86. EtherChannel is forced on. No PAgP or LACP is needed. Example 24-1 shows the configuration for Sw2.

**Example 24-1 EtherChannel Configuration**

```
Sw2(config)# interface range g0/1-2
Sw2(config-if-range)# channel-group 1 mode on
Creating a port-channel interface Port-channel 1
Sw2(config-if-range)# interface port-channel 1
Sw2(config-if)# switchport mode trunk
Sw2(config-if)# switchport trunk native vlan 86
Sw2(config-if)# switchport trunk allowed vlan 1,10,20,86
```

In configuring PAgP or LACP, use the appropriate mode keyword for the **channel-group** command. Just ensure that the commands on both sides of the channel are compatible, according to Tables 24-1 and 24-2.

# Verifying EtherChannel

If you configured management addressing, you can quickly verify both sides of an EtherChannel bundle by pinging across the trunk. The two switches should be able to ping each other. Devices configured as members of the various VLANs also should be able to ping each other.

To verify the configuration, use the **show run** command (see Example 24-2).

**Example 24-2 Verifying the EtherChannel Configuration**

```
Sw2# show run | begin interface Port
interface Port-channel1
 switchport trunk native vlan 86
 switchport trunk allowed vlan 1,10,20,86
 switchport mode trunk
!
<output omitted>
interface GigabitEthernet0/1
 switchport trunk native vlan 86
 switchport trunk allowed vlan 1,10,20,86
 switchport mode trunk
 channel-group 1 mode on
!
interface GigabitEthernet0/2
 switchport trunk native vlan 86
 switchport trunk allowed vlan 1,10,20,86
 switchport mode trunk
 channel-group 1 mode on
```

To get an overall summary of the EtherChannel configuration, use the **show etherchannel summary** command (see Example 24-3).

### Example 24-3 Verifying That EtherChannel Is Operational

```
Sw2# show etherchannel summary
Flags: D -  down            P - bundled in port-channel
         I -  stand-alone  s - suspended
         H -  Hot-standby  (LACP only)
         R -  Layer3        S - Layer2
         U -  in use        f - failed to allocate aggregator

         M - not in use, minimum links not met
         u - unsuitable for bundling
         w - waiting to be aggregated
         d - default port


Number of channel-groups in use:  1
Number of aggregators:            1

Group  Port-channel  Protocol   Ports
------+-------------+-----------+----------------------------------------------
1       Po1(SU)          -        Gig0/1(P)  Gig0/2(P)
```

To verify the operational status of a specific interface in the EtherChannel bundle, use the **show interface switchport** command (see Example 24-4).

### Example 24-4 Verifying an Interface's Port Channel Settings

```
Sw2# show interface fa0/1 switchport
Name: Fa0/1
Switchport: Enabled
Administrative Mode: trunk
Operational Mode: trunk (member of bundle Po1)
Administrative Trunking Encapsulation: dot1q
Operational Trunking Encapsulation: dot1q
Negotiation of Trunking: On
Access Mode VLAN: 1 (default)
Trunking Native Mode VLAN: 86 (VLAN0086)
Administrative Native VLAN tagging: enabled
Voice VLAN: none
Administrative private-vlan host-association: none
Administrative private-vlan mapping: none
Administrative private-vlan trunk native VLAN: none
Administrative private-vlan trunk Native VLAN tagging: enabled
Administrative private-vlan trunk encapsulation: dot1q
Administrative private-vlan trunk normal VLANs: none
Administrative private-vlan trunk associations: none
```

```
Administrative private-vlan trunk mappings: none
Operational private-vlan: none
Trunking VLANs Enabled: 1,10,20,86
Pruning VLANs Enabled: 2-1001
```

# Troubleshooting EtherChannel

All interfaces within an EtherChannel must have the same configuration of speed for the duplex mode, native and allowed VLANs on trunks, and access VLAN on access ports:

- Assign all ports in the EtherChannel to the same VLAN or configure them as trunks. Ports with different native VLANs cannot form an EtherChannel.
- When configuring a trunk on an EtherChannel, verify the trunking mode on the EtherChannel. Configuring trunking mode on individual ports that make up the EtherChannel is not recommended. However, if it is done, verify that the trunking configuration is the same on all interfaces.
- An EtherChannel supports the same allowed range of VLANs on all the ports. If the allowed range of VLANs is not the same, the ports do not form an EtherChannel even when PAgP is set to **auto** or **desirable** mode.
- The dynamic negotiation options for PAgP and LACP must be compatibly configured on both ends of the EtherChannel.

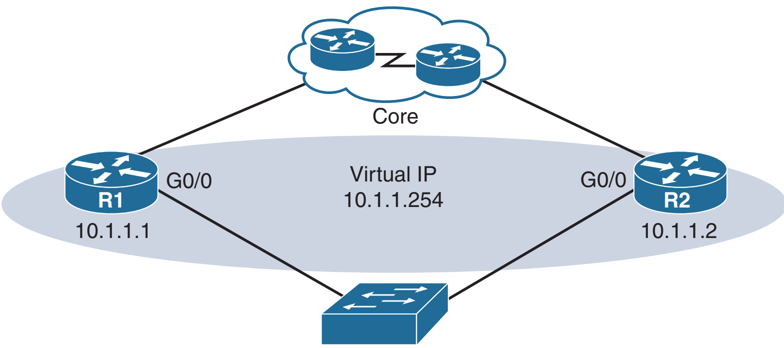
Configuration issues with the **channel-group** command include the following:

- Configuring the **on** keyword on one switch and **desirable**, **auto**, **active**, or **passive** on the other switch. The **on** keyword does not enable PAgP or LACP. Both switches should be configured on one of the acceptable PAgP or LACP modes.
- Configuring the **auto** keyword on both switches. This enables PAgP, but each switch waits on the other to begin negotiations.
- Configuring the **passive** keyword on both switches. This enables LACP, but each switch waits on the other to begin negotiations.
- Mixing keywords from PAgP and LACP, which are not compatible (for example, configuring **active** (LACP) on one switch and **desirable** or **auto** (PAgP) on the other switch).

# First-Hop Redundancy Concepts

FHRPs enable you to install multiple routers in a subnet to collectively act as a single default router. These routers share a virtual IP address, as Figure 24-3 shows.

**Figure 24-3 Redundant Default Gateway Example**




In the figure, the G0/0 interfaces on R1 and R2 are configured with the IP addresses shown. However, both routers are also configured with the virtual IP address. This virtual IP address is the default gateway address configured on end devices. A redundancy protocol provides the mechanism for determining which router should take the active role in forwarding traffic. It also determines when a standby router must take over the forwarding role. The transition from one forwarding router to another is transparent to the end devices. This capability of a network to dynamically recover from the failure of a device acting as a default gateway is known as *first-hop redundancy*.

Regardless of which FHRP is implemented, the following steps take place when the active router fails:

**Step 1.** The standby router stops seeing hello messages from the forwarding router.

**Step 2.** The standby router assumes the role of the forwarding router.

**Step 3.** Because the new forwarding router assumes both the IP and MAC addresses of the virtual router, the end stations do not recognize a disruption in service.

# FHRPs

The following list defines the three options available for FHRPs:

- **Hot Standby Router Protocol (HSRP):** A Cisco-proprietary FHRP designed to allow for transparent failover of a first-hop IPv4 device. The function of the HSRP standby router is to monitor the operational status of the HSRP group and to quickly assume packet-forwarding responsibility if the active router fails. HSRP for IPv6 provides support for IPv6 networks.

- **Virtual Router Redundancy Protocol (VRRP):** An IETF standard that dynamically assigns responsibility for one or more virtual routers to the VRRP routers on an IPv4 LAN. Its operation is similar to that of HSRP. VRRPv3 supports IPv4 and IPv6.

- **Gateway Load Balancing Protocol (GLBP):** A Cisco-proprietary FHRP that protects data traffic from a failed router or circuit, as in HSRP and VRRP, while also allowing load balancing (also called load sharing) between a group of redundant routers. GLBP for IPv6 provides support for IPv6 networks.

The CCNA exam covers HSRP.

# HSRP Operation

HSRP uses an active/standby model in which one router actively assumes the role of default gateway for devices on the subnet. One or more routers on the same subnet are then in standby mode. The HSRP active router implements a virtual IP address and matching virtual MAC address. This virtual IP address is part of the HSRP configuration and belongs to the same subnet as the physical interface IP address, but it is a different IP address. The router then automatically creates the virtual MAC address. All the cooperating HSRP routers know these virtual addresses, but only the HSRP active router uses these addresses at any one point in time.

Assume that you have two HSRP routers similar to R1 and R2 in Figure 24-3. These HSRP routers send each other messages to negotiate which router should be active. Then they continue to send each other messages so that the standby router can detect when the active router fails. If the active router fails, the standby router automatically assumes the virtual IP and MAC addresses and serves as the default gateway for the LAN. The new active router then sends out a gratuitous ARP so that the switches on the subnet will change their MAC address tables to reflect the correct port to reach the virtual MAC. This failover process is transparent to end devices, which are all configured with the virtual IP address as the default gateway.

So what about load balancing? Aren't we wasting the capacity of the standby router and the links connecting to it? Yes, if the routers are connected to only one subnet. However, if VLANs are configured, the routers can share the load by each serving as the active router for some of the VLANs. For example, in Figure 24-3, R1 is the active router for VLAN 10, and R2 is the active router for VLAN 20. Both routers are configured with subinterfaces for inter-VLAN routing and the two virtual IP addresses so that each can assume the role of active router if the other router fails.

## HSRP Versions

Cisco IOS defaults to HSRP version 1. Table 24-3 compares HSRP version 1 and version 2.

**Table 24-3 HSRP Version 1 and Version 2 Features**

| HSRP Feature | Version 1 | Version 2 |
|---|---|---|
| Group numbers supported | 0–255 | 0–4095 |
| Authentication | None | MD5 |
| Multicast addresses | IPv4: 224.0.0.2 | IPv4: 224.0.0.102<br>IPv6: FF02::66 |
| Virtual MAC ranges | 0000.0C07.AC00 to 0000.0C07.ACFF | IPv4: 0000.0C9F.F000 to 0000.0C9F.FFFF<br>IPv6: 0005.73A0.0000 to 0005.73A0.0FFF |

**NOTE:** The last three hexadecimal digits of the virtual MAC address indicate the configured group number. Group numbers are important for more advanced HSRP configurations, which are beyond the scope of the CCNA exam.

## HSRP Priority and Preemption

By default, the router with the numerically highest IPv4 address is elected as the active HSRP router. To configure a router to be the active router, regardless of IPv4 addressing, use the **standby priority** interface configuration command. The default priority is 100. The router with the highest priority will be the active HSRP router, assuming that no election has already occurred.

To force a new HSRP election, preemption must be enabled with the **standby preempt** interface configuration command.

# HSRP Configuration and Verification

Let's look at how to configure the topology in Figure 24-3. HSRP requires only one command on both routers:

```
Router(config-if)# standby group ip ip-address
```

The interface must be on the same subnet as the other HSRP router or routers. The *group* number and virtual *ip-address* must be the same on all HSRP routers.

Unless the **priority** command is used, the first router configured becomes the HSRP active router. Therefore, even though in Example 24-5 R1 is configured first, it includes a priority configuration to make sure that R1 is always the active router. Also, to make sure that R1 resumes the active router role after losing connectivity, the **standby preempt** command is configured.

**Example 24-5 Configuring HSRP**

```
R1(config)# interface g0/0
R1(config-if)# ip address 10.1.1.1 255.255.0.0
R1(config-if)# standby 1 ip 10.1.1.254
R1(config-if)# standby 1 priority 200
R1(config-if)# standby 1 preempt
```

```
R2(config)# interface g0/0
R2(config-if)# ip address 10.1.1.2 255.255.0.0
R2(config-if)# standby 1 ip 10.1.1.254
```

To verify that HSRP is up and running, use the **show standby** command or the **brief** version of the command, as in Example 24-6.

**Example 24-6 Verifying HSRP**

```
R1# show standby
GigabitEthernet0/0 - Group 1
  State is Active
    2 state changes, last state change 00:11:51
  Virtual IP address is 10.1.1.254
  Active virtual MAC address is 0000.0c07.ac01
```

```
    Local virtual MAC address is 0000.0c07.ac01 (v1 default)
  Hello time 3 sec, hold time 10 sec
    Next hello sent in 1.232 secs
  Preemption enabled
  Active router is local
  Standby router is 10.1.1.2, priority 100 (expires in 9.808 sec)
  Priority 200 (configured 200)
  Group name is "hsrp-Gi0/0-1" (default)
R1# show standby brief
                     P indicates configured to preempt.
                     |
Interface    Grp  Pri P State    Active  Standby    Virtual IP
Gi0/0        1    200   Active   local   10.1.1.2   10.1.1.254
```

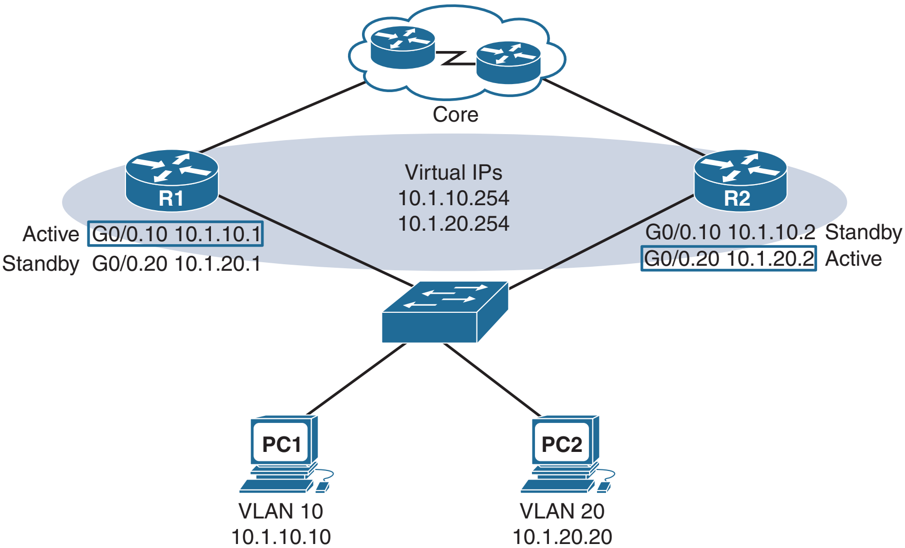
```
R2# show standby
GigabitEthernet0/0 - Group 1
  State is Standby
     1 state change, last state change 00:15:23
  Virtual IP address is 10.1.1.254
  Active virtual MAC address is 0000.0c07.ac01
     Local virtual MAC address is 0000.0c07.ac01 (v1 default)
  Hello time 3 sec, hold time 10 sec
    Next hello sent in 1.008 secs
  Preemption disabled
  Active router is 10.1.1.1, priority 200 (expires in 8.624 sec)
  Standby router is local
  Priority 100 (default 100)
  Group name is "hsrp-Gi0/0-1" (default)
R2# show standby brief
                       P  indicates configured to preempt.
                       |
Interface    Grp  Pri P State    Active     Standby   Virtual IP
Gi0/0        1    100   Standby 10.1.1.1   local     10.1.1.254
```

The **show standby brief** command displays the most pertinent information you might need in a few lines of output. The more verbose **show standby** command provides additional information, such as the number of state changes, the virtual MAC address, hellos, and the group name.

# HSRP Load Balancing

As with STP, you might want your HSRP routers to be configured in active/active state, with one router active for one set of VLANs and the other router active for the remaining VLANs. Figure 24-4 shows a topology with multiple VLANs.

**Figure 24-4 HSRP Load Balancing Example**




To implement HSRP load balancing for different VLANs, configure R1 as the active router for half the VLANs and R2 as the active router for the other half of the VLANs (see Example 24-7).

**Example 24-7 Configuring HSRP Load Balancing**

```
R1# show run | begin interface G

interface GigabitEthernet0/0
  no ip address
  duplex auto
  speed auto
!
interface GigabitEthernet0/0.10
  encapsulation dot1Q 10
  ip address 10.1.10.1 255.255.255.0
  standby version 2
  standby 1 ip 10.1.10.254
  standby 1 priority 150
  standby 1 preempt
!
interface GigabitEthernet0/0.20
  encapsulation dot1Q 20
  ip address 10.1.20.1 255.255.255.0
  standby version 2
  standby 1 ip 10.1.20.254
```

```
R2# show run | begin interface G

interface GigabitEthernet0/0
  no ip address
  duplex auto
  speed auto
!
interface GigabitEthernet0/0.10
  encapsulation dot1Q 10
  ip address 10.1.10.2 255.255.255.0
  standby version 2
  standby 1 ip 10.1.10.254
!
interface GigabitEthernet0/0.20
  encapsulation dot1Q 20
  ip address 10.1.20.2 255.255.255.0
  standby version 2
  standby 1 ip 10.1.20.254
  standby 1 priority 150
  standby 1 preempt
!
```

To verify that HSRP with load balancing is operational, use the **show standby** command or the **brief** version of the command (see Example 24-8).

**Example 24-8 Verifying HSRP Load Balancing**

```
R1# show standby brief
                     P indicates configured to preempt.
                     |
Interface  Grp  Pri  P  State    Active     Standby     Virtual IP
           1    150     Active   local      10.1. 10.2  10.1.10.254
           1    100     Standby  10.1.20.2  local       10.1.20.254
```

```
R2# show standby brief
                     P indicates configured to preempt.
                     |
Interface   Grp  Pri  P State    Active     Standby    Virtual IP
            1    100    Standby  10.1.10.1  local      10.1.20.254
            1    150    Active   local      10.1.20.1  10.1.20.254
```

# Troubleshooting HSRP

Issues with HSRP most likely result from one or more of the following:

- The active router that controls the virtual IP address for the group was not successfully elected.
- The standby router did not successfully keep track of the active router.
- No decision was made regarding when to hand another router control of the virtual IP for the group.
- End devices failed to successfully configure the virtual IP address as the default gateway.

Common HSRP configuration issues include the following:

- The HSRP routers are not connected to the same network segment.
- The HSRP routers are not configured with IPv4 addresses from the same subnet.
- The HSRP routers are not configured with the same virtual IPv4 address.
- The HSRP routers are not configured with the same HSRP group number.
- End devices are not configured with the correct default gateway address.

# Study Resources

For today's exam topics, refer to the following resources for more study.

| Resource | Module or Chapter |
|---|---|
| Cisco Network Academy: CCNA 2 | 6 |
| | 9 |
| CCNA 200-301 Official Cert Guide, Volume 1 | 9 |
| | 10 |
| | 17 |
| CCNA 200-301 Official Cert Guide, Volume 2 | 7 |
| Portable Command Guide | 17 |

# Day 23

# DHCP and DNS

## CCNA 200-301 Exam Topics

- Explain the role of DHCP and DNS within the network
- Configure and verify DHCP client and relay
- Verify IP parameters for Client OS (Windows, Mac OS, Linux)

## Key Topics

Imagine that you have to manually configure the IP addressing for every device you want to connect to the network. Furthermore, imagine that you have to type in the IP address for every website you want to visit. Today we review the two protocols that automate this process: Dynamic Host Configuration Protocol (DHCP) and Domain Name System (DNS). DHCP and DNS make the life of Internet users easier. We also review how to verify IP configuration of end devices for Windows, macOS, and Linux.

## DHCPv4

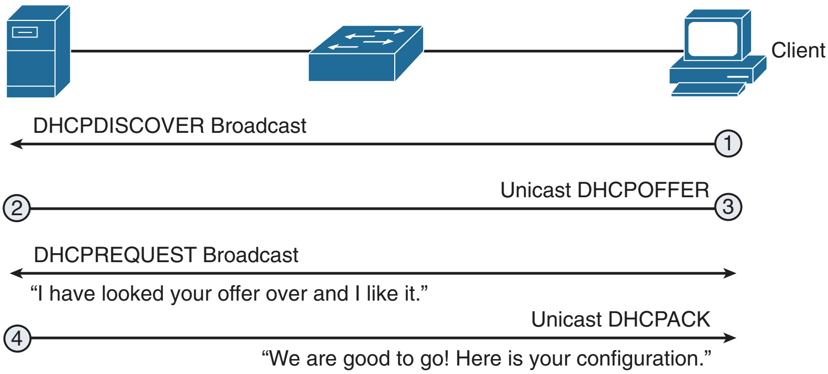
DHCPv4 allows a host to obtain an IP address dynamically when it connects to the network. The DHCPv4 client contacts the DHCPv4 server by sending a request for an IP address. The DHCPv4 server chooses an address from a configured range of addresses called a *pool* and assigns it to the host client for a set period. Figure 23-1 graphically shows the process by which a DHCPv4 server fulfills a request from a DHCPv4 client.

When a DHCPv4-configured device boots up or connects to the network, the client broadcasts a DHCPDISCOVER packet to identify any available DHCPv4 servers on the network. A DHCPv4 server replies with a DHCPOFFER, which is a lease offer message with an assigned IP address, subnet mask, DNS server, and default gateway information, as well as the duration of the lease.

The client can receive multiple DHCPOFFER packets if the local network has more than one DHCPv4 server. The client chooses the first offer and broadcasts a DHCPREQUEST packet that identifies the explicit server and lease offer that it is accepting.

Assuming that the IP address is still valid, the chosen server returns a DHCPACK (acknowledgment) message, finalizing the lease. If the offer is no longer valid for some reason, the chosen server responds to the client with a DHCPNAK (negative acknowledgment) message. After it is leased, the client renews before the lease expiration through another DHCPREQUEST. If the client is powered down or taken off the network, the address is returned to the pool for reuse.

**Figure 23-1 Allocating IP Addressing Information Using DHCPv4**




# DHCPv4 Configuration Options

A Cisco router can be configured to handle DHCP requests in two ways: as a DHCP server or as a DHCP relay agent. A Cisco router can also be configured as a DHCP client, requesting an IPv4 address from a DHCP server for one or more of its interfaces. All these options can be configured at the same time on the same device. For example, a router might be the DHCP server for a directly connected LAN while at the same time forwarding DHCP server requests to another DHCP server for other LANs. In addition, the router could have one or more of its interfaces configured to request DHCP addressing from a remote server.

## Configuring a Router as a DHCPv4 Server

A Cisco router running Cisco IOS Software can be configured to act as a DHCPv4 server. The Cisco IOS DHCPv4 server assigns and manages IPv4 addresses from specified address pools within the router to DHCPv4 clients.

The steps to configure a router as a DHCPv4 server follow:

**Step 1.** Use the **ip dhcp excluded-address** *low-address* [*high-address*] command to identify an address or range of addresses to exclude from the DHCPv4 pool. For example:

```
R1(config)# ip dhcp excluded-address 192.168.10.1 192.168.10.9
R1(config)# ip dhcp excluded-address 192.168.10.254
```

**Step 2.** Create the DHCPv4 pool by using the **ip dhcp pool** *pool-name* command, which places you in DHCP configuration mode:

```
R1(config)# ip dhcp pool LAN-POOL-10
R1(dhcp-config)#
```

**Step 3.** Configure the IP addressing parameter you need to automatically assign to requesting clients. Table 23-1 lists the required commands.

**Table 23-1 Required DHCPv4 Configuration Commands**

| Required Task | Command |
|---|---|
| Define the address pool | **network** *network-number* [*mask* \| /*prefix-length*] |
| Define the default router or gateway | **default-router** *address* [*address2...address8*] |

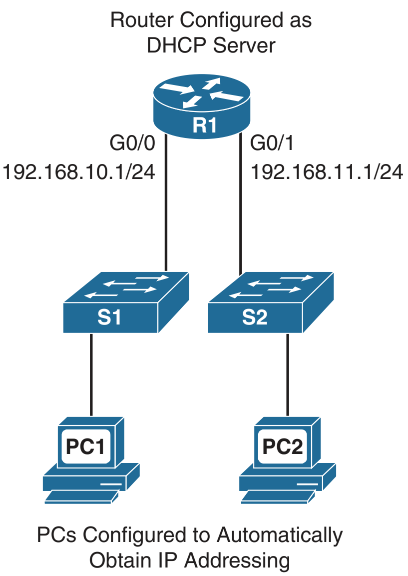
Table 23-2 lists some of the common optional DHCPv4 tasks.

**Table 23-2 Optional DHCPv4 Configuration Commands**

| Optional Task | Command |
|---|---|
| Define a DNS server | **dns-server** *address* [*address2...address8*] |
| Define the domain name | **domain-name** *domain* |
| Define the duration of the DHCPv4 lease | **lease** {*days* [*hours*] [*minutes*] \| **infinite**} |
| Define the NetBIOS WINS server | **netbios-name-server** *address* [*address2...address8*] |

Figure 23-2 shows a sample DHCPv4 topology.

**Figure 23-2 DHCPv4 Sample Topology**




Example 23-1 shows DHCPv4 required and optional commands to configure R1 as the DHCPv4 server for both LANs in Figure 23-2.

**Example 23-1 DHCPv4 Configuration Example**

```
!Configure IP addresses that you want excluded from the DHCPv4 pool of addresses
R1(config)# ip dhcp excluded-address 192.168.10.1 192.168.10.9
R1(config)# ip dhcp excluded-address 192.168.10.254
R1(config)# ip dhcp excluded-address 192.168.11.1 192.168.11.9
R1(config)# ip dhcp excluded-address 192.168.11.254
```

```
!R1 needs two DHCPv4 pools for the two LANs. Each pool is configured with required
   and optional commands.
R1(config)# ip dhcp pool LAN-POOL-10
R1(dhcp-config)# network 192.168.10.0 255.255.255.0
R1(dhcp-config)# default-router 192.168.10.1
R1(dhcp-config)# dns-server 192.168.50.195 209.165.202.158
R1(dhcp-config)# domain-name cisco.com
R1(dhcp-config)# lease 2
R1(dhcp-config)# netbios-name-server 192.168.10.254
R1(dhcp-config)# ip dhcp pool LAN-POOL-11
R1(dhcp-config)# network 192.168.11.0 255.255.255.0
R1(dhcp-config)# default-router 192.168.11.1
R1(dhcp-config)# dns-server 192.168.50.195 209.165.202.158
R1(dhcp-config)# domain-name cisco.com
R1(dhcp-config)# lease 2
R1(dhcp-config)# netbios-name-server 192.168.11.254
R1(dhcp-config)# end
```

Cisco IOS Software supports DHCPv4 service by default. To disable it, use the global command **no service dhcp**.

To verify DHCPv4 operations on R1 in Figure 23-2, use the commands in Example 23-2.

**Example 23-2 Verifying DHCPv4 Operation**

```
R1# show ip dhcp binding
Bindings from all pools not associated with VRF:
IP address      Client-ID/          Lease expiration      Type
                Hardware address/
                User name
192.168.10.10   0100.1641.aea5.a7   Jul 18 2008 08:17 AM Automatic
192.168.11.10   0100.e018.5bdd.35    Jul 18 2008 08:17 AM Automatic

R1# show ip dhcp server statistics
Memory usage         26455
Address pools        2
Database agents      0
Automatic bindings   2
Manual bindings      0
Expired bindings     0

Malformed messages   0
Secure arp entries   0
Message              Received
```

```
BOOTREQUEST          0
DHCPDISCOVER         2
DHCPREQUEST          2
DHCPDECLINE          0
DHCPRELEASE          0
DHCPINFORM           0

Message              Sent
BOOTREPLY            0
DHCPOFFER            2
DHCPACK              2
DHCPNAK              0
R1#
```

Because PC1 and PC2 are connected to the LANs, each automatically receives its IP addressing information from the router's DHCPv4 server. Example 23-3 shows the output from the **ipconfig/all** command on PC1.

**Example 23-3 DHCPv4 Client Configuration**

```
C:\> ipconfig/all

Windows IP Configuration

        Host Name . . . . . . . . . . . . : ciscolab
        Primary Dns Suffix . . . . . . .  :
        Node Type . . . . . . . . . . . . : Hybrid
        IP Routing Enabled. . . . . . . . : No
        WINS Proxy Enabled. . . . . . . . : No

Ethernet adapter Local Area Connection:

        Connection-specific DNS Suffix .  : cisco.com
        Description . . . . . . . . . . . : Intel(R) PRO/1000 PL
        Physical Address. . . . . . . . . : 00-7-41-AE-A5-A7
        Dhcp Enabled. . . . . . . . . . . : Yes
        Autoconfiguration Enabled . . . . : Yes
        IP Address. . . . . . . . . . . . : 192.168.10.11
        Subnet Mask . . . . . . . . . . . : 255.255.255.0
        Default Gateway . . . . . . . . . : 192.168.10.1

        DHCP Server . . . . . . . . . . . : 192.168.10.1
        DNS Servers . . . . . . . . . . . : 192.168.50.195
        209.165.202.158
```

```
      Primary WINS Server . . . . . . . . : 192.168.10.254
      Lease Obtained. . . . . . . . . . . : Wednesday, July 16, 2008 8:16:59 AM
      Lease Expires . . . . . . . . . . . : Friday, July 18, 2008 8:16:59 AM

C:\>
```

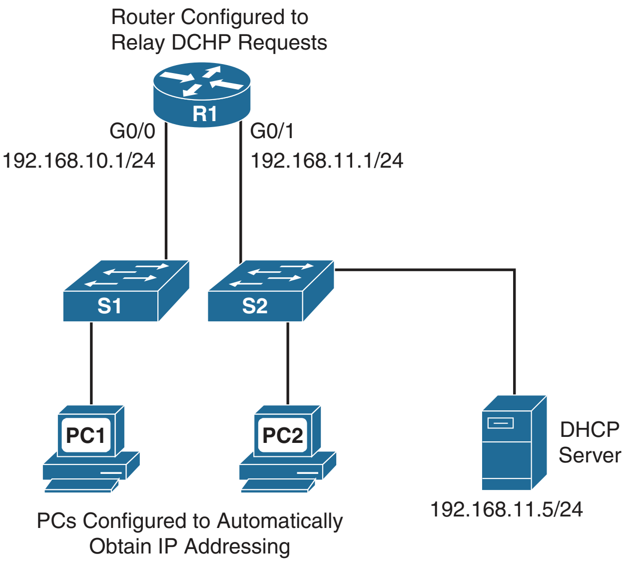
To release the DHCPv4 configuration on a Windows-based client, enter the **ipconfig/release** command. To renew the DHCPv4 configuration, enter the **ipconfig/renew** command.

## Configuring a Router to Relay DHCPv4 Requests

In a complex network, the DHCPv4 servers are usually contained in a server farm. Therefore, clients typically are not on the same subnet as the DHCPv4 server, as in the previous example. To ensure that broadcasted DHCPDISCOVER messages are sent to the remote DHCPv4 server, use the **ip helper-address** command.

For example, in Figure 23-3, the DHCPv4 server is located on the 192.168.11.0/24 LAN and is serving IP addressing information for both LANs.

**Figure 23-3 DHCPv4 Relay Topology**




Without the **ip helper-address** command, R1 would discard any broadcasts from PC1 requesting DHCPv4 services. To configure R1 to relay DHCPDISCOVER messages, enter the following commands:

```
R1(config)# interface gigabitethernet 0/0
R1(config-if)# ip helper-address 192.168.11.5
```

Notice that the commands are entered on the interface that will receive DHCPv4 broadcasts. R1 then forwards DHCPv4 broadcast messages as a unicast to 192.168.11.5. By default, the **ip helper-address** command forwards the following eight UDP services:

- **Port 37:** Time
- **Port 49:** TACACS
- **Port 53:** DNS
- **Port 67:** DHCP/BOOTP server
- **Port 68:** DHCP/BOOTP client
- **Port 69:** TFTP
- **Port 137:** NetBIOS name service
- **Port 138:** NetBIOS datagram service

To specify additional ports, use the global command **ip forward-protocol udp** [*port-number* | *protocol*]. To disable broadcasts of a particular protocol, use the **no** form of the command.

## Configuring a Router as a DHCPv4 Client

Cisco routers in small offices or branch sites are often configured as DHCPv4 clients. The method used depends on the ISP. However, in its simplest configuration, the interface used to connect to a cable or DSL modem is configured with the **ip address dhcp** interface configuration command.

For example, in Figure 23-4, the BRANCH router's GigabitEthernet 0/1 interface can be configured to request addressing from the ISP router.

**Figure 23-4 Router as a DHCP Client**


Example 23-4 shows the configuration and verification of DHCP addressing on BRANCH.

**Example 23-4 Configuring a Router as a DHCP Client**

```
BRANCH(config)# interface g0/1
BRANCH(config-if)# ip address dhcp
BRANCH(config-if)# no shutdown
*Mar 15 08:45:34.632: %DHCP-6-ADDRESS_ASSIGN: Interface GigabitEthernet0/1
  assigned
   DHCP address 209.165.201.12, mask 255.255.255.224, hostname BRANCH
BRANCH(config-if)# end
```

```
BRANCH# show ip interface g0/1
GigabitEthernet0/1 is up, line protocol is up
   Internet address is 209.165.201.12/27
   Broadcast address is 255.255.255.255
   Address determined by DHCP
   <output omitted>
BRANCH#
```

# DHCPv6

IPv6 has two methods for automatically obtaining a global unicast address:

- Stateless address autoconfiguration (SLAAC)
- Stateful DHCPv6 (Dynamic Host Configuration Protocol for IPv6)

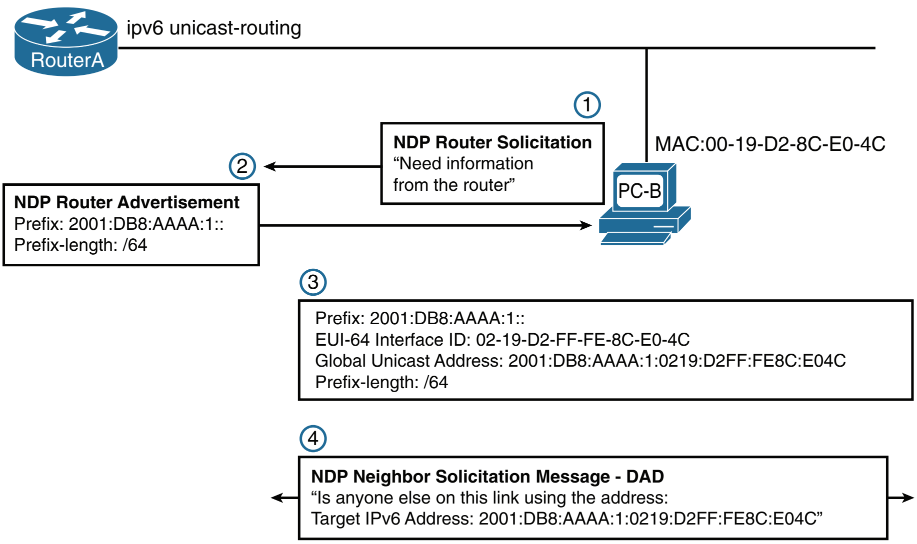
## SLAAC

SLAAC uses ICMPv6 Router Solicitation (RS) and Router Advertisement (RA) messages to provide addressing and other configuration information. A client then uses the RA information to build an IPv6 address and verify it with a special type of Neighbor Solicitation (NS) message through duplicate address detection (DAD). These three message types—RS, RA, and NS—belong to the Neighbor Discovery Protocol:

- **Router Solicitation (RS) message:** When a client is configured to obtain its addressing information automatically using SLAAC, the client sends an RS message to the router. The RS message is sent to the IPv6 all-routers multicast address, FF02::2.
- **Router Advertisement (RA) message:** A client uses this information to create its own IPv6 global unicast address. A router sends RA messages periodically or in response to RS messages. An RA message includes the prefix and prefix length of the local segment. By default, Cisco routers send RA messages every 200 seconds. RA messages are sent to the IPv6 all-nodes multicast address, FF02::1.
- **Neighbor Solicitation (NS) message:** An NS message is normally used to learn the data link layer address of a neighbor on the same network. In the SLAAC process, a host uses DAD by inserting its own IPv6 address as the destination address in an NS message. The NS message is sent out on the network to verify that a newly minted IPv6 address is unique. If a Neighbor Advertisement message is received, the host knows that the IPv6 address is not unique.

Figure 23-5 shows the SLAAC process using three messages of NDP.

**Figure 23-5 Neighbor Discovery and the SLAAC Process**




Let's briefly review the steps in Figure 23-5.

**Step 1.** PC-B sends an RS message to the all-routers multicast address, FF02::2, to inform the local IPv6 router that it needs an RA message.

**Step 2.** RouterA receives the RS message and responds with an RA message. Included in the RA message are the prefix and prefix length of the network. The RA message is sent to the IPv6 all-nodes multicast address, FF02::1, with the link-local address of the router as the IPv6 source address.

**Step 3.** PC-B uses this information to create its own IPv6 global unicast address. It appends the 64-bit prefix address to its own locally generated 64-bit interface ID, which it creates using either the EUI process (see Figure 23-5) or a random number generator. It uses RouterA's link-local address as the default gateway.

**Step 4.** Before PC-B can use this newly created IPv6 address, it uses the DAD process, sending out an NS message to verify that the address is unique.

---

**NOTE:** A client's operating system can be configured to ignore RA messages, in which case the client always opts to use the services of a DHCPv6 server.

---

An RA message informs a client how to obtain automatic IPv6 addressing: using SLAAC, DHCPv6, or a combination of the two. The RA message contains two flags to indicate the configuration option: the Managed Address Configuration flag (M flag) and the Other Configuration flag (O flag).

The default setting for these flags is 0, or both bits off. To the client, this means it is to use the SLAAC process exclusively to obtain all of its IPv6 addressing information. If either of these flags is set to 1 for some reason, you can use the **no** form of the following **ipv6 nd** commands in interface configuration mode to reset them to 0:

```
Router(config-if)# no ipv6 nd managed-config-flag
Router(config-if)# no ipv6 nd other-config-flag
```

## Stateless DHCPv6

In stateless DHCPv6, the client uses the RA message from the router to generate its global unicast address. However, the client then sends a request to the DHCPv6 server to obtain any additional information that the RA has not already supplied.

For stateless DHCPv6, the O flag is set to 1 so that the client is informed that additional configuration information is available from a stateless DHCPv6 server. Use the following command on the interface to modify the RA message:

```
Router(config-if)# ipv6 nd other-config-flag
```

## Stateful DHCPv6

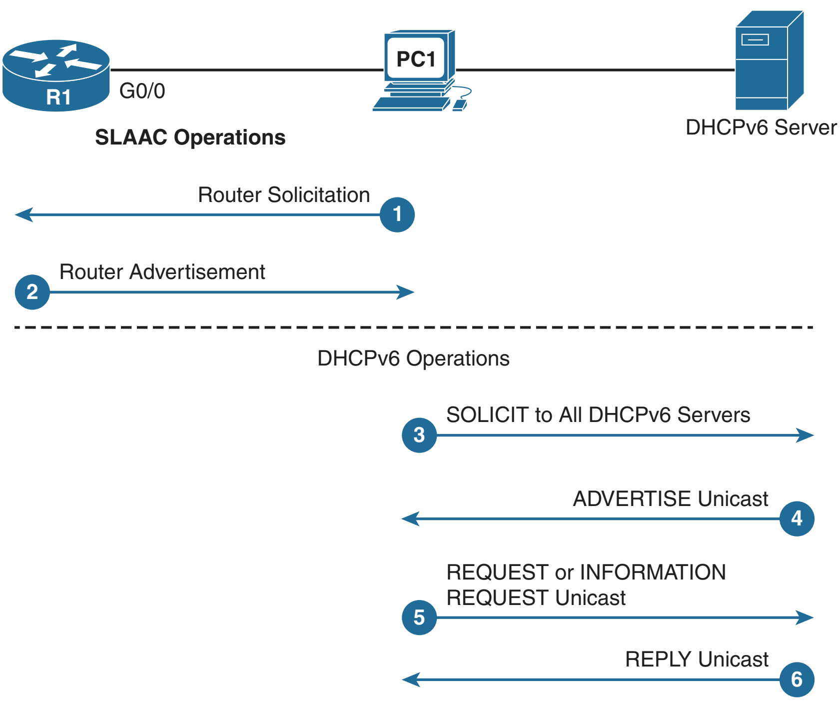
For stateful DHCPv6, the RA message tells the client to obtain all its addressing information from a DHCPv6 server. The M flag must be set on the interface with the following command:

```
Router(config-if)# ipv6 nd managed-config-flag
```

## Stateless and Stateful DHCPv6 Operation

Figure 23-6 shows the full operation of DHCPv6, regardless of the method used: SLAAC, stateless DHCPv6, or stateful DHCPv6.

**Figure 23-6 DHCPv6 Operations**



The following steps occur in Figure 23-6:

**Step 1.** PC1 sends an RS message on bootup to begin the process of obtaining IPv6 addressing.

**Step 2.** R1 replies with an RA message. If the M and O flags are not set, PC1 uses SLAAC. If either the M flag or the O flag is set, PC1 begins the DHCPv6 process.

**Step 3.** PC1 sends a DHCPv6 SOLICIT message to the all-DHCPv6-servers address, FF02::1:2—a link-local multicast address that will not be forwarded by routers.

**Step 4.** A DHCPv6 server responds with a DHCPv6 ADVERTISE unicast message informing the client of its presence.

**Step 5.** The client sends either a unicast DHCPv6 REQUEST (the M flag was set, and the client is using stateful DHCPv6) or a unicast DHCPv6 INFORMATION-REQUEST (the O flag was set, and the client is using stateless DHCPv6).

**Step 6.** The server replies with the information requested.

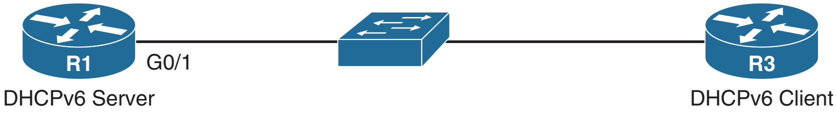
# DHCPv6 Configuration Options

A router can be configured as a stateless DHCPv6 server, a stateful DHCPv6 server, and a DHCPv6 client. As in DHCPv4, the router can be configured with all three, depending on what role it plays for its various interfaces.

## Configuring a Router as a Stateless DHCPv6 Server

We use Figure 23-7 for all the examples in this section. R1 is the DHCPv6 server, and R3 is the DHCPv6 client.

**Figure 23-7 DHCPv6 Server and Client Topology**




To configure R1 as a stateless DHCP server, you need to make sure that **ipv6 unicast-routing** is enabled. Then, in global configuration mode, configure the pool name, DNS server, and domain name. Finally, enable the DHCPv6 pool on the appropriate interface and set the O flag so that clients on that interface know to request DHCPv6 services from the router. Example 23-5 shows the configuration for R1.

**Example 23-5 Configuring a Router as a Stateless DHCPv6 Server**

```
R1(config)# ipv6 unicast-routing
R1(config)# ipv6 dhcp pool O-FLAG-SET
R1(config-dhcpv6)# dns-server 2001:db8:acad:1::5
R1(config-dhcpv6)# domain-name cisco.com
R1(config-dhcpv6)# exit
R1(config)# interface g0/1
R1(config-if)# ipv6 address 2001:db8:1:1::1/64
R1(config-if)# ipv6 dhcp server O-FLAG-SET
R1(config-if)# ipv6 nd other-config-flag
R1(config-if)# end
R1# show ipv6 dhcp pool
DHCPv6 pool: O-FLAG-SET
  DNS server: 2001:DB8:ACAD:1::5
  Domain name: cisco.com
  Active clients: 0
R1#
```

To configure a router interface as a DHCPv6 client, enable IPv6 on the interface and enter the **ipv6 address autoconfig** command, as in Example 23-6. Verify the configuration with the **show ipv6 interface** command.

**Example 23-6 Configuring an Interface as a DHCPv6 Client**

```
R3(config)# interface g0/1
R3(config-if)# ipv6 enable
R3(config-if)# ipv6 address autoconfig
R3(config-if)# end
R3# show ipv6 interface g0/1
GigabitEthernet0/1 is up, line protocol is up
  IPv6 is enabled, link-local address is FE80::32F7:DFF:FE25:2DE1
  No Virtual link-local address(es):
  Stateless address autoconfig enabled
  Global unicast address(es):
     2001:DB8:1:1:32F7:DFF:FE25:2DE1, subnet is 2001:DB8:1:1::/64 [EUI/CAL/PRE]
     valid lifetime 2591935 preferred lifetime 604735
  Joined group address(es):
    FF02::1
    FF02::1:FF25:2DE1
  MTU is 1500 bytes
  ICMP error messages limited to one every 100 milliseconds
  ICMP redirects are enabled
  ICMP unreachables are sent
```

```
  ND DAD is enabled, number of DAD attempts: 1
  ND reachable time is 30000 milliseconds (using 30000)
  ND NS retransmit interval is 1000 milliseconds
  Default router is FE80::D68C:B5FF:FECE:A0C1 on GigabitEthernet0/1
R3#
```

## Configuring a Router as a Stateful DHCPv6 Server

The main difference between a stateless configuration and a stateful configuration is that a stateful server includes IPv6 addressing information and keeps a record of the IPv6 addresses that are leased out. Also, for the client side, the **ipv6 address dhcp** command is used instead of the **ipv6 address autoconfig** command. Example 23-7 shows the stateful DHCPv6 server configuration with stateful address information added and the M bit set instead of the O bit.

**Example 23-7 Configuring a Router as a Stateful DHCPv6 Server**

```
R1(config)# ipv6 unicast-routing
R1(config)# ipv6 dhcp pool M-FLAG-SET
R1(config-dhcpv6)# address prefix 2001:db8:1:1::/64 lifetime infinite infinite
R1(config-dhcpv6)# dns-server 2001:db8:acad:1::5
R1(config-dhcpv6)# domain-name cisco.com
R1(config-dhcpv6)# exit
R1(config)# interface g0/1
R1(config-if)# ipv6 address 2001:db8:1:1::1/64
R1(config-if)# ipv6 nd managed-config-flag
R1(config-if)# end
!After R3 is configured as a DHCP client, verify DHCP with the following
  commands:
R1# show ipv6 dhcp pool
DHCPv6 pool: M-FLAG-SET
   Address allocation prefix: 2001:DB8:1:1::/64 valid 4294967295 preferred
  4294967295 (1 in use, 0 conflicts)
   DNS server: 2001:DB8:ACAD:1::5
   Domain name: cisco.com
   Active clients: 1
R1# show ipv6 dhcp binding
Client: FE80::32F7:DFF:FEA3:1640
   DUID: 0003000130F70DA31640
   Username: unassigned
   IA NA: IA ID 0x00060001, T1 43200, T2 69120
     Address: 2001:DB8:1:1:8902:60D6:E76:6C16
               preferred lifetime INFINITY, , valid lifetime INFINITY,
R1#
```

# DHCP Troubleshooting

DHCP problems can arise for a multitude of reasons, such as software defects in operating systems, NIC drivers, or DHCP relay agents. However, the most common problems are configuration issues.

## Resolving IPv4 Address Conflicts

An IPv4 address lease can expire on a client that is still connected to a network. If the client does not renew the lease, the DHCP server can reassign that IPv4 address to another client. When the client reboots, it requests an IPv4 address. If the DHCP server does not respond quickly, the client uses the last IPv4 address. Then two clients begin using the same IPv4 address, creating a conflict.

The **show ip dhcp conflict** command displays all address conflicts recorded by the DHCP server. The server uses the **ping** command to detect conflicts. The client uses Address Resolution Protocol (ARP) to detect clients. If an address conflict is detected, the address is removed from the pool and is not assigned until an administrator resolves the conflict.

## Testing Connectivity Using a Static IP Address

When troubleshooting any DHCP issue, verify network connectivity by configuring static IPv4 address information on a client workstation. If the workstation cannot reach network resources with a statically configured IPv4 address, the root cause of the problem is not the DHCP server. At this point, network connectivity troubleshooting is required.

## Verifying Switch Port Configuration

If the DHCP client cannot obtain an IPv4 address from the DHCP server at startup, attempt to obtain an IPv4 address from the DHCP server by manually forcing the client to send a DHCP request. If a switch lies between the client and the DHCP server and the client cannot obtain the DHCP configuration, switch port configuration issues might be the cause. These causes can include issues from trunking and channeling to STP and RSTP. PortFast configuration and edge port configurations resolve the most common DHCPv4 client issues that occur with an initial installation of a Cisco switch.

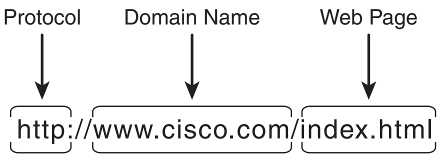
## Testing DHCPv4 Operation on the Same Subnet or VLAN

Distinguishing whether DHCP is functioning correctly is important when the client is on the same subnet or VLAN as the DHCP server. If DHCP is working correctly when the client is on the same subnet or VLAN, the problem might be the DHCP relay agent. If the problem persists even when testing DHCP on the same subnet or VLAN as the DHCP server, the problem might be with the DHCP server.

# DNS Operation

DNS is a distributed system of servers that resolve domain names to IP addresses. The domain name is part of the uniform resource identifier (URI), as Figure 23-8 shows.

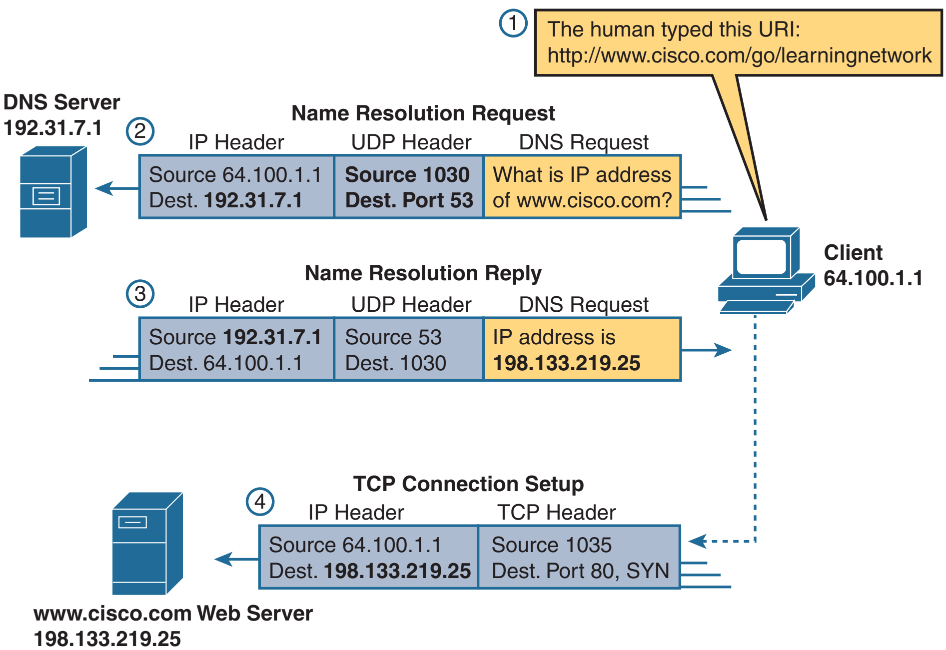
**Figure 23-8 URI Structure**




> **NOTE:** Many people use the terms *web address* and *universal (or uniform) resource locator (URL)*. However, *uniform resource identifier (URI)* is the correct formal term.

When you type a new URI in your browser, your computer uses DNS to send out a request to resolve the URI into an IP address. Figure 23-9 summarizes the DNS process.

**Figure 23-9 DNS Process**




The DNS server stores different types of resource records used to resolve names. These records contain the name, address, and type of record. Some of these record types follow:

- **A:** An end device IPv4 address
- **NS:** An authoritative name server
- **AAAA:** An end device IPv6 address (pronounced "quad-A")
- **MX:** A mail exchange record

When a client makes a query, the server's DNS process first looks at its own records to resolve the name. If it cannot resolve the name using its stored records, it contacts other servers to resolve the name.

DNS root servers manage the top-domain suffixes, such as these:

- **.com:** Commercial businesses
- **.edu:** Educational organizations
- **.gov:** Government organizations
- **.mil:** Military organizations
- **.net:** Networking organizations, such as ISPs
- **.org:** Noncommercial organizations

Top-level DNS servers also exist for each country code, such as .ca (Canada), .de (Germany), .ru (Russia), and .cn (China).

# Troubleshooting DNS

As a network administrator, your control over DNS issues is limited to two basic issues: DHCP server configurations and DNS server configurations.

In a small branch office, you are most likely using your ISP for all your DNS resolutions. Therefore, all the clients on your network will most likely have the IP address of the default gateway configured as the DNS server, as shown in the **ipconfig /all** output in Example 23-8.

**Example 23-8 DNS Server As the Default Gateway**

```
C:\> ipconfig /all

Windows IP Configuration
<output omitted>

   DHCP Enabled. . . . . . . . . . .   : Yes
   Autoconfiguration Enabled . . . .   : Yes
   IPv4 Address. . . . . . . . . . .   : 10.10.10.2(Preferred)
   Subnet Mask . . . . . . . . . . .   : 255.255.255.0
   Lease Obtained. . . . . . . . . .   : Sunday, November 13, 2016 1:28:51 PM
   Lease Expires . . . . . . . . . .   : Monday, November 14, 2016 1:28:50 PM
   Default Gateway . . . . . . . . .   : 10.10.10.1
   DHCP Server . . . . . . . . . . .   : 10.10.10.1
   DNS Servers . . . . . . . . . . .   :10.10.10.1
```

Therefore, issues with DNS are most likely due to issues with the default gateway router or the connection to your ISP. If you know the IP address of a publicly available server, you can verify that DNS is the issue if you can ping the IP address but not the URI.

In larger organizations, the network administrator is responsible for making sure the DHCP server is configured with accurate DNS IP addresses. Those DNS servers are most likely managed in-house to reduce the amount of outbound traffic to the public DNS servers. DNS server misconfiguration could be the cause if end-user devices cannot resolve URIs. Therefore, the hierarchy of DNS servers within the organization should ensure that there are backup DNS servers and that, when a record doesn't exist, the DNS server can accurately forward the request to another DNS server.

# Verifying Host IP Configuration

Whether manually configured or dynamically learned, every device on the network must have a valid IP address configuration. The following are some examples of those settings on Windows, Linux, and macOS.

## IP Settings

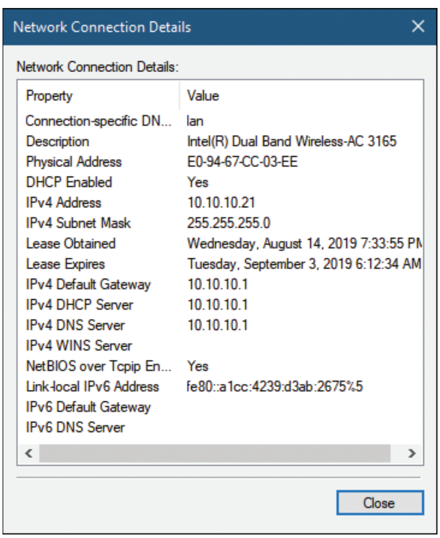
To work correctly, an IP host needs to know these values:

- DNS server IP addresses
- Default gateway (router) IP address
- Device's own IP address
- Device's own subnet mask

## Host IP Settings on Windows

In Windows 10, you can access IP address details from the Network and Sharing Center, as shown in Figure 23-10, to quickly view the four important settings: address, mask, router, and DNS.

**Figure 23-10 Windows 10 Network Connection Details**




However, network administrators typically open a command line window to verify IP settings. All Windows versions support **ipconfig** and **ipconfig /all**, as shown in Example 23-9. Both list the address, mask, and default gateway. But you need **ipconfig /all** to see DNS server settings.

Also notice in Example 23-9 that the Ethernet adapter does not have a default gateway. That is because the computer is currently using the wireless adapter for network connectivity.

**Example 23-9 Windows Command Line IP Settings Verification**

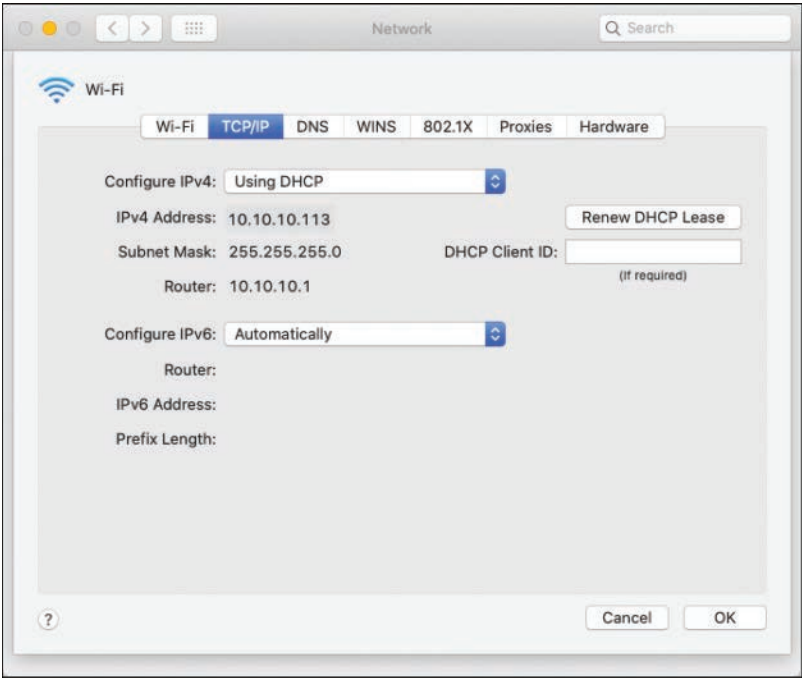
```
C:\> ipconfig
<some output omitted>

Windows IP Configuration

Ethernet adapter Ethernet 2:

   Connection-specific DNS Suffix  . : cisco.com
   IPv6 Address. . . . . . . . . . . : 2001:db8:acad:1008::3d
   Link-local IPv6 Address . . . . . : fe80::ad66:4abd:d554:f703%20
   IPv4 Address. . . . . . . . . . . : 10.24.247.53
   Subnet Mask . . . . . . . . . . . : 255.255.255.255
   Default Gateway . . . . . . . . . : ::
                                         0.0.0.0

Wireless LAN adapter Wi-Fi:

   Connection-specific DNS Suffix  . : lan
   Link-local IPv6 Address . . . . . : fe80::90cb:adf9:9331:8ded%13
   IPv4 Address. . . . . . . . . . . : 10.10.10.73
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . : 10.10.10.1


C:\> ipconfig /all
<some output omitted>

Windows IP Configuration

   Host Name . . . . . . . . . . . . : ALLANJ
   Primary Dns Suffix  . . . . . . . : cisco.com
   Node Type . . . . . . . . . . . . : Hybrid
   IP Routing Enabled. . . . . . . . : No
   WINS Proxy Enabled. . . . . . . . : No
   DNS Suffix Search List. . . . . . : cisco.com
                                       lan
 Wireless LAN adapter Wi-Fi:

   Connection-specific DNS Suffix  . : lan
   Description . . . . . . . . . . . : Intel(R) Dual Band Wireless-AC 8265
```

```
    Physical Address. . . . . . . . . . : 88-B1-11-77-4A-D9
    DHCP Enabled. . . . . . . . . . . . : Yes
    Autoconfiguration Enabled . . . . . : Yes
    Link-local IPv6 Address . . . . . . : fe80::90cb:adf9:9331:8ded%13(Preferred)
    IPv4 Address. . . . . . . . . . . . : 10.10.10.73(Preferred)
    Subnet Mask . . . . . . . . . . . . : 255.255.255.0
    Lease Obtained. . . . . . . . . . . : Saturday, August 31, 2019 12:17:12 PM
    Lease Expires . . . . . . . . . . . : Tuesday, September 3, 2019 11:03:11 AM
    Default Gateway . . . . . . . . . . : 10.10.10.1
    DHCP Server . . . . . . . . . . . . : 10.10.10.1
    DHCPv6 IAID . . . . . . . . . . . . : 92844305
    DHCPv6 Client DUID. . . . . . . . . : 00-01-00-01-21-8E-02-90-54-E1-AD-83-2C-77
    DNS Servers . . . . . . . . . . . . : 10.10.10.1
    NetBIOS over Tcpip. . . . . . . . . : Enabled

C:\Users\allan>
```

## Host IP Settings on macOS

On a Mac, open **Network Preferences > Advanced** to get the IP addressing information shown in Figure 23-11. The router IP address is the default gateway and also serves as the DNS server for this device.

**Figure 23-11 macOS Network Settings**



On the command line for macOS, use the **ifconfig** command to view IP addressing information, as shown in Example 23-10. Other useful commands include **networksetup -listallnetworkservices**, **networksetup -getinfo** <*network service*>, and **networksetup -getdnsservers** <*network service*> (not shown).

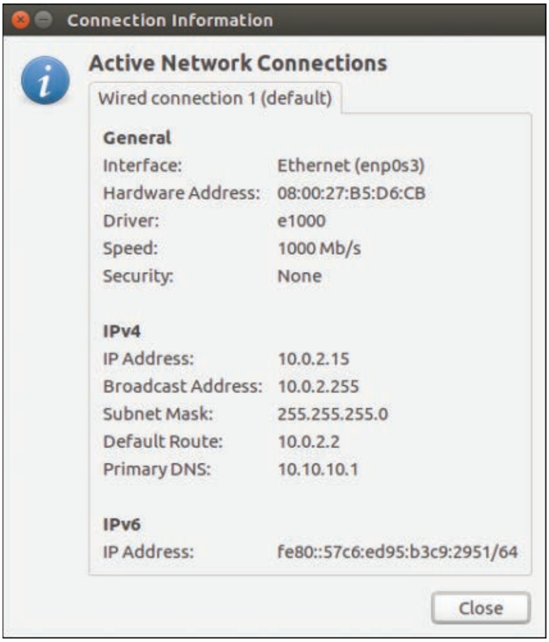
**Example 23-10 macOS Command Line IP Settings Verification**

```
MacBook-Air:~ Becky$ ifconfig en0
en0: flags=8863 mtu 1500
        ether c4:b3:01:a0:64:98
        inet6 fe80::c0f:1bf4:60b1:3adb%en0 prefixlen 64 secured scopeid 0x5
        inet 10.10.10.113 netmask 0xffffff00 broadcast 10.10.10.255
        nd6 options=201
        media: autoselect
        status: active
MacBook-Air:~ Becky$ networksetup -listallnetworkservices
An asterisk (*) denotes that a network service is disabled.
iPhone USB
Wi-Fi
Bluetooth PAN
Thunderbolt Bridge
MacBook-Air:~ Becky$ networksetup -getinfo Wi-Fi
DHCP Configuration
IP address: 10.10.10.113
Subnet mask: 255.255.255.0
Router: 10.10.10.1
Client ID:
IPv6: Automatic
IPv6 IP address: none
IPv6 Router: none
Wi-Fi ID: c4:b3:01:a0:64:98
MacBook-Air:~ Becky$
```

## Host IP Settings on Linux

Verifying IP settings using the GUI on a Linux machine differs depending on the Linux distribution and desktop interface. Figure 23-12 shows the Connection Information dialog box on the Ubuntu distro running the Gnome desktop.

**Figure 23-12 Linux Ubuntu Connection Information**




Example 23-11 shows the commands to verify the IP settings on a Linux machine.

**Example 23-11 Linux OS Command Line IP Settings Verification**

```
allan@allan-VirtualBox:~$ ifconfig enp0s3
enp0s3    Link encap:Ethernet  HWaddr 08:00:27:b5:d6:cb
          inet addr:10.0.2.15  Bcast:10.0.2.255  Mask:255.255.255.0
          inet6 addr: fe80::57c6:ed95:b3c9:2951/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:1332239 errors:0 dropped:0 overruns:0 frame:0
          TX packets:105910 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:1855455014 (1.8 GB)  TX bytes:13140139 (13.1 MB)

allan@allan-VirtualBox:~$ ip address
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
  qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
```

```
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP
  group default qlen 1000
    link/ether 08:00:27:b5:d6:cb brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic enp0s3
       valid_lft 86130sec preferred_lft 86130sec
    inet6 fe80::57c6:ed95:b3c9:2951/64 scope link
       valid_lft forever preferred_lft forever
```

# Study Resources

For today's exam topics, refer to the following resources for more study.

| Resource | Module or Chapter |
|---|---|
| Cisco Network Academy: CCNA 1 | 4 |
| | 6 |
| | 7 |
| CCNA 200-301 Official Cert Guide, Volume 1 | 5 |
| | 8 |
| Portable Command Guide | 17 |

# Day 22

# Wireless Concepts

## CCNA 200-301 Exam Topics

- Explain the role and function of network components
- Describe wireless principles
- Compare Cisco Wireless Architectures and AP modes
- Describe physical infrastructure connections of WLAN components (AP, WLC, access/trunk ports, LAG)
- Describe AP and WLC management access connections (Telnet, SSH, HTTP, HTTPS, console, TACACS+/Radius)
- Describe wireless security protocols (WPA, WPA2, and WPA3)

## Key Topics

Wireless specifications are detailed in the IEEE 802.11 family of standards, including wireless topologies, spectrum allocation, and wireless security. Today we review basic wireless network concepts.

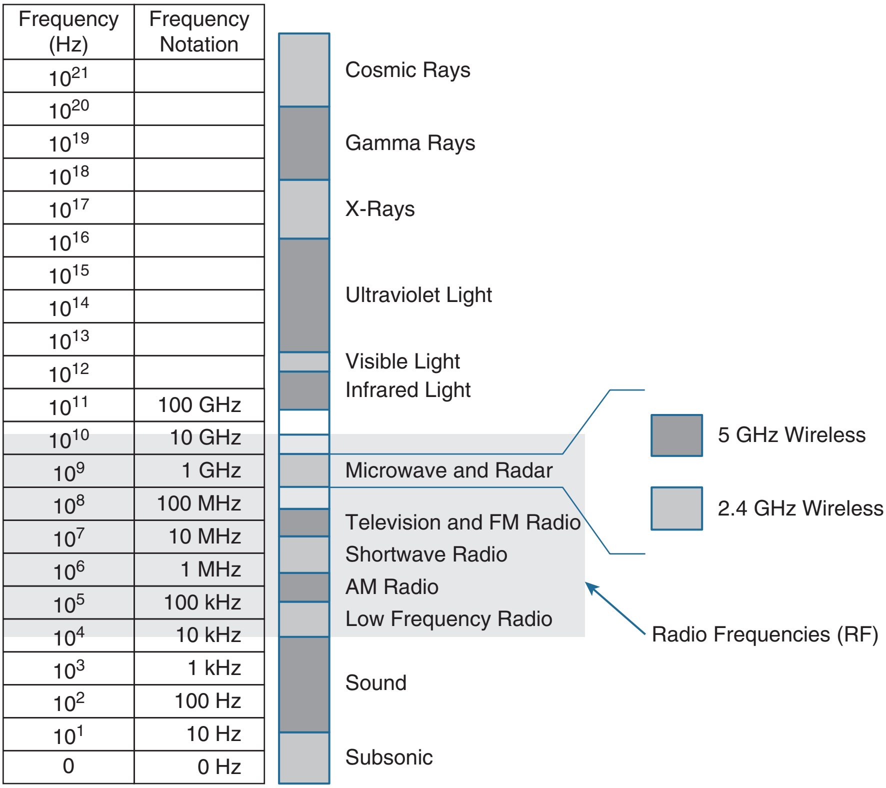
## Wireless Standards

The IEEE 802.11 WLAN standards define how radio frequencies (RFs) are used for wireless links. To avoid interference, different channels within an RF can be used.

### RF Spectrum

The RF spectrum, shown in Figure 22-1, includes all types of radio communications, including the 2.4-GHz and 5-GHz frequencies used by wireless devices.

**Figure 22-1 RF Spectrum**




## Channels

A frequency range is typically called a *band* of frequencies. For example, a wireless LAN device with a 2.4-GHz antenna can actually use any frequency from 2.4000 to 2.4835 GHz. The 5-GHz band lies between 5.150 and 5.825 GHz.

The bands are further subdivided into frequency channels. Channels become particularly important when the wireless devices in a specific area become saturated. Each channel is known by a channel number and is assigned to a specific frequency. As long as the channels are defined by a national or international standards body, they can be used consistently in all locations. Figure 22-2 and Figure 22-3 show the channel layouts for the 2.4- and 5-GHz bands, respectively.

**Figure 22-2 2.4-GHz Channels**

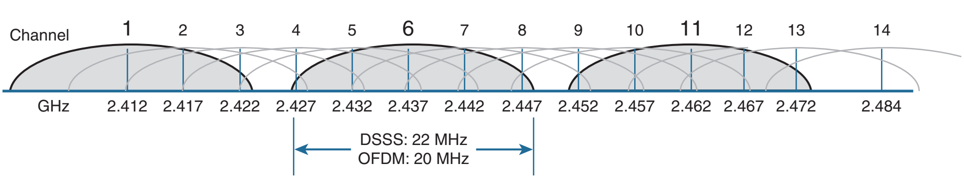
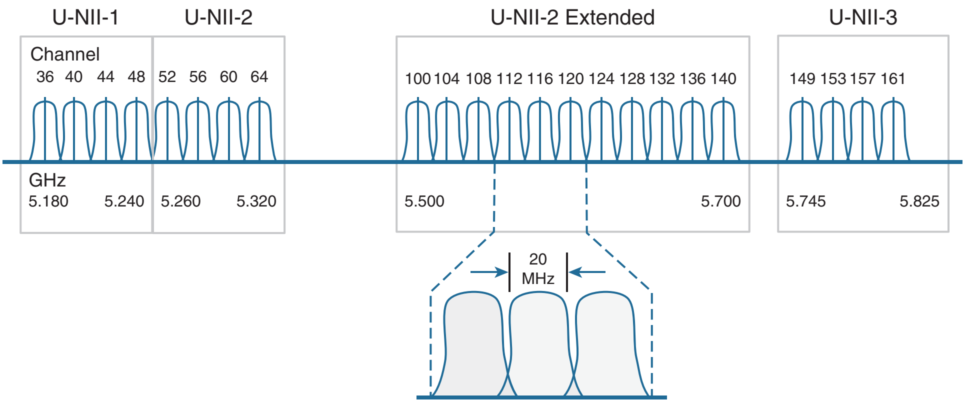


**Figure 22-3 5-GHz Channels**




Notice in Figure 22-3 that the 5-GHz band consists of nonoverlapping channels. Each channel is allocated a frequency range that does not encroach on or overlap the frequencies allocated for any other channel. The same is not true of the 2.4-GHz band in Figure 22-2. The only way to avoid any overlap between adjacent channels is to configure access points (APs) to use only channels 1, 6, and 11.

## 802.11 Standards

Most of the standards specify that a wireless device must have one antenna to transmit and receive wireless signals on the specified radio frequency (2.4 GHz or 5 GHz). Some of the newer standards that transmit and receive at higher speeds require APs and wireless clients to have multiple antennas using the multiple input, multiple output (MIMO) technology. MIMO uses multiple antennas as both the transmitter and receiver to improve communication performance. Up to four antennas can be supported.

Various implementations of the IEEE 802.11 standard have been developed over the years. Table 22-1 highlights these standards.

**Table 22-1 Summary of 802.11 Standards**

| IEEE WLAN Standard | Radio Frequency | Description |
|---|---|---|
| 802.11 | 2.4 GHz | Speeds of up to 2 Mbps |
| 802.11a | 5 GHz | Speeds of up to 54 Mbps<br>Small coverage area<br>Less effective at penetrating building structures<br>Not interoperable with 802.11b and 802.11g |

| IEEE WLAN Standard | Radio Frequency | Description |
|---|---|---|
| 802.11b | 2.4 GHz | Speeds of up to 11 Mbps<br>Longer range than 802.11a<br>Better able to penetrate building structures |
| 802.11g | 2.4 GHz | Speeds of up to 54 Mbps<br>Backward compatible with 802.11b with reduced bandwidth capacity |
| 802.11n | 2.4 GHz<br>5 GHz | Data rates ranging from 150 Mbps to 600 Mbps with a distance range of up to 70 m (230 feet)<br>APs and wireless clients require multiple antennas using MIMO technology<br>Backward compatible with 802.11a/b/g devices with limiting data rates |
| 802.11ac | 5 GHz | Provides data rates ranging from 450 Mbps to 1.3 Gbps (1300 Mbps) using MIMO technology<br>Up to eight antennas can be supported<br>Backward compatible with 802.11a/n devices with limiting data rates |
| 802.11ax | 2.4 GHz<br>5 GHz | Released in 2019 (latest standard)<br>Also known as High-Efficiency Wireless (HEW)<br>Higher data rates and increased capacity<br>Handles many connected devices<br>Improved power efficiency<br>1 GHz and 7 GHz capable when those frequencies become available |

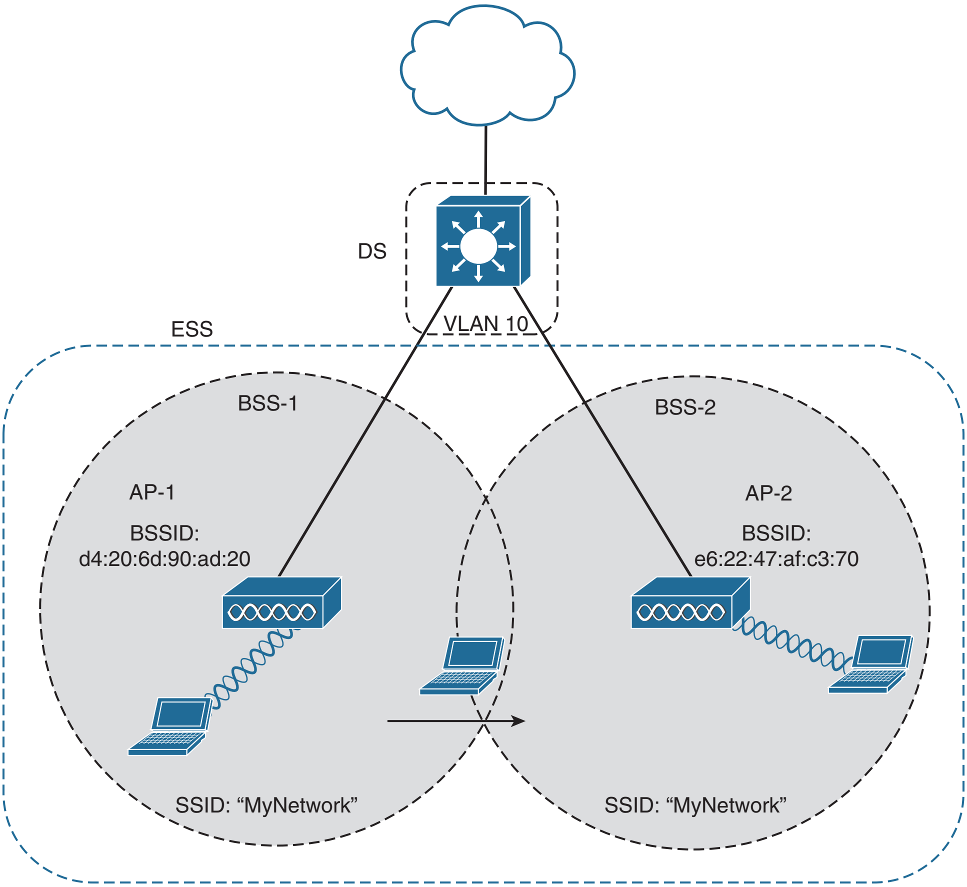
# Wireless Topologies

The 802.11 standard identifies two main wireless topology modes: infrastructure mode and Independent Basic Service Set (IBSS). IBSS is also knows as ad hoc mode. With the ubiquity of wireless networks, mesh topologies are now common.

## Infrastructure Mode

With infrastructure mode, wireless clients interconnect via an AP. Figure 22-4 illustrates infrastructure mode terminology. Notice that the configuration of the APs to share the same SSID allows wireless clients to roam between BSAs.

**Figure 22-4 Example of ESS Infrastructure Mode**




Infrastructure mode terminology includes the following:

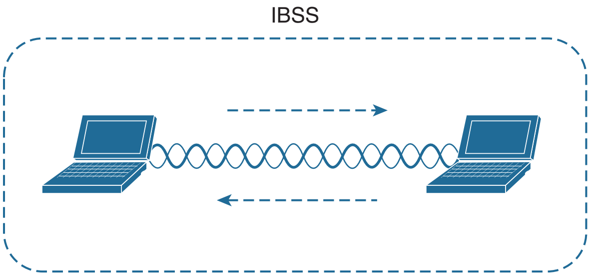
- **Basic service set (BSS):** This consists of a single AP interconnecting all associated wireless clients.
- **Basic service area (BSA):** This is the area that is bound by the reach of the AP's signal. The BSA is also called a *cell* (the gray area in Figure 22-4).
- **Basic service set identifier (BSSID):** This is the unique, machine-readable identifier for the AP that is in the format of a MAC address and is usually derived from the AP's wireless MAC address.
- **Service set identifier (SSID):** This is a human-readable, non-unique identifier used by the AP to advertise its wireless service.
- **Distribution system (DS):** APs connect to the network infrastructure using the wired DS, such as Ethernet. An AP with a wired connection to the DS is responsible for translating frames between 802.3 Ethernet and 802.11 wireless protocols.

- **Extended service set (ESS):** When a single BSS provides insufficient coverage, two or more BSSs can be joined through a common DS into an ESS. An ESS is the union of two or more BSSs interconnected by a wired DS. Each ESS is identified by its SSID, and each BSS is identified by its BSSID.

## IBSS, or Ad Hoc Mode

In the 802.11 standard, Independent Basic Service Set (IBSS) is defined as two devices connected wirelessly in a peer-to-peer (P2P) manner without the use of an AP. One device takes the role of advertising the wireless network to clients. The IBSS allows two devices to communicate directly without the need for any other wireless devices, as shown in Figure 22-5. IBSSs do not scale well beyond 8 to 10 devices.

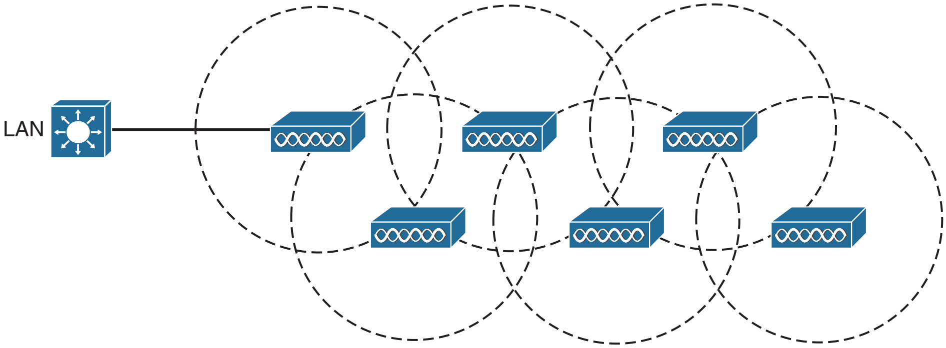
**Figure 22-5 802.11 Independent Basic Service Set**




## Mesh

Having a wired DS connecting all APs is not always practical or necessary. Instead, APs can be configured to connect in mesh mode. In this mode, APs bridge client traffic between each other, as shown in Figure 22-6.

**Figure 22-6 Example of a Wireless Mesh Network**




Each AP in the mesh maintains a BSS on one channel used by wireless clients. Then the APs bridge between each other using other channels. The mesh network runs its own dynamic routing protocol to determine the best path to the wired network.

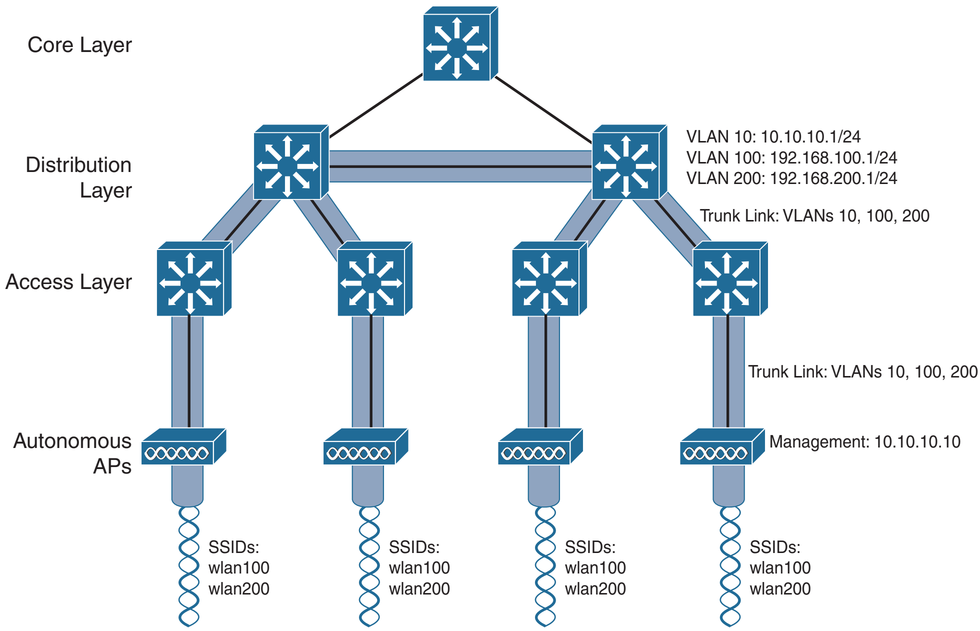
# AP Architectures

APs can be networked together in a variety of architectures. The size and scalability of the network determine which architecture is most suited for a given implementation.

## Autonomous AP Architecture

An autonomous AP is a self-contained device with both wired and wireless hardware so that it can bridge to the wired VLAN infrastructure wireless clients that belong to SSIDs, as shown in Figure 22-7. Each autonomous AP must be configured with a management IP address so that it can be remotely accessed using Telnet, SSH, or a web interface. Each AP must be individually managed and maintained unless you use a management platform such as Cisco DNA Center.

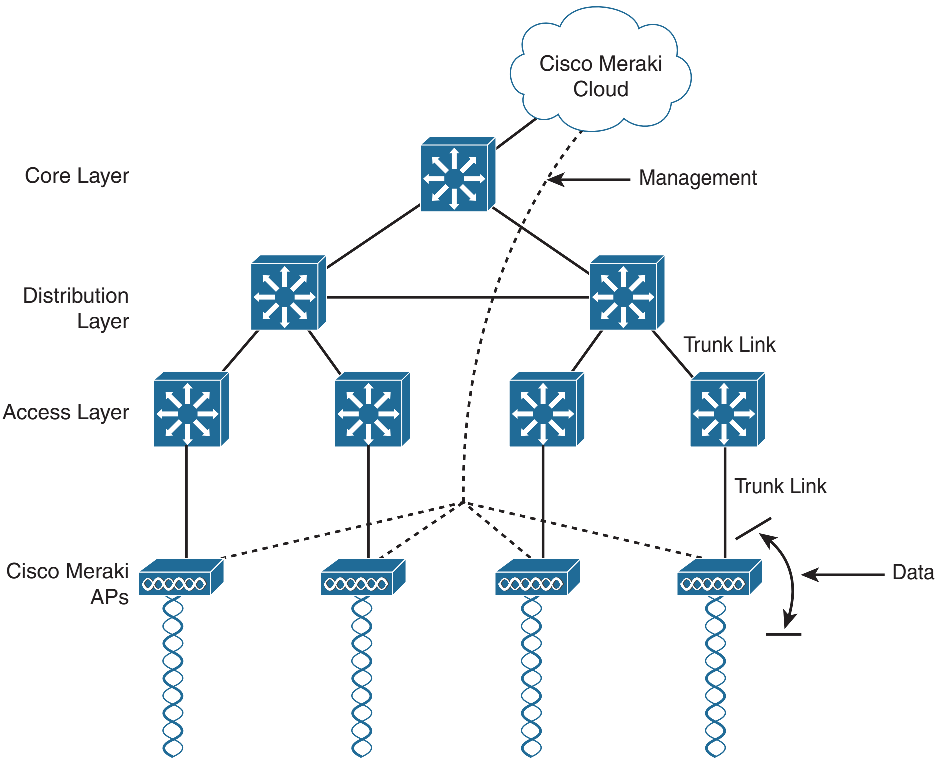
**Figure 22-7 Autonomous APs**




## Cloud-Based AP Architecture

Cloud-based AP management is an alternative to purchasing a management platform. The AP management function is pushed into the Internet cloud. For example, Cisco Meraki is a cloud-based AP management service that allows you to automatically deploy Cisco Meraki APs. These APs can then be managed from the Meraki cloud web interface (dashboard). In Figure 22-8, the same APs shown in Figure 22-7 are now managed in the cloud.

**Figure 22-8 Cisco Meraki Cloud-Based AP Management**



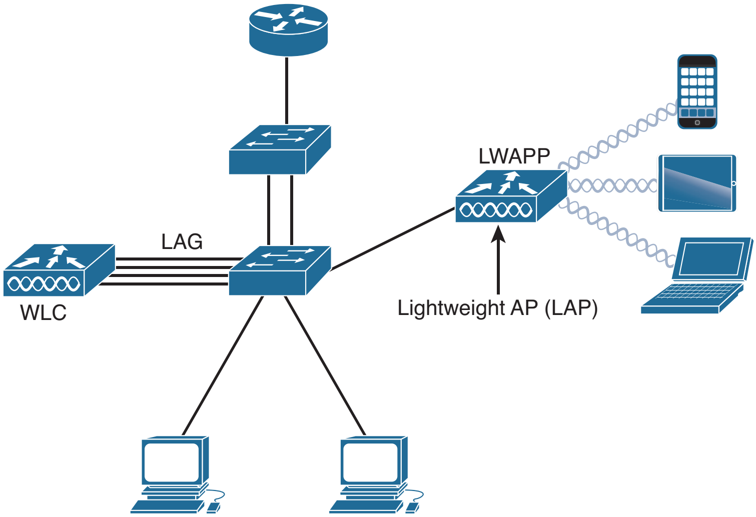

Notice that there are two distinct paths for data traffic and for management traffic, corresponding to the following two functions:

- **A control plane:** Traffic used to control, configure, manage, and monitor the AP itself
- **A data plane:** End-user traffic passing through the AP

## Lightweight AP Architectures

Wireless LAN controllers (WLCs) use Lightweight Access Point Protocol (LWAPP) to communicate with lightweight APs (LAPs), as shown in Figure 22-9. LAPs are useful in situations where many APs are required in the network. They are "lightweight" because they only perform the 802.11 wireless operation for wireless clients. Each LAP is automatically configured and managed by the WLC.

**Figure 22-9 Controller-Based AP Architecture**




Notice in Figure 22-9 that the WLC has four ports connected to the switching infrastructure. These four ports are configured as a link aggregation group (LAG) so they can be bundled together. Much like EtherChannel, LAG provides redundancy and load balancing.

## CAPWAP Operation

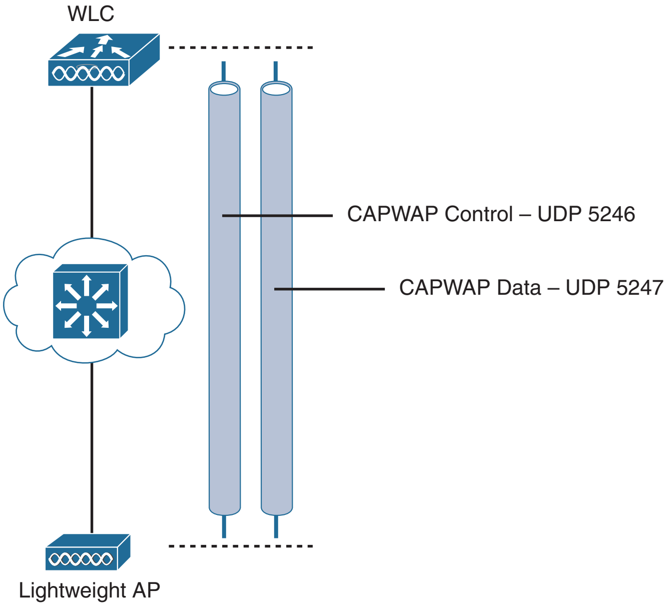
The division of labor between the WLC and LAPs is known as *split-MAC architecture*. The LAP must interact with wireless clients on some low level, known as the Media Access Control (MAC) layer. These functions must stay with the LAP hardware, closest to the clients. The management functions are not integral to handling frames but are things that should be centrally administered. Therefore, those functions can be moved to a centrally located platform away from the AP. Table 22-2 summarizes MAC functions of the LAP and WLC.

**Table 22-2 Split-MAC Functions of the AP and WLC**

| AP MAC Functions | WLC MAC Functions |
|---|---|
| Beacons and probe responses | Authentication |
| Packet acknowledgments and retransmissions | Association and re-association of roaming clients |
| Frame queueing and packet prioritization | Frame translation to other protocols |
| MAC layer data encryption and decryption | Termination of 802.11 traffic on a wired interface |

LWAPP has been replaced with the Control and Provisioning of Wireless Access Points (CAPWAP) tunneling protocol to implement these split-MAC functions. CAPWAP uses two tunnels—one for control and one for data—as shown in Figure 22-10 and described in the list that follows:

**Figure 22-10 CAPWAP Control and Data Tunnels**




- **CAPWAP control message tunnel:** Carries exchanges that are used to configure the LAP and manage its operation. The control messages are authenticated and encrypted, so the LAP is securely controlled by only the appropriate WLC and then transported over the control tunnel using UDP port 5246.
- **CAPWAP data tunnel:** Used for packets traveling to and from wireless clients that are associated with the AP. Data packets are transported over the data tunnel using UDP port 5247 but are not encrypted by default. When data encryption is enabled for a LAP, packets are protected with Datagram Transport Layer Security (DTLS).

# Wireless Security Protocols

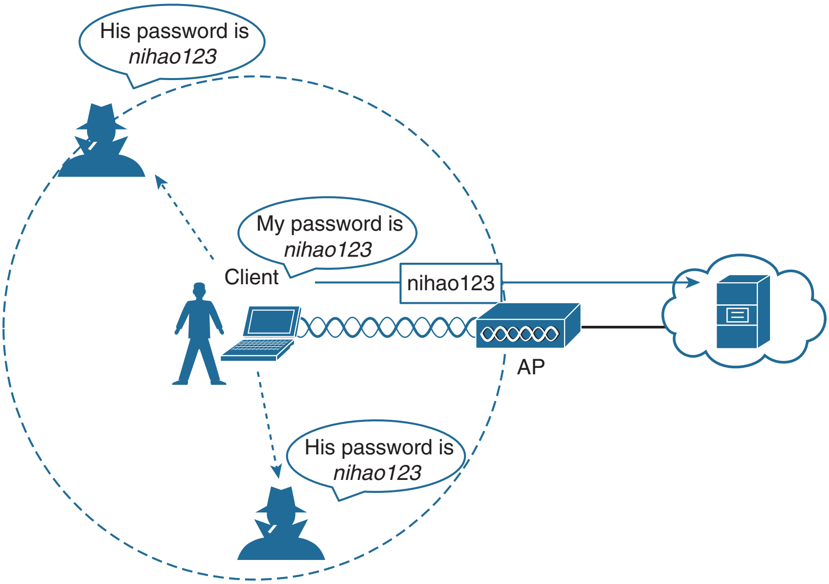
Wireless traffic is inherently different from traffic traveling over a wired infrastructure. Any wireless device operating in the same frequency can hear the frames and potentially read them. Therefore, WLANs need to be secured to allow only authorized users and devices and to prevent eavesdropping and tampering of wireless traffic.

## Wireless Authentication Methods

For wireless devices to communicate over a network, they must first associate with the AP. An important part of the 802.11 process is discovering a WLAN and subsequently connecting to it.

During this process, transmitted frames can reach any device within range. If the wireless connection is not secured, then others can read the traffic, as shown in Figure 22-11.

**Figure 22-11 Open Wireless Network**




The best way to secure a wireless network is to use authentication and encryption systems.

Two types of authentication were introduced with the original 802.11 standard:

- **Open system authentication:** Should only be used in situations where security is of no concern. The wireless client is responsible for providing security such as by using a virtual private network (VPN) to connect securely.
- **Shared key authentication:** Provides mechanisms shown in Table 22-3 to authenticate and encrypt data between a wireless client and an AP. However, the password must be pre-shared between the parties to allow connection.

**Table 22-3 Shared Key Authentication Methods**

| Authentication Method | Description |
|---|---|
| Wired Equivalent Privacy (WEP) | The original 802.11 specification designed to secure the data using the Rivest Cipher 4 (RC4) encryption method with a static key. However, the key never changes when exchanging packets. This makes WEP easy to hack. WEP is no longer recommended and should never be used. |
| Wi-Fi Protected Access (WPA) | A Wi-Fi Alliance standard that uses WEP but secures the data with the much stronger Temporal Key Integrity Protocol (TKIP) encryption algorithm. TKIP changes the key for each packet, making it much more difficult to hack. |

| Authentication Method | Description |
|---|---|
| WPA2 | The current industry standard for securing wireless networks. It uses the Advanced Encryption Standard (AES) for encryption. AES is currently considered the strongest encryption protocol. |
| WPA3 | The next generation of Wi-Fi security. All WPA3-enabled devices use the latest security methods, disallow outdated legacy protocols, and require the use of Protected Management Frames (PMF). However, devices with WPA3 are not yet readily available. |

## WPA and WPA2

Home routers typically have two choices for authentication: WPA and WPA2. WPA2 is the stronger of the two. WPA2 authentication methods included the following:

- **Personal:** Intended for home or small office networks, users authenticate using a pre-shared key (PSK). Wireless clients authenticate with the wireless router using a pre-shared password. No special authentication server is required.
- **Enterprise:** Intended for enterprise networks but requires a Remote Authentication Dial-In User Service (RADIUS) authentication server. Although more complicated to set up, it provides additional security. The device must be authenticated by the RADIUS server, and then users must authenticate using the 802.1X standard, which uses Extensible Authentication Protocol (EAP) for authentication.

## 802.1X/EAP

With open and WEP authentication, wireless clients are authenticated locally at the AP without further intervention. The scenario changes with 802.1X: The client uses open authentication to associate with the AP, and then the client authentication process occurs at a dedicated authentication server. Figure 22-11 shows the three-party 802.1X arrangement, which consists of the following entities:

- **Supplicant:** The client device that is requesting access.
- **Authenticator:** The network device that provides access to the network. In Figure 22-11, the AP forwards the supplicant's message to the WLC.
- **Authentication server (AS):** The device that permits or denies network access based on a user database and policies (usually a RADIUS server).

## WPA3

WPA3 includes four features:

- **WPA3-Personal:** In WPA2-Personal, threat actors can listen in on the "handshake" between a wireless client and the AP and use brute-force attacks to try to guess the PSK. WPA3-Personal thwarts such attacks by using Simultaneous Authentication of Equals (SAE), a feature specified in the IEEE 802.11-2016. The PSK is never exposed, making it impossible for the threat actor to guess.

- **WPA3-Enterprise:** WPA3-Enterprise still uses 802.1X/EAP authentication. However, it requires the use of a 192-bit cryptographic suite and eliminates the mixing of security protocols for previous 802.11 standards. WPA3-Enterprise adheres to the Commercial National Security Algorithm (CNSA) suite, which is commonly used in high-security Wi-Fi networks.
- **Open networks:** Open networks in WPA2 send user traffic in unauthenticated plaintext. In WPA3, open or public Wi-Fi networks still do not use any authentication. However, they do use Opportunistic Wireless Encryption (OWE) to encrypt all wireless traffic.
- **IoT onboarding:** Although WPA2 included Wi-Fi Protected Setup (WPS) to quickly onboard devices that were not previously configured, WPS is vulnerable to a variety of attacks and is not recommended. Furthermore, IoT devices are typically headless, meaning they have no built-in GUI for configuration and need any easy way to get connected to the wireless network. Device Provisioning Protocol (DPP) was designed to address this need. Each headless device has a hard-coded public key. The key is typically stamped on the outside of the device or its packaging as a Quick Response (QR) code. The network administrator can scan the QR code and quickly onboard the device. Although DPP is not strictly part of the WPA3 standard, it will replace WPS over time.

## Wireless Encryption Methods

Encryption is used to protect data. An intruder may be able to captured encrypted data, but he or she would not be able to decipher it in any reasonable amount of time. The following encryption protocols are used with wireless authentication:

- **Temporal Key Integrity Protocol (TKIP):** TKIP is the encryption method used by WPA. It provides support for legacy WLAN equipment and addresses the original flaws associated with the 802.11 WEP encryption method. It makes use of WEP but encrypts the Layer 2 payload using TKIP and carries out a message integrity check (MIC) in the encrypted packet to ensure that the message has not been altered.
- **Advanced Encryption Standard (AES):** AES is the encryption method used by WPA2. It is the preferred method because it is a very strong method of encryption. It uses Counter Cipher Mode with Block Chaining Message Authentication Code Protocol (CCMP), which allows destination hosts to recognize if the encrypted and nonencrypted bits have been altered.
- **The Galois/Counter Mode Protocol (GCMP):** This is a robust authenticated encryption suite that is more secure and more efficient than CCMP. GCMP is used in WPA3.

Table 22-4 summarizes the basic differences between WPA, WPA2, and WPA3. Each successive version is meant to replace prior versions and offer better security features. You should avoid using WPA and use WPA2 instead—at least until WPA3 becomes widely available on wireless client devices, APs, and WLCs.

**Table 22-4 Wireless Authentication and Encryption Comparison**

| Feature | WPA | WPA2 | WPA3 |
|---|---|---|---|
| Authentication with pre-shared keys? | Yes | Yes | Yes |
| Authentication with 802.1X? | Yes | Yes | Yes |
| Encryption and MIC with TKIP? | Yes | No | No |
| Encryption and MIC with AES and CCMP? | Yes | Yes | No |
| Encryption and MIC with AES and GCMP? | No | No | Yes |

# Study Resources

For today's exam topics, refer to the following resources for more study.

| Resource | Module or Chapter |
|---|---|
| Switching, Routing, and Wireless Essentials | 12 |
| CCNA 200-301 Official Cert Guide, Volume 1 | 26 |
| | 27 |
| | 28 |
| Portable Command Guide | 23 |

# WLAN Configuration

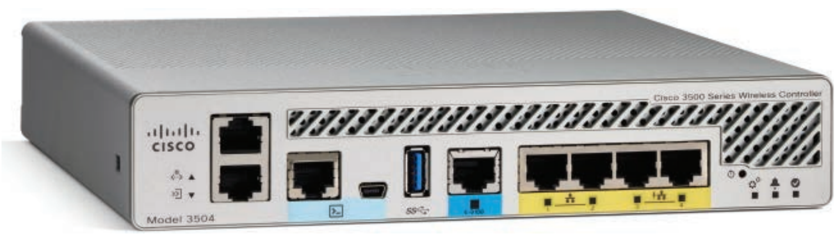
## CCNA 200-301 Exam Topics

- Describe AP and WLC management access connections (Telnet, SSH, HTTP, HTTPS, console, TACACS+/RADIUS)
- Configure the components of a wireless LAN access for client connectivity using GUI only such as WLAN creation, security settings, QoS profiles, and advanced WLAN settings
- Configure WLAN using WPA2 PSK using the GUI

## Key Topics

Today we review the steps to configure a wireless LAN controller (WLC). The figures show the graphical user interface (GUI) and menus from a Cisco 3504 Wireless Controller (see Figure 21-1). However, other WLC models have similar menus and features.

**Figure 21-1 Cisco 3504 Wireless Controller**




## Logging Into a Cisco WLC

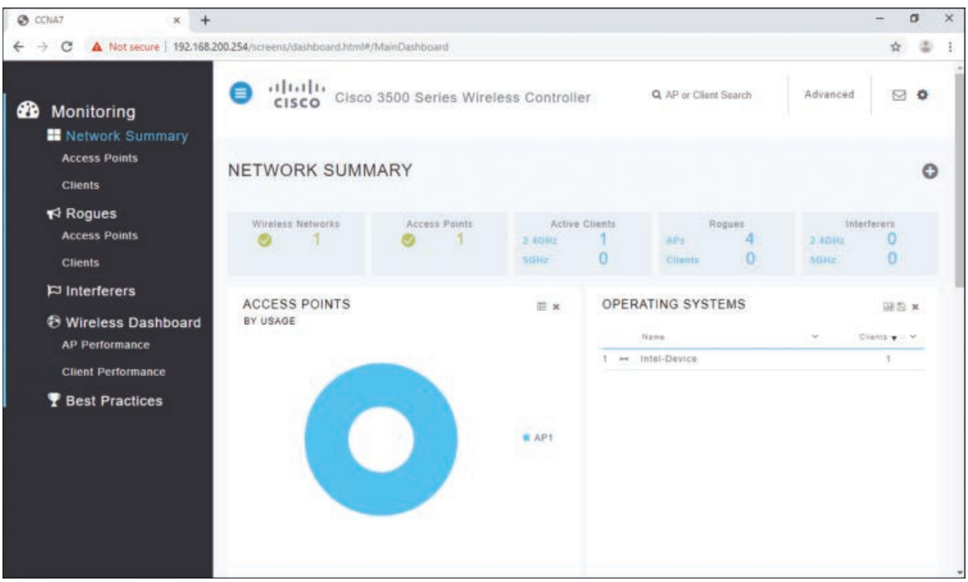
In order to configure a WLC, you need to be able to access it. The WLC requires an initial configuration and a management IP address before you can access it with a web browser through HTTP or HTTPS. This initial configuration requires a console connection. The WLC can also be further configured from the command-line interface (CLI) using Telnet or SSH. However, the CCNA exam focuses on GUI access to the WLC. Therefore, the rest of this day focuses on logging into and configuring a WLC that already has its basic configuration.

Log in to the WLC web interface, as shown in Figure 21-2.

**Figure 21-2 WLC Login Window**

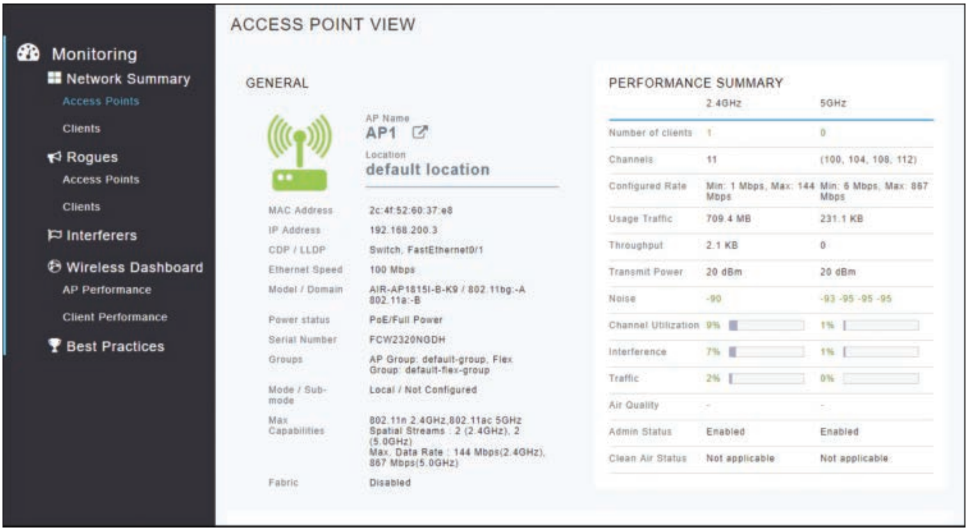

The Network Summary page is a dashboard that provides a quick overview of the number of configured wireless networks, associated access points (APs), and active clients, as shown in Figure 21-3.

**Figure 21-3 Network Summary Dashboard**



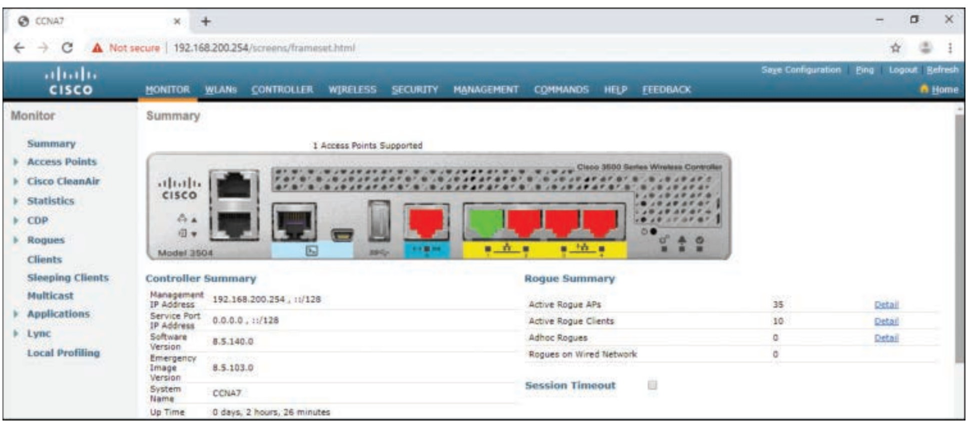

In the menu on the left side of the Network Summary page, click **Access Points** to view an overall picture of AP system information and performance, as shown in Figure 21-4.

**Figure 21-4 Access Points**




Click **Advanced** to access the advanced Summary page, as shown in Figure 21-5. From here, you can access all the features of the WLC.

**Figure 21-5 WLC Advanced Features**



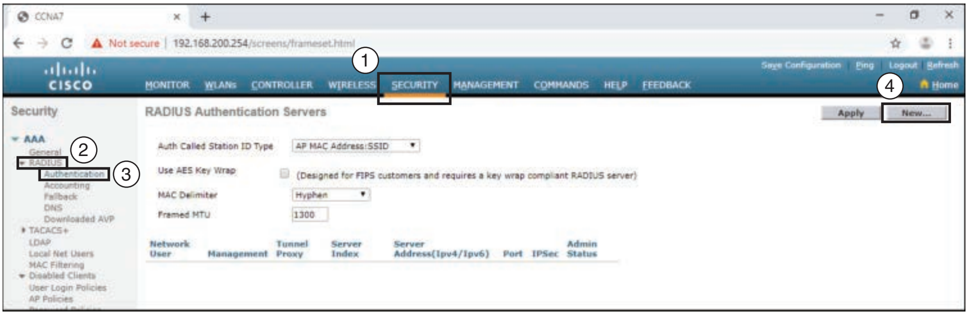

# Configuring a WLC with a WLAN

You can configure a WLAN directly on the Cisco 3504 Wireless Controller so that it serves as an AP for wireless clients. However, a WLC is more commonly used in enterprise networks to manage a number of APs.

## Configuring a RADIUS Server

An enterprise WLAN typically uses a RADIUS server for user and device authentication before allowing wireless clients to associate with an AP. To configure the WLC with the RADIUS server information, click the **SECURITY** tab > **RADIUS > Authentication** to navigate to the screen in Figure 21-6. Click **New** to add the RADIUS server.

**Figure 21-6 Accessing a RADIUS Authentication Server's Configuration**

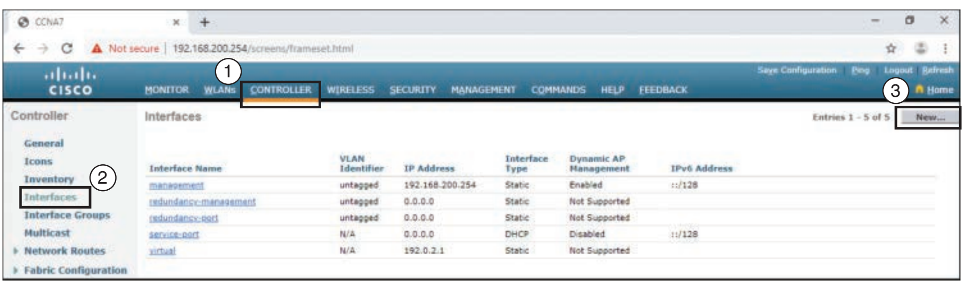



## Configuring a New Interface

Each WLAN configured on the WLC needs its own virtual interface. The WLC has five physical ports for data traffic. Each physical port can be configured to support multiple WLANs, each on its own virtual interface. The virtual interface is typically named with a VLAN number and associated to that VLAN. Use the following steps to configure a new interface:

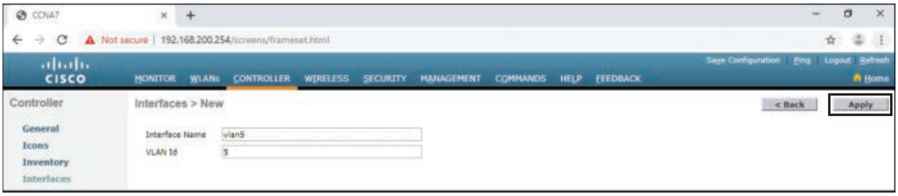
**Step 1.** Create a new interface by clicking **CONTROLLER > Interfaces > New**, as shown in Figure 21-7.

**Figure 21-7 Creating a New Virtual Interface**



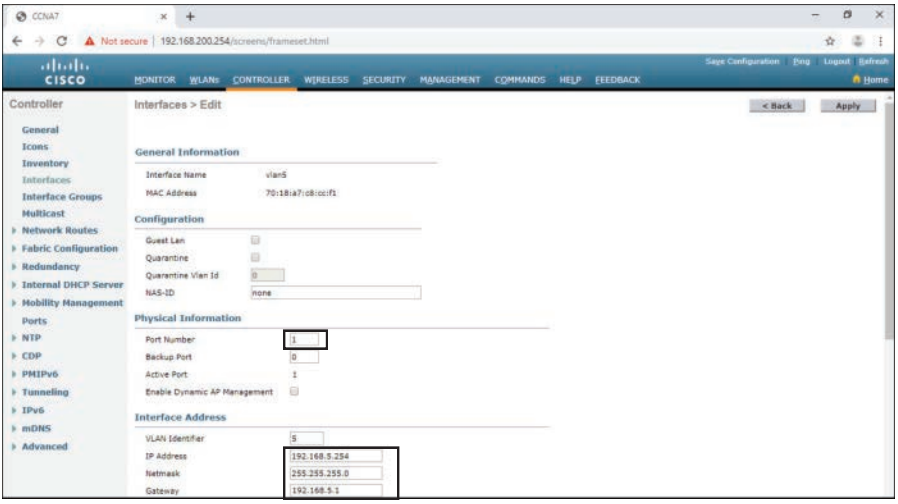

**Step 2.** Configure an interface name and VLAN ID as shown in Figure 21-8, which shows the interface name being set to **vlan5** and the VLAN ID being set to **5**. Click **Apply** to create the new interface.

**Figure 21-8 Configuring the Interface Name and VLAN ID**



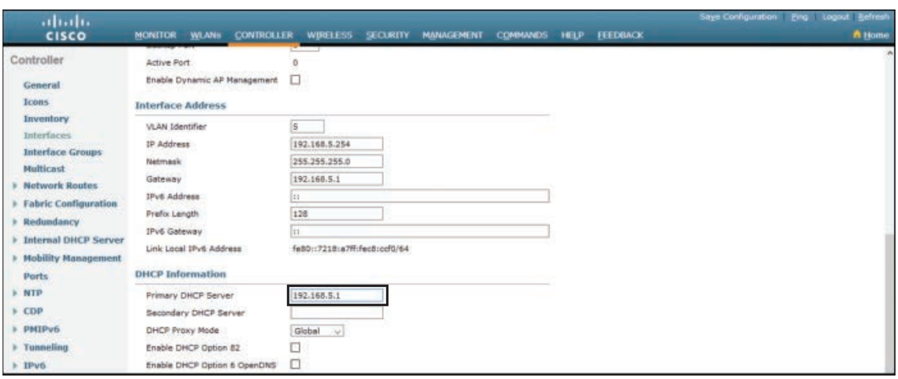

**Step 3.** On the Edit page for the interface, configure the physical port number and IP addressing information (see Figure 21-9).

**Figure 21-9 Configuring Port and IP Addressing**




**Step 4.** In order to forward DHCP messages to a dedicated DHCP server, configure the DHCP server address as shown in Figure 21-10.

**Figure 21-10 Configuring the DHCP Server Address**



**Step 5.** Scroll to the top and click **Apply**, as shown in Figure 21-11. Click **OK** in the warning message.

**Figure 21-11 Applying a New Virtual Interface**


**Step 6.** To verify the newly configured virtual interface, click **Interfaces**. The new vlan5 interface is now shown in the list of interfaces with its IPv4 address, as shown in Figure 21-12.

**Figure 21-12 Verifying a New Virtual Interface**


## Configuring a WPA2 Enterprise WLAN

By default, all newly created WLANs on the WLC use WPA2 with Advanced Encryption System (AES). 802.1X is the default key management protocol used to communicate with the RADIUS server. The WLC is already configured with the IP address of the RADIUS server.

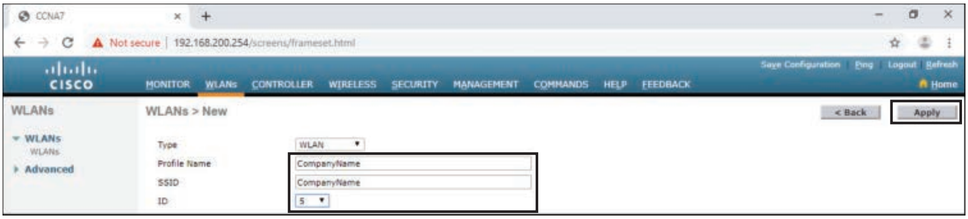
Configuring a new WLAN for interface **vlan5** on the WLC involves the following steps:

**Step 1.** To create a new WLAN, click the **WLANs** tab and then **Go**, as shown Figure 21-13.

**Figure 21-13 Creating a New WLAN**

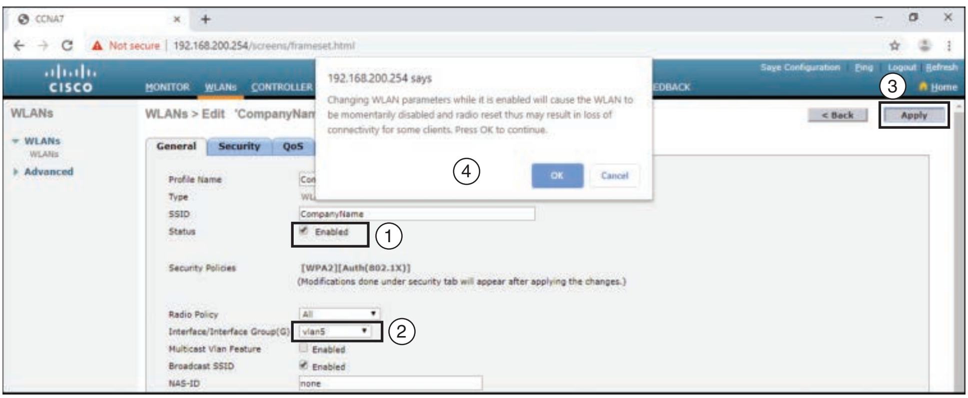
**Step 2.** Configure the WLAN name and SSID. In Figure 21-14, the SSID is also used as the profile name and uses the same ID as vlan5, created earlier.

**Figure 21-14 Setting the Profile Name and SSID**



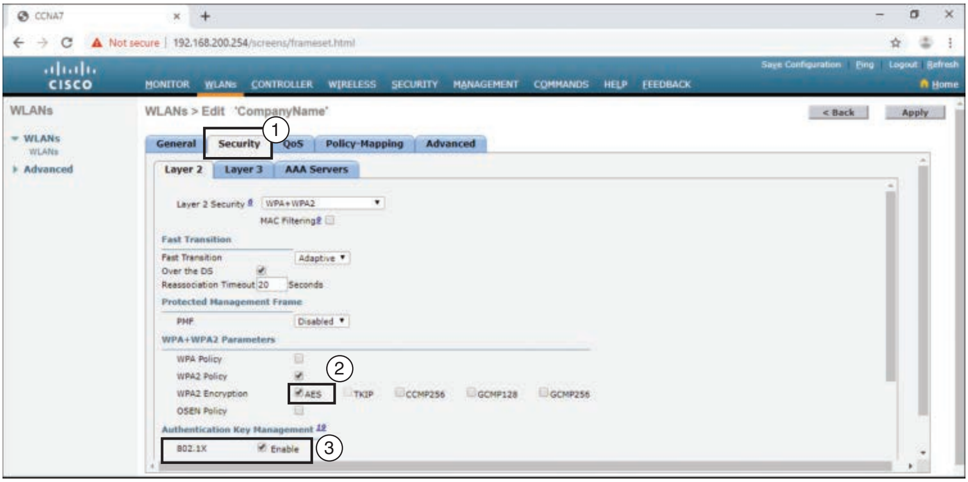

**Step 3.** To enable the WLAN for vlan5, change the status to **Enabled** and choose **vlan5** from the Interface/Interface Group(G) dropdown list. Click **Apply** and click **OK** to accept the popup message, as shown Figure 21-15.

**Figure 21-15 Enabling the WLAN**



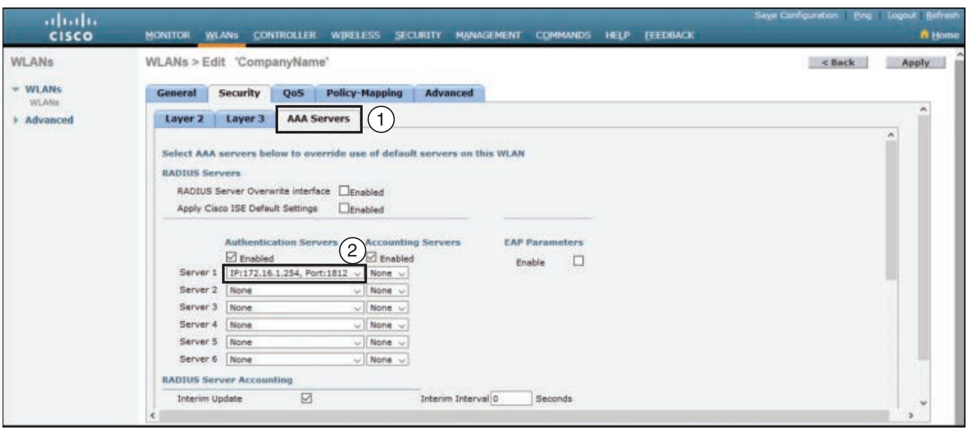

**Step 4.** To verify AES and the 802.1X defaults, click the **Security** tab to view the default security configuration for the new WLAN, as shown in Figure 21-16. The WLAN should use WPA2 security with AES encryption. Authentication traffic is handled by 802.1X between the WLC and the RADIUS server.

**Figure 21-16 Verifying Security**



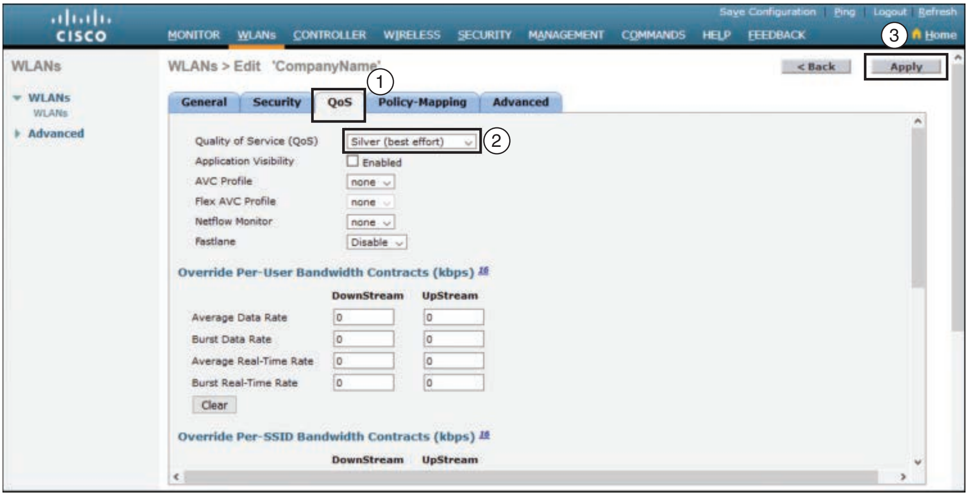

**Step 5.** To configure WLAN security to use the RADIUS server, click the **AAA Servers** tab, as shown in Figure 21-17. In the dropdown box, select the RADIUS server that was configured on the WLC previously.

**Figure 21-17 Associating the RADIUS Server to the WLAN**




**Step 6.** To configure a QoS profile, click the **QoS** tab, as shown in Figure 21-18. From here, you can configure a QoS profile that adheres to the company policy. Silver (best effort) is currently selected. Click **Apply** to apply the changes.

**Figure 21-18 Configuring a QoS Profile**




**Step 7.** To verify that the new WLAN is listed and enabled, click the **WLANs** submenu on the left. In Figure 21-19, notice that the WLAN **CompanyName** is enabled and is using WPA2 security with 802.1X authentication.

**Figure 21-19 Verifying the New WLAN**


# Study Resources

For today's exam topics, refer to the following resources for more study.

| Resource | Module or Chapter |
|---|---|
| Switching, Routing, and Wireless Essentials | 13 |
| CCNA 200-301 Official Cert Guide, Volume 1 | 29 |
| Portable Command Guide | 23 |

*This page intentionally left blank*

# LAN Security and Device Hardening

## CCNA 200-301 Exam Topics

- Configure device access control using local passwords
- Configure network devices for remote access using SSH
- Differentiate authentication, authorization, and accounting concepts
- Configure Layer 2 security features (DHCP snooping, dynamic ARP inspection, and port security)

## Key Topics

Today's review is a whirlwind of topics related to LAN security and device hardening. We will review endpoint security, access control, port security, and LAN threat mitigation techniques.

## Endpoint Security

Endpoints are hosts including laptops, desktops, servers, and IP phones. In addition, a network that has a bring your own device (BYOD) policy includes employee-owned devices. Endpoints are particularly susceptible to malware-related attacks that originate through email or web browsing. If an endpoint is infiltrated, it can become a point from which a threat actor can gain access to critical system devices, such as servers and sensitive information.

Endpoints are best protected by host-based Cisco Advanced Malware Protection (AMP) software. AMP products include endpoint solutions such as Cisco AMP for Endpoints. In addition, content security appliances provide fine-grained control over email and web browsing for an organization's users.

Cisco has two content security appliance products:

- Cisco Email Security Appliance (ESA)
- Cisco Web Security Appliance (WSA)

### Cisco ESA

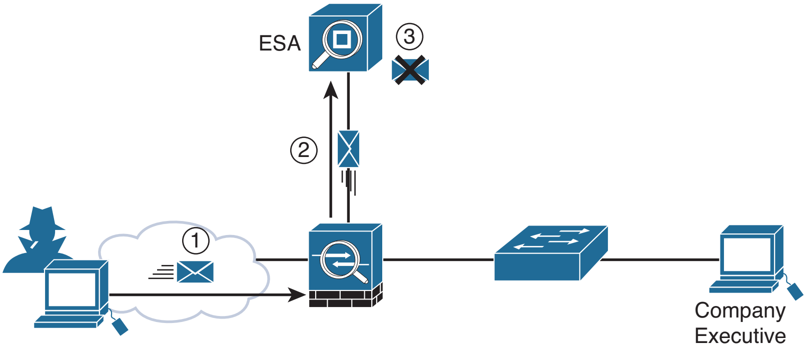
Cisco ESA is special device designed to monitor email's primary protocol, Simple Mail Transfer Protocol (SMTP). Cisco ESA can do the following:

- Block known threats
- Remediate against stealth malware that evades initial detection
- Discard emails with bad links

- Block access to newly infected sites
- Encrypt content in outgoing email to prevent data loss

Figure 20-1 shows the Cisco ESA process of discarding a targeted phishing attack.

**Figure 20-1 Cisco ESA Discards Bad Emails**




The process shown in Figure 20-1 is as follows:

**Step 1.** Threat actor sends a phishing attack to an important host on the network.

**Step 2.** The firewall forwards all email to the ESA.

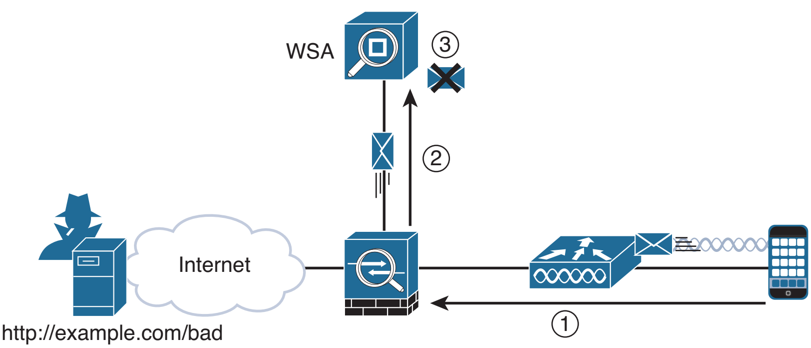
**Step 3.** The ESA analyzes the email, logs it, and discards it.

## Cisco WSA

Cisco WSA combines advanced malware protection, application visibility and control, acceptable use policy controls, and reporting. Cisco WSA provides complete control over how users access the Internet. Certain features and applications, such as chat, messaging, video, and audio can be allowed, restricted with time and bandwidth limits, or blocked, according to the organization's requirements. WSA can perform blacklisting of URLs, URL filtering, malware scanning, URL categorization, Web application filtering, and encryption and decryption of web traffic.

Figure 20-2 shows a corporate user attempting to connect to a known blacklisted site.

**Figure 20-2 Cisco WSA Discard Packet Destined for a Blacklisted Site**



The process shown in Figure 20-2 is as follows:

**Step 1.** A user attempts to connect to a website.

**Step 2.** The firewall forwards the website request to the WSA.

**Step 3.** The WSA evaluates the URL and determines it is a known blacklisted site. The WSA discards the packet and sends an access denied message to the user.

# Access Control

Many types of authentication can be performed on networking devices to control access, and each method offers varying levels of security.

## Local Authentication

The simplest method of remote access authentication is to configure a login and password combination on console, vty lines, and aux ports, as shown in Example 20-1. This method, however, provides no accountability, and the password is sent in plaintext. Anyone with the password can gain entry to the device.

**Example 20-1 Local Password Only Authentication**

```
R1(config)# line vty 0 4
R1(config-line)# password ci5c0
R1(config-line)# login
```

Instead of using a shared password with no usernames, you can use the **username** *username* **secret** *password* command to configure local username/password pairs. Require a username/password pair with the **login local** line configuration command. Use the **no password** line configuration command to remove any configured passwords. In Example 20-2, a username/password pair is configured and applied to the lines, and then Telnet access is tested from a switch. Notice that the password has been hashed using MD5 encryption, indicated by the **5** following **secret** in the output from the **show run** command.

**Example 20-2 Local Username/Password Authentication**

```
R1(config)# username allanj secret 31daysCCNA
R1(config)# line console 0
R1(config-line)# login local
R1(config-line)# no password
R1(config-line)# line vty 0 15
R1(config-line)# login local
R1(config-line)# no password
```

```
S1# telnet 10.10.10.1
```

```
Trying 10.10.10.1 ...Open


User Access Verification

Username: allanj
Password:
R1> enable
Password:
R1# show run | include username
username allanj secret 5 $1$mERr$e/edsAr7D0CyM/z3tMvyL/
R1#
```

## SSH Configuration

Secure Shell (SSH) is considered a security best practice because Telnet (port 23) uses insecure plaintext transmission of both the login and the data across the connection. SSH (port 22) is a more secure form of remote access:

- It requires a username and a password, both of which are encrypted during transmissions.
- The username and password can be authenticated using the local database method.
- It provides more accountability because the username is recorded when a user logs in.

Example 20-3 illustrates SSH and local database methods of remote access.

**Example 20-3 Configuring SSH Remote Access on a Switch**

```
S1# show ip ssh
SSH Disabled-version 1.99
%Please create RSA keys to enable SSH (of at least 768 bits size) to enable SSH v2.
Authentication timeout: 120 secs; Authentication retries:3
S1# conf t
S1(config)# ip domain-name cisco.com
S1(config)# crypto key generate rsa
The name for the keys will be: S1.cisco.com
Choose the size of the key modulus in the range of 360 to 4096 for your
   General Purpose Keys. Choosing a key modulus greater than 512 may take
   a few minutes.

How many bits in the modulus [512]:1024
% Generating 1024 bit RSA keys, keys will be non-exportable...
[OK] (elapsed time was 4 seconds)
```

```
*Mar 1 02:20:18.529: %SSH-5-ENABLED: SSH 1.99 has been enabled
S1(config)# line vty 0 15
S1(config-line)# login local
S1(config-line)# transport input ssh
S1(config-line)# username allanj secret 31daysCCNA
!The following commands are optional SSH configurations.
S1(config)# ip ssh version2
S1(config)# ip ssh authentication-retries 5
S1(config)# ip ssh time-out 60
S1(config)# end
S1# show ip ssh
SSH Enabled - version 2.0
Authentication timeout: 60 secs; Authentication retries: 5
S1#
```

The following steps occur in Example 20-3:

**Step 1.** Verify that the switch supports SSH using the **show ip ssh** command. If the command is not recognized, you know that SSH is not supported.

**Step 2.** Configure a DNS domain name with the **ip domain-name** global configuration command.

**Step 3.** Configure the switch using the **crypto key generate rsa** command to generate an RSA key pair and automatically enable SSH. When generating RSA keys, you are prompted to enter a modulus length. Cisco recommends a minimum modulus size of 1024 bits, as in Example 20-3.

**NOTE:** To remove the RSA key pair, use the **crypto key zeroize rsa** command. This disables the SSH service.

**Step 4.** Change the vty lines to use usernames, with either locally configured usernames or an authentication, authorization, and accounting (AAA) server. In Example 20-3, the **login local** vty subcommand defines the use of local usernames, replacing the **login** vty subcommand.

**Step 5.** Configure the switch to accept only SSH connections with the **transport input** ssh vty subcommand. (The default is **transport input telnet**.)

**Step 6.** Add one or more **username password** global configuration commands to configure username/password pairs.

**Step 7.** If desired, modify the default SSH configuration to change the SSH version to 2.0, the number of authentication tries, and the timeout, as in Example 20-3.

**Step 8.** Verify your SSH parameters by using the **show ip ssh** command.

## Switch Port Hardening

Router interfaces must be activated with the **no shutdown** command before they become operational. The opposite is true for Cisco Catalyst switches: an interface is activated when a device is connected to the port. To provide out-of-the-box functionality, Cisco chose a default configuration that includes interfaces that work without any configuration, including automatically negotiating speed and duplex. In addition, all interfaces are assigned to the default VLAN 1.

This default configuration exposes switches to some security threats. The following are security best practices for unused interfaces:

- Administratively disable the interface by using the **shutdown** interface subcommand.
- Prevent VLAN trunking by making the port a nontrunking interface using the **switchport mode** access interface subcommand.
- Assign the port to an unused VLAN by using the **switchport access vlan** *number* interface subcommand.
- Set the native VLAN to not be VLAN 1 but to instead be an unused VLAN, using the **switchport trunk native vlan** *vlan-id* interface subcommand.

Even when you shut down unused ports on the switches, if a device is connected to one of those ports and the interface is enabled, trunking can occur. In addition, all ports are in VLAN 1 by default. A good practice is to put all unused ports in a black hole VLAN. Example 20-4 demonstrates this best practice, assuming that ports 20–24 are unused.

**Example 20-4 Assigning Unused Ports to a Black Hole VLAN**

```
S1(config)# vlan 999
S1(config-vlan)# name BlackHole
S1(config-vlan)# interface range fa0/20 - 24
S1(config-if-range)# shutdown
S1(config-if-range)# switchport mode access
S1(config-if-range)# switchport access vlan 999
S1(config-if-range)#
```

## AAA

Configuring usernames and passwords on all your network devices is not very scalable. A better option is to use an external server to centralize and secure all username/password pairs. To address this issue, Cisco devices support the authentication, authorization, and accounting (AAA) framework to help secure device access.

Cisco devices support two AAA authentication protocols:

- Terminal Access Controller Access Control System Plus (TACACS+, pronounced as "tack-axe plus")
- Remote Authentication Dial-In User Service (RADIUS)

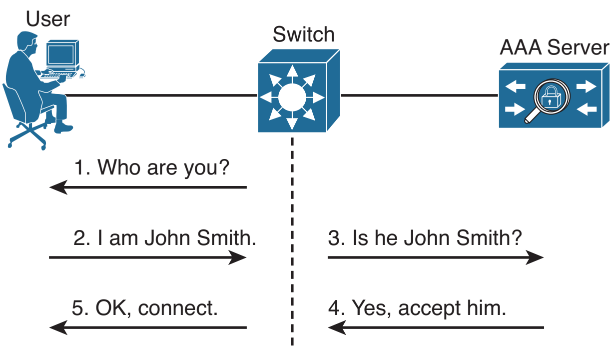
The choice of TACACS+ or RADIUS depends on the needs of the organization. For example, a large ISP might select RADIUS because it supports the detailed accounting required for billing users. An organization with various user groups might select TACACS+ because it requires authorization policies to be applied on a per-user or per-group basis. Table 20-1 compares TACACS+ and RADIUS.

**Table 20-1 Comparison of TACACS+ and RADIUS**

| Feature | TACACS+ | RADIUS |
|---|---|---|
| Most often used for | Network devices | Users |
| Transport protocol | TCP | UDP |
| Authentication port number(s) | 49 | 1645, 1812 |
| Protocol encrypts the password | Yes | Yes |
| Protocol encrypts entire packet | Yes | No |
| Supports function to authorize each user to a subset of CLI commands | Yes | No |
| Defined by | Cisco | RFC 2865 |

Both TACACS+ and RADIUS use a client/server model, where an authenticating device is the client talking to an AAA server. Figure 20-3 shows a simplified view of the process, where a user is attempting to connect to a switch for management purposes.

**Figure 20-3 A Simplified View of AAA**

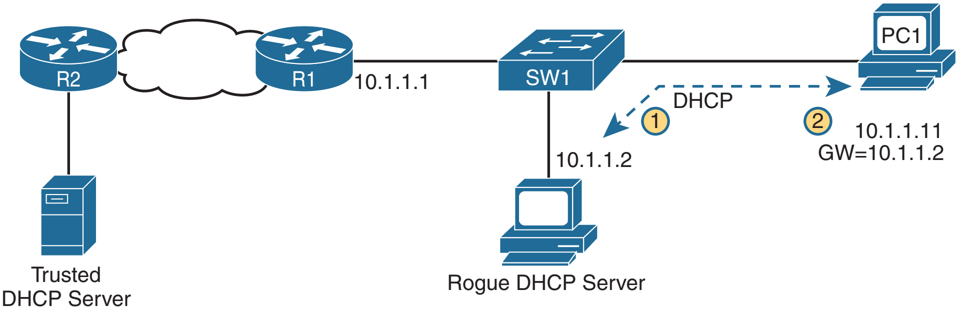



## 802.1X

IEEE 802.1X is a standard port-based access control and authentication protocol. It is ideal for restricting unauthorized access through publicly available LAN devices, such as switches and wireless access points.

802.1X defines three roles for devices in the network, as Figure 20-4 shows:

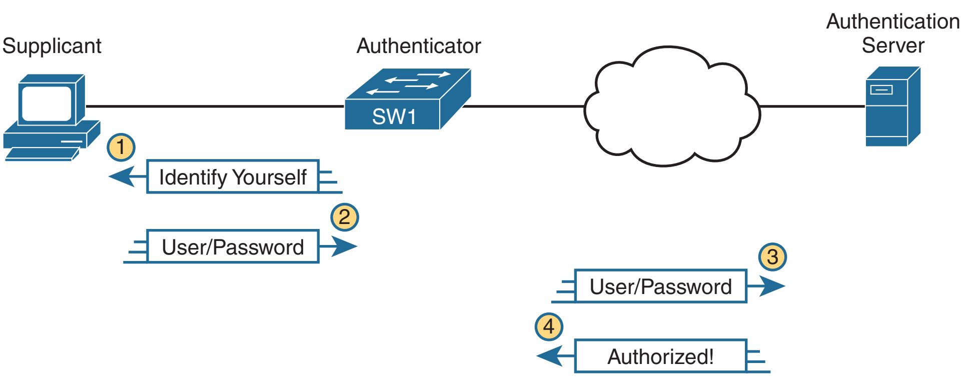
**Figure 20-4 802.1X Roles**




- **Client (supplicant):** This is usually the 802.1X-enabled port on the device that requests access to LAN and switch services and responds to requests from the switch. In Figure 20-4, the device is a PC running 802.1X-compliant client software.
- **Switch (authenticator):** The switch controls physical access to the network, based on the authentication status of the client. The switch acts as a proxy between the client and the authentication server. It requests identifying information from the client, verifies that information with the authentication server, and relays a response to the client.
- **Authentication server:** The authentication server performs the actual authentication of the client. The authentication server validates the identity of the client and notifies the switch about whether the client is authorized to access the LAN and switch services. Because the switch acts as the proxy, the authentication service is transparent to the client. RADIUS is the only supported authentication server.

Figure 20-5 shows the authentication flows for a typical 802.1X process.

**Figure 20-5 802.1X Authentication Flows**




The 802.1X process is summarized as follows:

- The RADIUS authentication server is configured with usernames and passwords.
- Each LAN switch is enabled as an 802.1X authenticators, is configured with the IP address of the authentication server, and has 802.1X enabled on all required ports.

- Users that connect devices to 802.1X-enabled ports must know the username/password before they can access the network.

# Port Security

If you know which devices should be cabled and connected to particular interfaces on a switch, you can use port security to restrict that interface so that only the expected devices can use it. This reduces exposure to some types of attacks in which the attacker connects a laptop to the wall socket or uses the cable attached to another end device to gain access to the network.

## Port Security Configuration

Port security configuration involves several steps. Basically, you need to make the port an access port, which means the port is not doing any VLAN trunking. You then need to enable port security and configure the Media Access Control (MAC) addresses of the devices allowed to use that port. The following list outlines the steps in port security configuration, including the configuration commands used:

**Step 1.** Configure the interface for static access mode by using the **switchport mode access** interface subcommand.

**Step 2.** Enable port security by using the **switchport port-security** interface subcommand.

**Step 3.** (Optional) Override the maximum number of allowed MAC addresses associated with the interface (1) by using the **switchport port-security maximum** *number* interface subcommand.

**Step 4.** (Optional) Override the default action when there is a security violation (shutdown) by using the **switchport port-security violation {protect | restrict | shutdown}** interface subcommand.

**Step 5.** (Optional) Predefine any allowed source MAC address(es) for this interface by using the **switchport port-security mac-address** *mac-address* command. Use the command multiple times to define more than one MAC address.

**Step 6.** (Optional) Instead of taking step 5, configure the interface to dynamically learn and configure the MAC addresses of currently connected hosts by configuring the **switchport port-security mac-address sticky** interface subcommand.

When an unauthorized device attempts to send frames to the switch interface, the switch can issue informational messages, discard frames from that device, or even discard frames from all devices by effectively shutting down the interface. Exactly which action the switch port takes depends on the option you configure in the **switchport port-security violation** command. Table 20-2 lists actions that the switch will take based on whether you configure the option **protect**, **restrict**, or **shutdown** (default).

**Table 20-2 Actions When Port Security Violation Occurs**

| Option on the switchport port-security violation Command | protect | restrict | shutdown |
|---|---|---|---|
| Discards offending traffic | Yes | Yes | Yes |
| Sends log and SNMP messages | No | Yes | Yes |
| Disables the interface, discarding all traffic | No | No | Yes |

Example 20-5 shows a port security configuration in which each access interface is allowed a maximum of three MAC addresses. If a fourth MAC address is detected, only the offending device's traffic is discarded. If the violation option is not explicitly configured, the traffic for devices that are allowed on the port also is discarded because the port would be shut down by default.

**Example 20-5 Port Security Configuration Example**

```
S1(config)# interface range fa 0/5 - fa 0/24
S1(config-if-range)# switchport mode access
S1(config-if-range)# switchport port-security
S1(config-if-range)# switchport port-security maximum 3
S1(config-if-range)# switchport port-security violation restrict
S1(config-if-range)# switchport port-security mac-address sticky
```

To verify port security configuration, use the more general **show port-security** command or the more specific **show port-security interface** *type number* command. Example 20-6 demonstrates the use of both commands. In the examples, notice that only one device is currently attached to an access port on S1.

**Example 20-6 Port Security Verification Command Output Examples**

```
S1# show port-security
Secure Port MaxSecureAddr CurrentAddr SecurityViolation Security Action
             (Count)          (Count)       (Count)
Fa0/5        3                1             0                  Restrict
Fa0/6        3                0             0                  Restrict
Fa0/7        3                0             0                  Restrict
Fa0/8        3                0             0                  Restrict
Fa0/9        3                0             0                  Restrict
Fa0/10       3                0             0                  Restrict
Fa0/11       3                0             0                  Restrict
Fa0/12       3                0             0                  Restrict
Fa0/13       3                0             0                  Restrict
Fa0/14       3                0             0                  Restrict
Fa0/15       3                0             0                  Restrict
Fa0/16       3                0             0                  Restrict
Fa0/17       3                0             0                  Restrict
Fa0/18       3                0             0                  Restrict
Fa0/19       3                0             0                  Restrict
```

```
Fa0/20        3              0             0                  Restrict
Fa0/21        3              0             0                  Restrict
Fa0/22        3              0             0                  Restrict
Fa0/23        3              0             0                  Restrict
Fa0/24        3              0             0                  Restrict
Total Addresses in System (excluding one mac per port) : 0
Max Addresses limit in System (excluding one mac per port) : 8320
S1# show port-security interface fastethernet 0/5
Port Security                  :Enabled
Port Status                    :Secure-down
Violation Mode                 :Restrict
Aging Time                     :0 mins
Aging Type                     :Absolute
SecureStatic Address Aging     :Disabled
Maximum MAC Addresses          :3
Total MAC Addresses            :1
Configured MAC Addresses       :0
Sticky MAC Addresses           :1
Last Source Address:Vlan       :0014.22dd.37a3:1
Security Violation Count       :0
```

## Port Security Aging

Port security aging can be used to set the aging time for static and dynamic secure addresses on a port. Two types of aging are supported per port:

- **Absolute:** The secure addresses on the port are deleted after the specified aging time.
- **Inactivity:** The secure addresses on the port are deleted only if they are inactive for the specified aging time.

Use the **switchport port-security aging** command to enable or disable static aging for the secure port or to set the aging time or type:

```
Switch(config-if)# switchport port-security aging { static | time time |
  type {absolute | inactivity}}
```

Table 20-3 describes the parameters for this command.

**Table 20-3 Parameters for the *port-security aging* Command**

| Parameter | Description |
|---|---|
| **static** | Enable aging for statically configured secure addresses on this port. |
| **time** *time* | Specify the aging time for this port. The range is 0 to 1440 minutes. If the time is 0, aging is disabled for this port. |
| **type absolute** | Set the absolute aging time. All the secure addresses on this port age out exactly after the time (in minutes) specified and are removed from the secure address list. |
| **type inactivity** | Set the inactivity aging type. The secure addresses on this port age out only if there is no data traffic from the secure source address for the specified time period. |

Example 20-7 shows an administrator configuring the aging type to 10 minutes of inactivity and using the **show port-security interface** command to verify the configuration.

**Example 20-7 Configuring and Verifying Port Security Aging**

```
S1(config)# interface fa0/1
S1(config-if)# switchport port-security aging time 10
S1(config-if)# switchport port-security aging type inactivity
S1(config-if)# end
S1# show port-security interface fa0/1
Port Security              : Enabled
Port Status                : Secure-shutdown
Violation Mode             : Restrict
Aging Time                 : 10 mins
Aging Type                 : Inactivity
SecureStatic Address Aging : Disabled
Maximum MAC Addresses      : 4
Total MAC Addresses        : 1
Configured MAC Addresses   : 1
Sticky MAC Addresses       : 0
Last Source Address:Vlan   : 0050.56be.e4dd:1
Security Violation Count   : 1
```

## Port Restoration After a Violation

When port security is activated on an interface, the default action when a violation occurs is to shut down the port. A security violation can occur in one of two ways:

- The maximum number of secure MAC addresses has been added to the address table for that interface, and a station whose MAC address is not in the address table attempts to access the interface.
- An address learned or configured on one secure interface is seen on another secure interface in the same VLAN.

When a violation occurs, a syslog message is sent to the console, stating that the interface is now in the **err-disable** state. The console messages include the port number and the MAC address that caused the violation, as Example 20-8 shows.

**Example 20-8 Port Security Violation Verification and Restoration**

```
S1#
Sep 20 06:44:54.966: %PM-4-ERR_DISABLE: psecure-violation error detected on
  Fa0/18,
   putting Fa0/18 in err-disable state
Sep 20 06:44:54.966: %PORT_SECURITY-2-PSECURE_VIOLATION: Security violation
   occurred, caused by MAC address 000c.292b.4c75 on port FastEthernet0/18.
Sep 20 06:44:55.973: %LINEPROTO-5-PPDOWN: Line protocol on Interface
```

```
FastEthernet0/18, changed state to down
Sep 20 06:44:56.971: %LINK-3-UPDOWN: Interface FastEthernet0/18, changed state
   to down
!The two following commands can be used to verify the port status.
S1# show interface fa0/18 status
Port    Name  Status         Vlan Duplex  Speed   Type
Fa0/18        err-disabled   5    auto     auto    10/100BaseTX
S1# show port-security interface fastethernet 0/18
Port Security                      : Enabled
Port Status                        : Secure-shutdown
Violation Mode                     : Shutdown
Aging Time                         : 0 mins
Aging Type                         : Absolute
SecureStatic Address Aging         : Disabled
Maximum MAC Addresses              : 1
Total MAC Addresses                : 0
Configured MAC Addresses           : 0
Sticky MAC Addresses               : 0
Last Source Address:Vlan           : 000c.292b.4c75:1
Security Violation Count           : 1
!To restore a port, manually shut it down and then reactivate it.
S1(config)# interface FastEthernet 0/18
S1(config-if)# shutdown
Sep 20 06:57:28.532: %LINK-5-CHANGED: Interface FastEthernet0/18, changed state to
   administratively down
S1(config-if)# no shutdown
Sep 20 06:57:48.186: %LINK-3-UPDOWN: Interface FastEthernet0/18, changed state to up
Sep 20 06:57:49.193: %LINEPROTO-5-UPDOWN: Line protocol on Interface
FastEthernet0/18, changed state to up
```

You can use the **show interface** *type number* **status** or **show port-security interface** *type number* command to verify the current state of the port. To restore the port, you must first manually shut down the interface and then reactivate it, as in Example 20-8.

# LAN Threat Mitigation

This section reviews LAN threats and mitigation techniques for VLAN attacks, DHCP attacks, and ARP attacks.

## Native and Management VLAN Modification

The IEEE 802.1Q specification defines a native VLAN to maintain backward compatibility with untagged traffic that is common in legacy LAN scenarios. A native VLAN serves as a common identifier on opposite ends of a trunk link. VLAN 1 is the native VLAN by default.

A management VLAN is any VLAN configured to access the management capabilities of a switch. VLAN 1 is the management VLAN by default. The management VLAN is assigned an IP address and subnet mask, allowing the switch to be managed through HTTP, Telnet, SSH, or SNMP.

It is a best practice to configure the native VLAN as an unused VLAN distinct from VLAN 1 and other VLANs. In fact, it is not unusual to dedicate a fixed VLAN to serve the role of the native VLAN for all trunk ports in the switched domain. Likewise, the management VLAN should be configured as something other than VLAN 1. The management and native VLANs can be configured as the same VLAN, as in Example 20-9.

**Example 20-9 Configuring the Native and Management VLAN**

```
S1(config)# vlan 86
S1(config-vlan)# name Management&Native
S1(config-vlan)# interface vlan 86
*Jul 13 14:14:04.840: %LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan86,
   changed state to down
S1(config-if)# ip address 10.10.86.10 255.255.255.0
S1(config-if)# no shutdown
S1(config-if)# ip default-gateway 10.10.86.254
S1(config)# interface range fa0/21 - 24
S1(config-if-range)# switchport mode trunk
S1(config-if-range)# switchport trunk native vlan 86
S1(config-if-range)#
*Jul 13 14:15:55.499: %LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan86,
   changed state to up
S1(config-if-range)#
```

First, a VLAN is created that will be used for the management and native VLAN. Next, by activating interface VLAN 86, the switch can be remotely managed. Finally, the trunk ports are statically configured, and VLAN 86 is set as the native VLAN for all untagged traffic. After it is configured, the interface VLAN 86 comes up.

## VLAN Attacks

VLAN attacks can be launched in one of three ways:

- **Spoofing Dynamic Trunking Protocol (DTP) messages:** Spoofing DTP messages from the attacking host can cause the switch to enter trunking mode. From here, the attacker can send traffic tagged with the target VLAN, and the switch then delivers the packets to the destination.
- **Introducing a rogue switch and enabling trunking:** After doing this, an attacker can access all the VLANs on the victim switch from the rogue switch.
- **Mounting a double-tagging (or double-encapsulated) attack:** This type of VLAN hopping attack takes advantage of the way hardware on most switches operates. A threat actor in specific situations could embed a hidden 802.1Q tag inside the frame that already has an 802.1Q tag. This tag allows the frame to go to a VLAN that the original 802.1Q tag did not specify.

## VLAN Attack Mitigation

Use the following steps to mitigate VLAN hopping attacks:

**Step 1.** Disable DTP (auto trunking) negotiations on non-trunking ports by using the **switchport mode access** interface configuration command.

**Step 2.** Disable unused ports and put them in an unused VLAN.

**Step 3.** Manually enable the trunk link on a trunking port by using the **switchport mode trunk** command.

**Step 4.** Disable DTP (auto trunking) negotiations on trunking ports by using the **switchport nonegotiate** command.

**Step 5.** Set the native VLAN to a VLAN other than VLAN 1 by using the **switchport trunk native vlan** *vlan_number* command.

For example, assume the following:

- FastEthernet ports 0/1 through fa0/16 are active access ports.
- FastEthernet ports 0/17 through 0/24 are not currently in use.
- FastEthernet ports 0/21 through 0/24 are trunk ports.

VLAN hopping can be mitigated by implementing the following configuration, as shown in Example 20-10:

- Trunking is disabled on FastEthernet ports 0/1 to 0/16.
- FastEthernet ports 0/17 to 0/20 are assigned an unused VLAN.
- FastEthernet ports 0/21 to 0/24 are manually enabled as trunks with DTP disabled. The native VLAN is also changed from the default VLAN 1 to VLAN 86.

**Example 20-10 VLAN Hopping Attack Mitigation**

```
S1(config)# interface range fa0/1 - 16
S1(config-if-range)# switchport mode access
S1(config-if-range)# exit
S1(config)#
S1(config)# interface range fa0/17 - 20
S1(config-if-range)# switchport mode access
S1(config-if-range)# switchport access vlan 999
S1(config-if-range)# exit
S1(config)#
S1(config)# interface range fa0/21 - 24
S1(config-if-range)# switchport mode trunk
S1(config-if-range)# switchport nonegotiate
S1(config-if-range)# switchport trunk native vlan 86
S1(config-if-range)# end
S1#
```

# DHCP Attacks

Two types of DHCP attacks are DHCP starvation and DHCP spoofing. Both attacks are mitigated by implementing DHCP snooping.

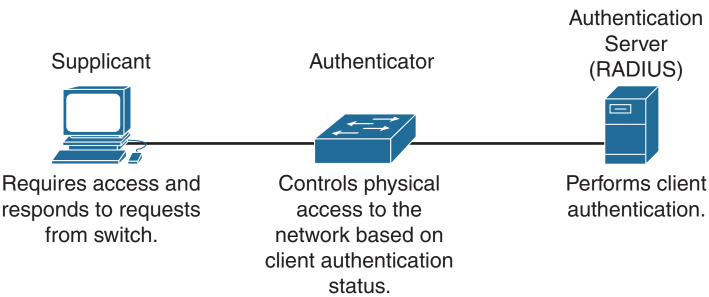
## DHCP Starvation Attacks

The goal of a DHCP starvation attack is to create a denial-of-service condition for connecting clients. DHCP starvation attacks require an attack tool such as Gobbler. Gobbler looks at the entire scope of leasable IP addresses and tries to lease them all. Specifically, it creates DHCP discovery messages with bogus MAC addresses.

## DHCP Spoofing Attacks

A DHCP spoofing attack occurs when a rogue DHCP server is connected to the network and provides false IP configuration parameters to legitimate clients. For example, in Figure 20-6, R1 is configured to relay DHCP requests to the DHCP server attached to R2.

**Figure 20-6 Rogue DHCP Server Intercepting DHCP Requests**

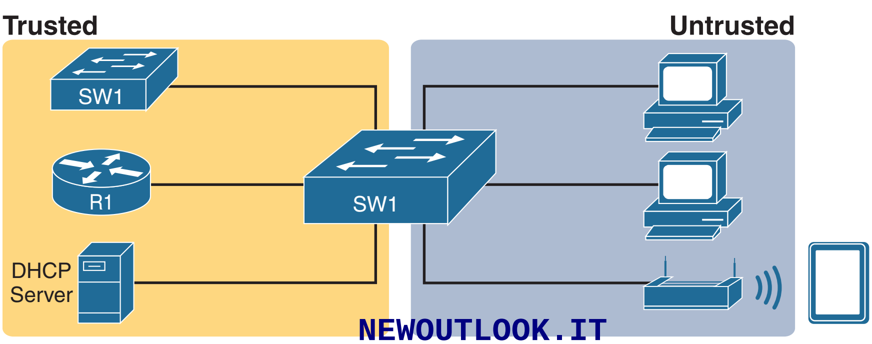



However, the rogue DHCP server attached to SW1 responds to the DHCP request from PC1 first. PC1 accepts the DHCP offer and sets the rogue DHCP server as the default gateway.

# DHCP Snooping

To protect against DHCP attacks, DHCP snooping uses the concept of trusted and untrusted ports. As Figure 20-7 shows, SW2, R1, and the DHCP server are attached to trusted ports on SW1. The other devices, including the wireless access point, are connected to untrusted ports.

**Figure 20-7 Trusted and Untrusted Ports**



Some critical features of a DHCP snooping configuration include the following:

- **Trusted ports:** Trusted ports allow all incoming DHCP messages.
- **Untrusted ports, server messages:** Untrusted ports discard all incoming messages that are considered server messages.
- **Untrusted ports, client messages:** Untrusted ports apply more complex logic for messages considered client messages. They check whether each incoming DHCP message conflicts with existing DHCP binding table information; if so, they discard the DHCP message. If the message has no conflicts, the switch allows the message through, which typically results in the addition of new DHCP binding table entries.
- **Rate limiting:** This feature optionally limits the number of received DHCP messages per second per port.

Use the following steps to enable DHCP snooping:

**Step 1.** Enable DHCP snooping by using the **ip dhcp snooping** global configuration command.

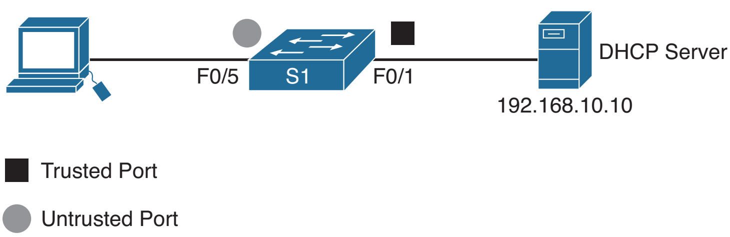
**Step 2.** On trusted ports, use the **ip dhcp snooping trust** interface configuration command.

**Step 3.** Limit the number of DHCP discovery messages that can be received per second on untrusted ports by using the **ip dhcp snooping limit rate** *number* interface configuration command. This helps mitigate DHCP starvation attacks.

**Step 4.** Enable DHCP snooping by VLAN or by a range of VLANs by using the **ip dhcp snooping** *vlan* global configuration command.

For a simple scenario, consider the topology in Figure 20-8.

**Figure 20-8 DHCP Snooping Configuration Topology**




Example 20-11 shows how to configure and verify DHCP snooping on S1.

**Example 20-11 Configuring and Verifying DHCP Snooping**

```
S1(config)# ip dhcp snooping
S1(config)# interface f0/1
S1(config-if)# ip dhcp snooping trust
S1(config-if)# exit
S1(config)# interface range f0/5 - 24
```

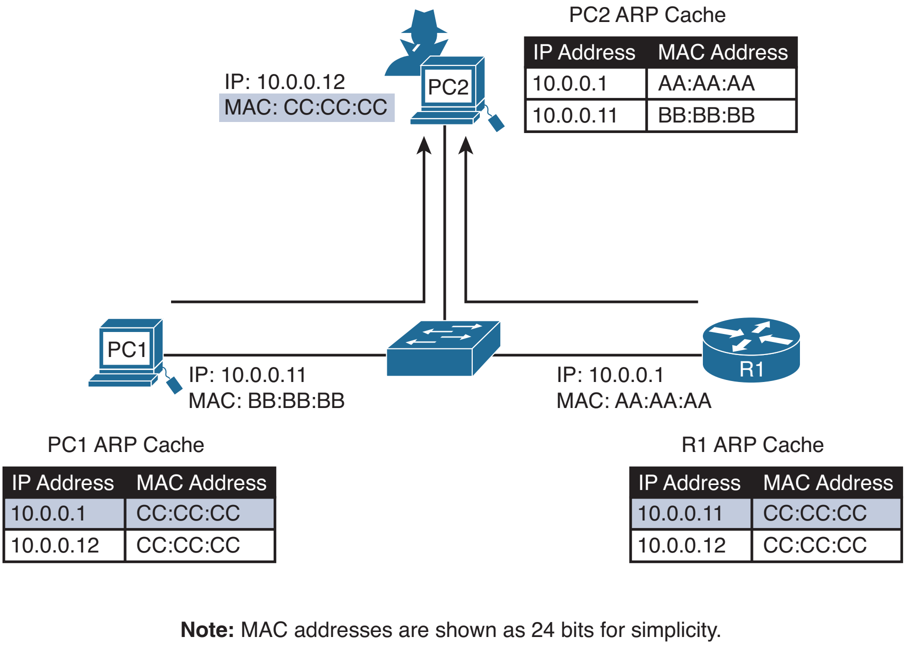
```
S1(config-if-range)# ip dhcp snooping limit rate 6
S1(config-if)# exit
S1(config)# ip dhcp snooping vlan 5,10,50-52
S1(config)# end
S1# show ip dhcp snooping
Switch DHCP snooping is enabled
DHCP snooping is configured on following VLANs:
5,10,50-52
DHCP snooping is operational on following VLANs:
none
DHCP snooping is configured on the following L3 Interfaces:
Insertion of option 82 is enabled
   circuit-id default format: vlan-mod-port
   remote-id: 0cd9.96d2.3f80 (MAC)
Option 82 on untrusted port is not allowed
Verification of hwaddr field is enabled
Verification of giaddr field is enabled
DHCP snooping trust/rate is configured on the following Interfaces:
Interface                   Trusted    Allow option    Rate limit (pps)
-----------------------    -------    ------------    ----------------
FastEthernet0/1             yes        yes              unlimited
  Custom circuit-ids:
FastEthernet0/5             no         no               6
  Custom circuit-ids:
FastEthernet0/6             no         no               6
  Custom circuit-ids:
S1# show ip dhcp snooping binding
MacAddress          IpAddress        Lease(sec)  Type           VLAN  Interface
------------------  ---------------  ----------  -------------  ----  ----------------
00:03:47:B5:9F:AD   192.168.10.10    193185      dhcp-snooping  5     FastEthernet0/5
S1#
```

## ARP Attacks

On Ethernet LANs, hosts are allowed to send an unsolicited Address Resolution Protocol (ARP) reply called a *gratuitous ARP* message. These ARP messages cause all other hosts on the LAN to store the MAC address and IP address in their ARP caches. The problem is that an attacker can send a gratuitous ARP message containing a spoofed MAC address to a switch, and the switch would update its MAC table accordingly. Therefore, any host can claim to be the owner of any IP and MAC address combination.

For example, in Figure 20-9, R1 and PC1 have removed the correct entry for each other's MAC address and replaced it with PC2's MAC address. The threat actor has poisoned the ARP caches of all devices on the subnet. ARP poisoning leads to various man-in-the-middle attacks, posing a serious security threat to the network. All traffic between R1 and PC1 will now flow through the threat actor's PC2.

**Figure 20-9 Successful ARP Poisoning Attack**




## Dynamic ARP Inspection

To prevent ARP spoofing and then ARP poisoning, a switch must ensure that only valid ARP requests and replies are relayed. Dynamic ARP inspection (DAI) requires DHCP snooping and helps prevent ARP attacks by doing the following:

- Not relaying invalid or gratuitous ARP replies out to other ports in the same VLAN
- Intercepting all ARP requests and replies on untrusted ports
- Verifying each intercepted packet for a valid IP-to-MAC binding
- Dropping and logging ARP replies coming from invalid source to prevent ARP poisoning
- Error disabling the interface if the configured DAI number of ARP packets is exceeded

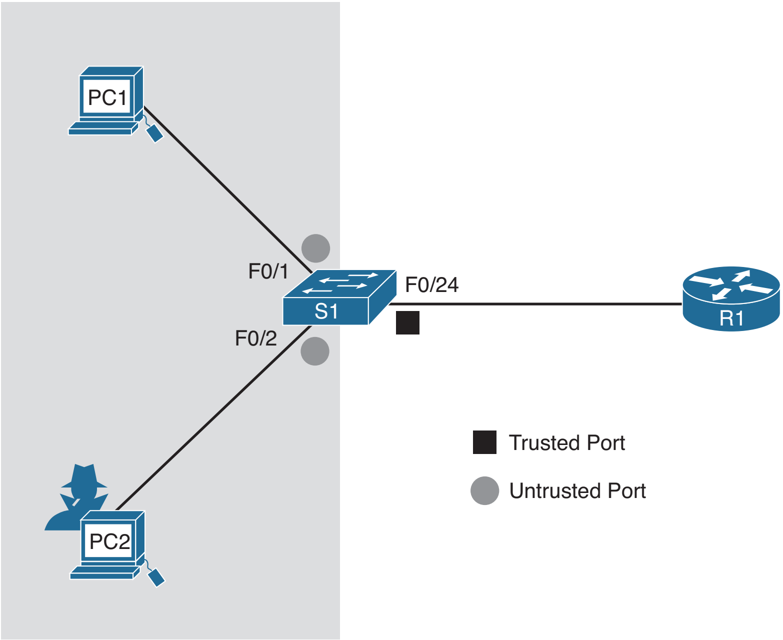
To mitigate the chances of ARP spoofing and ARP poisoning, follow these DAI implementation guidelines:

- Enable DHCP snooping globally.
- Enable DHCP snooping on selected VLANs.

- Enable DAI on selected VLANs.
- Configure trusted interfaces for DHCP snooping and ARP inspection.

The topology in Figure 20-10 identifies trusted and untrusted ports

**Figure 20-10 Trusted and Untrusted Ports for DAI Configuration**




In Figure 20-10, S1 is connecting two users on VLAN 10. In Example 20-12, DAI is configured to mitigate against ARP spoofing and ARP poisoning attacks. Notice that DHCP snooping is enabled because DAI requires the DHCP snooping binding table to operate.

**Example 20-12 DAI Configuration**

```
S1(config)# ip dhcp snooping
S1(config)# ip dhcp snooping vlan 10
S1(config)# ip arp inspection vlan 10
S1(config)# interface fa0/24
S1(config-if)# ip dhcp snooping trust
S1(config-if)# ip arp inspection trust
```

DAI can also be configured to check for both destination or source MAC and IP addresses with the **ip arp inspection validate** command. Only one command can be configured. Entering multiple **ip arp inspection validate** commands overwrites the previous command. To include more than one validation method, enter them on the same command line, as shown and verified in Example 20-13.

**Example 20-13 Configuring DAI to Validate MAC and IP Addresses**

```
S1(config)# ip arp inspection validate ?
  dst-mac  Validate destination MAC address
  ip       Validate IP addresses
  src-mac  Validate source MAC address
S1(config)# ip arp inspection validate src-mac
S1(config)# ip arp inspection validate dst-mac
S1(config)# ip arp inspection validate ip
S1(config)# do show run | include validate
ip arp inspection validate ip
S1(config)# ip arp inspection validate src-mac dst-mac ip
S1(config)# do show run | include validate
ip arp inspection validate src-mac dst-mac ip
S1(config)#
```

# Study Resources

For today's exam topics, refer to the following resources for more study.

| Resource | Module or Chapter |
|---|---|
| Cisco Network Academy: CCNA2 | 10 |
| | 11 |
| CCNA 200-301 Official Cert Guide, Volume 1 | 6 |
| CCNA 200-301 Official Cert Guide, Volume 2 | 4 |
| | 8 |
| Portable Command Guide | 20 |
| | 22 |

*This page intentionally left blank*

# Day 19

# Basic Routing Concepts

## CCNA 200-301 Exam Topics

- Explain the role and function of network components
- Determine how a router makes a forwarding decision by default

## Key Topics

Today we review basic routing concepts, including exactly how a packet is processed by intermediary devices (routers) on its way from source to destination. We then review the basic routing methods, including connected, static, and dynamic routes. We conclude the day's review with a deep dive into the operation of dynamic routing protocols.

## Packet Forwarding

Packet forwarding by routers is accomplished through path determination and switching functions. The path determination function is the process the router use to determine which path to use when forwarding a packet. To determine the best path, the router searches its routing table for a network address that matches the packet's destination IP address.

This search results in one of three path determinations:

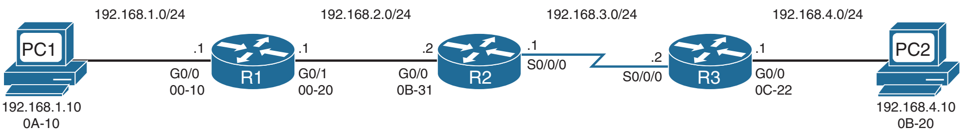
- **Directly connected network:** If the destination IP address of the packet belongs to a device on a network that is directly connected to one of the router's interfaces, that packet is forwarded directly to that device. This means the destination IP address of the packet is a host address on the same network as this router's interface.
- **Remote network:** If the destination IP address of the packet belongs to a remote network, the packet is forwarded to another router. Remote networks can be reached only by forwarding packets to another router.
- **No route determined:** If the destination IP address of the packet does not belong to a connected or remote network and the router does not have a default route, the packet is discarded. The router sends an Internet Control Message Protocol (ICMP) Unreachable message to the source IP address of the packet.

In the first two results, the router completes the process by switching the packet out the correct interface. It does this by reencapsulating the IP packet into the appropriate Layer 2 data-link frame format for the exit interface. The type of interface determines the type of Layer 2 encapsulation. For example, if the exit interface is Fast Ethernet, the packet is encapsulated in an Ethernet frame. If the exit interface is a serial interface configured for PPP, the IP packet is encapsulated in a PPP frame.

## Path Determination and Switching Function Example

Let's review the process of path determination and switching functions that routers perform as a packet travels from source to destination. Consider the topology in Figure 19-1 and the following steps:

**Figure 19-1 Packet Forwarding Sample Topology**




**NOTE:** For brevity, Figure 19-1 shows only the last two octets of the MAC address.

**Step 1.** PC1 has a packet to send to PC2. Using the AND operation on the destination's IP address and PC1's subnet mask, PC1 has determined that the IP source and IP destination addresses are on different networks. Therefore, PC1 checks its Address Resolution Protocol (ARP) table for the IP address of the default gateway and its associated MAC address. It then encapsulates the packet in an Ethernet header and forwards it to R1.

**Step 2.** Router R1 receives the Ethernet frame. Router R1 examines the destination MAC address, which matches the MAC address of the receiving interface, G0/0. R1 therefore copies the frame into its buffer to be processed.

R1 decapsulates the Ethernet frame and reads the destination IP address. Because it does not match any of R1's directly connected networks, the router consults its routing table to route this packet.

R1 searches the routing table for a network address and subnet mask that include this packet's destination IP address as a host address on that network. It selects the entry with the longest match (longest prefix). R1 encapsulates the packet in the appropriate frame format for the exit interface and switches the frame to the interface (G0/1 in this example). The interface then forwards it to the next hop.

**Step 3.** The packet arrives at router R2. R2 performs the same functions as R1, but this time, the exit interface is a serial interface—not Ethernet. Therefore, R2 encapsulates the packet in the appropriate frame format for the serial interface and sends it to R3. For this example, assume that the interface is using High-Level Data Link Control (HDLC), which uses the data-link address 0x8F. Remember that serial interfaces do not use MAC addresses.

**Step 4.** The packet arrives at R3. R3 decapsulates the data-link HDLC frame. The search of the routing table results in a network that is one of R3's directly connected networks. Because the exit interface is a directly connected Ethernet network, R3 needs to resolve the destination IP address of the packet with a destination MAC address.

R3 searches for the packet's destination IP address, 192.168.4.10, in its ARP cache. If the entry is not in the ARP cache, R3 sends an ARP request out its G0/0 interface.

PC2 sends back an ARP reply with its MAC address. R3 updates its ARP cache with an entry for 192.168.4.10 and the MAC address returned in the ARP reply.

The IP packet is encapsulated into a new data-link Ethernet frame and sent out R3's G0/0 interface.

**Step 5.** The Ethernet frame with the encapsulated IP packet arrives at PC2. PC2 examines the destination MAC address, which matches the MAC address of the receiving interface—that is, its own Ethernet NIC. PC2 therefore copies the rest of the frame. PC2 sees that the Ethernet Type field is 0x800, which means that the Ethernet frame contains an IP packet in the data portion of the frame. PC2 decapsulates the Ethernet frame and passes the IP packet to its operating system's IP process.

# Routing Methods

A router can learn routes from three basic sources:

- **Directly connected routes:** Automatically entered in the routing table when an interface is activated with an IP address
- **Static routes:** Manually configured by the network administrator and entered in the routing table if the exit interface for the static route is active
- **Dynamic routes:** Learned by the routers through sharing routes with other routers that use the same routing protocol

In many cases, the complexity of the network topology, the number of networks, and the need for the network to automatically adjust to changes require the use of a dynamic routing protocol. Dynamic routing certainly has several advantages over static routing; however, networks still use static routing. In fact, networks typically use a combination of static and dynamic routing.

Table 19-1 compares dynamic and static routing features. From this comparison, you can list the advantages of each routing method. The advantages of one method are the disadvantages of the other.

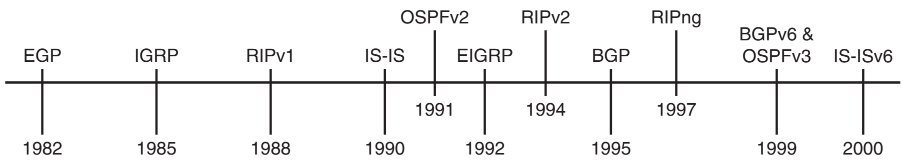
**Table 19-1 Dynamic Versus Static Routing**

| Feature | Dynamic Routing | Static Routing |
|---|---|---|
| Configuration complexity | Generally remains independent of the network size | Increases with network size |
| Required administrator knowledge | Requires advanced knowledge | Requires no extra knowledge |
| Topology changes | Automatically adapts to topology changes | Requires administrator intervention |
| Scaling | Suitable for simple and complex topologies | Suitable for simple topologies |
| Security | Less secure | More secure |
| Resource usage | Uses CPU, memory, and link bandwidth | Requires no extra resources |
| Predictability | Uses a route that depends on the current topology | Always uses the same route to the destination |

# Classifying Dynamic Routing Protocols

Figure 19-2 shows a timeline of IP routing protocols, along with a chart to help you memorize the various ways to classify routing protocols.

**Figure 19-2 Evolution and Classification of Routing Protocols**




| | Interior Gateway Protocols | | | | Exterior Gateway Protocols |
|---|---|---|---|---|---|
| | Distance Vector Routing Protocols | | Link State Routing Protocols | | Path Vector |
| Classful | RIP | IGRP | | | EGP |
| Classless | RIPv2 | EIGRP | OSPFv2 | IS-IS | BGPv4 |
| IPv6 | RIPng | EIGRP for IPv6 | OSPFv3 | IS-IS for IPv6 | MP BGP-4 (IPv6) |

Routing protocols are classified into different groups according to their characteristics:

- IGP or EGP
- Distance vector or link state
- Classful or classless

## IGP and EGP

An autonomous system (AS) is a collection of routers under a common administration that presents a common, clearly defined routing policy to the Internet. Typical examples are a large company's internal network and an ISP's network. Most company networks are not autonomous systems; in most cases, a company network is a network within its ISP's autonomous system. Because the Internet is based on the autonomous system concept, two types of routing protocols are required:

- **Interior gateway protocols (IGP):** Used for intra-AS routing—that is, routing inside an AS
- **Exterior gateway protocols (EGP):** Used for inter-AS routing—that is, routing between autonomous systems

## Distance Vector Routing Protocols

*Distance vector* means that routes are advertised as vectors of distance and direction. Distance is defined in terms of a metric such as hop count, and direction is the next-hop router or exit interface. Distance vector protocols typically use the Bellman-Ford algorithm for the best-path route determination.

Some distance vector protocols periodically send complete routing tables to all connected neighbors. In large networks, these routing updates can become enormous, causing significant traffic on the links.

Although the Bellman-Ford algorithm eventually accumulates enough knowledge to maintain a database of reachable networks, the algorithm does not allow a router to know the exact topology of an internetwork. The router knows only the routing information received from its neighbors.

Distance vector protocols use routers as signposts along the path to the final destination. The only information a router knows about a remote network is the distance or metric to reach that network and which path or interface to use to get there. A distance vector routing protocol does not have a map of the network topology.

Distance vector protocols work best in these situations:

- When the network is simple and flat and does not require a hierarchical design
- When the administrators do not have enough knowledge to configure and troubleshoot link-state protocols
- When specific types of networks, such as hub-and-spoke networks, are being implemented
- When worst-case convergence times in a network are not a concern

## Link-State Routing Protocols

In contrast to distance vector routing protocol operation, a router configured with a link-state routing protocol can create a complete view, or topology, of the network by gathering information from all the other routers. Think of a link-state routing protocol as having a complete map of the network topology. The signposts along the way from source to destination are not necessary because all link-state routers are using an identical map of the network. A link-state router uses the link-state information to create a topology map and to select the best path to each destination network in the topology.

With some distance vector routing protocols, routers periodically send updates of their routing information to their neighbors. Link-state routing protocols do not use periodic updates. After the network has converged, a link-state update is sent only when the topology changes.

Link-state protocols work best in these situations:

- When the network design is hierarchical, which is typically the case in large networks
- When the administrators have good knowledge of the implemented link-state routing protocol
- When fast convergence of the network is crucial