Operating Juniper Networks Switches in the Enterprise

9.a

Student Guide

1194 North Mathilda Avenue
Sunnyvale, CA 94089
USA
408-745-2000
www.juniper.net

Course Number: EDU-JUN-OJXE

Operating Juniper Networks Switches in the Enterprise Student Guide, Revision 9.a

Revision History:

Revision 9.a—May 2008.

The information in this document is current as of the date listed above.

The information in this document has been carefully verified and is believed to be accurate for software Release 9.0R2. Juniper Networks assumes no responsibilities for any inaccuracies that may appear in this document. In no event will Juniper Networks be liable for direct, indirect, special, exemplary, incidental or consequential damages resulting from any defect or omission in this document, even if advised of the possibility of such damages.

Juniper Networks reserves the right to change, modify, transfer or otherwise revise this publication without notice.

YEAR 2000 NOTICE

Juniper Networks hardware and software products do not suffer from Year 2000 problems and hence are Year 2000 compliant. The JUNOS software has no known time-related limitations through the year 2038. However, the NTP application is known to have some difficulty in the year 2036.

SOFTWARE LICENSE

The terms and conditions for using Juniper Networks software are described in the software license provided with the software, or to the extent applicable, in an agreement executed between you and Juniper Networks, or Juniper Networks agent. By using Juniper Networks software, you indicate that you understand and agree to be bound by its license terms and conditions. Generally speaking, the software license restricts the manner in which you are permitted to use the Juniper Networks software, may contain prohibitions against certain uses, and may state conditions under which the license is automatically terminated. You should consult the software license for further details.

Contents

Course Overview

This four-day course discusses the configuration of Juniper Networks EX-series switches in a typical network environment. Key topics include a platform overview, user configuration interfaces, initial and secondary system configuration tasks, operational monitoring of an EX-series switch, Ethernet switching concepts, the Spanning Tree Protocol (STP), inter-VLAN routing, switching security protocols and features, IP telephony features, and design and implementation considerations. This course is based upon JUNOS software for EX-series switches Release 9.0R2.

Through demonstrations and hands-on labs, you will gain experience in configuring and monitoring the EX-series switches.

Objectives

After successfully completing this course, you should be able to:

- Describe common deployment options for the EX-series switches.
- Perform operational monitoring tasks typically associated with EX-series switches.
- Install and configure an EX-series switch in a network.
- Form a Virtual Chassis system by connecting multiple EX-4200 switches.
- Configure and monitor virtual LANs (VLANs).
- Create a loop-free network environment using STP.
- Implement inter-VLAN routing using static and dynamic methods.
- Define and apply routing policy and firewall filters.
- Secure a switch port using available port access security features.
- Configure and monitor IP telephony features such as Power over Ethernet (PoE), voice VLAN, and the Link Layer Discovery Protocol (LLDP).
- Design and implement a Layer 2 network using EX-series switches.

Intended Audience

The primary audiences for this course are end users of EX-series switches, which include the following:

- Network engineers;
- Support personnel;
- Reseller support; and
- Others responsible for implementing Juniper Networks enterprise switching products.

Course Level

Operating Juniper Networks Switches in the Enterprise is an introductory-level course.

Prerequisites

The prerequisite for this course is a basic understanding of the TCP/IP protocols.

Although not required, familiarity with the command-line interface (CLI) of a switching platform or UNIX system is helpful.

Course Agenda

Day 1

Chapter 1: Course Introduction

Chapter 2: Introduction to Juniper Networks Enterprise Switches

Chapter 3: User Interface Options

Lab 1: User Interface Options

Chapter 4: Installation and Initial Configuration

Lab 2: Initial Configuration

Chapter 5: Secondary System Configuration

Lab 3: Secondary System Configuration

Chapter 6: Operational Monitoring and Maintenance

Lab 4: Operational Monitoring

Day 2

Chapter 7: Virtual Chassis Systems

Lab 5: Virtual Chassis System

Chapter 8: Interface Support, Configuration, and Monitoring

Lab 6: Interface Configuration

Chapter 9: Ethernet Switching and Virtual LANs

Lab 7: Ethernet Switching and VLANs

Chapter 10: Spanning Tree Protocol

Lab 8: Spanning Tree

Day 3

Chapter 11: Inter-VLAN Routing

Lab 9, Parts 1–3: RVI and Static Routing

Lab 9, Part 4: Single-Area OSPF

Lab 9, Part 5: VRRP

Chapter 12: Routing Policy and Firewall Filters

Lab 10: Routing Policy and ACLs

Chapter 13: Switching Security

Lab 11: Switching Security

Day 4

Chapter 14: IP Telephony Services

Lab 12: IP Telephony Services

Chapter 15: Design and Implementation of Layer 2 Networks

Lab 13: Design and Implementation

Document Conventions

CLI and GUI Text

Frequently throughout this course, we refer to text that appears in a CLI or a graphical user interface (GUI). To make the language of these documents easier to read, we distinguish GUI and CLI text from chapter text according to the following table.

Style	Description	Usage Example
Franklin Gothic	Normal text.	Most of what you read in the Lab Guide and Student Guide.

`Courier New`	Console text:
	• Screen captures
	• Noncommand-related syntax
	GUI text elements:
	• Menu names
	• Text field entry
		`commit complete`
		`Exiting configuration mode`
		Select `File > Open`, and then click `Configuration.conf` in the `Filename` text box.

Input Text Versus Output Text

You will also frequently see cases where you must enter input text yourself. Often this will be shown in the context of where you must enter it. We use bold style to distinguish text that is input versus text that is simply displayed.

Style	Description	Usage Example
`Normal CLI` `Normal GUI`	No distinguishing variant.	`Physical interface:fxp0, Enabled` View configuration history by clicking `Configuration > History`.
`CLI Input` **`GUI Input`**	Text that you must enter.	`lab@San_Jose>` **`show route`** Select `File > Save`, and enter **`config.ini`** in the `Filename` field.

Defined and Undefined Syntax Variables

Finally, this course distinguishes between regular text and syntax variables, and it also distinguishes between syntax variables where the value is already assigned (defined variables) and syntax variables where you must assign the value (undefined variables). Note that these styles can be combined with the input style as well.

Style	Description	Usage Example
CLI Variable *GUI Variable*	Text where variable value is already assigned.	`policy` *`my-peers`* Click on *`my-peers`* in the dialog.
CLI Undefined *GUI Undefined*	Text where the variable's value is the user's discretion and text where the variable's value as shown in the lab guide might differ from the value the user must input.	Type **`set policy`** ***`policy-name`***. **`ping 10.0.`*****`1.1`*** Select `File > Save`, and enter ***`filename`*** in the `Filename` field.

Additional Information

Education Services Offerings

You can obtain information on the latest Education Services offerings, course dates, and class locations from the World Wide Web by pointing your Web browser to: http://www.juniper.net/training/education/.

About This Publication

The *Operating Juniper Networks Switches in the Enterprise Student Guide* was developed and tested using software version 9.0R2. Previous and later versions of software may behave differently so you should always consult the documentation and release notes for the version of code you are running before reporting errors.

This document is written and maintained by the Juniper Networks Education Services development team. Please send questions and suggestions for improvement to training@juniper.net.

Technical Publications

You can print technical manuals and release notes directly from the Internet in a variety of formats:

- Go to http://www.juniper.net/techpubs/.
- Locate the specific software or hardware release and title you need, and choose the format in which you want to view or print the document.

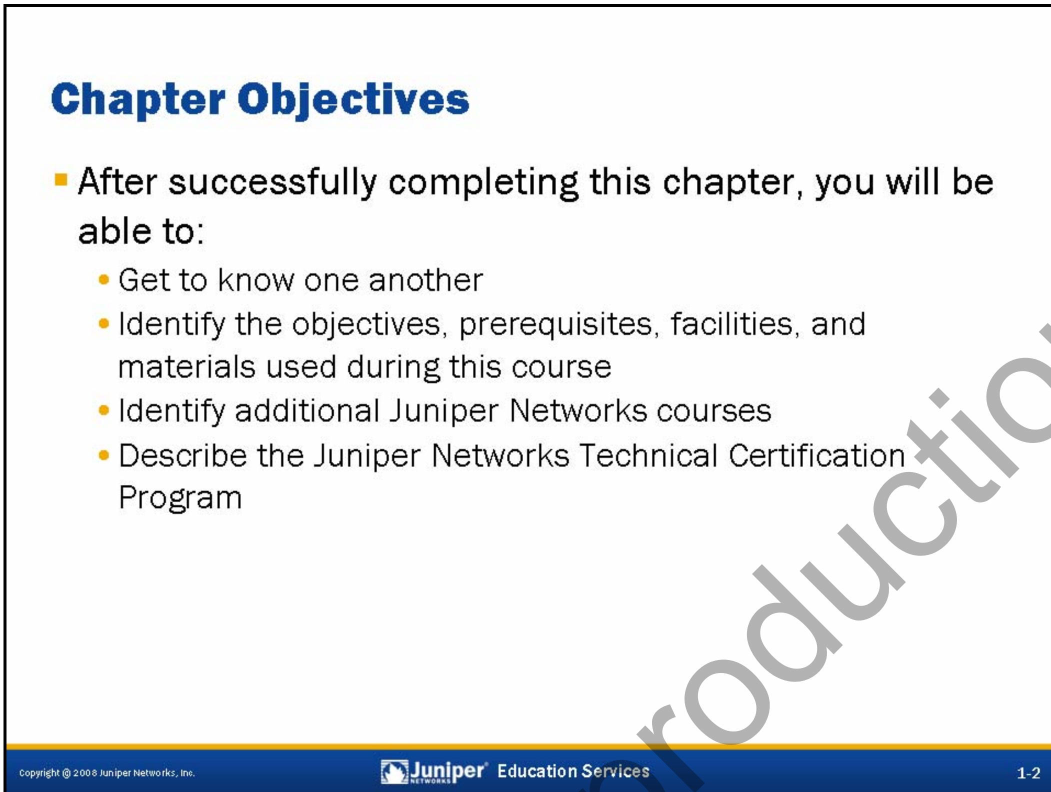
Documentation sets and CDs are available through your local Juniper Networks sales office or account representative.

Juniper Networks Support

For technical support, contact Juniper Networks at http://www.juniper.net/customers/support/, or at 1-888-314-JTAC (within the United States) or 408-745-2121 (from outside the United States).

Operating Juniper Networks Switches in the Enterprise

Chapter 1: Course Introduction



This Chapter Discusses:

- Objectives and course content information;
- Additional Juniper Networks, Inc. courses; and
- Juniper Networks Technical Certification Program (JNTCP).

Introductions

This slide asks several questions for you to answer during class introductions.

Course Contents

- Contents:
  - Chapter 1: Course Introduction
  - Chapter 2: Introduction to Juniper Networks Enterprise Switches
  - Chapter 3: User Interface Options
  - Chapter 4: Installation and Initial Configuration
  - Chapter 5: Secondary System Configuration
  - Chapter 6: Operational Monitoring and Maintenance
  - Chapter 7: Virtual Chassis Systems
  - Chapter 8: Interface Support and Configuration
  - Chapter 9: Ethernet Switching and VLANs
  - Chapter 10: Spanning Tree Protocol
  - Chapter 11: Inter-VLAN Routing
  - Chapter 12: Routing Policy and Firewall Filters
  - Chapter 13: Switching Security
  - Chapter 14: IP Telephony Services
  - Chapter 15: Design and Implementation of Layer 2 Networks

Juniper Education Services
1-4

Course Contents

This slide lists the topics we discuss in this course.

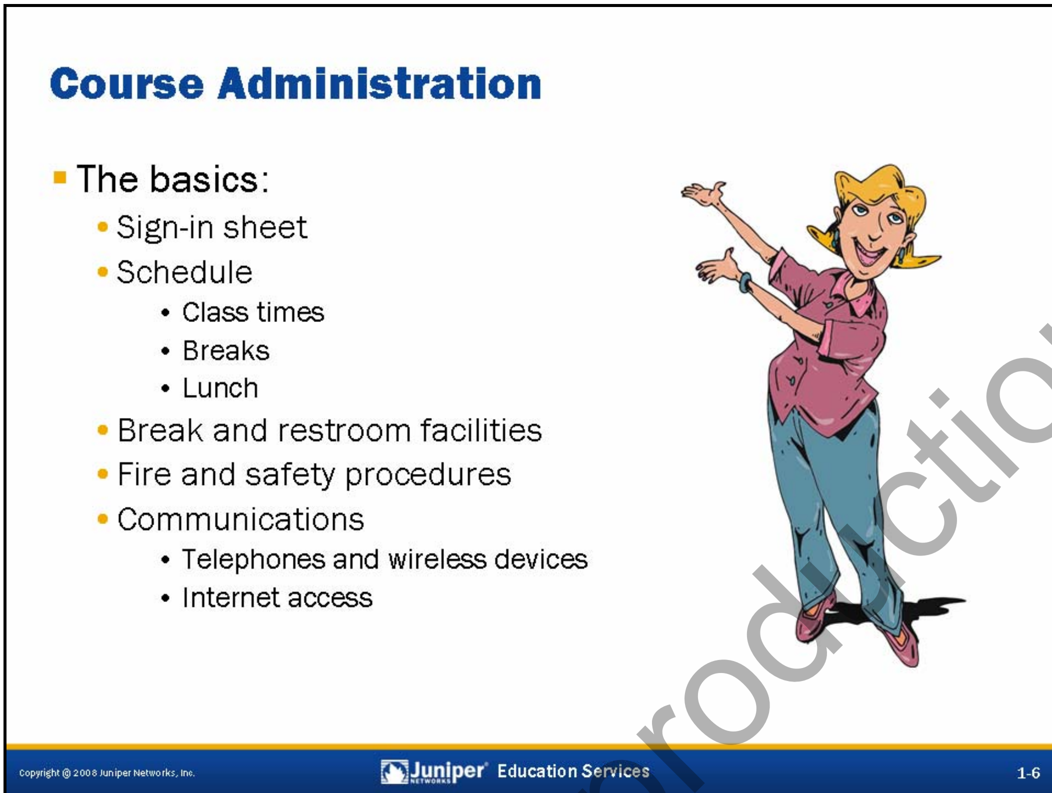
Prerequisites

- The prerequisites for this course are the following:
  - Basic understanding of the TCP/IP protocols
  - Basic understanding of Layer 3 routing and Layer 2 switching concepts
  - While not required, familiarity with the command-line interface of a routing platform or UNIX system is helpful

Juniper Education Services
1-5

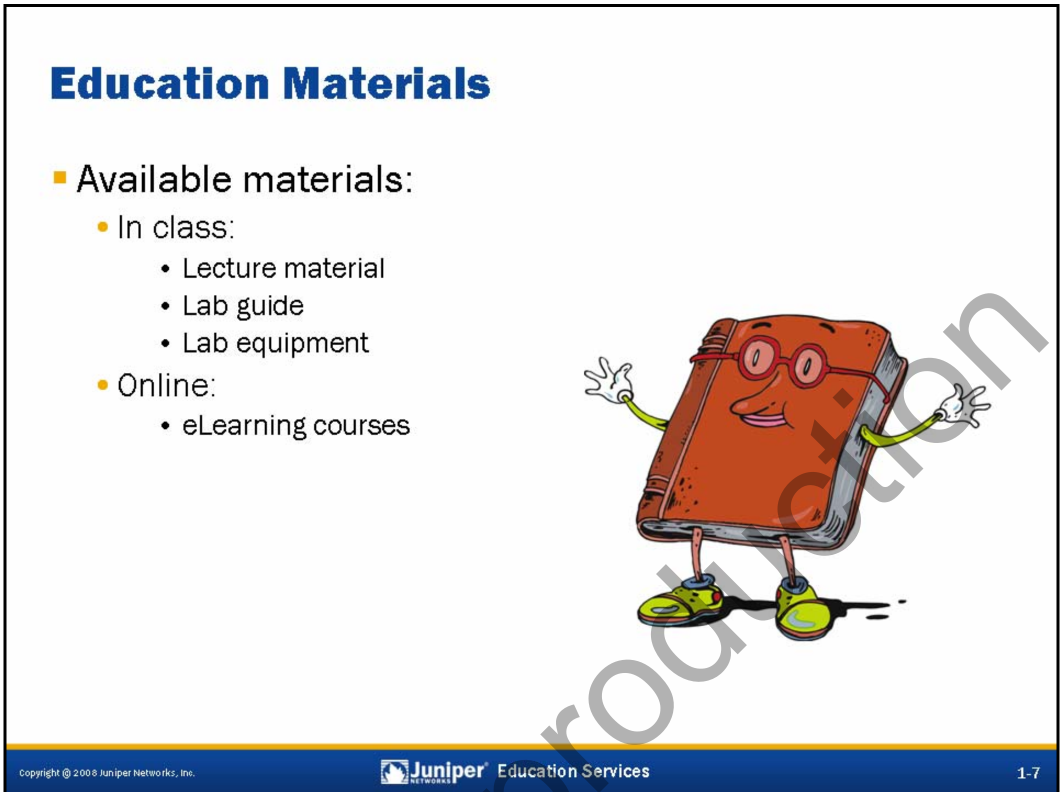
Prerequisites

This slide lists the prerequisites for this course.



General Course Administration

This slide documents general aspects of classroom administration.

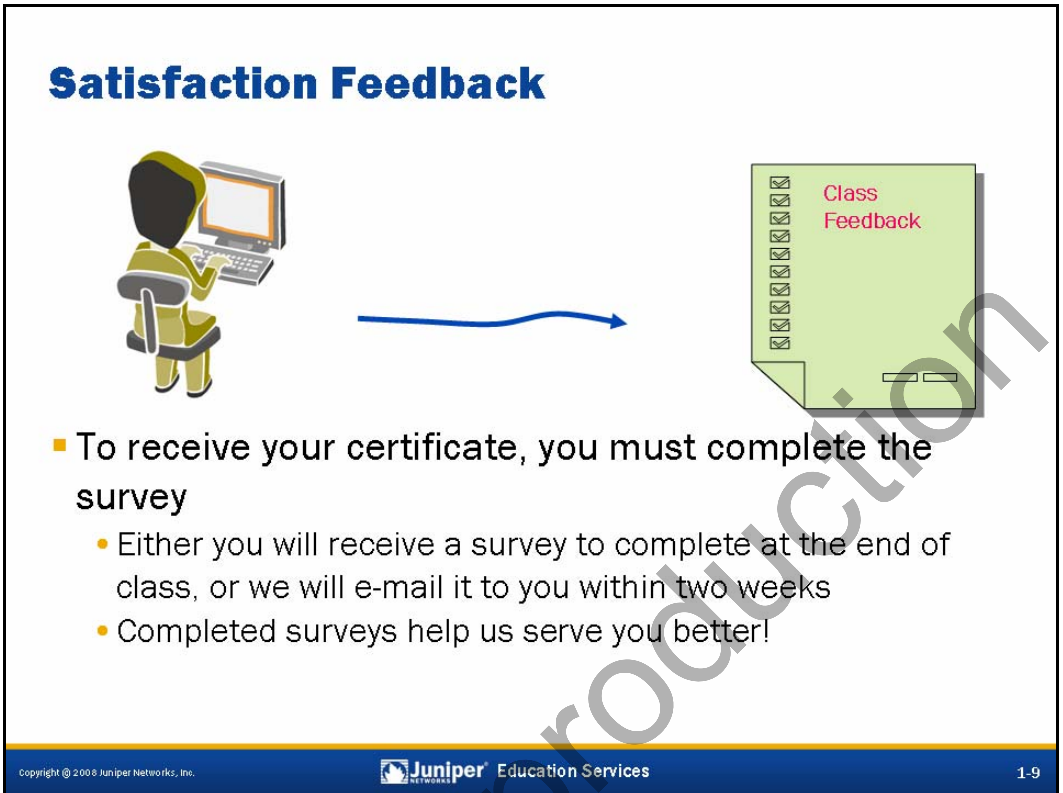


Training and Study Materials

This slide describes Education Services materials that are available for reference both in the classroom and online.

Additional Resources

This slide describes additional resources available to assist you in the installation, configuration, and operation of Juniper Networks products.



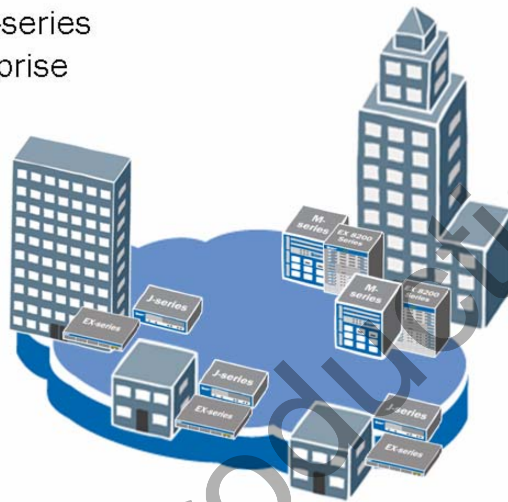
Satisfaction Feedback

Juniper Networks uses an electronic survey system to collect and analyze your comments and feedback. Depending on the class you are taking, please complete the survey at the end of the class, or be sure to look for an e-mail about two weeks from class completion that directs you to complete an online survey form. (Be sure to provide us with your current e-mail address.)

Submitting your feedback entitles you to a certificate of class completion. We thank you in advance for taking the time to help us improve our educational offerings.

Enterprise Curriculum (1 of 3)

- Enterprise Routing
  - Supports J-series and M-series technologies in an enterprise environment
- Enterprise Switching
  - Supports EX-series technologies in an enterprise environment



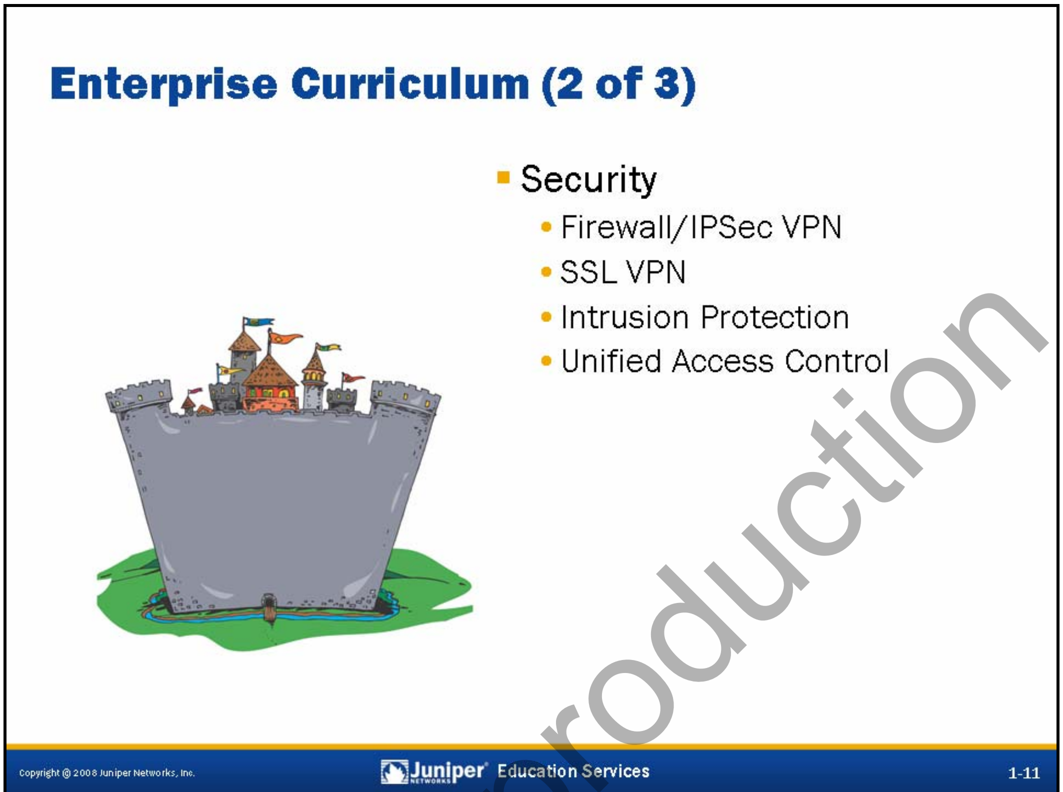
Juniper Networks Education Services

Enterprise Routing Curriculum

You can access the latest Education Services offerings that support Juniper Networks J-series and M-series technologies in an enterprise environment at http://www.juniper.net/training/technical_education/#enterprise.

Enterprise Switching Curriculum

You can access the latest Education Services offerings that support Juniper Networks EX-series technologies in an enterprise environment at http://www.juniper.net/training/technical_education/#enterprise.



Enterprise Security Curriculum

You can access the latest Education Services offerings that support Juniper Networks security technologies in an enterprise environment at http://www.juniper.net/training/technical_education/#sec.

Enterprise Curriculum (3 of 3)

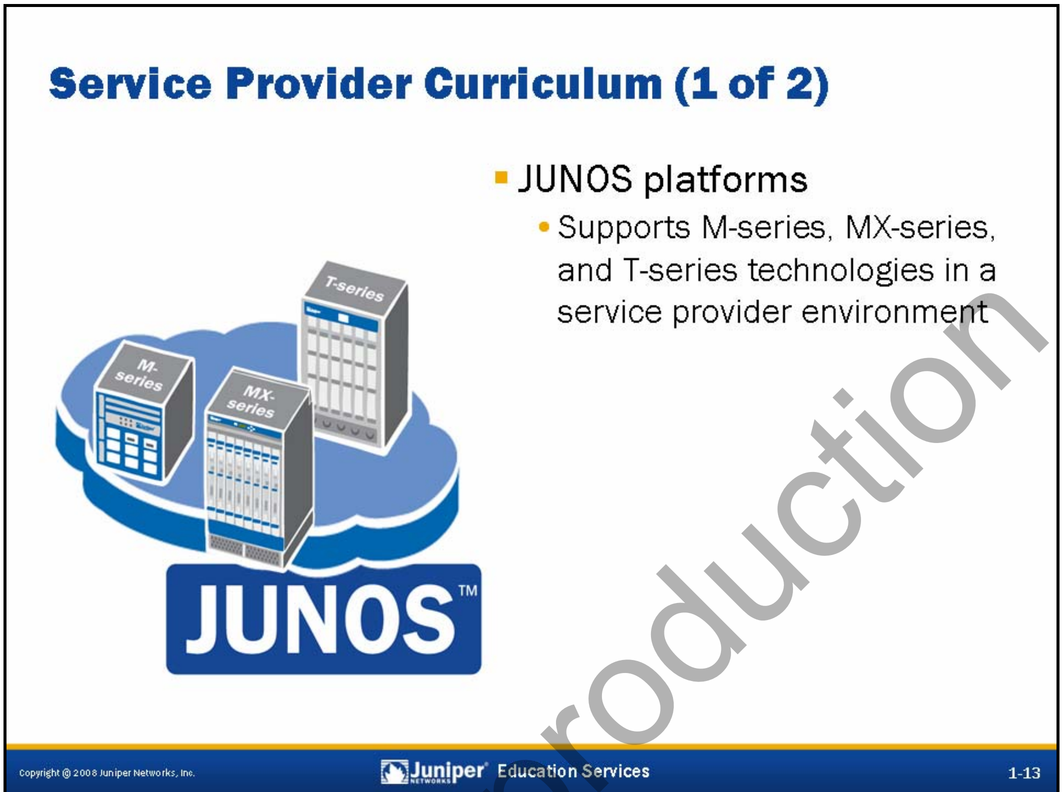
- Application Acceleration
  - WX WAN Acceleration
  - DX Application Acceleration

Juniper Education Services

1-12

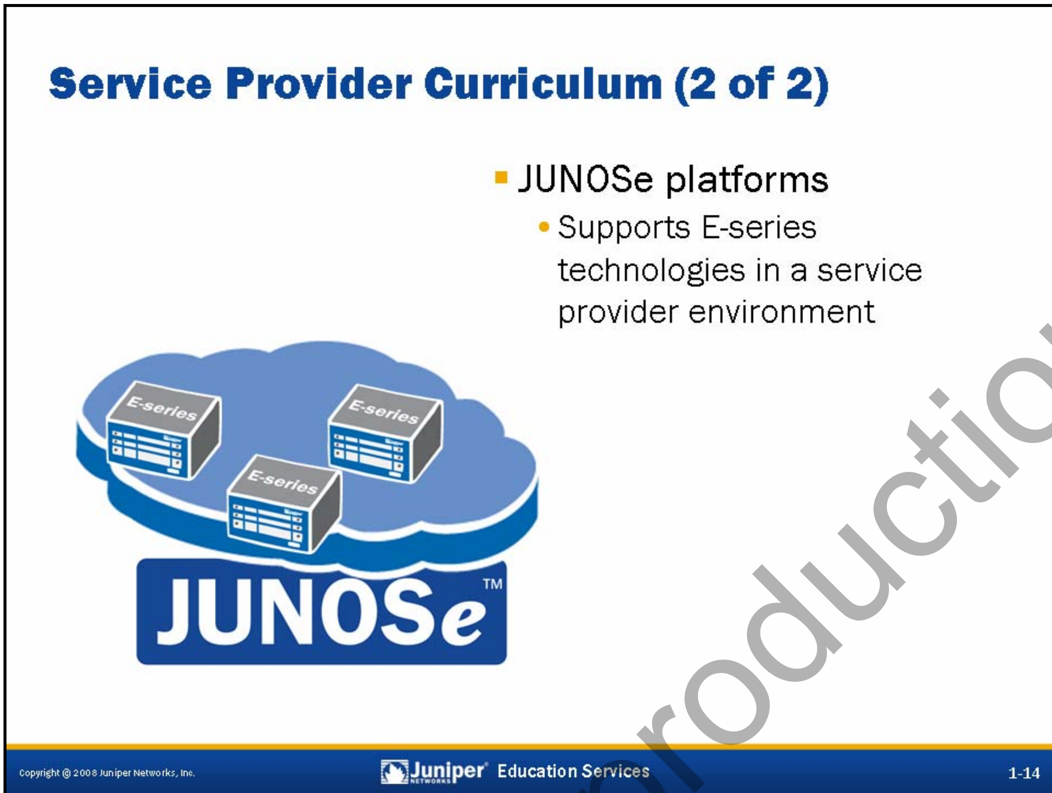
Enterprise Application Acceleration Curriculum

You can access the latest Education Services offerings that support Juniper Networks application acceleration technologies in an enterprise environment at http://www.juniper.net/training/technical_education/#appaccel.



Service Provider Curriculum: JUNOS Platforms

You can access the latest Education Services offerings that support Juniper Networks M-series, MX-series, and T-series technologies in a service provider environment by going to http://www.juniper.net/training/technical_education/ and clicking the `Service Provider` tab.



Service Provider Curriculum: JUNOSe Platforms

You can access the latest Education Services offerings that support Juniper Networks E-series technologies in a service provider environment by going to http://www.juniper.net/training/technical_education/ and clicking the `Service Provider` tab.

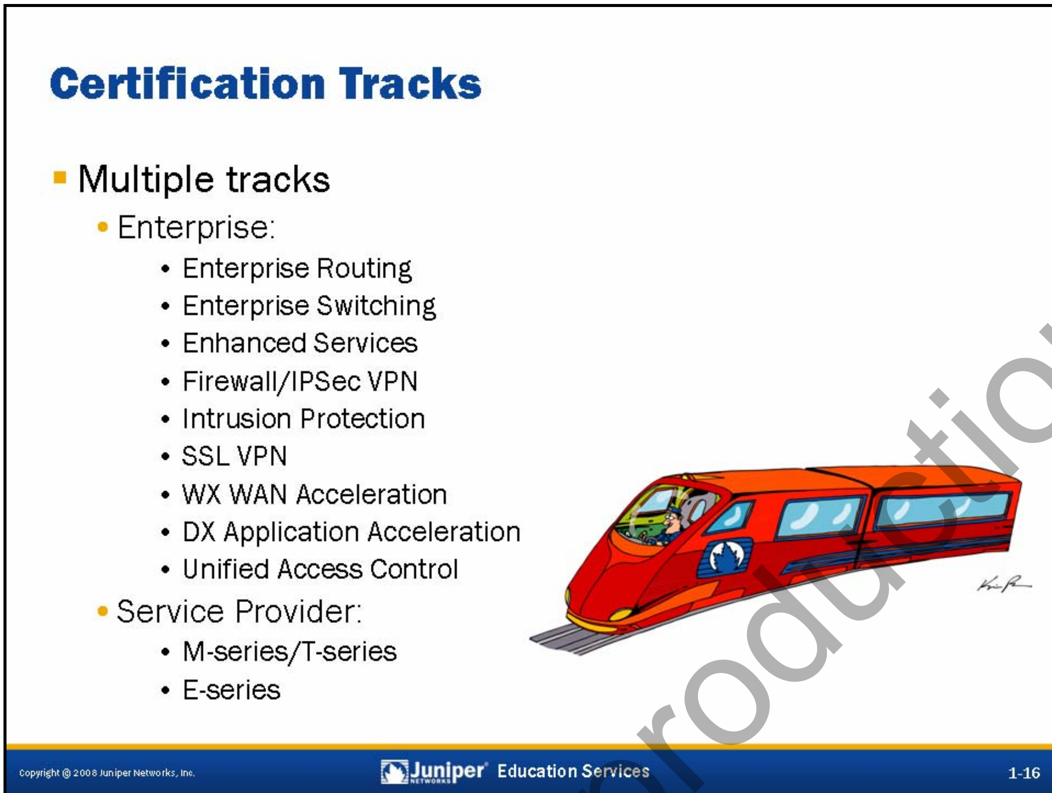
Technical Certification Programs

- Demonstrate competence with Juniper Networks technology
  - Multiple tracks
  - Multiple certification levels
  - Written proficiency exams
  - Hands-on configuration and troubleshooting exams

Juniper Education Services
1-15

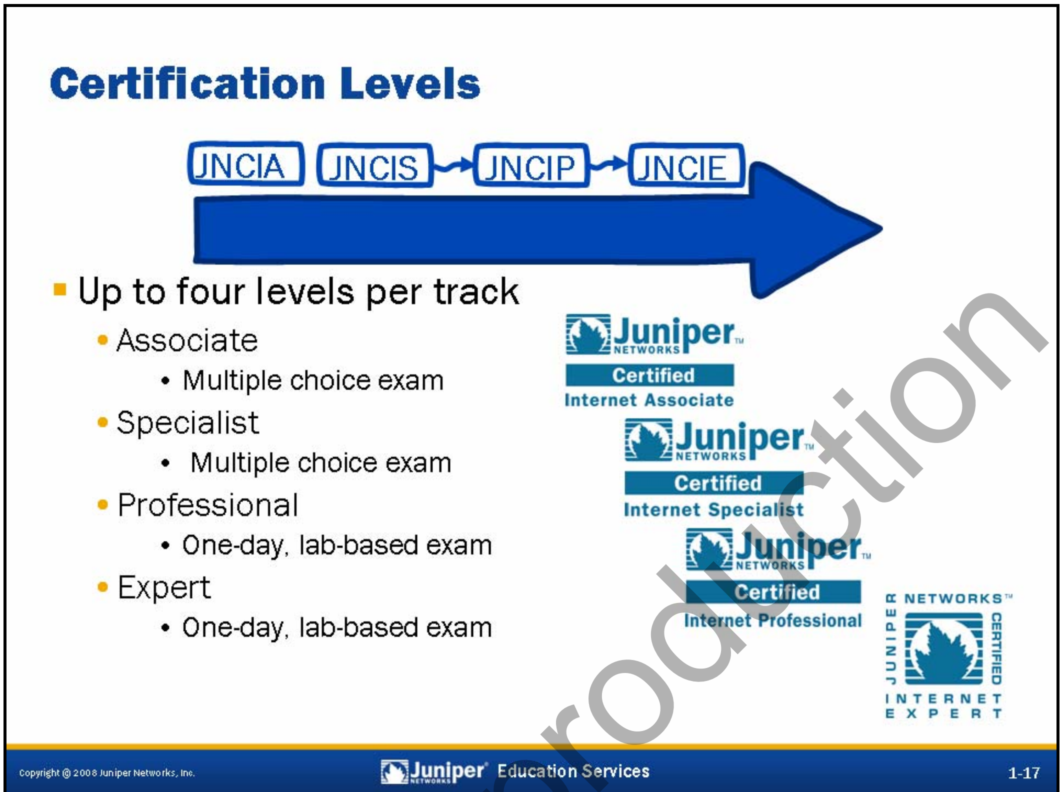
JNTCP

The Juniper Networks Technical Certification Program (JNTCP) consists of platform-specific, multitiered tracks that enable participants to demonstrate, through a combination of written proficiency exams and hands-on configuration and troubleshooting exams, competence with Juniper Networks technology. Successful candidates demonstrate thorough understanding of Internet and security technologies and Juniper Networks platform configuration and troubleshooting skills. You can learn more information about the JNTCP at http://www.juniper.net/training/certification/.



Multiple Tracks

This slide details the different JNTCP tracks. You can access more details on each of these tracks on the JNTCP Web site at http://www.juniper.net/training/certification/.



Certification Levels

Each JNTCP track has one to four certification levels. Associate-level and Specialist-level exams are computer-based exams composed of multiple choice questions. These computer-based exams are administered at Prometric testing centers worldwide and have no prerequisite certification requirements.

Professional-level and Expert-level exams are composed of hands-on lab exercises that are administered at select Juniper Networks testing centers. Professional-level and Expert-level exams require that you first obtain the next lower certification in the track. Please visit the JNTCP Web site at http://www.juniper.net/training/certification/ for detailed exam information, exam pricing, and exam registration.

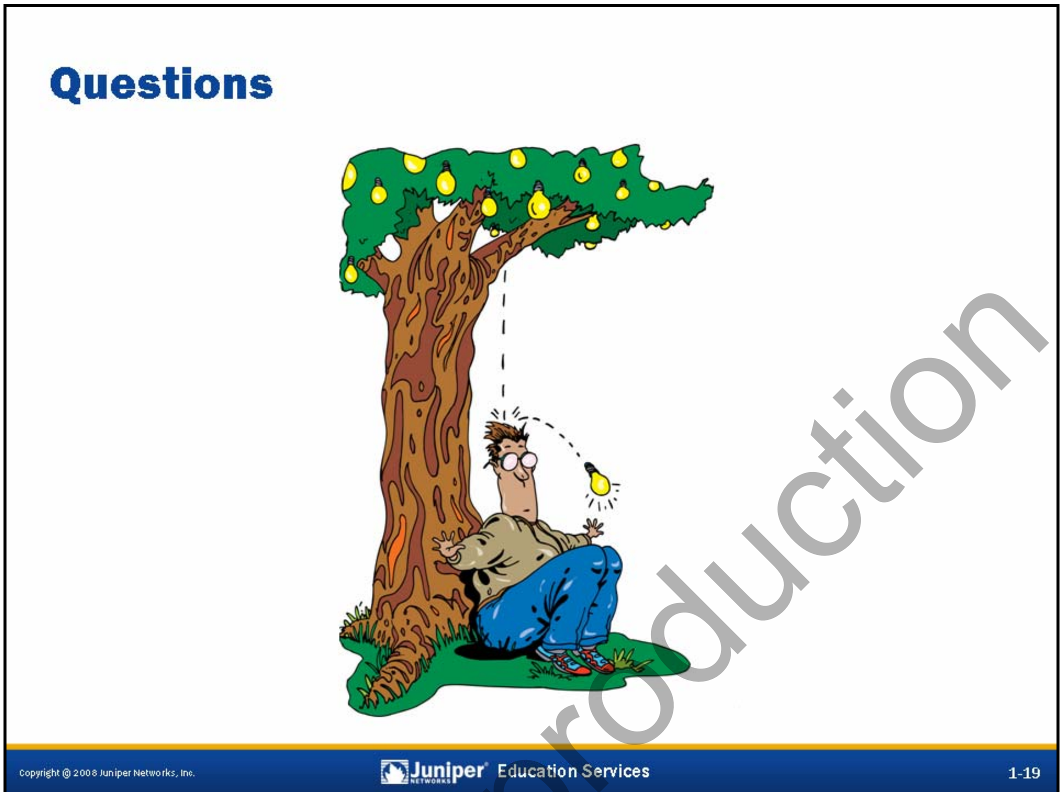
Certification Preparation

- How to prepare:
  - Training and study resources
    - JNTCP Web site
      http://www.juniper.net/training/certification/
    - Education Services training classes
      http://www.juniper.net/training/technical_education/
    - Juniper networks documentation and white papers
      http://www.juniper.net/techpubs/
  - Practical exams: lots of hands-on practice
    - On-the-job experience
    - Education Services training classes
    - Equipment access

 Juniper Education Services 1-18

Prepping and Studying

This slide lists some options for those interested in prepping for Juniper Networks certification.



Any Questions?

If you have any questions or concerns about the class you are attending, we suggest that you voice them now so that your instructor can best address your needs during class.

Operating Juniper Networks Switches in the Enterprise

Chapter 2: Introduction to Juniper Networks Enterprise Switches

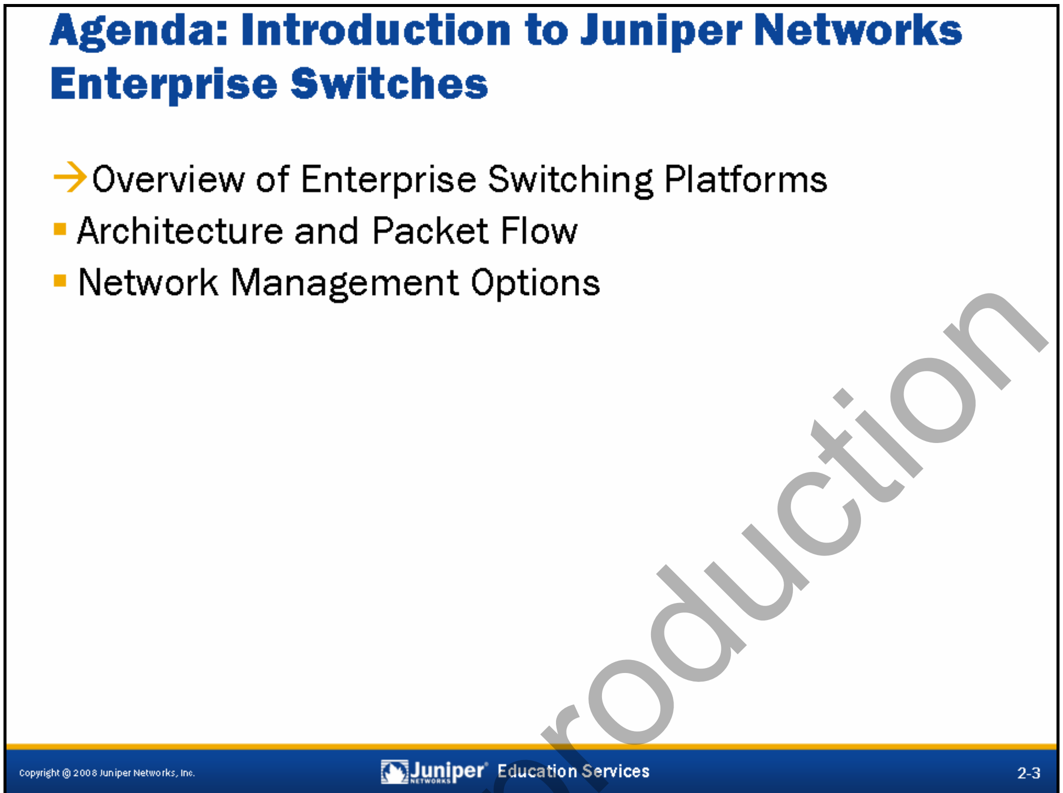
Chapter Objectives

- After successfully completing this chapter, you will be able to:
  - Describe the EX-series switching platforms
  - List the components of an EX-series switch
  - Explain the design architecture of EX-series switches
  - Describe packet flow through an EX-series switch
  - List the management options for EX-series switches

Juniper Networks Education Services
2-2

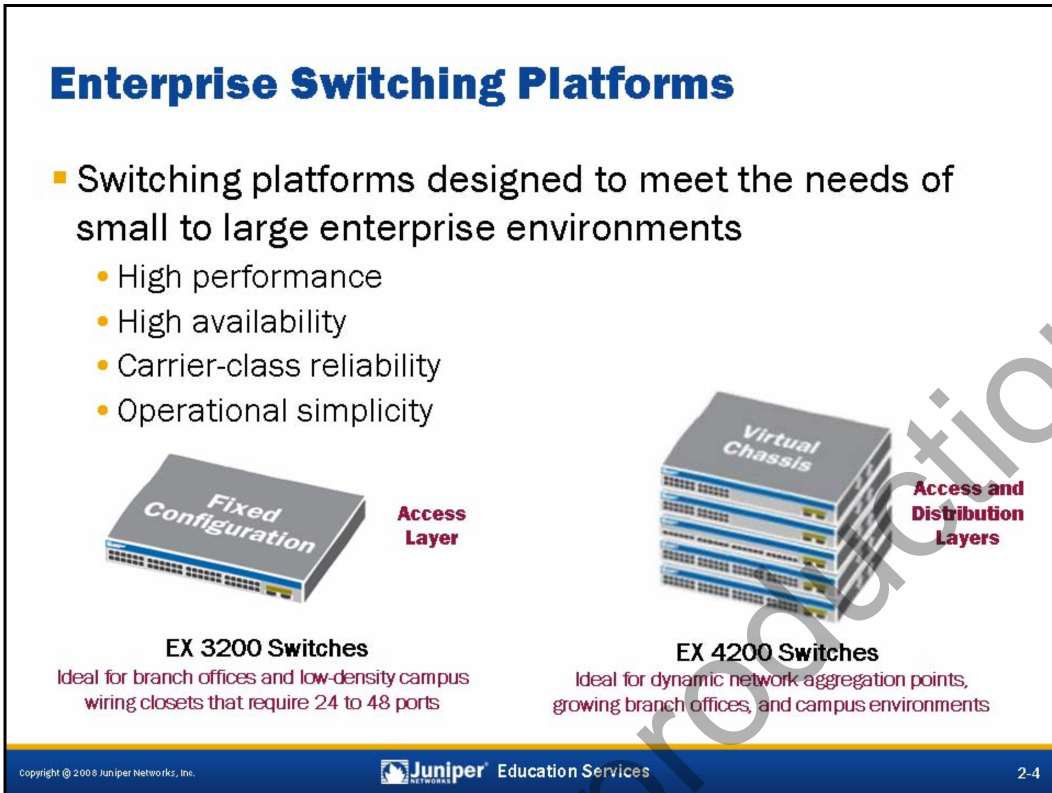
This Chapter Discusses:

- EX-series switching platforms;
- Components of EX-series switches;
- The design and architecture of EX-series switches;
- Packet flow through an EX-series switch; and
- Some common management options for EX-series switches.



Overview of Enterprise Switching Platforms

The slide lists the topics we cover in this chapter. We discuss the highlighted topic first.



Enterprise Switching Platforms

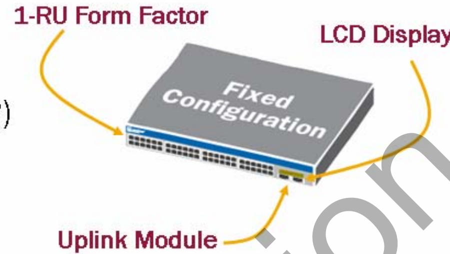
The slide outlines the Juniper Networks switches targeted for the enterprise market. These switches offer various combinations of price, performance, and redundancy to match the needs of both small and large enterprises.

The Juniper Networks EX 3200 Ethernet switches deliver best-in-class, high-performance connectivity for branch office environments and low-density campus wiring closets. The EX 3200 switches are available with either 24 or 48 10/100/1000Base-T Ethernet ports with either partial or full Power over Ethernet (PoE) configurations.

The Juniper Networks EX 4200 Ethernet switches provide a high-performance, scalable solution for medium- and high-density environments where incremental growth and high availability are absolute requirements. The Virtual Chassis switch family includes 24- and 48-port 10/100/1000Base-T platforms, as well as a 24-port 100Base-FX/1000Base-X SFP fiber platform. With the exception of the 24-port 100Base-FX/1000Base-X SFP fiber platform, all models offer either partial or full PoE options. EX 4200 switches can be interconnected to form a Virtual Chassis system, whereas EX 3200 series switches cannot. We cover the Virtual Chassis deployment option system in a subsequent chapter.

Fixed-Configuration Switches (EX 3200)

- Uplink modules:
  - 4-port 1 Gigabit Ethernet (SFP)
  - 2-port 10 Gigabit Ethernet (XFP)
- Power and cooling:
  - Field-replaceable AC PSU
  - Full Class 3 PoE (15.4 W)
  - Field-replaceable fan tray



Model	Access Port Configuration	PoE Ports	Switching Capacity	Height	System Power (with PoE)
EX 3200-24T	24 port 10/100/1000BASE-T	8	88 Gbps	1 RU	320W AC PSU
EX 3200-24P	24 port 10/100/1000BASE-T	24	88 Gbps	1 RU	600W AC PSU
EX 3200-48T	48 port 10/100/1000BASE-T	8	136 Gbps	1 RU	320W AC PSU
EX 3200-48P	48 port 10/100/1000BASE-T	48	136 Gbps	1 RU	930W AC PSU

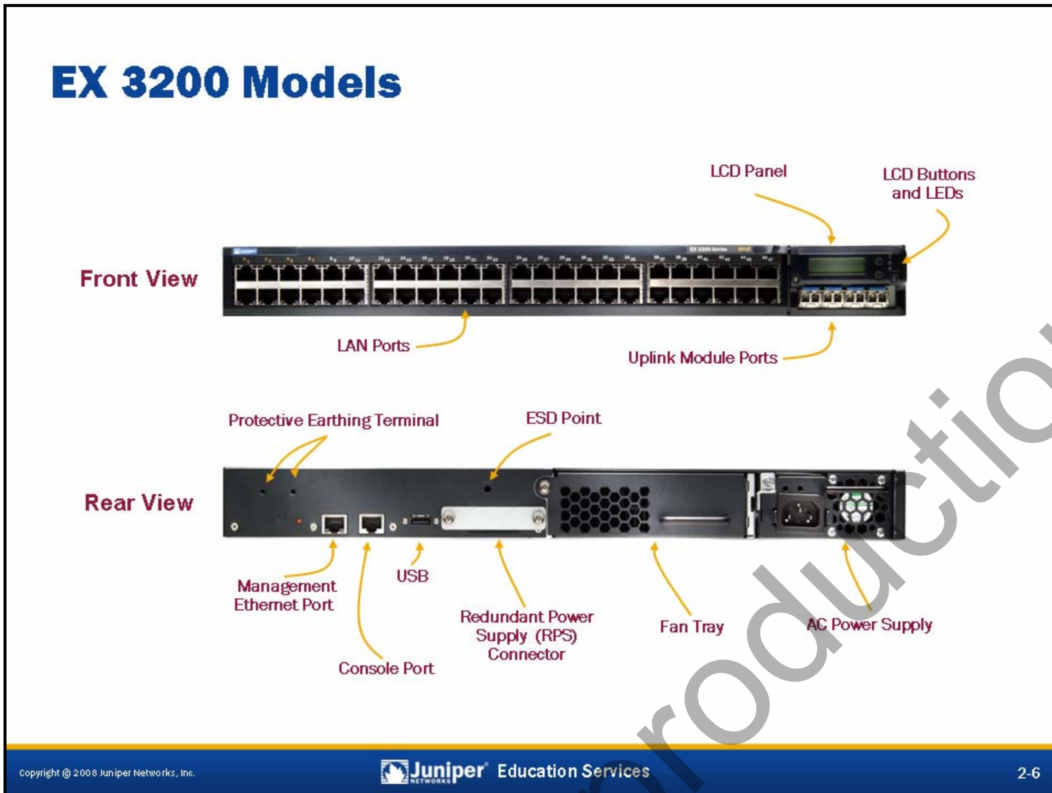
 Juniper Education Services 2-5

Uplink Modules

Optional uplink modules are available for all EX 3200 models. Uplink modules provide either two 10-gigabit small form-factor pluggable transceivers (XFPs) or four 1-gigabit small form-factor pluggable transceivers (SFPs). These ports are often used to connect an access switch to a distribution switch or a customer edge (CE) router.

Power and Cooling

The slide highlights the power and cooling components used in the EX 3200 switches. These platforms offer a field-replaceable AC power supply and a field-replaceable fan tray.



EX 3200 Switches

Juniper Networks EX 3200 fixed-configuration Ethernet switches are ideal for wiring closets and branch offices where a single switch is required. The EX 3200 platforms have the following specifications and features:

- *Compact 1-rack unit (RU) form factor.*
- *USB port*: This port accepts a USB storage device for use as a secondary storage device.
- *LAN ports*: These ports are fixed, autosensing 10/100/1000 Base-X Gigabit Ethernet ports. Options include 24 or 48 network ports.
- *Console port*: This port is a data terminal equipment (DTE) RS-232 serial port with an RJ-45 connector used to access the switch's command-line interface (CLI).
- *Management Ethernet port*: This Gigabit Ethernet port is used for out-of-band (OoB) management access.
- *Redundant power supply (RPS) connector*: This connector is used to connect an external redundant power supply.
- *Protective earthing terminal*: This terminal is the attachment point for a grounding cable that connects the switch to earth ground.
- *Electrostatic discharge (ESD) point:* This attachment point is used to ground a user to the device during maintenance operations.

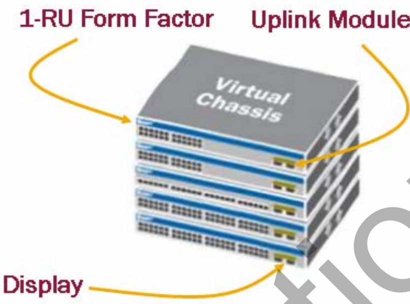
Continued on next page.

EX 3200 Switches (contd.)

- *Power supply and fan exhaust*: The AC power supply is field replaceable. The power supply contains a built-in exhaust and cooling fan.
- *Fan tray:* This tray contains the fan that cools the switch and is field replaceable.
- *LCD panel and buttons:* This panel allows the user to verify status information visually, while the buttons provide for various configuration operations.
- *LEDs:* These LEDs indicate status and alarm conditions.
- *Uplink module slot:* This uplink module slot is used as an insertion point for either a 4-port 1 Gigabit Ethernet module or a 2-port 10 Gigabit Ethernet module. You must order these modules separately.

Virtual Chassis Switches (EX 4200)

- Uplink modules:
  - 4-port 1 Gigabit Ethernet (SFP)
  - 2-port 10 Gigabit Ethernet (XFP)
- Power and cooling:
  - Dual, hot-swap AC PSU
  - Full Class 3 PoE (15.4 W)
  - Fan FRU, multiple blowers



Model	Access Port Configuration	PoE Ports	Switching Capacity	Height	System Power (with PoE)
EX 4200-24T	24 port 10/100/1000BASE-T	8	88 Gbps	1 RU	320W AC PSU
EX 4200-24P	24 port 10/100/1000BASE-T	24	88 Gbps	1 RU	600W AC PSU
EX 4200-24F	24 port 100Base-FX/1000BASE-X (SFP)	N/A	136 Gbps	1 RU	320W AC PSU
EX 4200-48T	48 port 10/100/1000BASE-T	8	136 Gbps	1 RU	320W AC PSU
EX 4200-48P	48 port 10/100/1000BASE-T	48	136 Gbps	1 RU	930W AC PSU

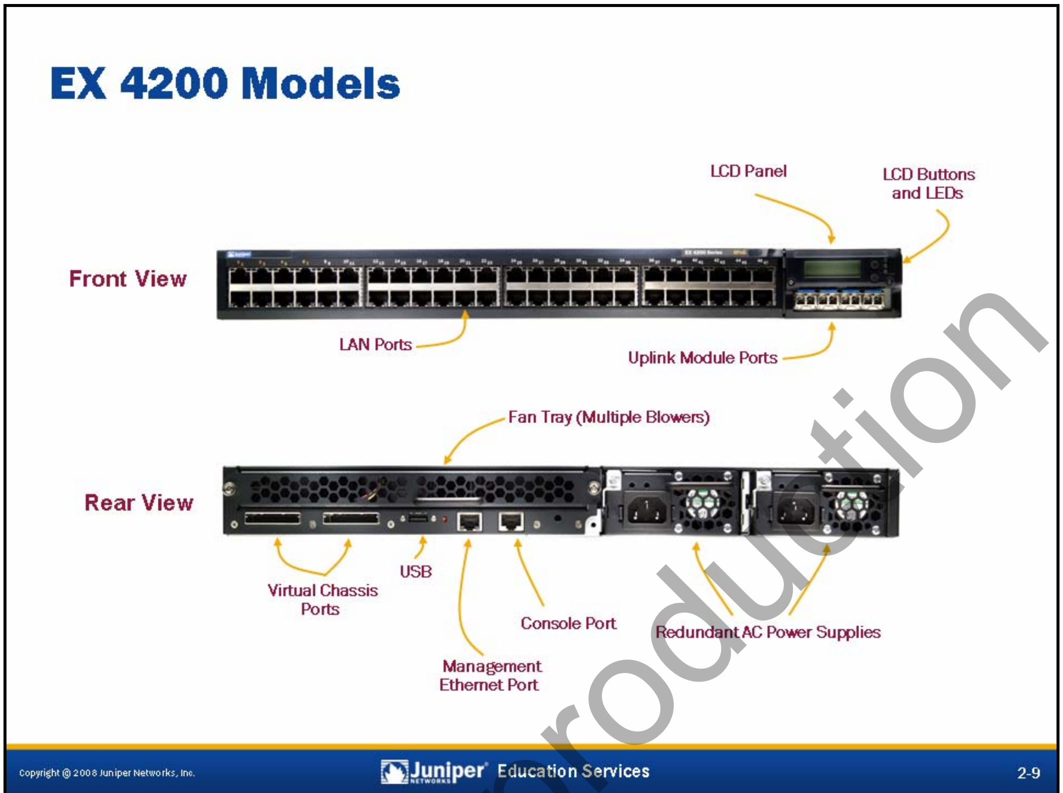
Juniper Education Services
2-8

Uplink Modules

Optional uplink modules are available for all EX 4200 models. Uplink modules provide either two XFPs or four SFPs. These ports are often used to connect an access switch to a distribution switch, or to interconnect member switches of a Virtual Chassis system across multiple wiring closets.

Power and Cooling

The slide highlights the power and cooling components used in the EX 4200 switches. These platforms offer dual, hot-swappable internal AC power supply units (PSUs), full Class 3 PoE (15.4 W) ports, and a field-replaceable fan tray with multiple blowers.



EX 4200 Switches

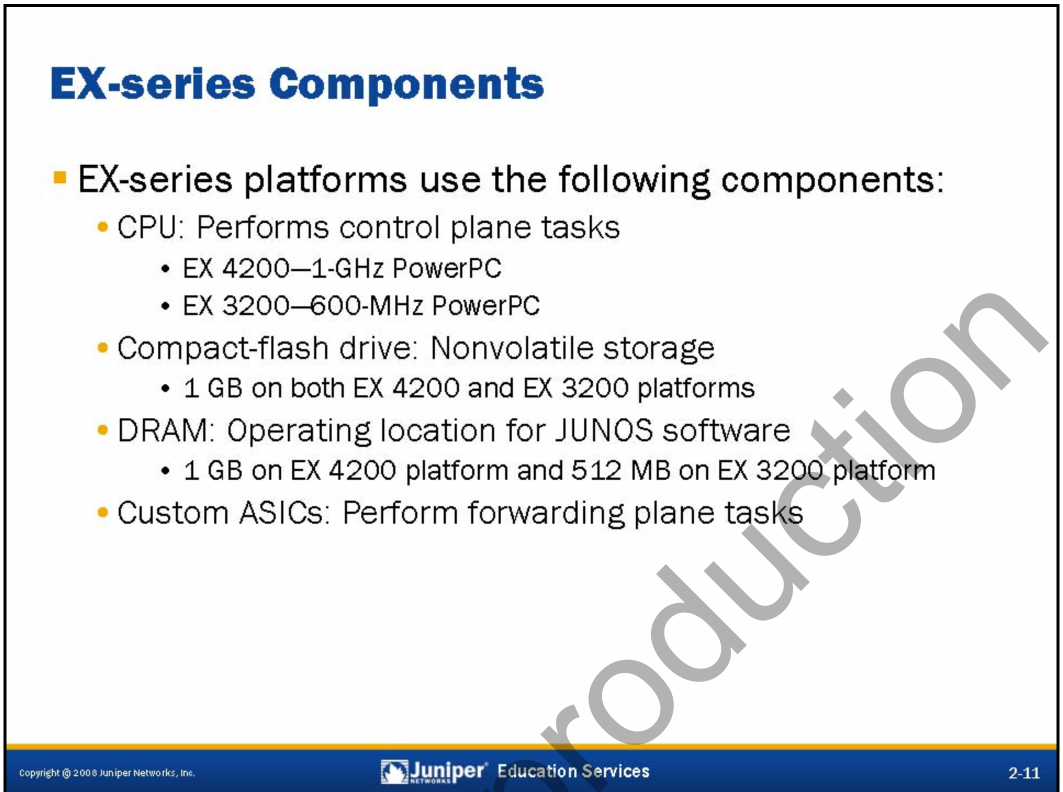
Juniper Networks EX-series Virtual Chassis Ethernet switches deliver a powerful, scalable solution for dynamic headquarters environments, network aggregation points, and growing office locations. The EX 4200 platforms have the following specifications and features:

- *Compact 1-RU form factor.*
- *USB port*: This port accepts a USB storage device for use as a secondary storage device.
- *LAN ports*: The EX 4200 switches offer fixed, autosensing 24- or 48-port 10/100/1000 Base-X as well as 24-port 100Base-FX/1000Base-FX (SFP) ports.
- *Console port*: This port is a DTE RS-232 serial port with an RJ-45 connector used to access the switch’s CLI.
- *Management Ethernet port*: This Gigabit Ethernet port is used for OoB management access.

Continued on next page.

EX 4200 Switches (contd.)

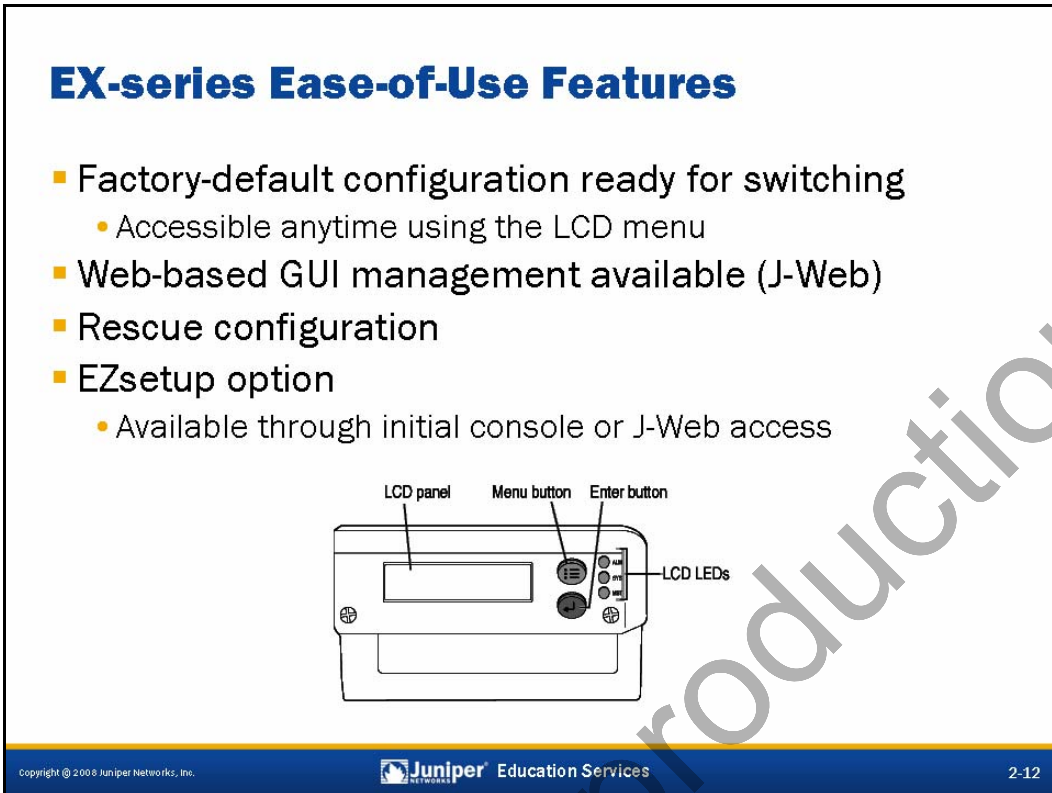
- *Power supply*: These dual, load-sharing, redundant power supplies are field replaceable and hot swappable.
- *Fan tray*: This field-replaceable fan tray contains three fans. The switch remains operational if a single fan fails.
- *LCD panel and buttons*: This panel allows the user to verify status information visually, while the buttons provide for various configuration operations.
- *LEDs*: These LEDs indicate status and alarm conditions.
- *Uplink module slot*: This uplink module slot is used as an insertion point for either a 4-port 1 Gigabit Ethernet module or a 2-port 10 Gigabit Ethernet module. You must order these modules separately.
- *Virtual Chassis ports*: These ports are used to interconnect multiple EX 4200 switches to create a Virtual Chassis system.



EX-series Components

EX 3200 and EX 4200 platforms use the following components:

- *CPU*: Performs control plane tasks such as protocol maintenance and routing and forwarding updates. The CPU is the primary component in the Routing Engine (RE).
- *Compact-flash drive*: Provides nonvolatile storage on EX-series platforms.
- *DRAM*: Runs JUNOS software and stores route and forwarding table information.
- *Custom application-specific integrated circuits (ASICs)*: Perform the various forwarding plane tasks such as next-hop lookup services and forwarding. These ASICs are the primary components of the Packet Forwarding Engine (PFE).



Ready for Switching, Right Out of the Box

EX-series switches come from the factory with a working switching configuration. Plug the switch in, attach network cables, and the switch is ready for your network. All network ports are enabled for switching in the factory-default configuration. You can load the factory-default configuration anytime using the LCD menu or by using the CLI.

J-Web GUI Management

J-Web is a graphical user interface (GUI) used to configure and monitor EX-series platforms and is installed by default on all EX-series switches.

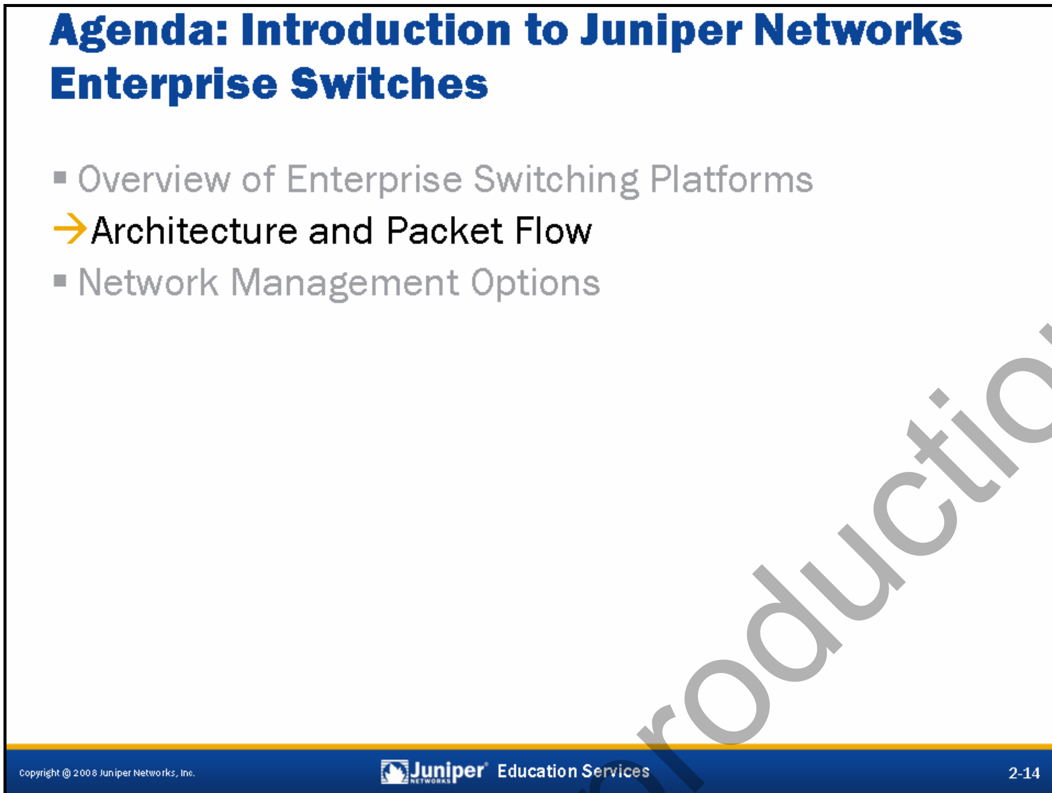
Rescue Configuration

The rescue configuration is a user-defined configuration that is known to be good and will allow user access. Once you have a known working configuration, save the configuration using J-Web or the CLI. You can load the rescue configuration through the CLI.

Continued on next page.

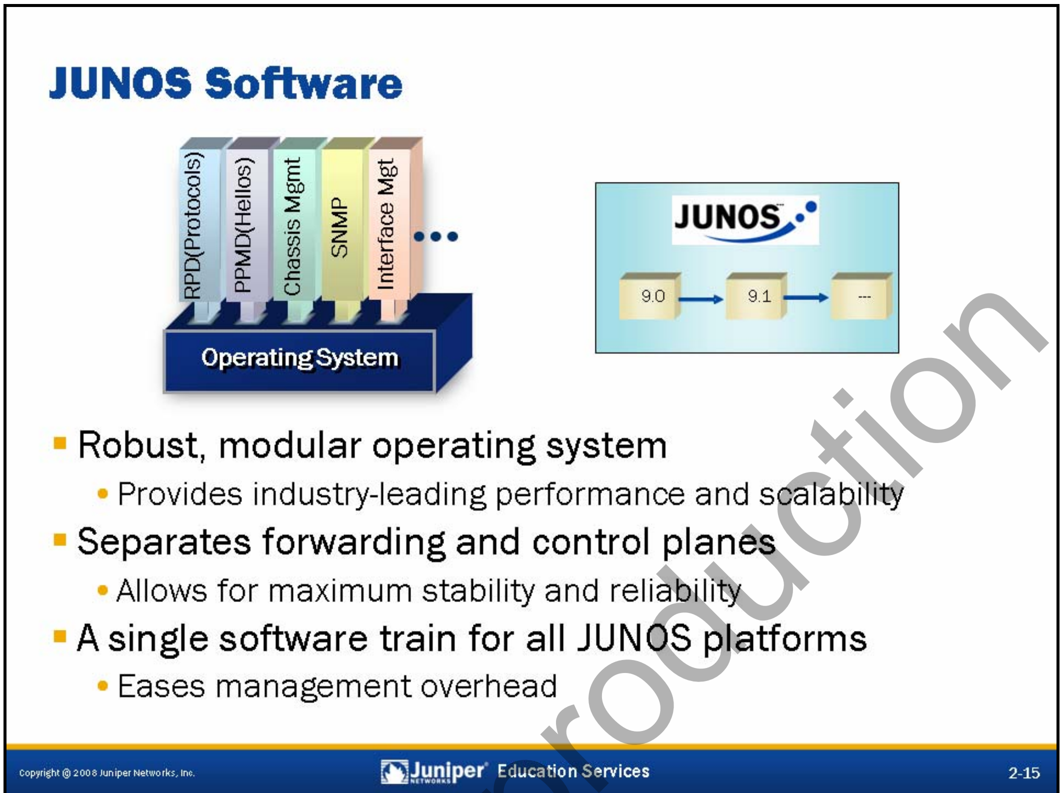
Initial Setup

You can use the EZsetup option, available through the console connection or the J-Web GUI. To access the EZsetup utility using the console connection, simply type **`ezsetup`** at the shell prompt when logged in with the `root` account. The J-Web EZsetup option is initiated by attaching a DHCP-enabled PC to the ge-0/0/0 or me0 interface and selecting `Enter EZsetup` from the LCD menu. Both EZsetup options require the switch to be in a factory-default state. These configuration features were added to the JUNOS software to ease initial configuration and to lessen the support overhead in remote locations that might not have full-time networking staff on site.



Architecture and Packet Flow

The slide highlights the topic we discuss next.



Robust, Modular, and Scalable

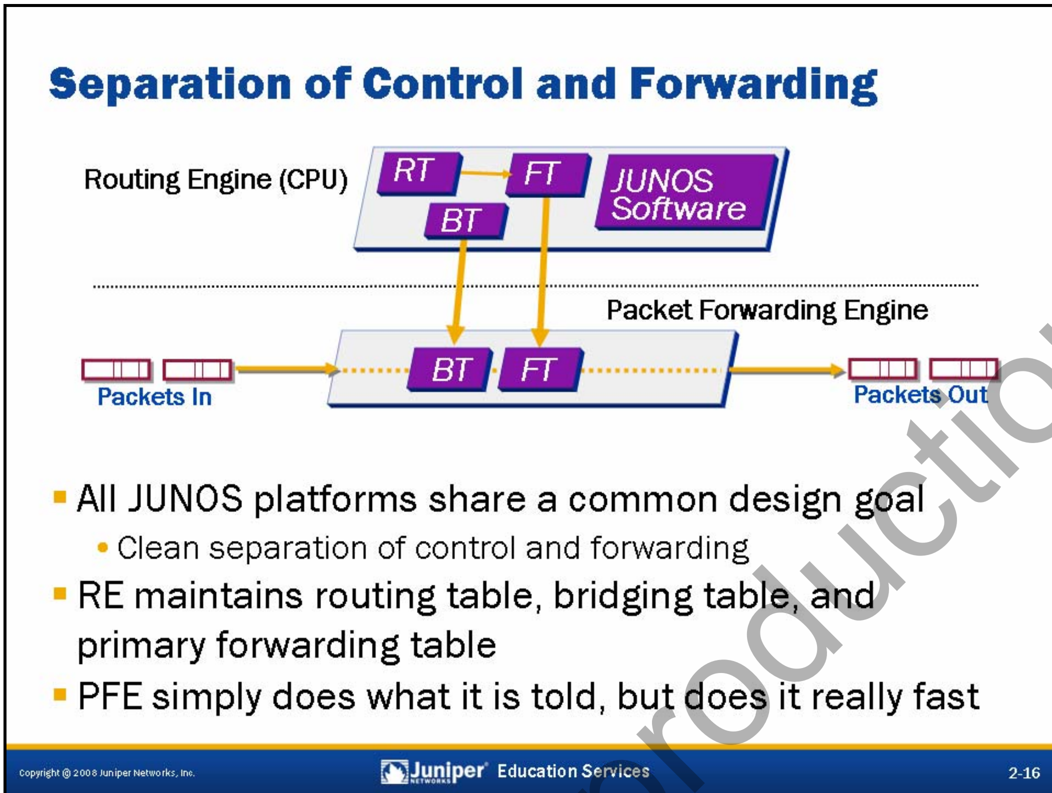
JUNOS software functionality is compartmentalized into multiple software processes. Each process handles a portion of the switch's functionality. Each process runs in its own protected memory space, ensuring that one process cannot directly interfere with another. When a single process fails, the entire system does not necessarily fail. This modularity also ensures that new features can be added with less likelihood of breaking current functionality.

Separate Forwarding and Control Planes

Another aspect of the JUNOS software's modularity is the separation of the forwarding plane and the control plane. The processes that control routing and switching protocols are cleanly separated from the processes that forward packets through the switch. This design allows each process to be tuned for maximum performance and reliability.

Single Software Source Code

JUNOS software on the EX-series platforms uses the same source code that is used on the J-series, M-series, and T-series platforms. This design ensures that features work the same across all JUNOS-based platforms. Enabling new software features does not require changing to a different JUNOS binary.



Architectural Philosophy

Architecturally, all JUNOS-based platforms share a common design that separates the switch's or router's control and forwarding planes. To this end, all EX-series, J-series, M-series, T-series, and MX-series platforms consist of two major components:

- *The Routing Engine (RE)*: The RE is the brains of the platform; it is responsible for performing protocol updates and system management. The RE runs various protocol and management software processes that reside inside a protected memory environment. The RE is based on the PowerPC architecture. The RE maintains the routing tables, bridging table and primary forwarding table and is connected to the PFE through an internal link.
- *The Packet Forwarding Engine (PFE)*: The PFE is responsible for forwarding transit packets through the switch. The PFE is implemented using ASICs on the EX-series platforms. Because this architecture separates control operations—such as protocol updates and system management—from packet forwarding, the switch can deliver superior performance and highly reliable deterministic operation.

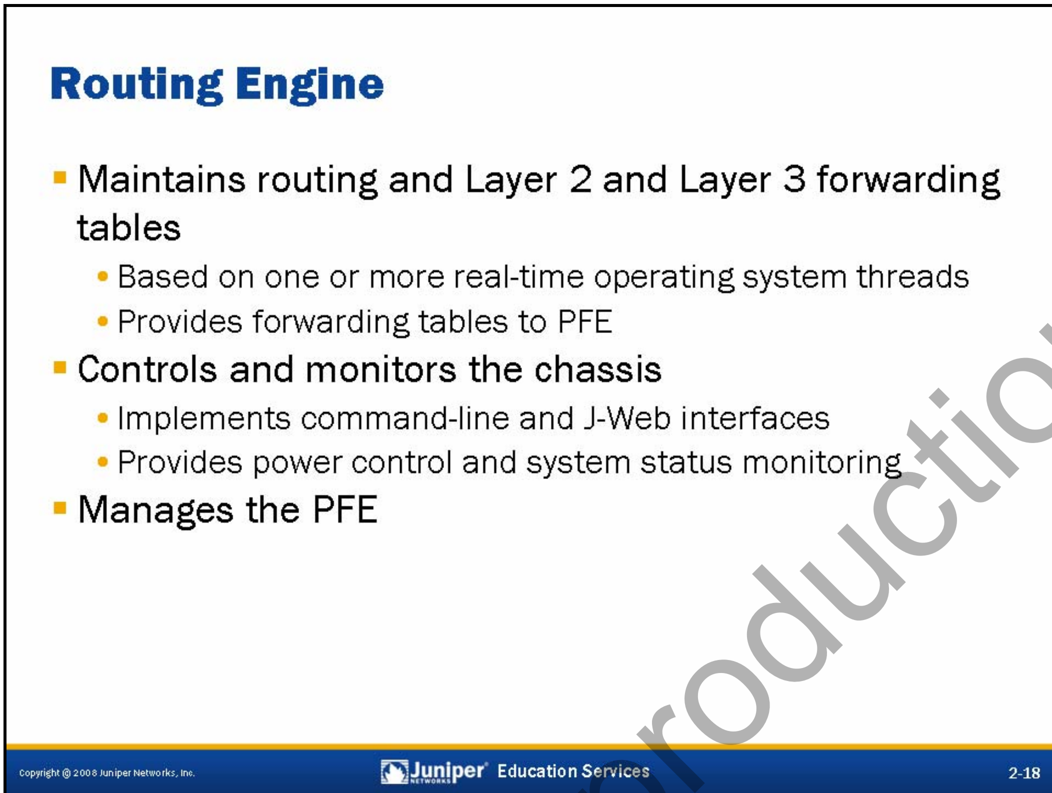
Continued on next page.

RE and PFE Interaction

The PFE receives the forwarding table (FT) and bridging table (BT) from the RE by means of an internal link. FT and BT updates are a high priority for the JUNOS software kernel and are performed incrementally.

The PFE Does What It Is Told

Because the RE provides the intelligence side of the equation, the PFE can simply do what it is told to do—that is, it forwards packets with a high degree of stability and deterministic performance.



Routing Engine Intelligence

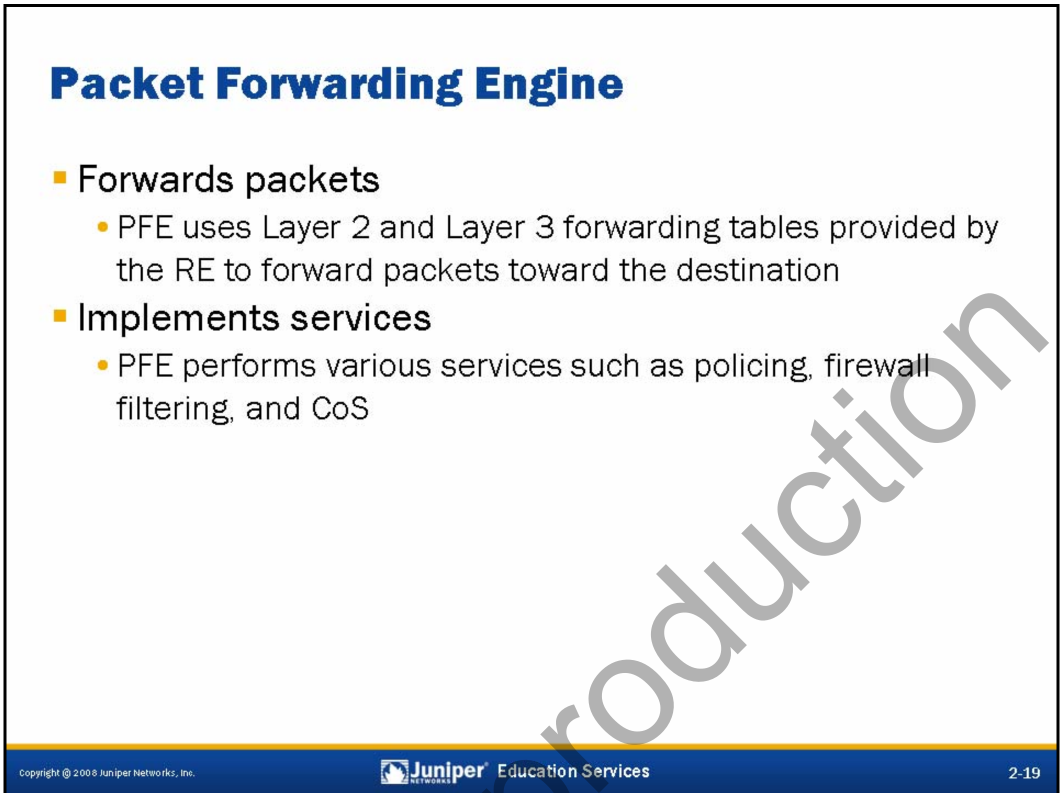
The RE handles all switching and routing protocol processes as well as other software processes that control the switch's interfaces, the chassis components, system management, and user access to the switch. These software processes run on top of the JUNOS kernel that interacts with the PFE. The switch directs all switching and routing protocol packets from the network to the RE.

Controls and Monitors

The RE provides the CLI as well as the J-Web GUI. These user interfaces run on top of the JUNOS kernel and provide user access and control of the switch.

Packet Forwarding Engine Management

The RE controls the PFE by providing accurate and up-to-date Layer 2 and Layer 3 forwarding tables and by downloading microcode and managing software processes that reside in the PFE's microcode. The RE receives hardware and environmental status messages from the PFE and acts upon them as appropriate.



Packet Forwarding

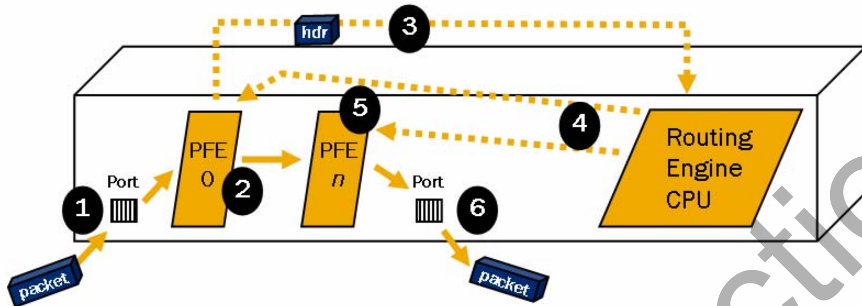
Each EX-series switch contains one or more PFEs. The PFE forwards packets based on the Layer 2 and Layer 3 forwarding tables provided by the RE. The 24-port EX 3200 models contain a single PFE, whereas the 48-port EX 3200 models contain two PFEs. The 24-port EX 4200 models contain two PFEs, whereas the 48-port EX 4200 models contain three PFEs.

Advanced Services

In addition to forwarding packets, the PFEs also implement a number of advanced services. Some examples of advanced services implemented through the PFE include policers that provide rate limiting, firewall filters, and class of service (CoS).

Packet Flow Overview (1 of 3)

- Transit packets with known destination MAC address:



1.	Packet enters port and attached ingress PFE.
2.	PFE performs MAC address lookup. If source MAC address is known, packet is forwarded to egress PFE.
3.	If source MAC address is unknown, header information is extracted and sent to RE, where it is added or discarded (MAC limiting).
4.	Newly learned source MAC address is programmed into PFE(s).
5.	Egress forwards packet to egress port. No additional lookup is needed.
6.	Packet is sent towards destination.

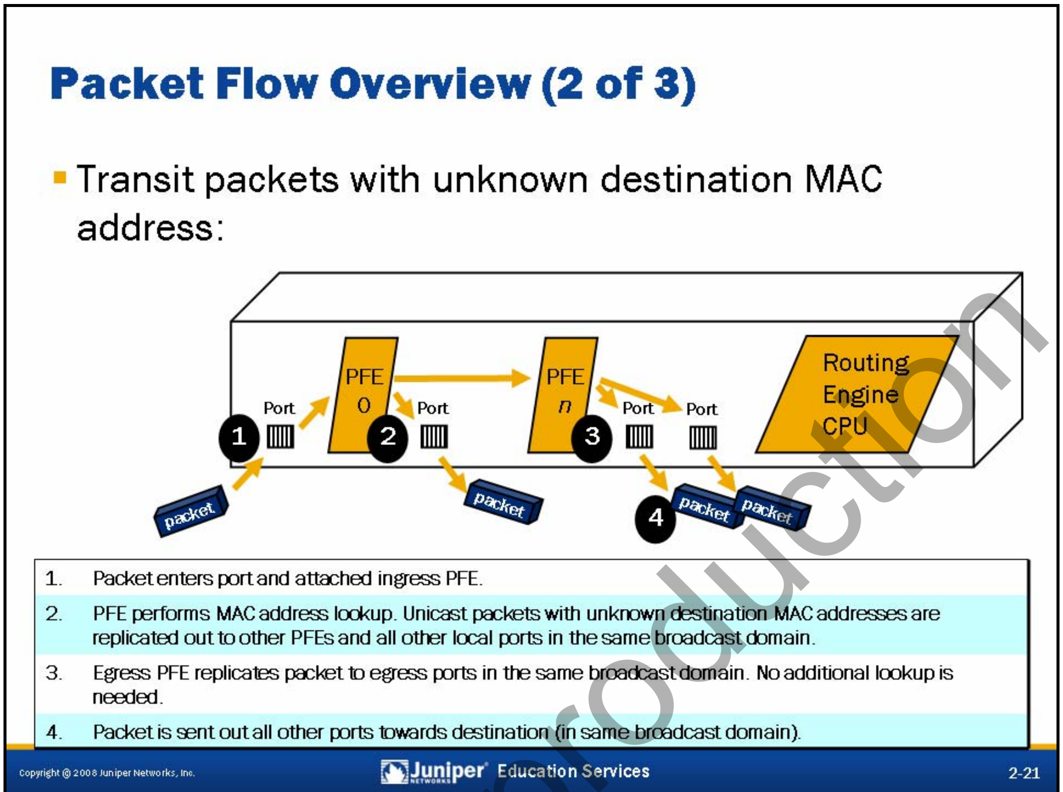
 Juniper Education Services 2-20

Switching Transit Packets with a Known Destination MAC Address

When packets enter a switch port, they are processed by the ingress PFE associated with that port. The ingress PFE determines how transit packets are processed and which lookup table is used when determining next-hop information. The PFE performs a lookup on the source and destination MAC address. In the example illustrated on the slide, the destination MAC address exists in the current bridging table.

If the egress port belongs to the ingress PFE, the packet is switched locally. If the egress port belongs to a PFE other than the ingress PFE, the packet is forwarded on through the switch fabric to the egress PFE where the egress switch port resides. This PFE might be a different PFE on the same switch or a remote PFE belonging to a separate member switch within the same Virtual Chassis system. We cover the Virtual Chassis details in a subsequent chapter.

If the source MAC address does not exist in the current bridging table, the PFE extracts and sends the header to the RE to update the bridging table, which is part of the MAC learning process.

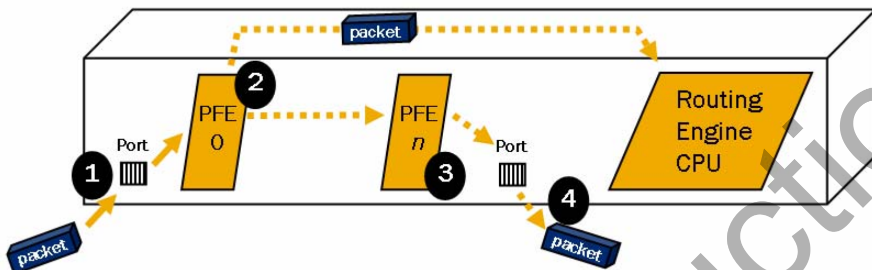


Switching Transit Packets with an Unknown Destination MAC Address

When the ingress PFE performs a lookup on the destination MAC address and no entry exists in the bridging table, the packet is flooded out all ports in the same broadcast domain. The packet is also flooded to other PFEs. However, the packet is not flooded out the port on which it was received. Once the switch sees return traffic from this MAC address, the address is added to the bridging table. Packets with broadcast and multicast destination MAC addresses are also flooded in a similar fashion.

Packet Flow Overview (3 of 3)

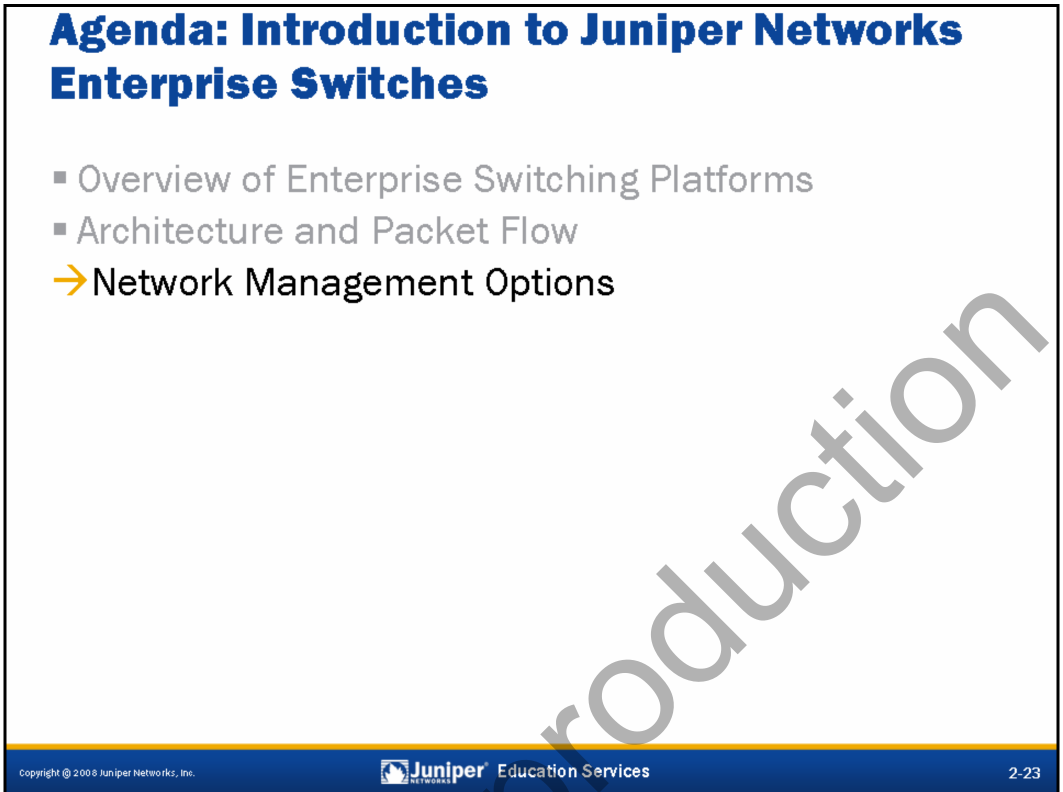
- Routed packets (destination MAC address is the switch's MAC address):



1.	Packet enters port and attached ingress PFE.
2.	PFE performs MAC address lookup. Because the destination MAC address is itself, Layer 3 lookup is performed. If destination IP address is the switch itself, packet is sent to RE. If the destination IP address is not the switch, packet is forwarded to egress PFE.
3.	Egress forwards packet to egress port. No additional lookup is needed.
4.	Packet is sent towards destination.

Routing Packets

When the PFE detects its own address as the destination MAC address, a Layer 3 lookup is performed. If the destination IP address belongs to the switch, the packet is forwarded to the RE. If the destination address is not the switch but exists in the forwarding table, the packet is forwarded to the egress PFE. If the destination IP address is not the switch and it is not listed in the forwarding table, the packet is discarded.



Network Management Options

The slide highlights the topic we discuss next.

Management Solutions

- Device management:
  - JUNOS CLI (Telnet, SSH, and XML)
  - J-Web (HTTP, HTTPS, and XML)
- Network management:
  - SNMP, RMON, NETCONF, and syslog
  - Third-party options include:
    - HP—OpenView NNM
    - IBM—Tivoli NetView and NetCool
    - CA—Unicenter

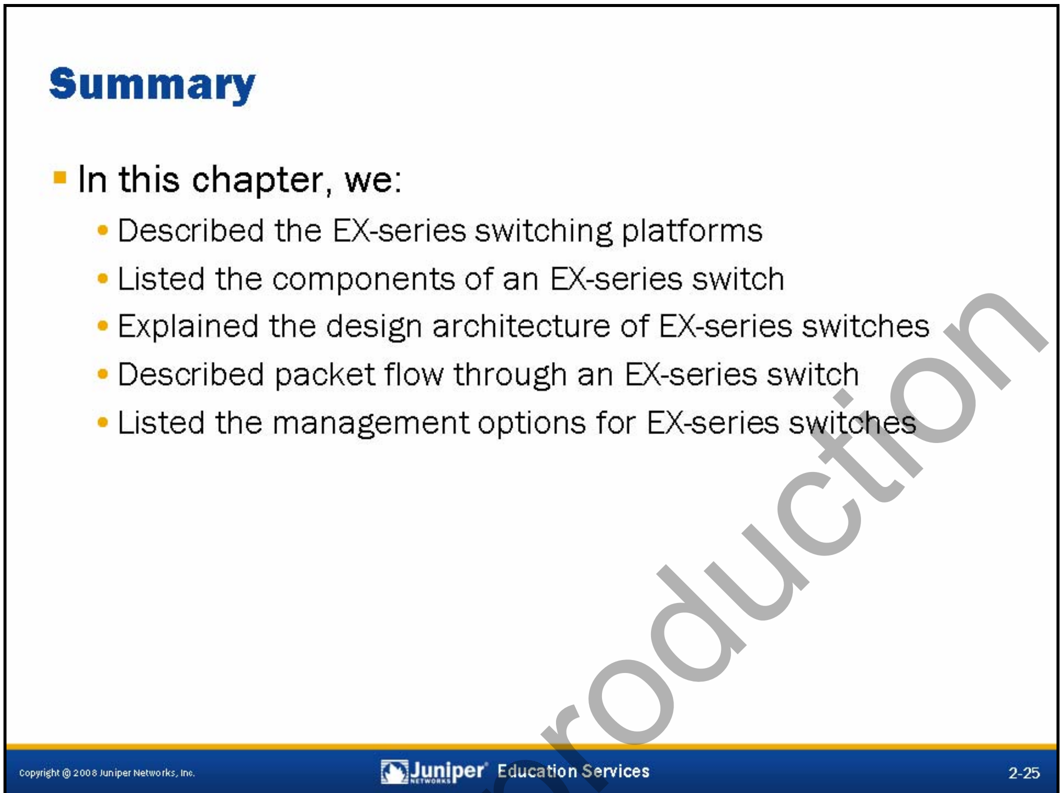
Juniper Education Services 2-24

Device Management

The traditional JUNOS software CLI gives access to all features. The J-Web user interface provides a graphical tool for common configuration tasks. The J-Web user interface is not intended to provide the full functionality found in the CLI.

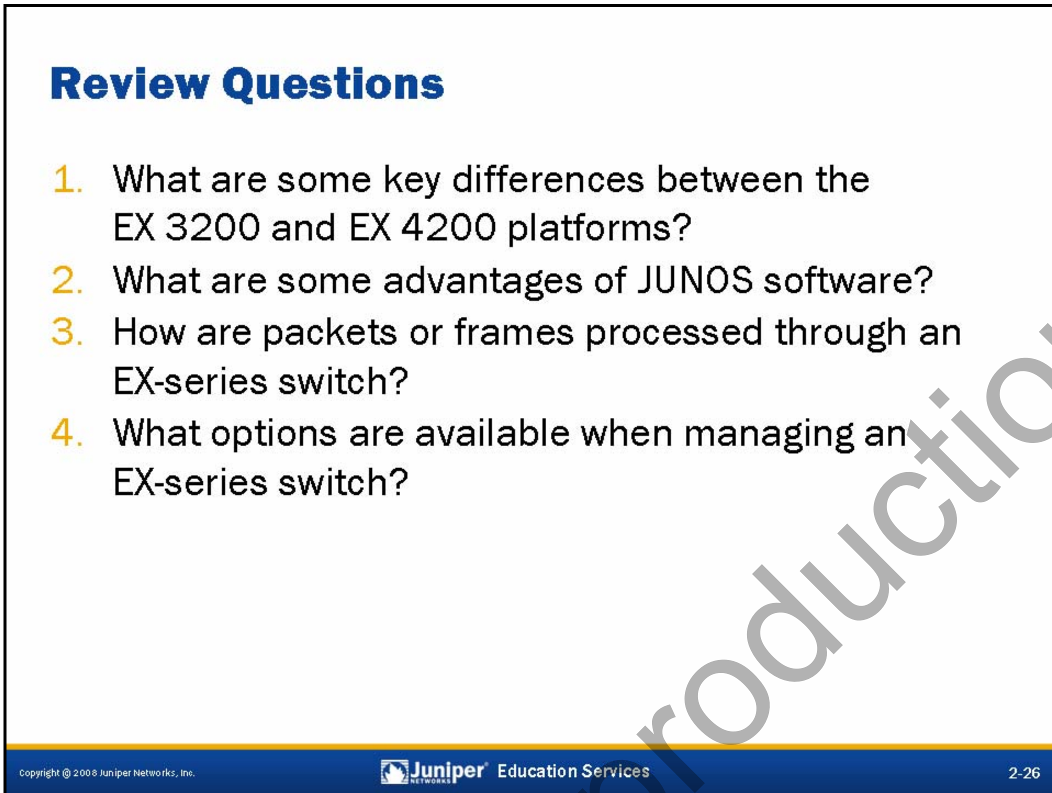
Network Management

The JUNOS software can act as an SNMP agent. It supports SNMP versions 1, 2c, and 3. Several standard and Juniper Networks enterprise-specific MIBs are supported. Refer to the Juniper Networks Web site for details about supported MIBs. In addition to SNMP, JUNOS software also supports Remote Monitoring (RMON), an SNMP extension. Other management options include NETCONF and syslog. The screen highlights some third-party options that make use of SNMP, RMON, NETCONF, or syslog.



This Chapter Discussed:

- EX-series switching platforms;
- Components of an EX-series switch;
- The design architecture of an EX-series switch;
- Packet flow through an EX-series switch; and
- Some common management options for EX-series switches.



Review Questions

1.

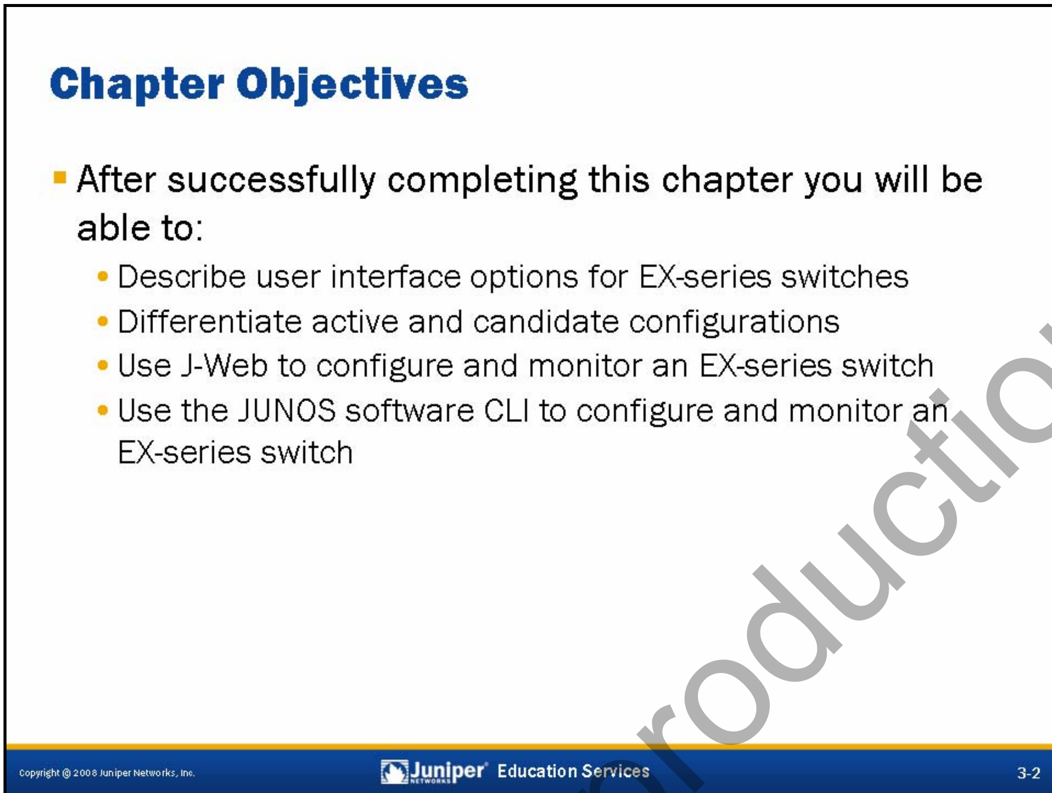
2.

3.

4.

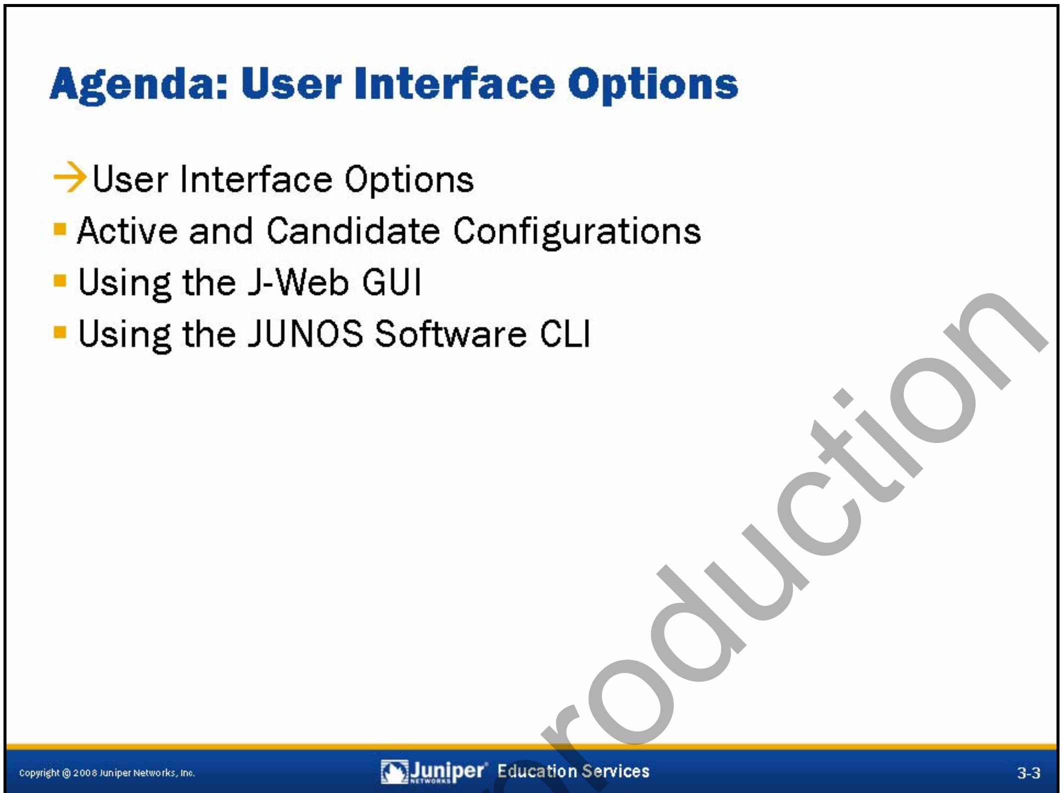
Operating Juniper Networks Switches in the Enterprise

Chapter 3: User Interface Options



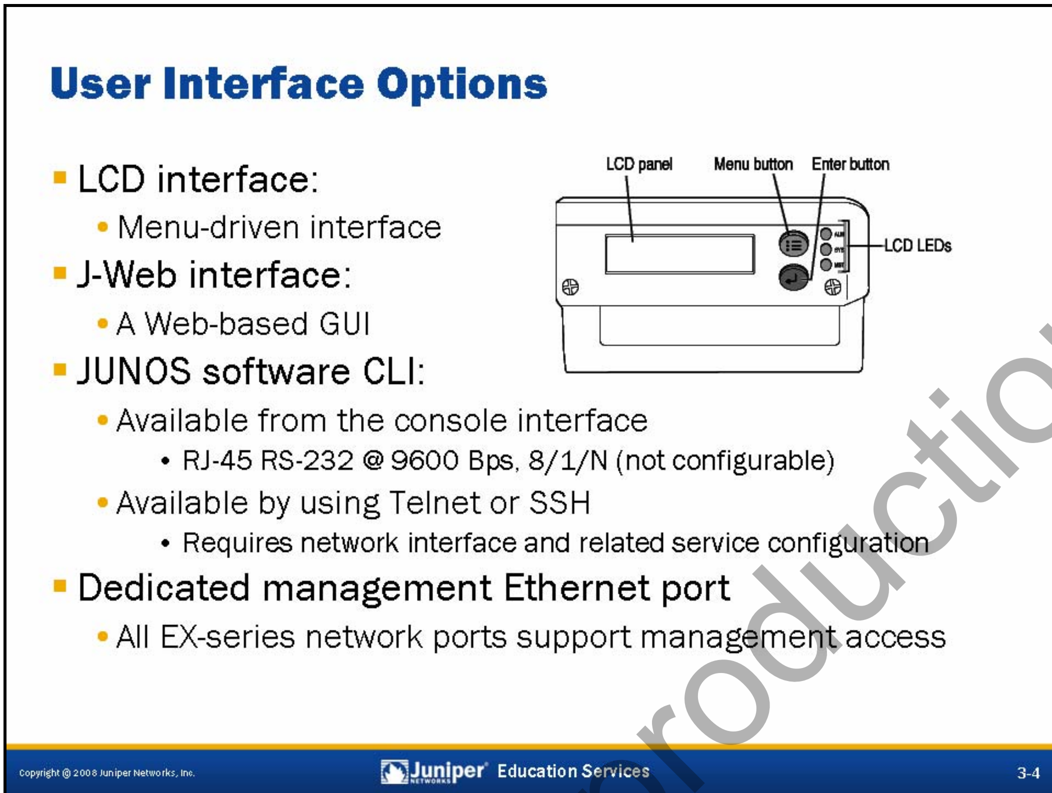
This Chapter Discusses:

- User interface options;
- Active and candidate configurations;
- Using the J-Web graphical user interface (GUI) to configure and monitor an EX-series switch; and
- Using the command-line interface (CLI) to configure and monitor an EX-series switch.



User Interface Options

The slide lists the topics we cover in this chapter. We discuss the highlighted topic first.



LCD Interface

Juniper Networks EX-series switches have a menu-driven LCD interface located on the right side of the front panel. Two buttons next to the LCD allow navigation and selection. You can use this interface in three modes:

- *Idle mode*: In this mode, the LCD displays the system temperature and the status of power supplies and fans.
- *Navigation mode*: This mode enables you to reboot the switch, restore the factory-default configuration, perform initial setup operations, and view the status of LEDs.
- *Alarm mode*: The LCD switches to this mode for the viewing of alarms when alarms occur.

J-Web Interface

J-Web is a Web-based GUI that you can access by using either Hypertext Transfer Protocol (HTTP) or HTTP over Secure Sockets Layer (HTTPS). It provides quick configuration wizards to simplify the most common configuration tasks. For more complicated configurations, the J-Web GUI allows you to directly edit the switch’s text configuration file. The J-Web GUI is installed and enabled by default on EX-series switches.

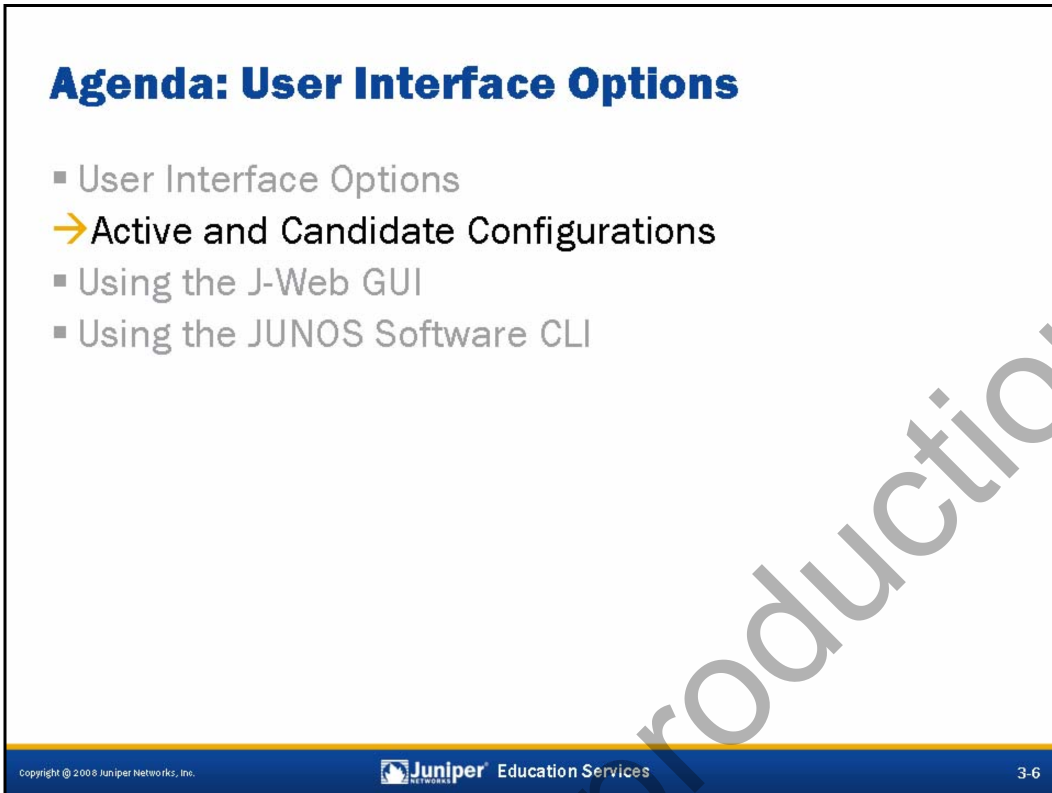
Continued on next page.

JUNOS Software CLI

You can access the JUNOS software CLI over the network (in band) by using the Telnet or SSH protocols. SSH versions 1 and 2 are supported. JUNOS software CLI access is also available using an out-of-band (OoB) serial console connection.

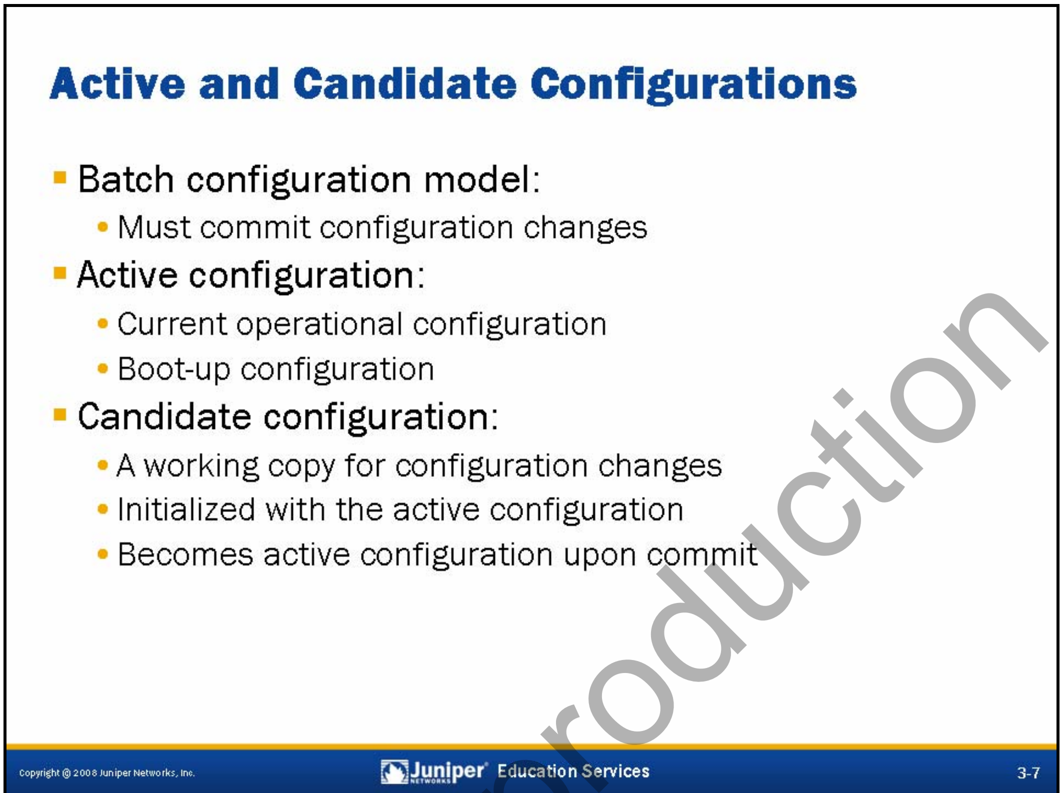
Dedicated Management Port

The EX-series switch has a dedicated management port named me0. This port provides OoB access; therefore, transit traffic cannot be forwarded over me0. You can also use network ports to manage the switch once the ports are configured.



Active and Candidate Configurations

The slide highlights the topic we discuss next.



Batch Configuration Changes

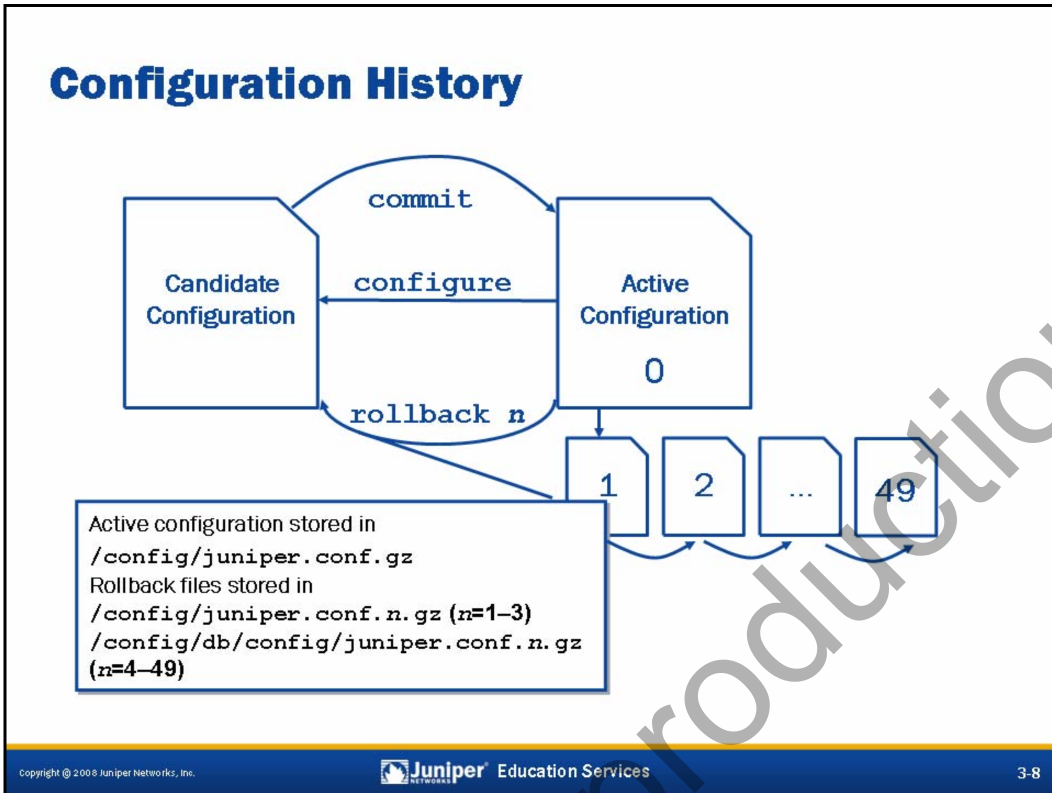
Unlike some vendors' software, configuration changes to the JUNOS software do not take affect immediately. This design feature allows you to group together and apply multiple configuration changes to the running configuration as a single unit.

Active Configuration

The active configuration is the configuration currently operational on the switch. It is also the configuration the switch loads during the boot sequence. This concept is analogous to both the *running configuration* and *startup configuration* in other vendors' software.

Candidate Configuration

The candidate configuration is a temporary configuration that might possibly become the active configuration. When you configure the switch, JUNOS software creates a candidate configuration and initially populates it with the switch's active configuration. You then modify the candidate configuration. Once satisfied with your modifications, you can apply or commit the changes. This action causes the candidate configuration to become the active configuration.



Configuration Files and Configuration History

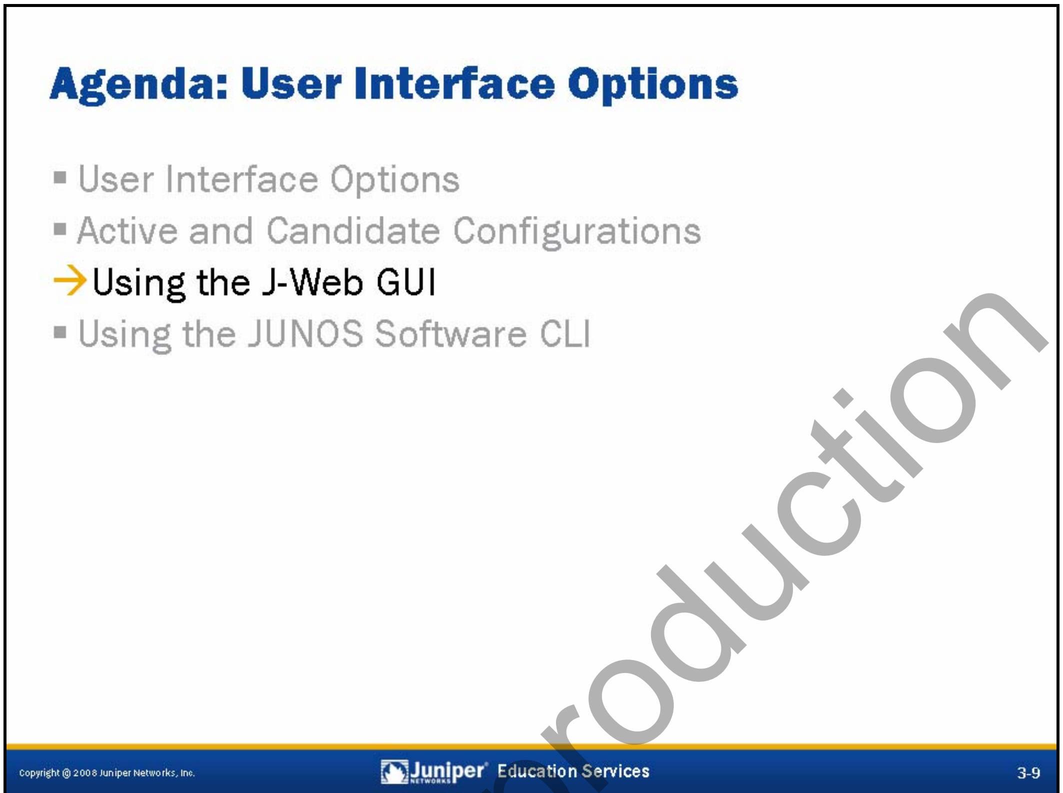
The **`configure`** command causes a *candidate* configuration to be created and populated with the contents of the *active* configuration. You can then modify the candidate configuration with your changes.

To have a candidate configuration take effect, you must commit the changes. At this time, JUNOS software checks the candidate configuration for proper syntax, and it installs it as the *active* configuration. If the syntax is not correct, an error message indicates the location of the error, and no part of the configuration is activated. You must correct the errors before recommitting the configuration.

Changes you make to the candidate configuration are visible immediately. By default, only one candidate configuration exists. If multiple users are editing the configuration at the same time, all users can see all changes. If more than one user is modifying the configuration, committing it saves and activates the changes of all the users.

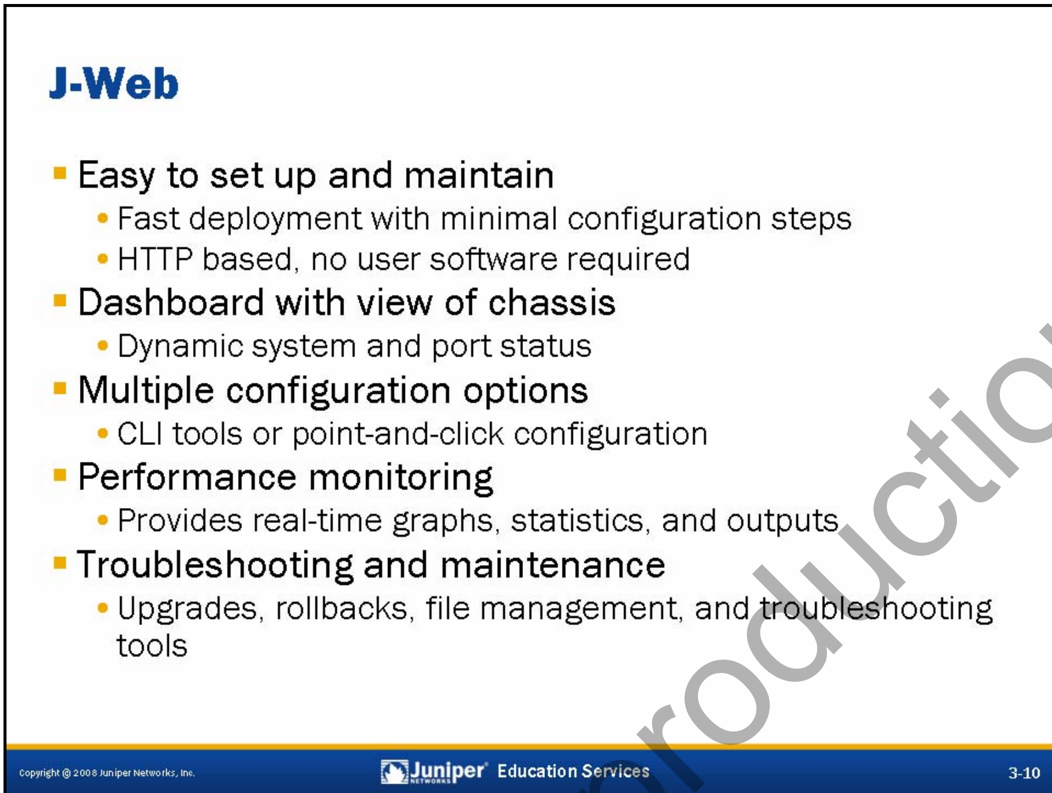
JUNOS software maintains a configuration history by storing previously active configurations. A maximum of 50 configurations are saved. This number includes the current *active* configuration, which is also known as `rollback 0`. You can easily recover previous configurations with a **`rollback`** ***n*** command.

Committing a configuration causes the old active configuration to become `rollback 1`. Each existing backup is renumbered and pushed further out, storing the oldest copy as number 49. The first three rollbacks (1–3) are stored in the `/config` directory, and the remainder are stored in the `/config/db/config` directory.



Using the J-Web Graphical User Interface

The slide highlights the topic we discuss next.



Ease of Use and Maintenance

The J-Web makes initial deployment a snap. With the factory-default configuration, simply connect your laptop or PC to the ge-0/0/0 port and initiate the *EZsetup* feature using the LCD menu. A wizard walks you through a series of steps to create your initial configuration. No client software is necessary other than a standard Web browser. After initial configuration, you can return to J-Web through any network port using the username and password you created.

Viewing Your Dashboard

When you log in to J-Web, you always start by viewing the J-Web `Dashboard`. The `Dashboard` provides a quick glance of system status, ports, alarms, and utilization information.

Configuration Options

The `Configuration` tab allows you to configure the switch in a point-and-click fashion or by a direct edit of the configuration in text format. Help is available by clicking the question mark (`?`) next to the various configuration options.

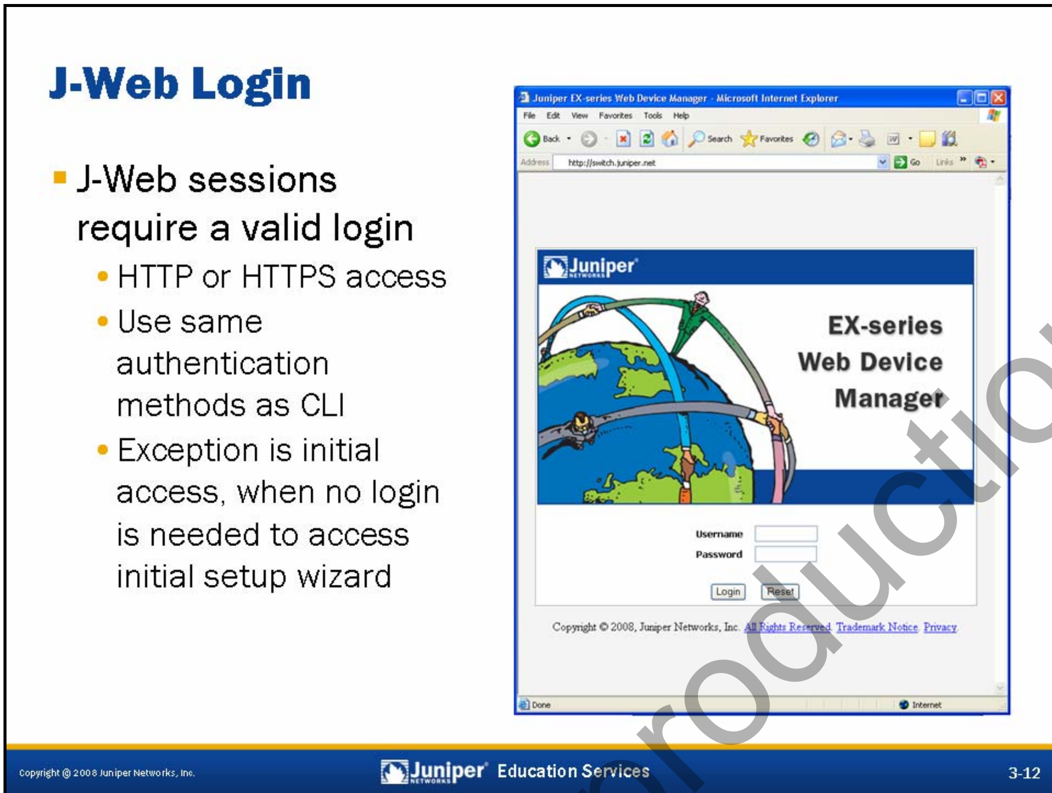
Continued on next page.

Monitoring Your Switch’s Performance

The `Monitor` tab provides real-time graphs and statistics. You can also view the results of configuration changes such as routing table entries or bridging information. You can find most of the **`show`** commands from the CLI here in a point-and-click fashion.

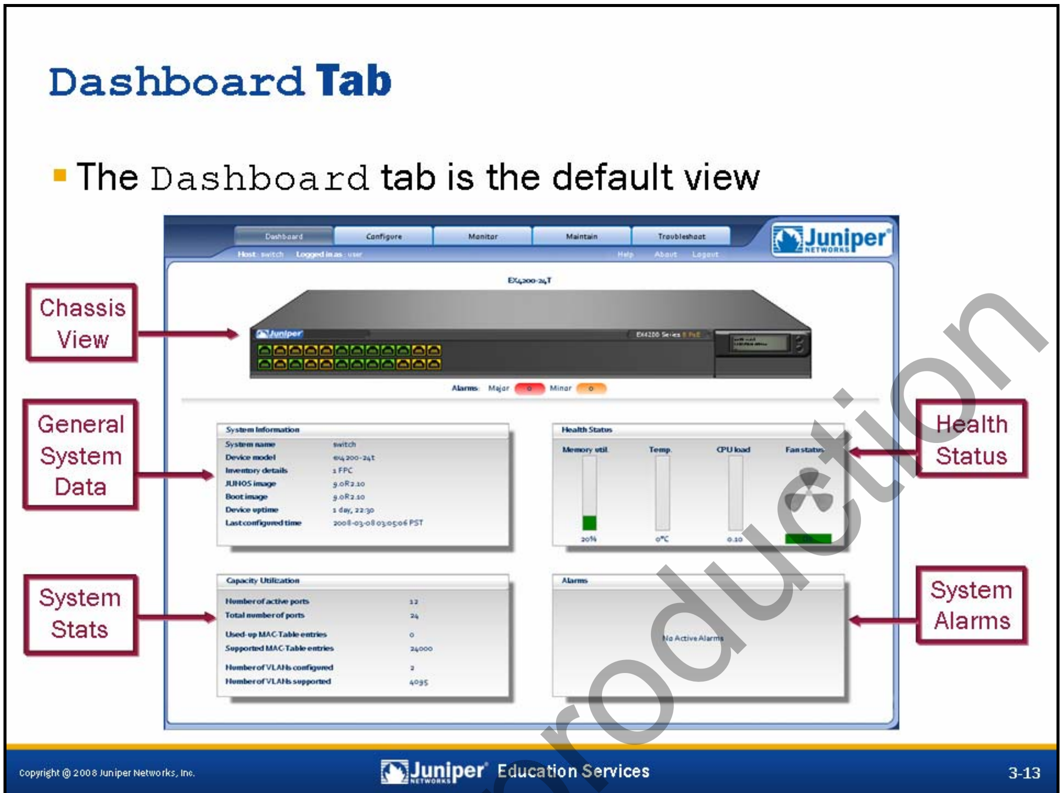
Troubleshooting and Software Management

The `Troubleshoot` tab provides common network tools such as ping and traceroute to quickly assess network issues. It is easy to perform software upgrades and file system maintenance using the `Maintain` tab.



Logging In to J-Web

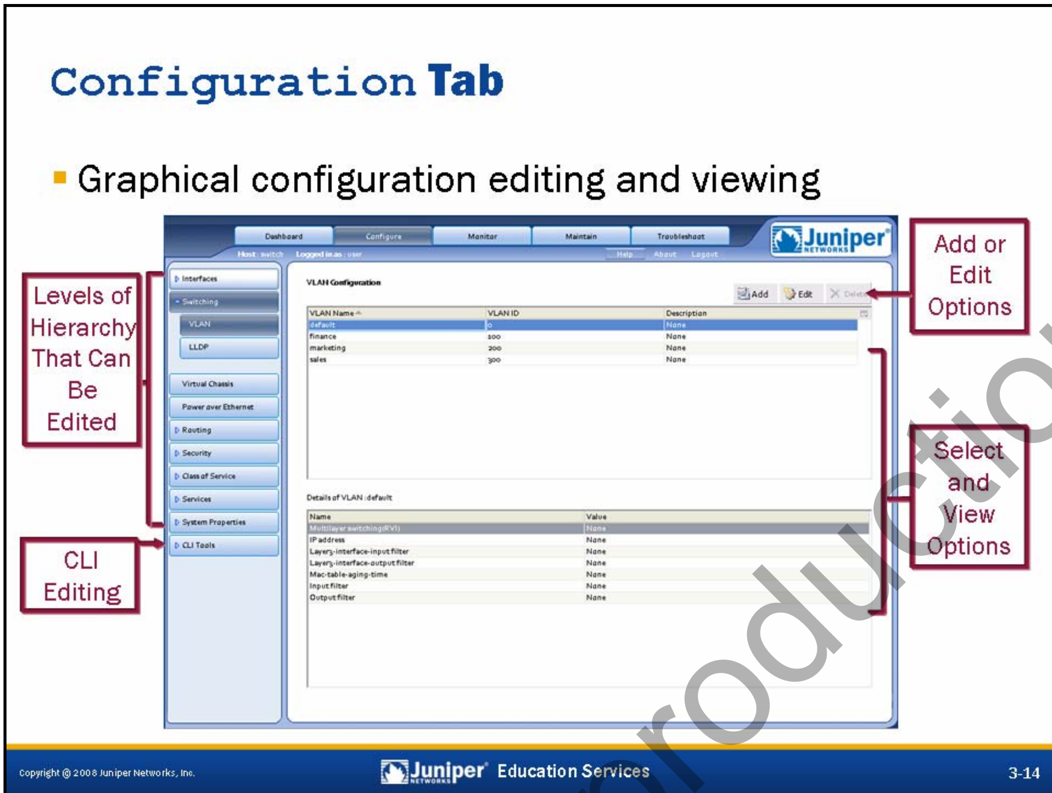
Once you configure the switch for access, you can log in by connecting to any network port using your Web browser. You must have enabled the HTTP or HTTPS service. If you configured the switch to use an external authentication mechanism such as a RADIUS server, J-Web will also use that mechanism for authentication. Otherwise, you use your configured username and password. The exception to this process is when you invoke the EZsetup wizard through the LCD menu while the switch has the factory-default configuration. In this scenario, the switch automatically invokes Dynamic Host Configuration Protocol (DHCP) services that enables your laptop or PC to initiate a Web browsing session while connected to the ge-0/0/0 or me0 interfaces.



It's Go Time!

When you get into a vehicle and start it up, the first items you check are your dashboard gauges. J-Web's default tab is the `Dashboard` tab. The `Dashboard` provides a quick view of the switch's current status. The graphical depiction of the chassis quickly illustrates port status, alarms, and the LCD screen. Hover your mouse pointer over components for more detailed information.

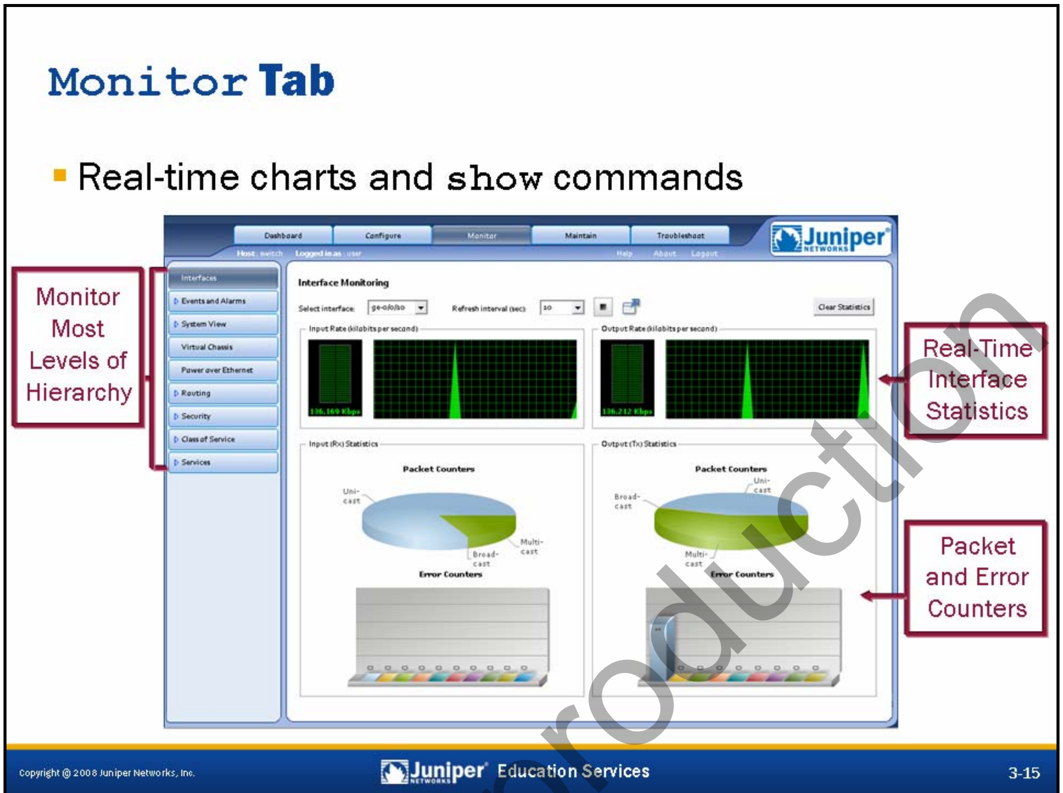
The `System Information` box shows information such as the configured name of the switch, the switch model, software version, and system uptime. The `Capacity Utilization` box shows the usage and maximum number of network ports, bridging table entries, and virtual LANs (VLAN). The `Health Status` box displays environmental conditions such as system temperature along with memory and CPU utilization. The `Alarms` box outputs red and yellow alarms that the switch is reporting.



Configuring an EX-series Switch with J-Web

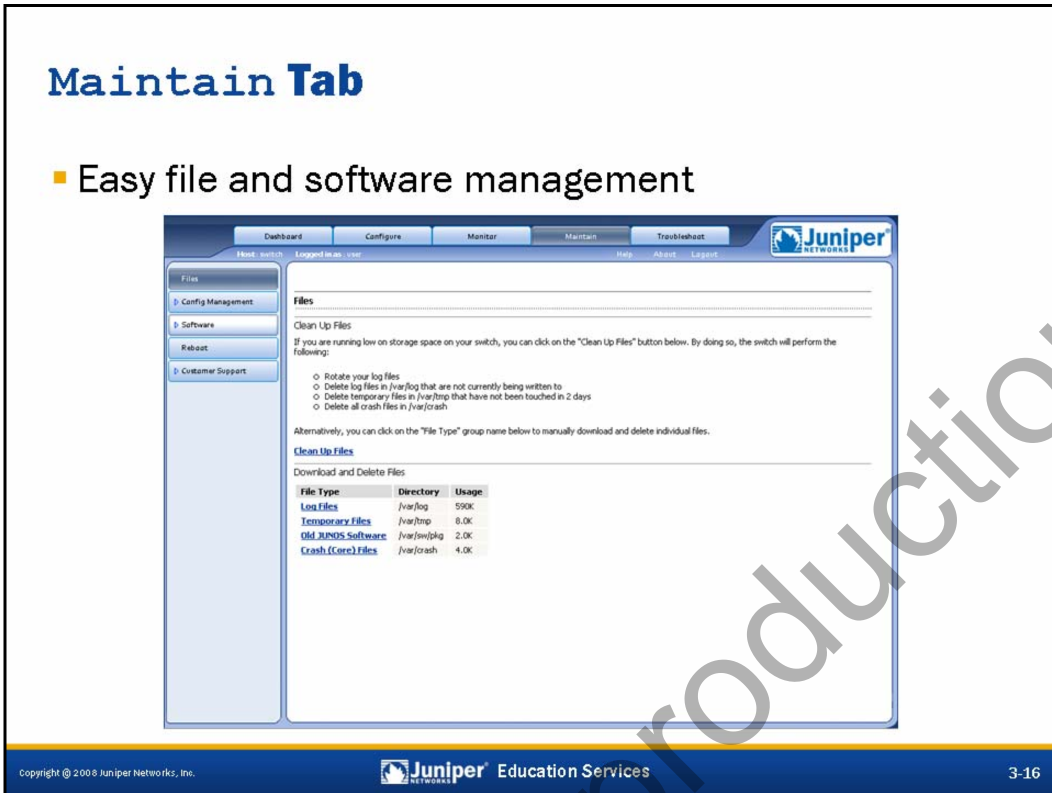
J-Web offers an easy-to-use interface for configuring your switch. Choose what configuration hierarchy you want to view or edit on the left side of the screen. Information about that hierarchy appears on the main portion of the screen. Here you can select various options for viewing or editing. You can add new configuration options with the `Add` button or edit existing configuration options with the `Edit` button. These buttons and a `Delete` button are located near the top right of the screen.

If you prefer to manipulate your configuration with a text-based approach, choose the `CLI Tools` option on the bottom left.



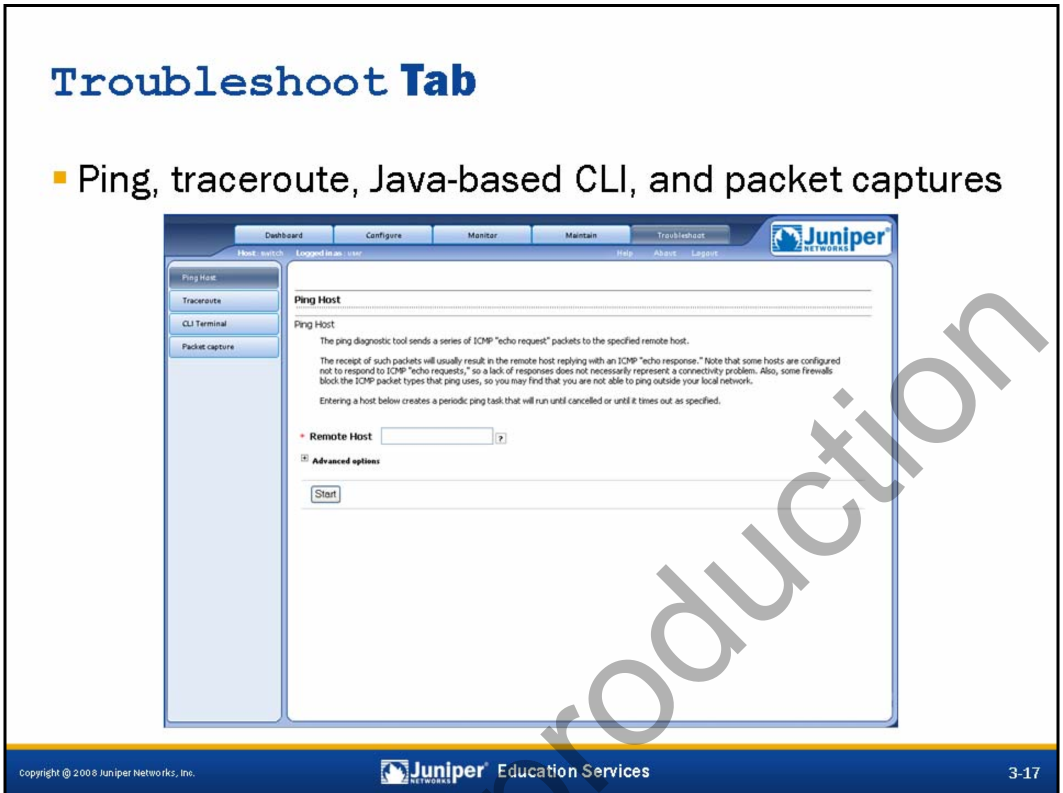
Performance Monitoring

On the `Monitor` tab, you can view detailed real-time statistics and the results of configuration-related activity. As seen on this slide, the `Interfaces` hierarchy provides statistics in a graphical fashion using colorful pie charts and graphs. Use the drop-down menus to customize your view. Again, hovering the mouse pointer over various parts of the screen presents you with more detailed information. Most of the hierarchies on the left side of the screen are carry-overs from the `Configure` tab. Selecting these options provides a point-and-click alternative over CLI **`show`** commands.



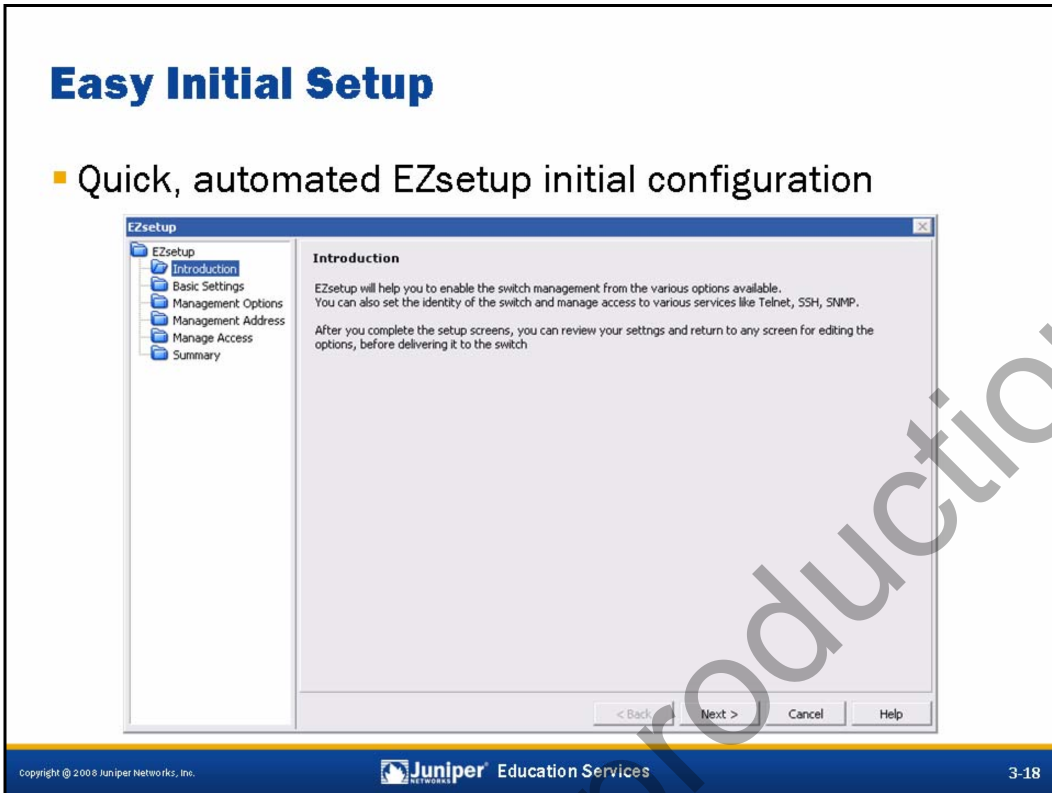
Maintenance of Your EX-series Switch

The `Maintain` tab provides an interface to manage file systems, JUNOS software, and configuration files. Under the `Files` section you can download and delete log files, memory dump files, and other temporary files so as to keep your compact-flash device from becoming too full. `Config Management` allows you to retrieve historical configuration files and to compare differences between configurations. Choosing `Software` provides methods for upgrading and downgrading the JUNOS software. You can automate the upgrade process by specifying a remote FTP server to retrieve the JUNOS software. The switch then upgrades with the retrieved software and issues a reboot of the system to complete the upgrade process. The `Reboot` section allows you to schedule reboots and provides other options for rebooting the switch. `Customer Support` provides a quick method to register your EX-series switch.



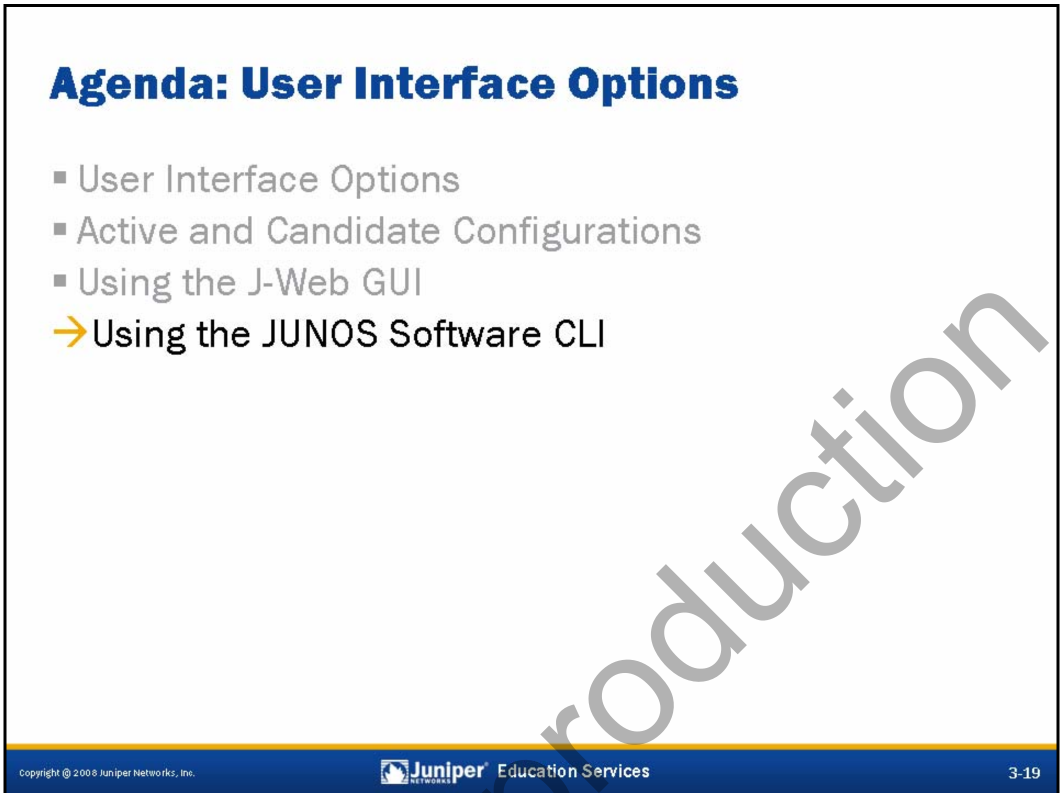
Tools for Troubleshooting

While you are making configuration changes or investigating the source of network problems with J-Web, it is helpful to have basic troubleshooting tools without having to open a separate terminal session to the switch. The `Troubleshoot` tab allows you to issue Internet Control Message Protocol (ICMP) echo requests with its ping tool. Expand the `Advanced Options` to change options for your ping packets such as size and time to live (TTL). You can also perform a traceroute, capture packet dumps, and even open an embedded Java-based terminal session to your switch.



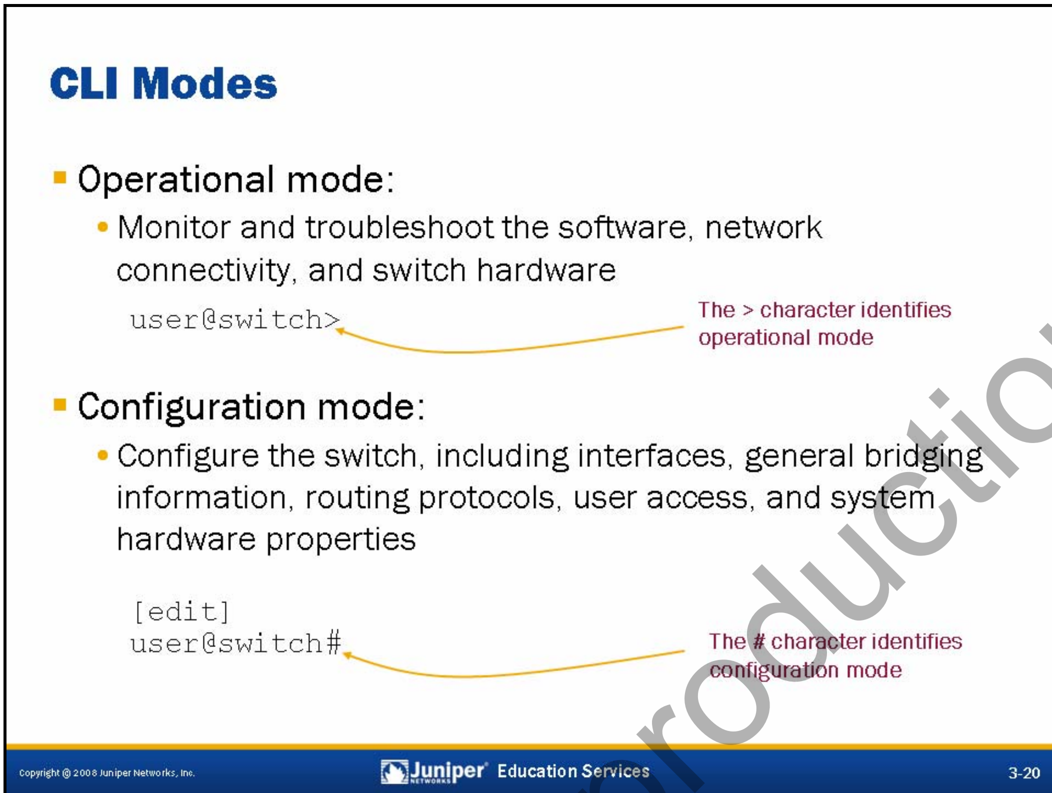
EZsetup Initial Configuration Wizard

JUNOS software for EX-series switches comes with an EZsetup initial configuration wizard to help get your switch operational quickly. EZsetup automates the initial configuration by presenting a series of basic initial configuration options such as the management IP address, the root password, system time settings, and management VLAN settings. The EZsetup wizard is available in both a J-Web GUI version and a CLI-based version.



Using the JUNOS Software Command-Line Interface

The slide highlights the topic we discuss next.

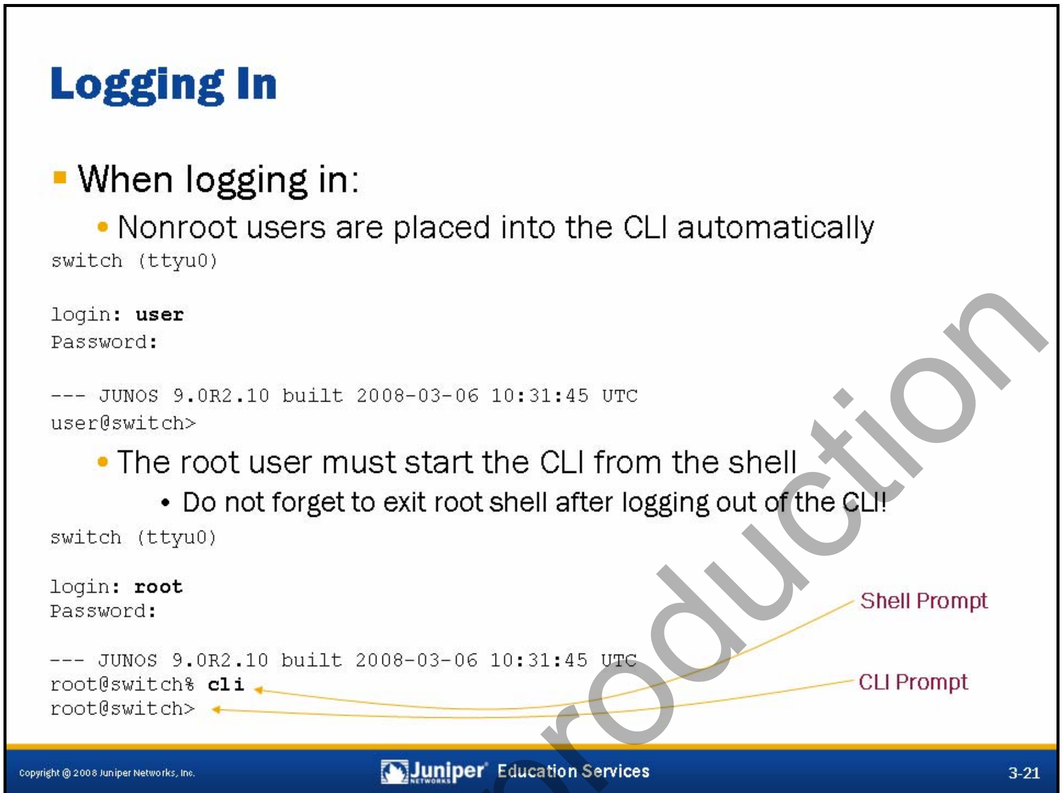


Operational Mode

In operational mode, you use the CLI to monitor and troubleshoot the switch. The **monitor**, **ping**, **show**, **test**, and **traceroute** commands let you display information and statistics about the software running on the switch, such as routing table entries, and these commands let you test network connectivity.

Configuration Mode

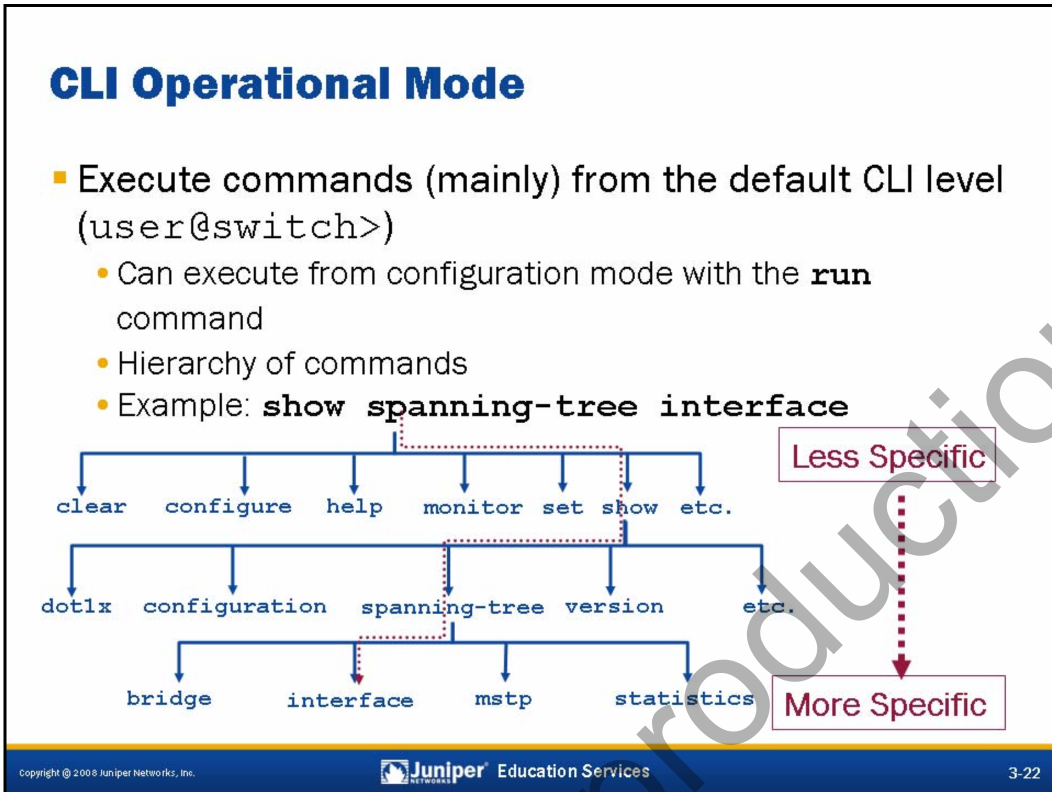
You configure JUNOS software by entering configuration mode and creating a hierarchy of configuration statements. From within configuration mode, you can configure all properties of JUNOS software, including interfaces, general bridging information, routing protocols, and user access, as well as several system hardware properties.



Logging In

JUNOS software requires a username and password for access. The switch administrator creates user accounts and assigns permissions. All platforms running JUNOS software have only the root user configured by default, without any password.

When you log in as the root user, you are placed at the UNIX shell. You must start the CLI by typing the **`cli`** command. When you exit the CLI, you return to the UNIX shell. For security reasons, make sure you also log out of the shell by using the **`exit`** command.



Operational Mode

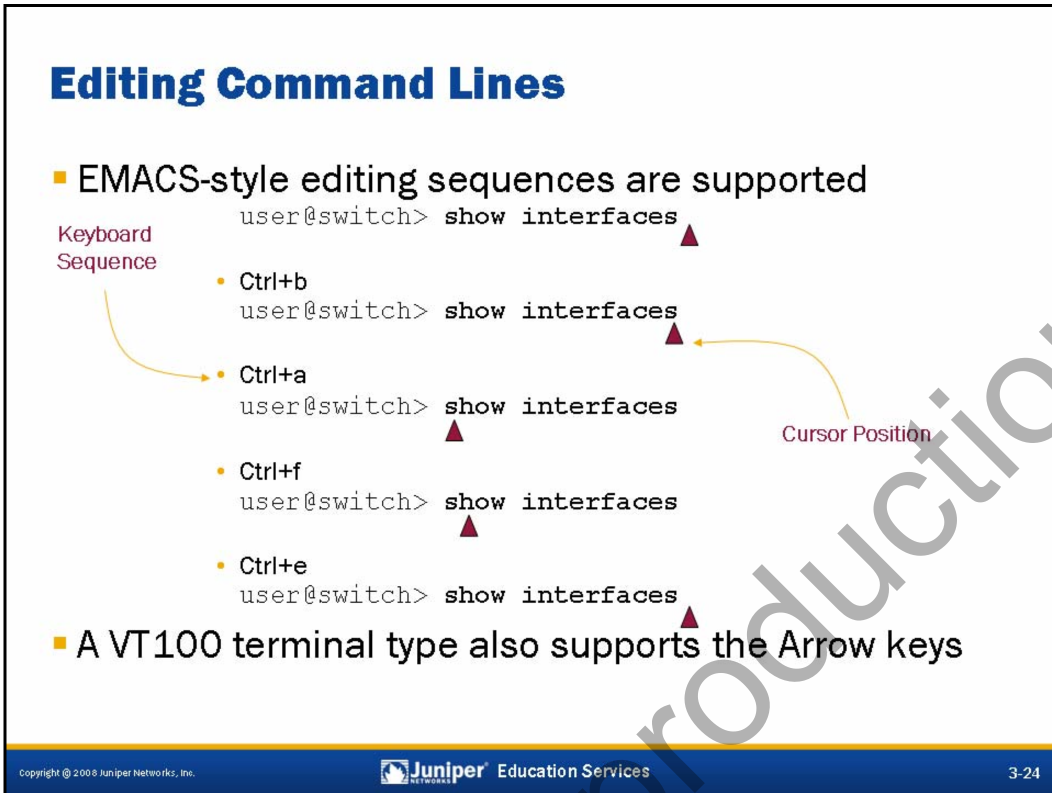
You use operational-mode CLI commands to monitor and control the operation of the switch. The operational-mode commands are hierarchically structured, as shown on the slide. For example, the **show** command displays various types of information about the system and its environment. One of the possible options for the **show** command is **spanning-tree**, which displays information about the Spanning Tree Protocol (STP). Specifying the **interface** option, as in the **show spanning-tree interface** command, outputs information on STP interfaces.

Continued on next page.

Operational Mode (contd.)

Key operational-mode capabilities include the following:

- Entering configuration mode;
- Controlling the CLI environment;
- Exiting the CLI;
- Monitoring and troubleshooting:
  - **`clear`**
  - **`monitor`**
  - **`mtrace`**
  - **`ping`**
  - **`show`**
  - **`test`**
  - **`traceroute`**
- Connecting to other network systems;
- Copying files;
- Restarting software processes; and
- Performing system-level operations.



EMACS-Style Control Keys

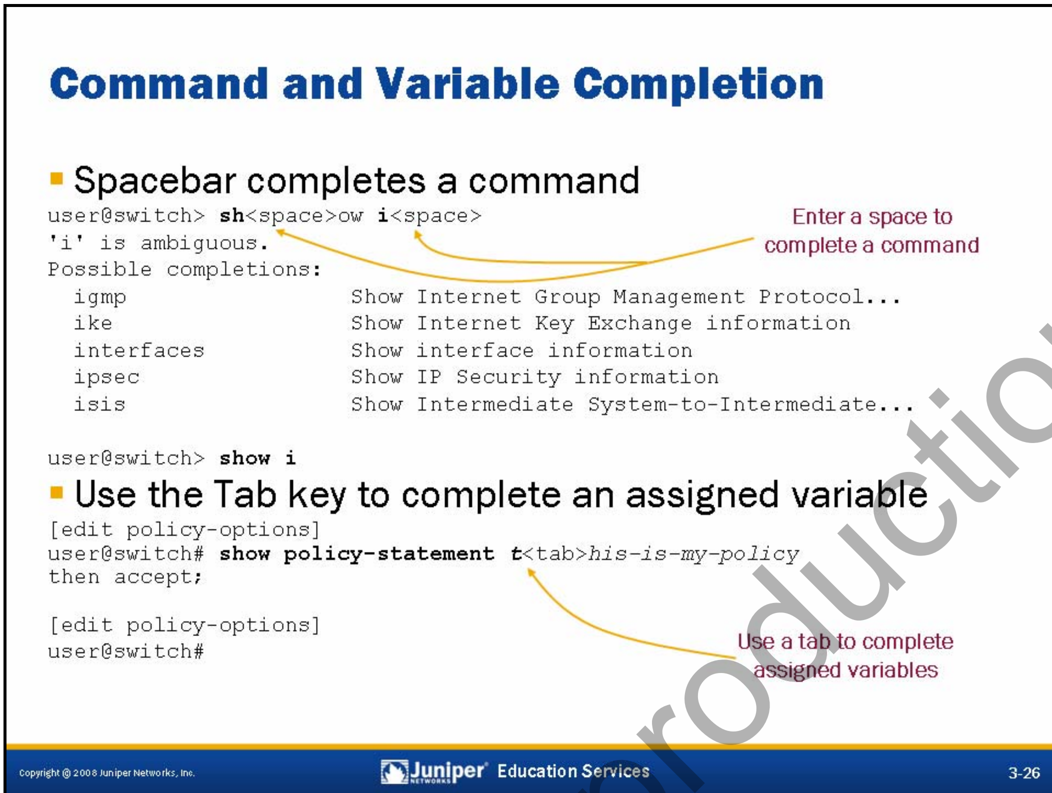
The CLI supports EMACS-style keyboard sequences that allow you to move around on a command line and delete specific characters or words. The following sequences are supported:

- Ctrl+b: Moves the cursor left one character;
- Ctrl+a: Moves the cursor to the beginning of the command line;
- Ctrl+f: Moves the cursor right one character;
- Ctrl+e: Moves the cursor to the end of the command line;
- Delete or Backspace: Deletes the character before the cursor;
- Ctrl+d: Deletes the character over the cursor;
- Ctrl+k: Deletes from the cursor to the end of the line;
- Ctrl+u: Deletes all characters and negates the current command;
- Ctrl+w: Deletes the entire word to the left of the cursor;
- Ctrl+l: Redraws the current line; and
- Ctrl+p or Ctrl+n: Repeats the previous and next command in the command history.

Continued on next page.

VT100 Terminal Type

JUNOS software defaults to a VT100 terminal type. This terminal type enables the use of keyboard Arrow keys without any additional session or configuration modification.



Spacebar Completion for Commands

The CLI provides a completion function. Therefore, you do not always have to type the full command or the command option name for the CLI to recognize it.

To complete a command or option that you have partially typed, press the Spacebar. If the partially typed letters begin a string that uniquely identifies a command, the CLI displays the complete command name. Otherwise, the CLI beeps to indicate that you have entered an ambiguous command, and it displays the possible completions.

The command completion option is on by default, but you can turn it off.

Tab Completion for Variables and Commands

You can also use the Tab key to complete variables. Examples of variables include policy names, AS paths, community names, and IP addresses. The Tab key also offers a list of possible completions, should multiple, ambiguous options exist.

Context-Sensitive Help

- Type ? anywhere on the command line

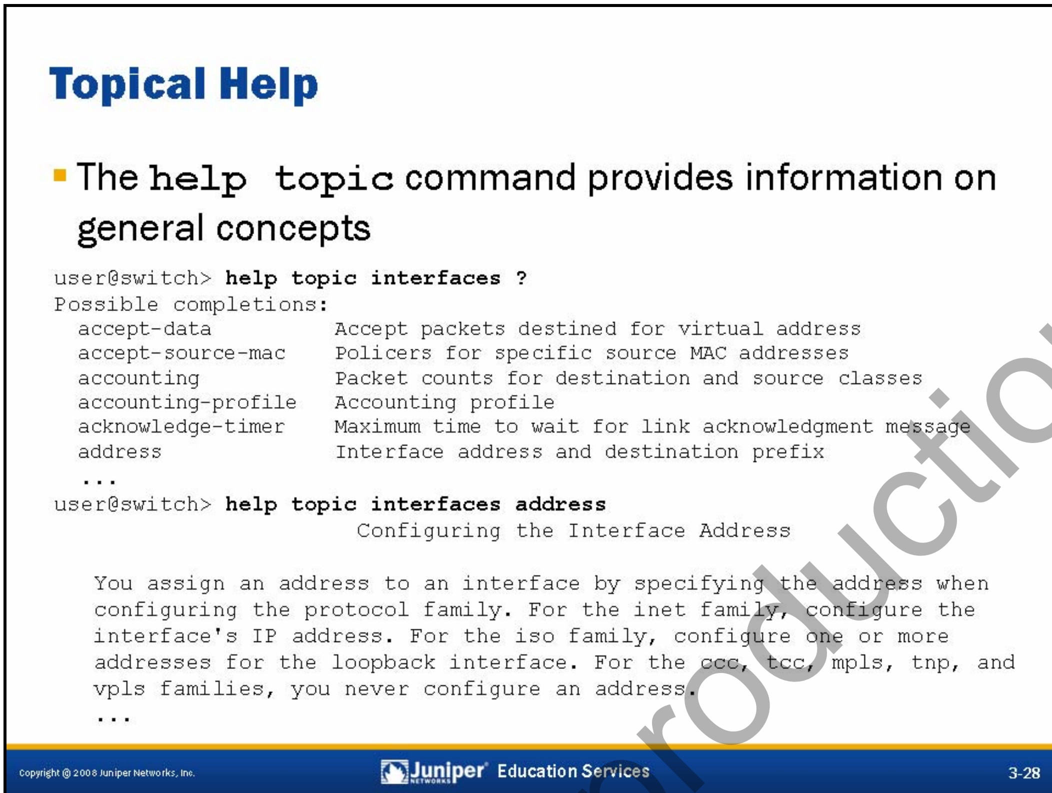
```
user@switch> ?
Possible completions:
  clear                  Clear information in the system
  configure              Manipulate software configuration information
  file                   Perform file operations
  help                   Provide help information
  . . .
user@switch> clear ?
Possible completions:
  arp                    Clear address resolution information
  bfd                    Clear Bidirectional Forwarding Detection
                         information
  bgp                    Clear Border Gateway Protocol information
  dhcp                   Clear DHCP information
  . . .
```

 Juniper Education Services 3-27

Need Help?

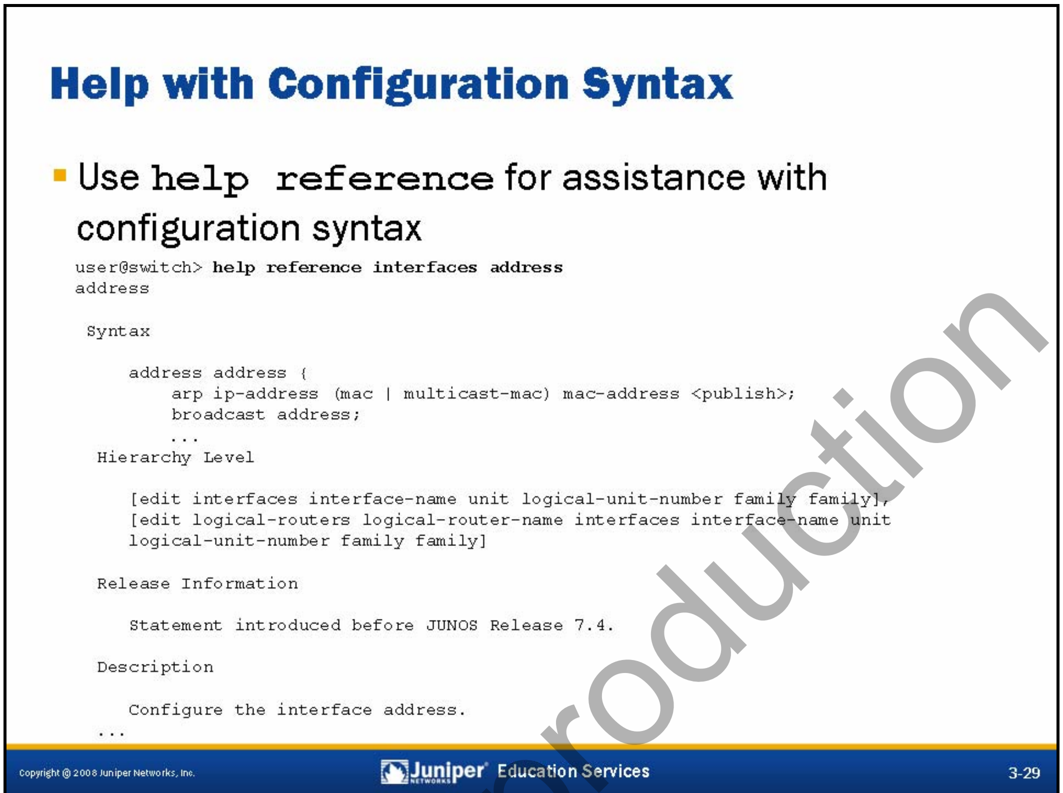
The CLI provides context-sensitive help at any point in a command line. Help tells you which options are acceptable at the current point in the command and provides a brief description of each command or command option.

To receive help at any time while in the JUNOS CLI, type a question mark (**?**). You do not need to press Enter. If you type the question mark at the command-line prompt, the CLI lists the available commands and options. If you type the question mark after entering the complete name of a command or an option, the CLI lists the available commands and options and then redisplays the command name and options that you typed. If you type the question mark in the middle of a command name, the CLI lists possible command completions that match the letters you have entered so far, then redisplays the letters that you typed.



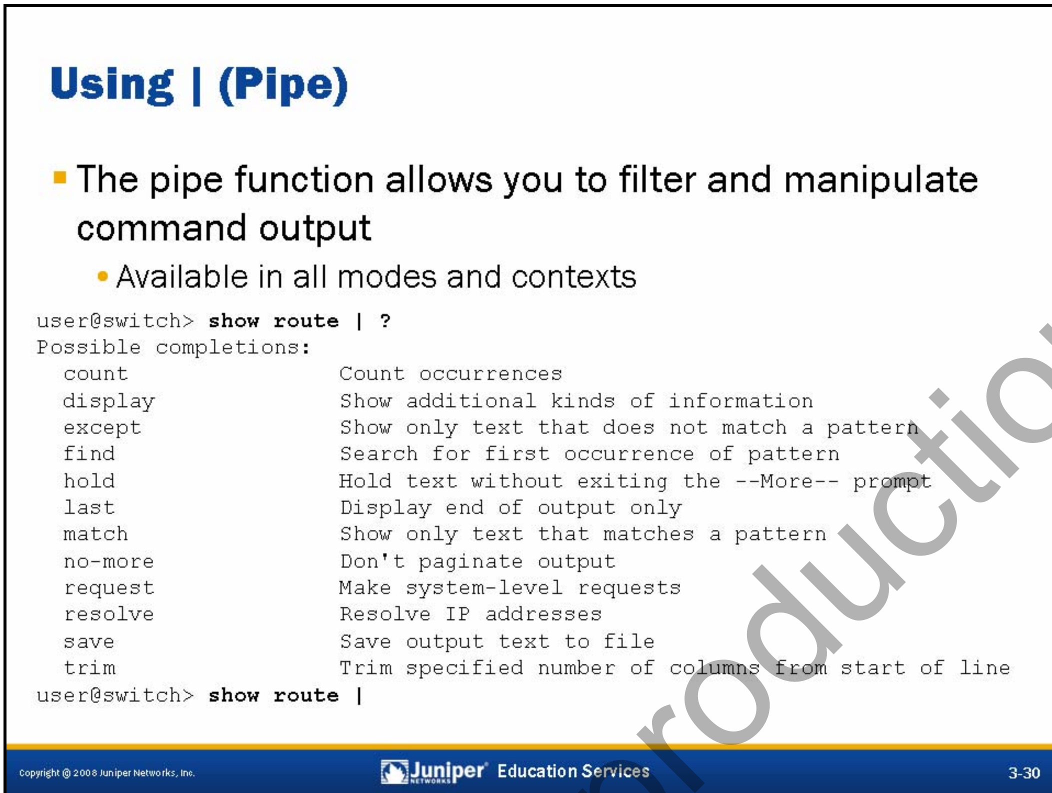
Help on General Concepts

There are various ways to use the **help** command. The **help topic** command displays usage guidelines for the statement. In the example on the slide, we are receiving information on configuring an interface address.



Help with JUNOS Software Configuration

The **`help reference`** command displays summary information for the statement. In other words, it contains JUNOS software-specific, configuration-related information. In the example on the slide, once again, we are using the **`help`** command for information on interface addressing. Notice the difference between the **`help reference`** command shown here and the **`help topic`** command from the previous slide.



Using Pipe

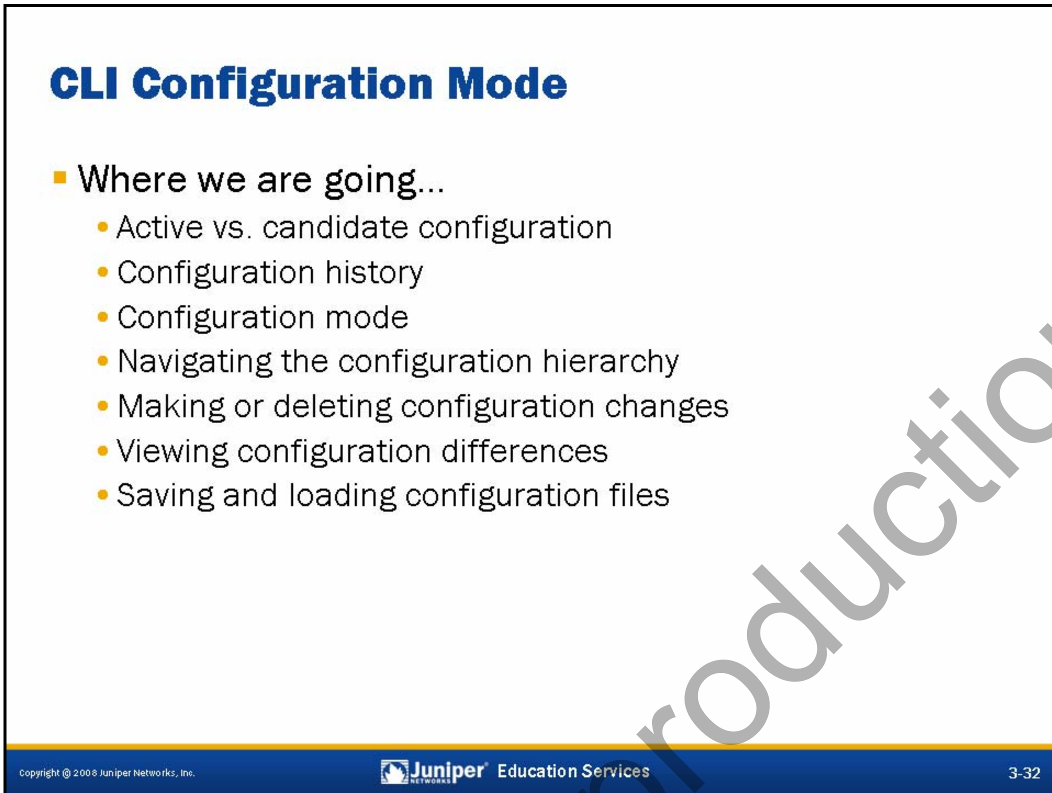
For operational and configuration commands that display output, such as the **show** commands, you can filter the output. When help is displayed for these commands, one of the options listed is |, called a pipe, which allows the command output to be filtered. To filter the output of an operational-mode or a configuration-mode command, add a pipe and an option to the end of the command. The options are the following:

- **compare** (***filename*** | **rollback** ***n***): Available in configuration mode using only the **show** command. Compares configuration changes with another configuration file.
- **count**: Displays the number of lines in the output.
- **display changed**: Available in configuration mode only. Tags changes with `junos:changed` attribute for XML use only.
- **display commit-scripts**: Shows data after JUNOS software applies commit scripts.
- **display detail**: Available in configuration mode only. Displays additional information about the contents of the configuration.

Continued on next page.

Using Pipe (contd.)

- **display inheritance**: Available in configuration mode only. Displays inherited configuration data and source group.
- **display omit**: Available in configuration mode only. Omits configuration statements with the **omit** option.
- **display set**: Available in configuration mode only. Shows **set** commands that created configuration statements.
- **display xml**: Displays the output in JUNOScript XML format.
- **except *regular-expression***: Ignores text matching a regular expression when searching the output. If the regular expression contains spaces, operators, or wildcard characters, you must enclose it in quotation marks.
- **find *regular-expression***: Displays the output starting at the first occurrence of text matching a regular expression. If the regular expression contains spaces, operators, or wildcard characters, you must enclose it in quotation marks.
- **hold**: Holds text without exiting the `--(more)--` prompt.
- **last**: Displays the last screen of information.
- **match *regular-expression***: Searches for text matching a regular expression. If the regular expression contains spaces, operators, or wildcard characters, you must enclose it in quotation marks.
- **no-more**: Displays output all at once rather than one screen at a time.
- **request message**: Displays output to multiple users.
- **resolve**: Converts IP addresses to Domain Name System (DNS) names. Truncates to fit original size unless you specify **full-names**.
- **save *filename***: Saves the output to a file or URL.
- **trim**: Trims specified number of columns from the start line.



CLI Configuration Mode

The slide lists the topics we examine on the following pages.

Review: Active Versus Candidate Configuration

- Batch configuration model:
  - Must commit configuration changes
- Active configuration:
  - Current operational configuration
  - Boot-up configuration
- Candidate configuration:
  - A working copy for configuration changes
  - Initialized with the active configuration
  - Becomes active configuration upon commit

Juniper Networks Education Services
3-33

Batch Configuration Changes

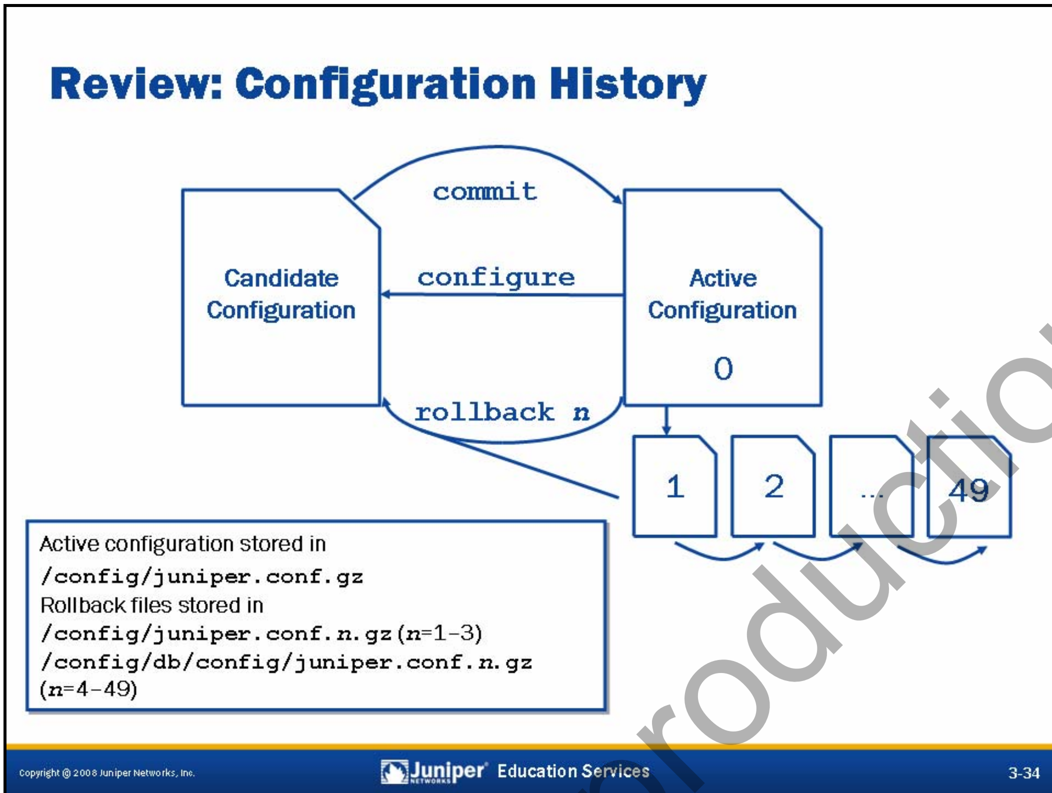
Unlike some switch software, configuration changes to the JUNOS software do not take affect immediately. This design feature allows you to group together and apply multiple configuration changes to the running configuration as a single unit.

Active Configuration

The active configuration is the configuration currently operational on the switch. It is also the configuration the switch loads during the boot sequence. This concept is analogous to both the *running configuration* and *startup configuration* in other switch software.

Candidate Configuration

The candidate configuration is a temporary configuration that might possibly become the active configuration. When you configure the switch, JUNOS software creates a candidate configuration and initially populates it with the switch's active configuration. You then modify the candidate configuration. Once satisfied with your modifications, you can apply or commit the changes. This action causes the candidate configuration to become the active configuration.



Configuration Files and Configuration History

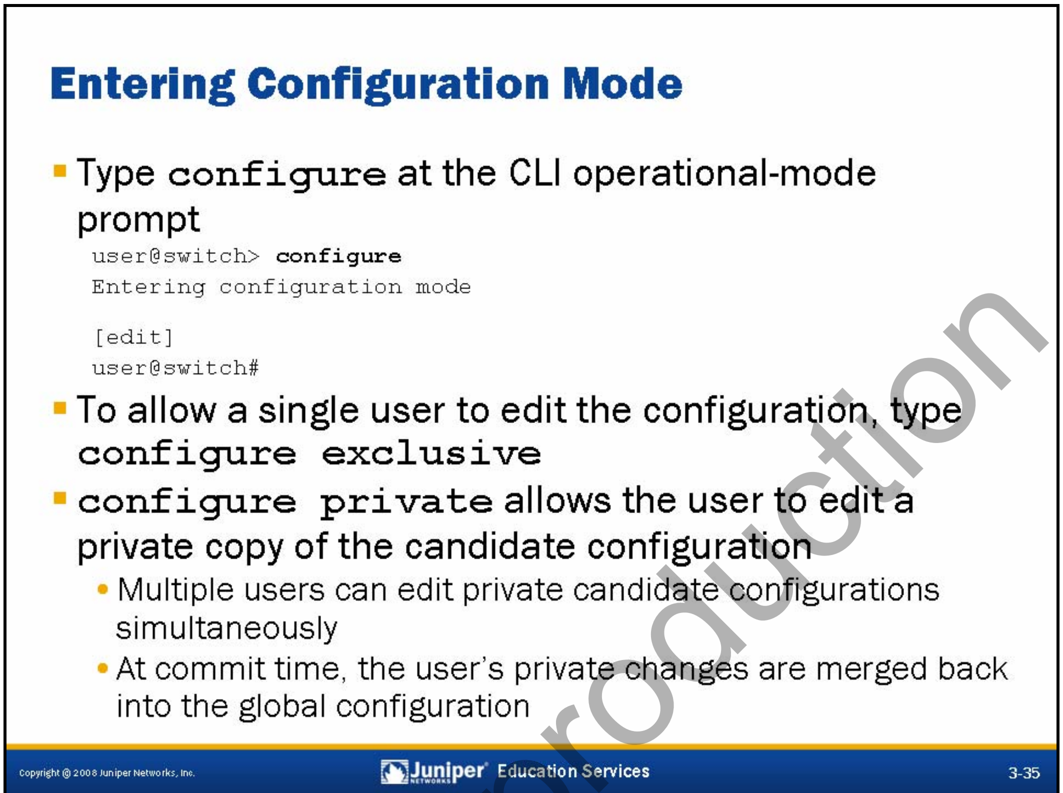
The **`configure`** command causes a *candidate* configuration to be created and populated with the contents of the *active* configuration. You can then modify the candidate configuration with your changes.

To have a candidate configuration take effect, you must commit the changes. At this time, JUNOS software checks the candidate configuration for proper syntax and it installs it as the *active* configuration. If the syntax is not correct, an error message indicates the location of the error, and no part of the configuration is activated. You must correct the errors before recommitting the configuration.

Changes you make to the candidate configuration are visible immediately. By default, only one candidate configuration exists. If multiple users are editing the configuration at the same time, all users can see all changes. If more than one user is modifying the configuration, committing it saves and activates the changes of all the users.

JUNOS software maintains a configuration history by storing previously active configurations. A maximum of 50 configurations are saved. This number includes the current *active* configuration, which is also known as `rollback 0`. You can easily recover previous configurations with a **`rollback n`** command.

Committing a configuration causes the old active configuration to become `rollback 1`. Each existing backup is renumbered and pushed further out, storing the oldest copy as number 49. The first three rollbacks (1–3) are stored in the `/config` directory, and the remainder are stored in the `/config/db/config` directory.



Starting Configuration Mode

You enter configuration mode by issuing the **`configure`** command from the CLI's operational mode. If, when you enter configuration mode, another user is also in configuration mode, a message indicates who the user is and what portion of the configuration the user is viewing or editing.

In configuration mode, the prompt changes from the angle bracket (>) of operational mode to the pound sign (#), preceded by the name of the user and the name of the switch.

The portion of the prompt in brackets, such as `[edit]`, is a banner indicating that you are in configuration mode and specifying your location within the statement hierarchy.

Exclusive Configuration

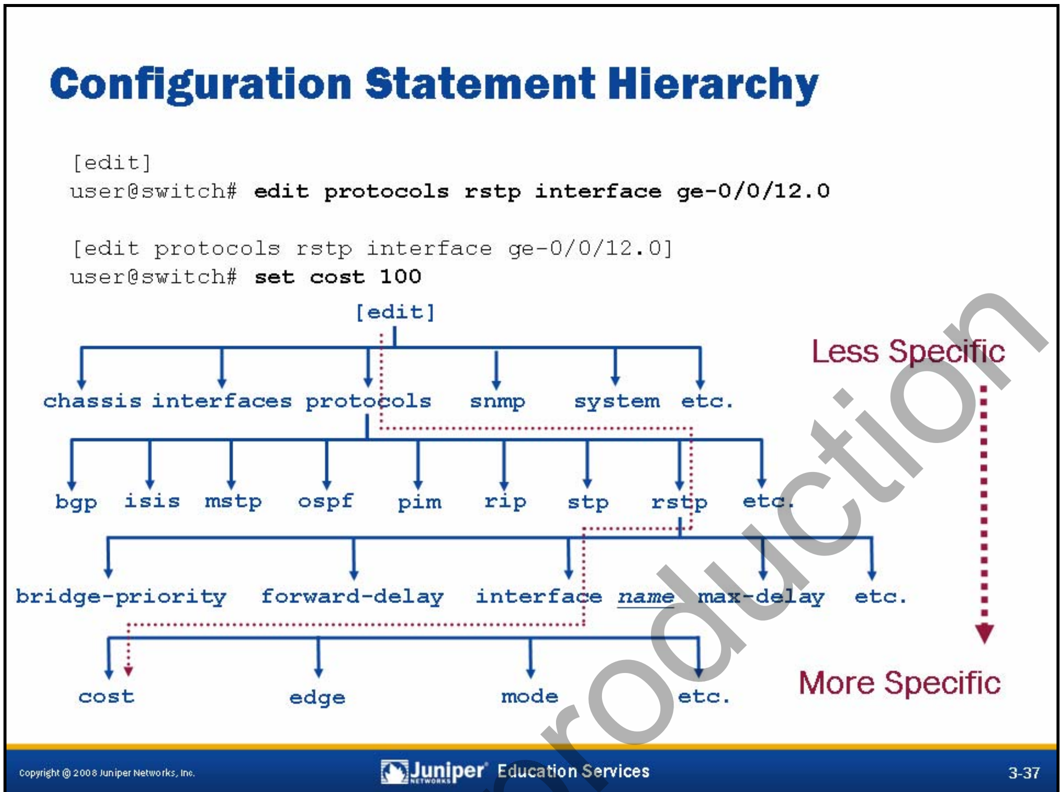
By default, multiple users can enter configuration mode and commit changes. To allow only a single user to edit the configuration, use the **`configure exclusive`** command. Exiting exclusive configuration without committing changes results in the loss of any modifications made to the candidate configuration.

Continued on next page.

Private Configuration

Entering configuration mode using the **`configure private`** command allows multiple users to edit the configuration while committing their private changes only (you must issue a **`commit`** command from the `[edit]` hierarchy). If private users issue a **`rollback 0`** command, only their changes are discarded. If two users are in private mode and both make the same change (`user_1` changes the system hostname to *`foo`* while `user_2` sets the name to *`bar`*), the second **`commit`** will fail with an error message to avoid configuration conflicts. The second user's changes are placed into effect if a second **`commit`** is issued by the second user, however.

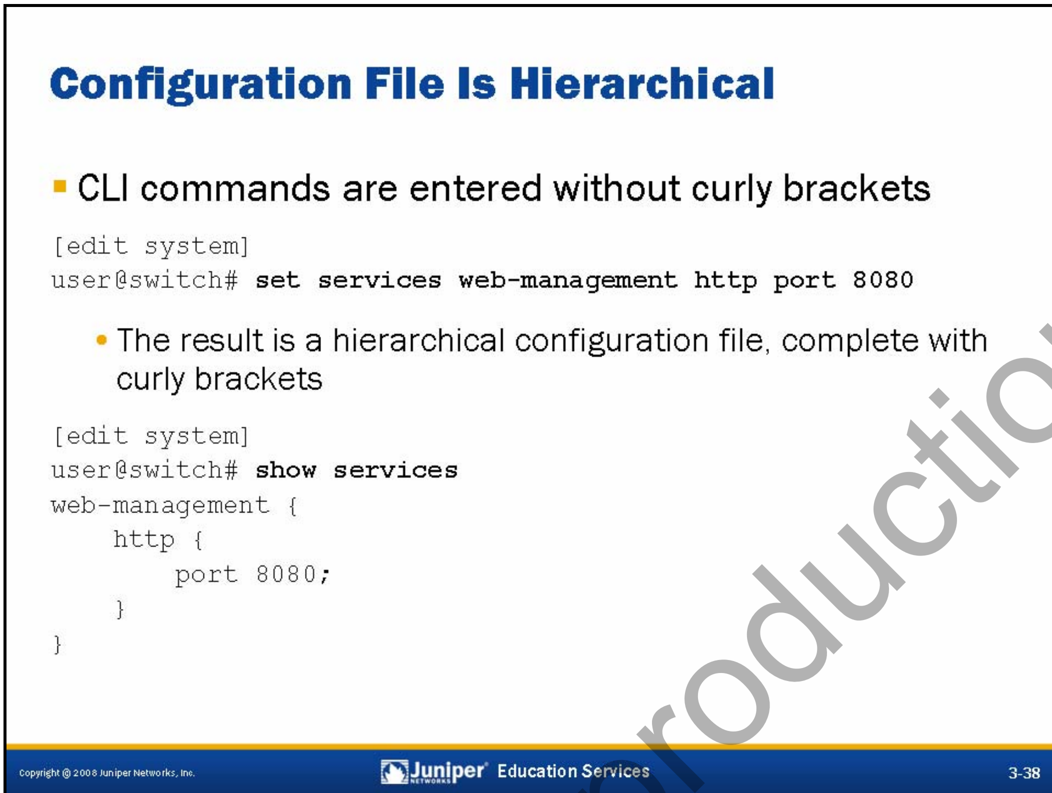
When a user is in private mode, other users must enter private mode or use **`configure exclusive`** to become the master, or they cannot modify the candidate configuration. Exiting private configuration without committing changes results in the loss of any modifications made to the private candidate configuration.



Statement Hierarchy

In configuration mode, you enter commands that affect the statement hierarchy. The statement hierarchy stores configuration information and is independent of the CLI operational-mode command hierarchy. The commands available in configuration mode are also independent of the commands available in operational mode. For example, CLI operational mode includes a **show** command to display specific operational information, while the CLI configuration mode provides a **show** command to display the statement hierarchy. The two commands are independent of each other.

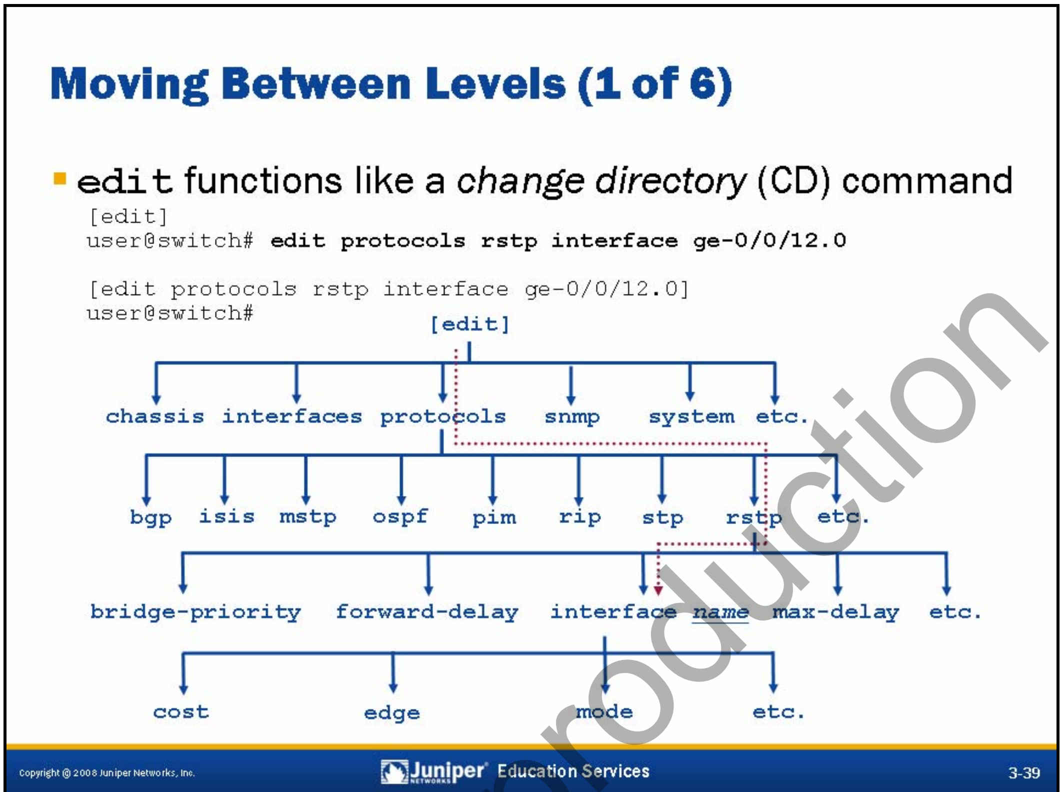
The statement hierarchy is organized in a tree structure similar to Windows folders or UNIX directories, grouping related information into a particular branch of the tree.



Hierarchical Configuration

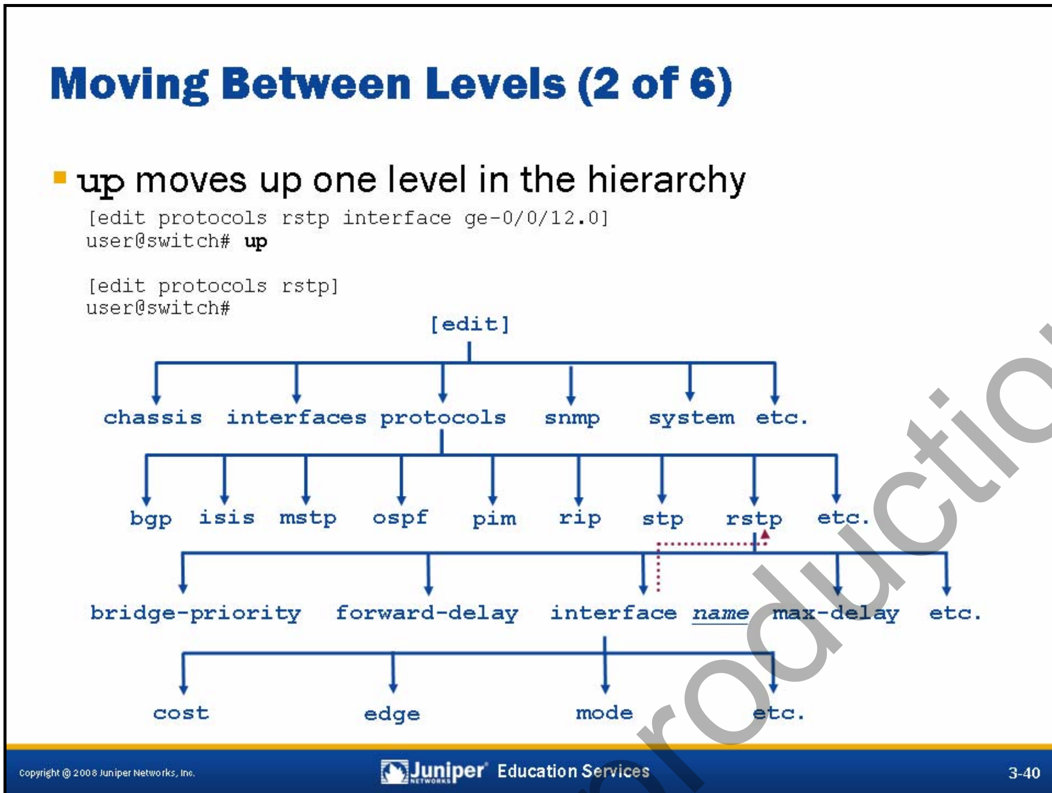
Use the **set** command in the CLI configuration mode to modify the candidate configuration. Use the **show** command to display the candidate configuration. Both commands are relative to the current configuration hierarchy, shown by the `[edit]` prompt.

Configuration files use curly brackets (`{ }`) and indentation to visually display the hierarchical structure of the configuration. Terminating—or leaf—statements in the configuration hierarchy are displayed with a trailing semicolon (`;`). You enter neither the curly brackets nor the semicolons as part of the **set** command.



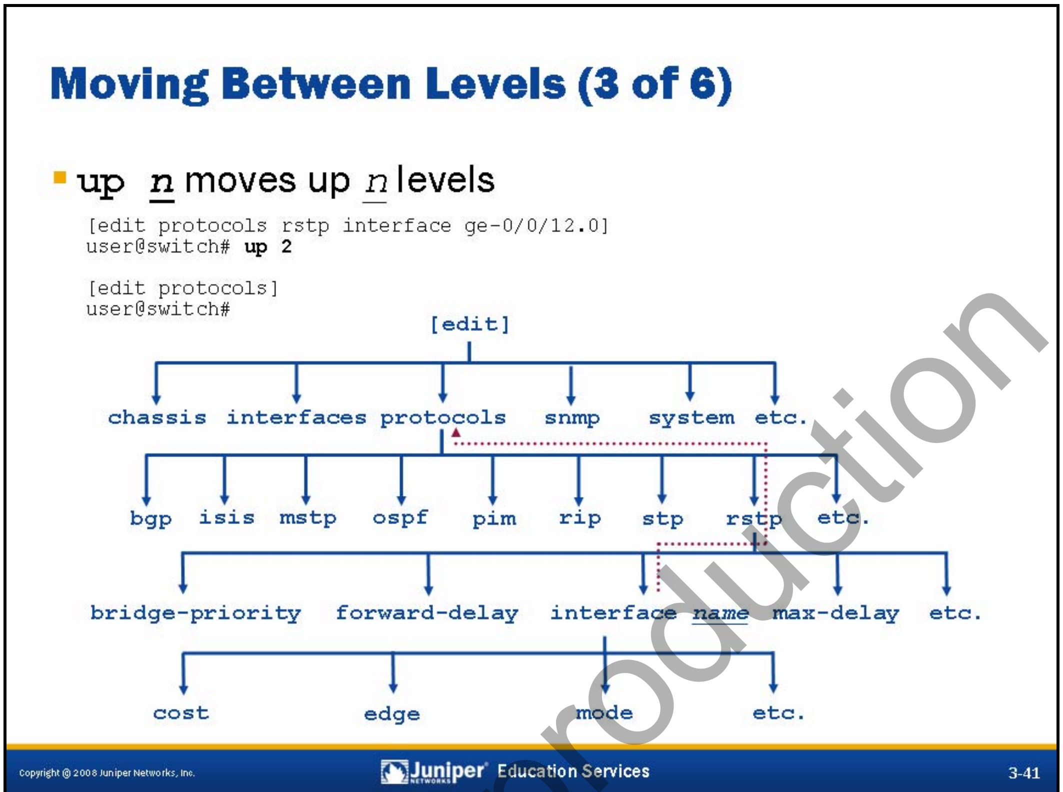
Moving Between Levels Is Like Changing Directories

To move down through an existing configuration statement hierarchy or to create a hierarchy and move down to that level, use the **edit** command, specifying your desired hierarchy level. After you issue an **edit** command, the configuration mode banner changes to indicate your current level in the hierarchy.



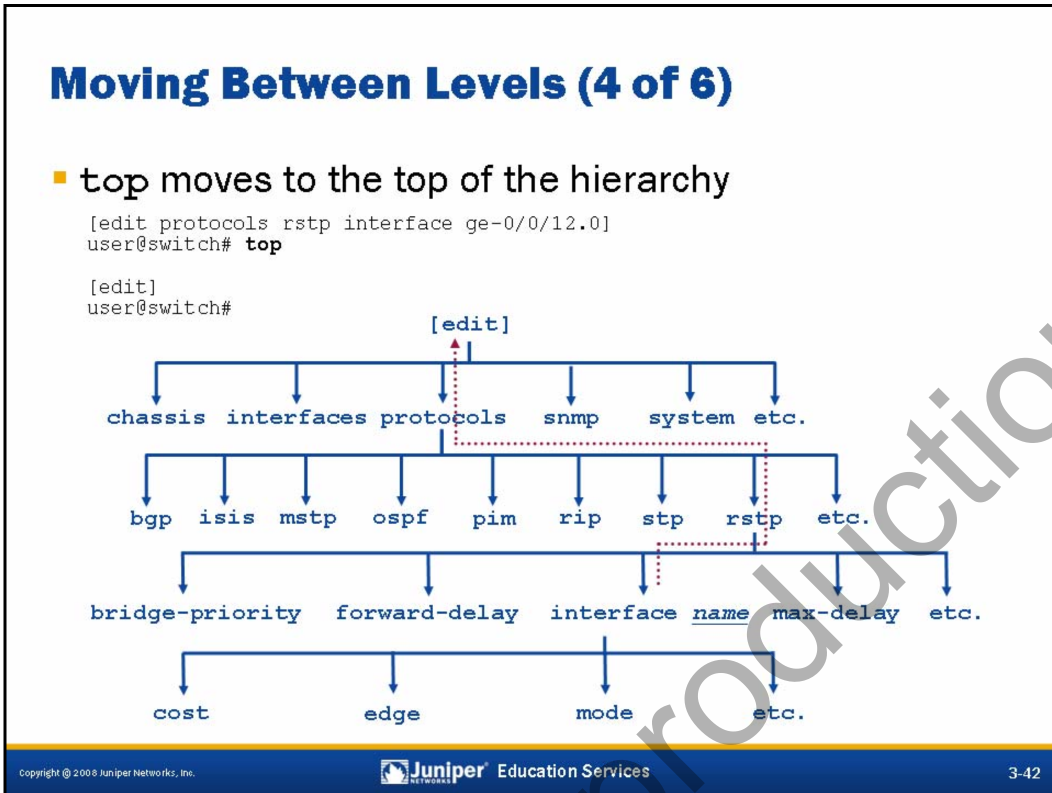
Moving Up One Level

To move up one level from the current position in the hierarchy, use the **up** command.



Moving Up More Than One Level

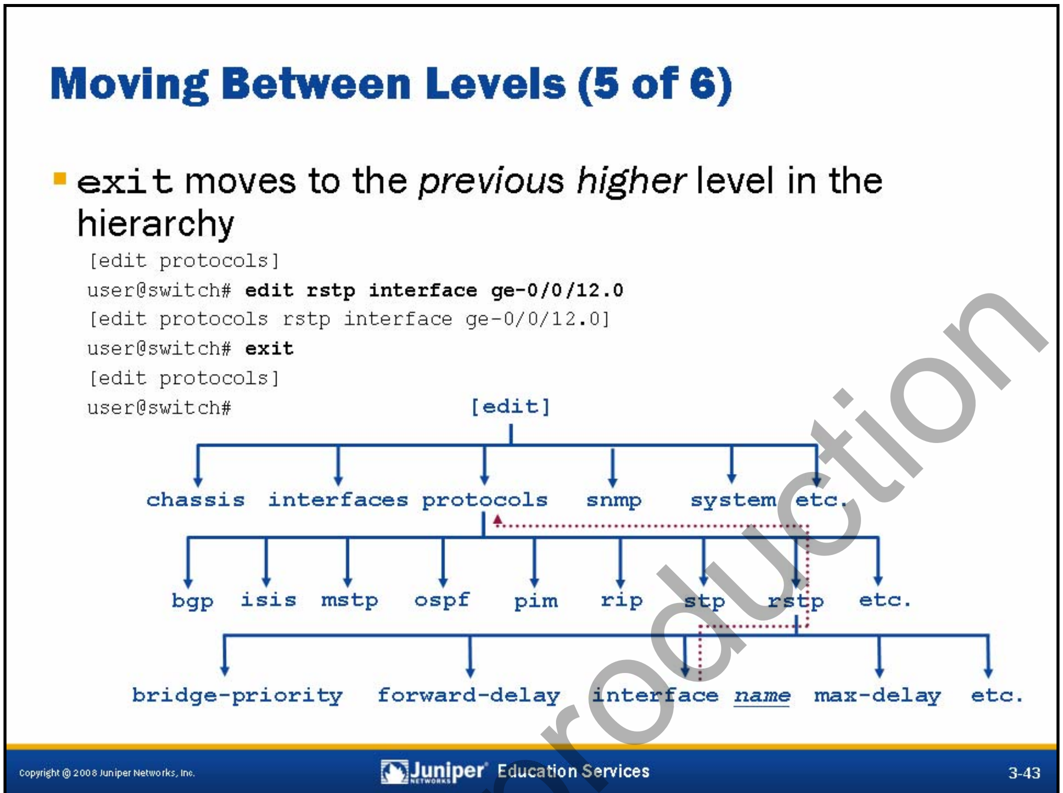
To move up more than one level from the current position in the hierarchy, supply an optional count to the **up** command. You will be moved up the number of levels specified or to the top of the hierarchy if there are fewer levels than specified.



Take Me to the Top

The **top** command quickly moves you to the top of the configuration hierarchy. You can combine **top** with **edit** to quickly move to a different hierarchy or with **show** to display a different hierarchy:

```
[edit protocols rstp interface ge-0/0/12.0]
user@switch# top edit system login

[edit system login]
user@switch#


[edit protocols rstp interface ge-0/0/12.0]
user@switch# top show system services
web-management {
    http {
        port 8080;
    }
}
```



Back to Where I Was Before

As the example on the slide illustrates, the **exit** command returns the user to the most recent higher level of the hierarchy. Entering **exit** at the top level of the hierarchy exits configuration mode. You can exit configuration mode from any level of the hierarchy by supplying the **configuration-mode** argument to the **exit** command:

```
[edit]
user@switch# exit
Exiting configuration mode


[edit protocols rstp interface ge-0/0/12.0]
user@switch# exit configuration-mode
Exiting configuration mode

user@switch>
```

Moving Between Levels (6 of 6)

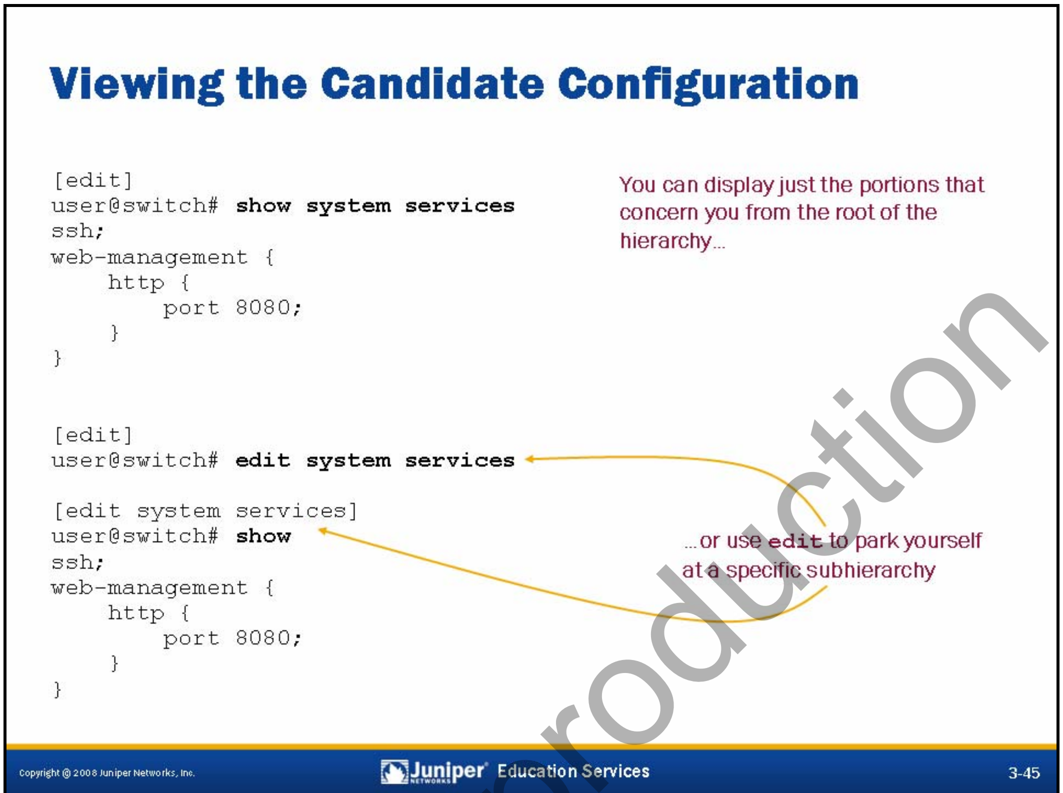
- Summary of moving between levels:
  - **edit** functions like a CD command
  - **up** moves up one level
  - **up** ***n*** moves up *n* levels
  - **top** moves to the top of the hierarchy
  - **exit** moves to the *previous higher* level in the hierarchy or exits configuration mode if at the top level of the hierarchy

```
[edit]
user@switch# edit protocols rstp interface ge-0/0/12.0
[edit protocols rstp interface ge-0/0/12.0]
user@switch# up
[edit protocols rstp]
user@switch# up 2
[edit]
user@switch# top
warning: already at top of configuration; use 'exit' to exit
[edit]
user@switch# exit
Exiting configuration mode
user@switch>
```

 Juniper Education Services 3-44

In Summary

The **edit**, **up**, **top**, and **exit** commands let you quickly navigate between levels of the configuration hierarchy.



Displaying the Candidate Configuration

To display the candidate configuration, use the configuration-mode **show** command. This command displays the configuration at the current hierarchy level or at the specified level below the current location.

The **show** command has the following syntax: **show** ***statement-path***. When displaying the configuration, the CLI indents each subordinate hierarchy level, inserts curly brackets to indicate the beginning and end of each hierarchy level, and places a semicolon at the end of statements that are at the lowest level of the hierarchy. The display format is the same format you use when creating an ASCII configuration file, and it is also the same format that the CLI uses when saving a configuration to an ASCII file.

In cases where an empty statement leads to an invalid configuration because it is incomplete or meaningless, the **show** command does not display any of the statement path.

Configuration File Differences (1 of 2)

- Change the candidate configuration:

```
[edit system]
user@switch# set services telnet
[edit system]
user@switch# delete services web-management
[edit system]
user@switch# delete services ssh
```

- Display differences between the candidate and active configurations:

```
user@switch# show | compare
[edit system services]
-   ssh;
+   telnet;
-   web-management {
-       http {
-           port 8080;
-       }
-   }
```

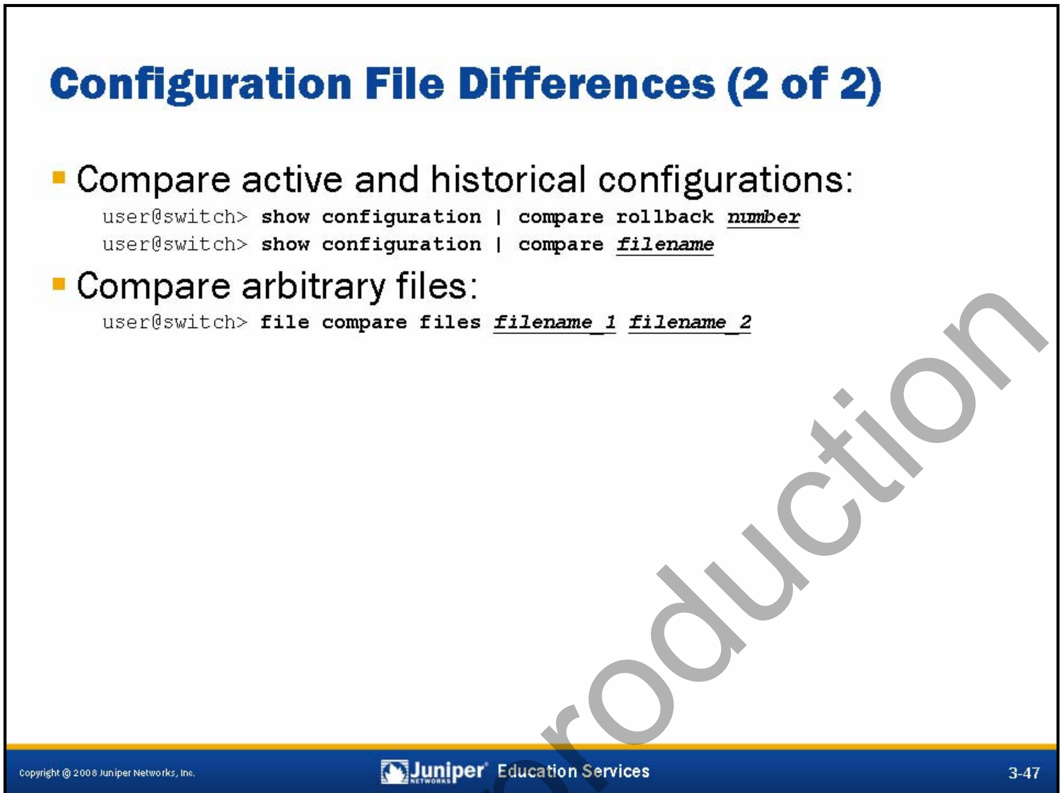
 Juniper NETWORKS Education Services 3-46

Modifying a Candidate Configuration

The example on the slide modifies a candidate configuration by enabling Telnet access and removing SSH and J-Web access. **set** and **delete** commands are relative to the current hierarchy.

Viewing Differences

Piping the output of a **show** command to the CLI **compare** function displays the differences between the candidate configuration file and the active configuration, also known as `rollback 0`. Configuration comparison is *patch*-like. Thus, instead of showing the entire configuration and where changes were made, only the actual changes are shown. By using the pipe switch, you can save the configuration differences to the file and location of your choosing. Once saved, you can issue a **load patch *filename*** command to merge the contents of the patch file into the candidate configuration where they can be viewed, edited, and, ultimately, committed.

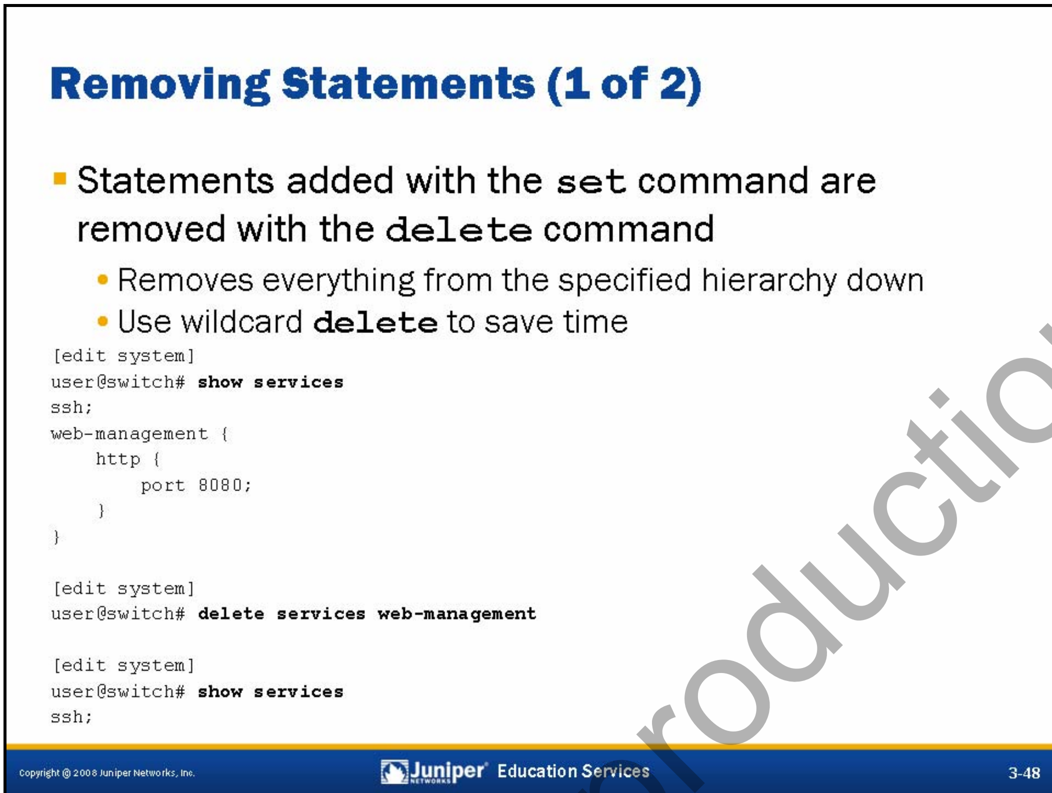


Comparing Active and Rollback Configurations

Using the operational-mode **show configuration | compare rollback *number*** command, as shown on the slide, allows you to view differences between the active configuration and any of the 49 rollback configurations. Similarly, the **show configuration | compare *filename*** command allows you to compare the active configuration to an arbitrary file. You can also use **show | compare rollback *number*** and **show | compare *filename*** in configuration mode to compare the *candidate* configuration with rollback configurations and arbitrary files respectively.

Viewing Differences in Other Files

The operational-mode **file compare files** command allows you to view differences between any two text files, including log files. The output of this command is in the same patch-like format as the **show configuration | compare** command.



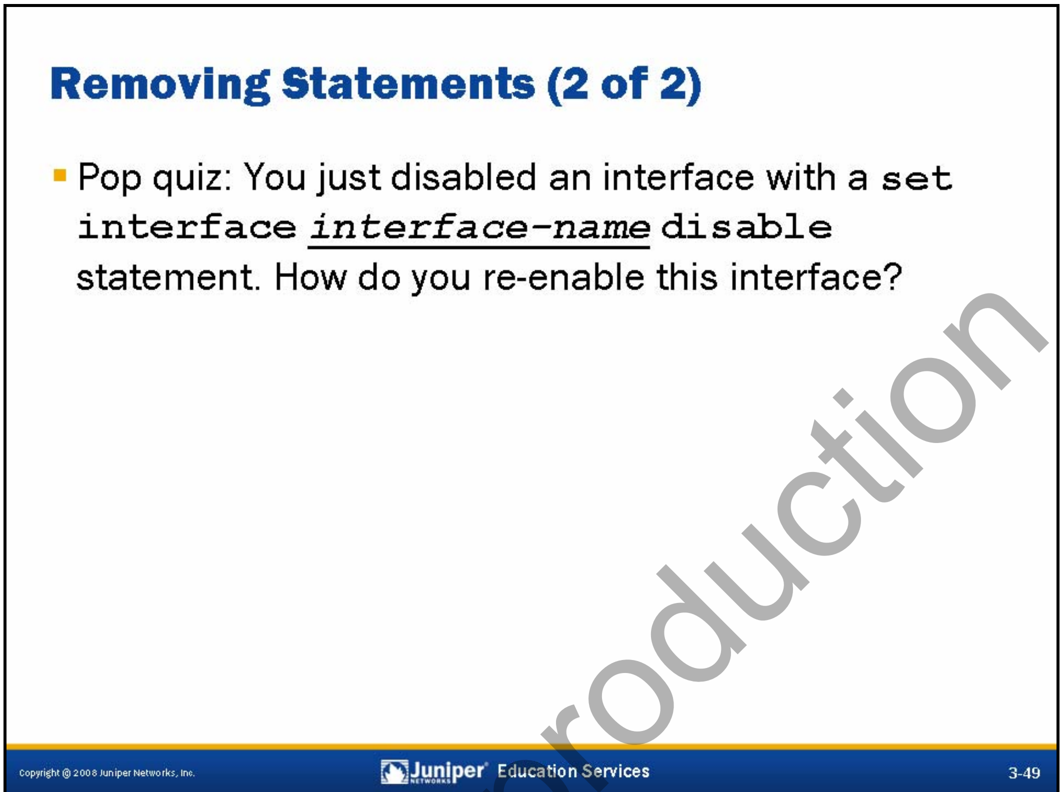
Removing Configuration Statements

Use the configuration-mode **`delete`** command to remove statements that were added to the configuration with a **`set`** command. This command deletes the statement and all its subordinate statements and identifiers. Deleting a statement or an identifier effectively *unconfigures* the functionality associated with that statement or identifier, returning that functionality to its default condition.

Consider using the **`wildcard delete`** function when deleting individual statements is too arduous and deleting an entire configuration subhierarchy lacks the granularity that is needed. Sample syntax for a **`wildcard delete`** is shown:

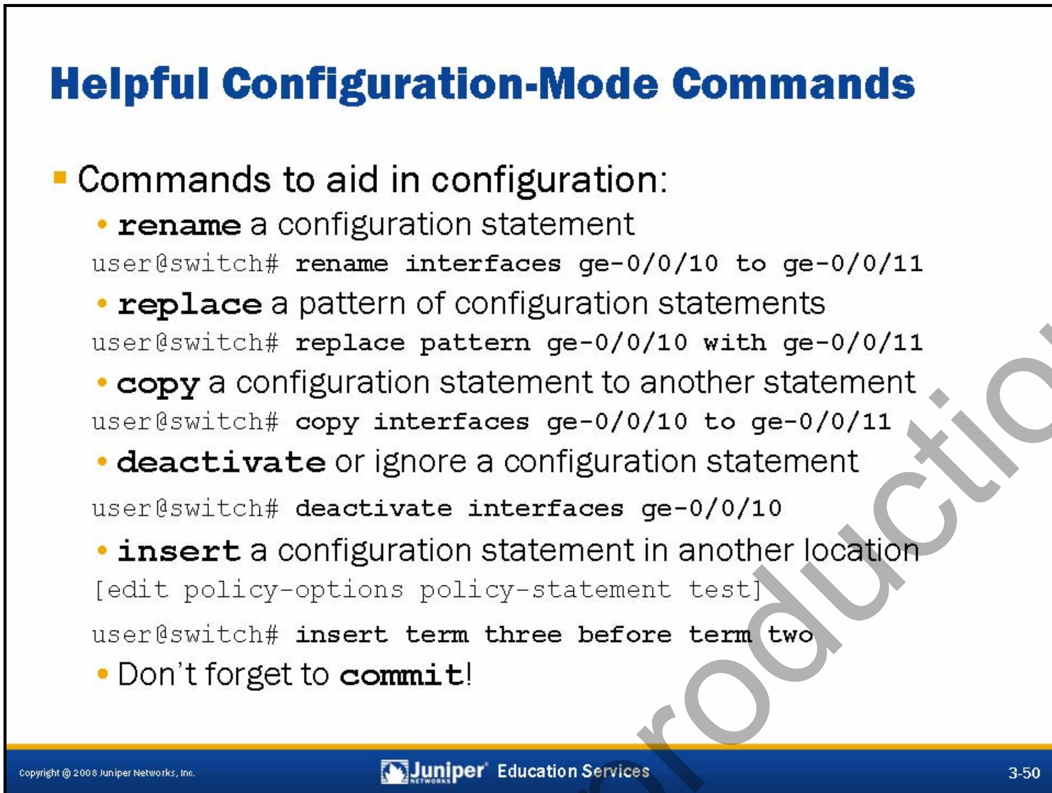
```
[edit]
user@switch# wildcard delete interfaces ge-*
 matched: ge-0/0/12
Delete 1 objects? [yes,no] (no) yes
```

In addition to deleting configuration statements, you should also consider the use of **`deactivate`** to cause the specified portion of the configuration hierarchy to be ignored while still retaining the original configuration. Issue an **`activate`** command to place the configuration back into effect. Also consider the use of **`disable`** for interfaces. Use the **`set`** command to add a **`disable`** statement to flag a given interface as being administratively disabled.



Pop Quiz!

Issue a **`delete interface`** ***`interface-name`*** **`disable`** command to delete the disable statement placed into effect with a **`set`** command. This syntax has been known to strike some folks as being more than a bit on the double-negative side; then again, these same folks tend to agree that a **`no shutdown`** statement, as used for similar functionality on other vendors' equipment, is equally counter-intuitive!



Using Configuration Mode Efficiently

Using the configuration commands shown on the slide can save time and increase accuracy. The full list of configuration mode commands is demonstrated here:

```
user@switch# ?
Possible completions:
  <[Enter]>            Execute this command
  activate             Remove the inactive tag from a statement
  annotate             Annotate the statement with a comment
  commit               Commit current set of changes
  copy                 Copy a statement
  deactivate           Add the inactive tag to a statement
  delete               Delete a data element
  edit                 Edit a sub-element
  exit                 Exit from this level
  extension            Extension operations
  help                 Provide help information
  insert               Insert a new ordered data element
  load                 Load configuration from ASCII file
  quit                 Quit from this level
  rename               Rename a statement
  replace              Replace character string in configuration
  rollback             Roll back to previous committed configuration
  run                  Run an operational-mode command
```

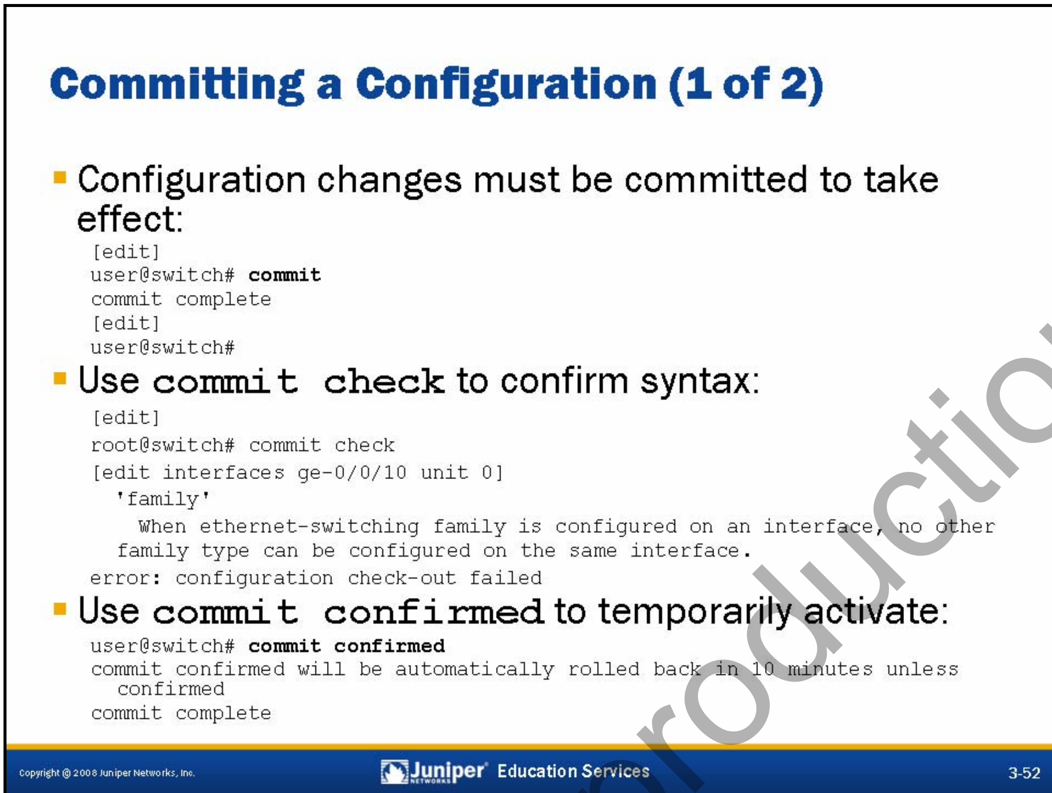
Continued on next page.

Using Configuration-Mode Efficiently (contd.)

```
  save                    Save configuration to ASCII file
  set                     Set a parameter
  show                    Show a parameter
  status                  Show users currently editing configuration
  top                     Exit to top level of configuration
  up                      Exit one level of configuration
  wildcard                Wildcard operations
```

Just like with any configuration change, the change will not become part of the active configuration until it is committed:

```
[edit]
user@switch# deactivate interfaces ge-0/0/0

[edit]
user@switch# commit
commit complete

[edit]
user@switch# show interfaces ge-0/0/0
##
## inactive: interfaces ge-0/0/0
##
unit 0 {
    family ethernet-switching;
}

[edit]
user@switch# activate interfaces ge-0/0/0

[edit]
user@switch# commit
commit complete

[edit]
user@switch# show interfaces ge-0/0/0
unit 0 {
    family ethernet-switching;
}
```



Don’t Forget to Commit

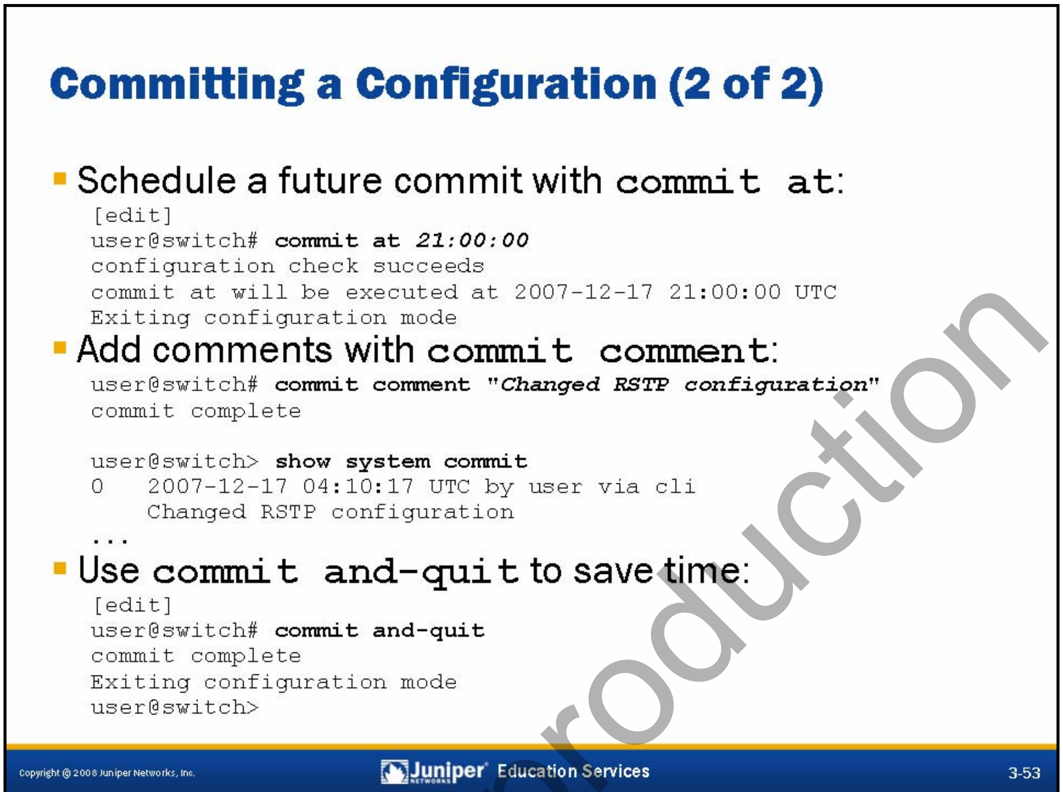
Remember, the switch does not automatically apply your configuration changes. You must use the **`commit`** command to activate your candidate configuration.

Checking Configuration Syntax

When you commit a candidate configuration (which you can do from any hierarchy level), you commit the entire configuration in its current form. Use the **`commit check`** command to validate the syntax of a candidate configuration without actually placing it into effect.

Remote Configuration Is Risky

Of course, **`commit check`** cannot catch logical errors in your configuration. What happens when you are configuring a switch remotely and make a mistake that leaves the switch inaccessible to remote connections? This scenario is solved by the **`commit confirmed`** command. When you issue a **`commit confirmed`** ***`time-out`*** command, the system starts a timer, during which time it expects to see another **`commit`**. If a second **`commit`** does not occur within the time-out value specified (a range of 1 to 65,535 minutes is supported, with 10 minutes being the default), the system performs a **`rollback 1`**, **`commit`** sequence on your behalf. After the automatic rollback you can load the `rollback 1` file to look for your mistake.



Scheduled Commits

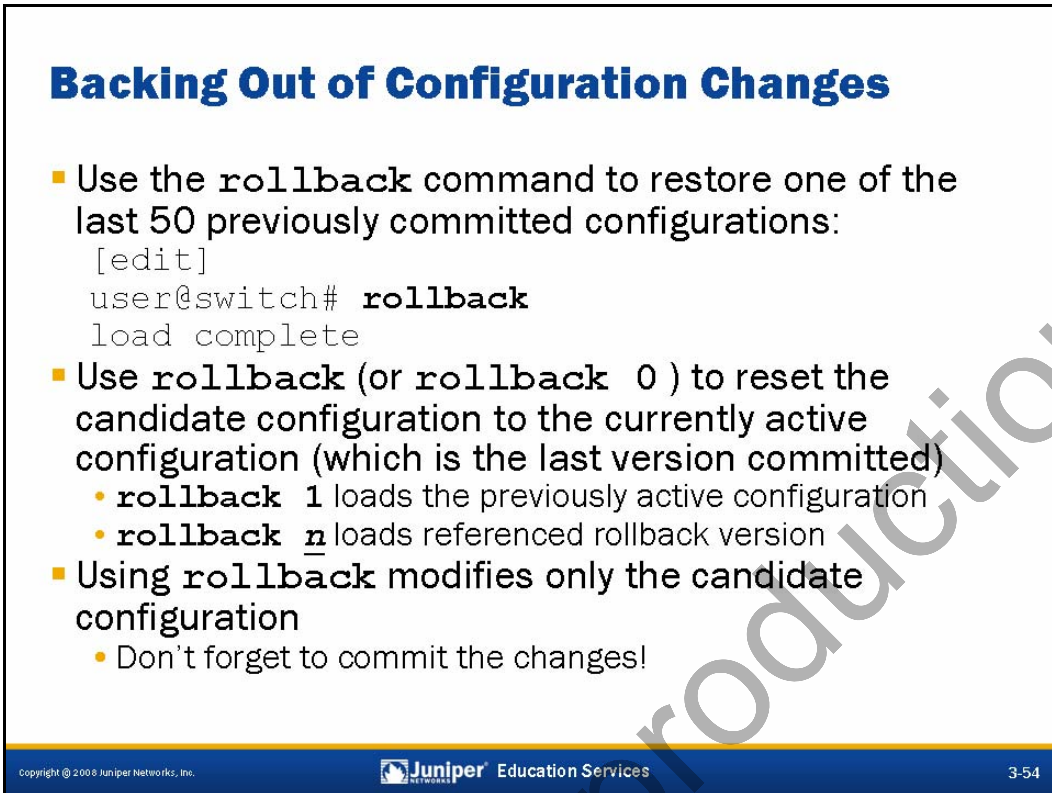
You can also schedule a commit that occurs at a specific time using the **commit at *time*** command. To view any pending commits (and the commit history), use the **show system commit** command. You can cancel a pending commit with the **clear system commit** command.

Adding a Log Entry to Your Commit

You can also add a log entry to your commit using the **commit comment "*comment-string*"** option. These logs are visible in the output of the **show system commit** command.

Exiting Configuration Mode

You can add the **and-quit** option to the **commit** command to activate your changes and exit configuration mode in a single step.



Backing Out of Changes

The software saves the last 50 committed versions of the configuration. To overwrite the candidate configuration with one of these previously committed versions, use the CLI configuration **rollback** command. By default, the system returns to the most recently committed configuration.

Specifying Rollback Files

To return to a version prior to the configuration most recently committed, include the version number in the **rollback** command:

```
[edit]
user@switch# rollback version
load complete
[edit]
user@switch#
```

The ***version*** argument can be a number in the range 0 through 49. The most recently saved configuration is version 0, which is a copy of the current active configuration. The oldest committed configuration that is now automatically saved is now version 49.

Continued on next page.

You Must Commit

The **rollback** command modifies only the candidate configuration. To activate the changes that you loaded, issue the **commit** command:

```
[edit]
user@switch# commit
```

Saving Configuration Files

- Save the current candidate configuration using the **save** command:

```
[edit]
user@switch# save filename
```

  - File is saved to user's home directory unless full path name is specified
  - Saves only from the current hierarchy down
- Filename can specify:
  - A path and filename on the local router's file system
  - A URL (FTP and SCP)
- Miscellaneous features:
  - **terminal** option for **save** commands
    - Simplifies load operations from terminal buffers
  - Pipe option for **display set**
    - Displays the **set** statements used to create a configuration
  - Periodic saves to a remote host

 Juniper Education Services 3-56

Saving Files

You can save the candidate configuration from your current configuration session to an ASCII file. Doing this procedure saves the configuration in its current form, including any uncommitted changes. If more than one user is modifying the configuration, saving it saves the changes made by all the users.

Note that configuration statements only at the current hierarchy level and below are saved. To save the entire candidate configuration, you must be at the top level of the configuration hierarchy. By default, JUNOS software saves the configuration to the specified file in your home directory. For example, user *doug* would store files in the `/var/home/doug` directory. You can change this default by specifying a path name.

Specifying File Names

You can specify a filename in one of the following ways:

- *filename* or *path/filename*.
- `ftp://user:password@host/path/filename`: Puts the file in the location explicitly described by this URL using the FTP protocol. Substituting the word "prompt" for the password causes the router to prompt you for the user's password.
- `scp://user@host/path/filename`: Puts the file on a remote system using the SSH protocol. You will be prompted for user's password.

Continued on next page.

Miscellaneous Features

JUNOS software supports saving configuration data to a terminal device. With this option, the appropriate configuration hierarchy name, curly brackets, and `replace` tag are added to readily accommodate pasting into another switch's configuration using some form of load-terminal operation. You can also save the output to a file for later use in a file load operation. An example of **`load terminal`** at work is provided here:

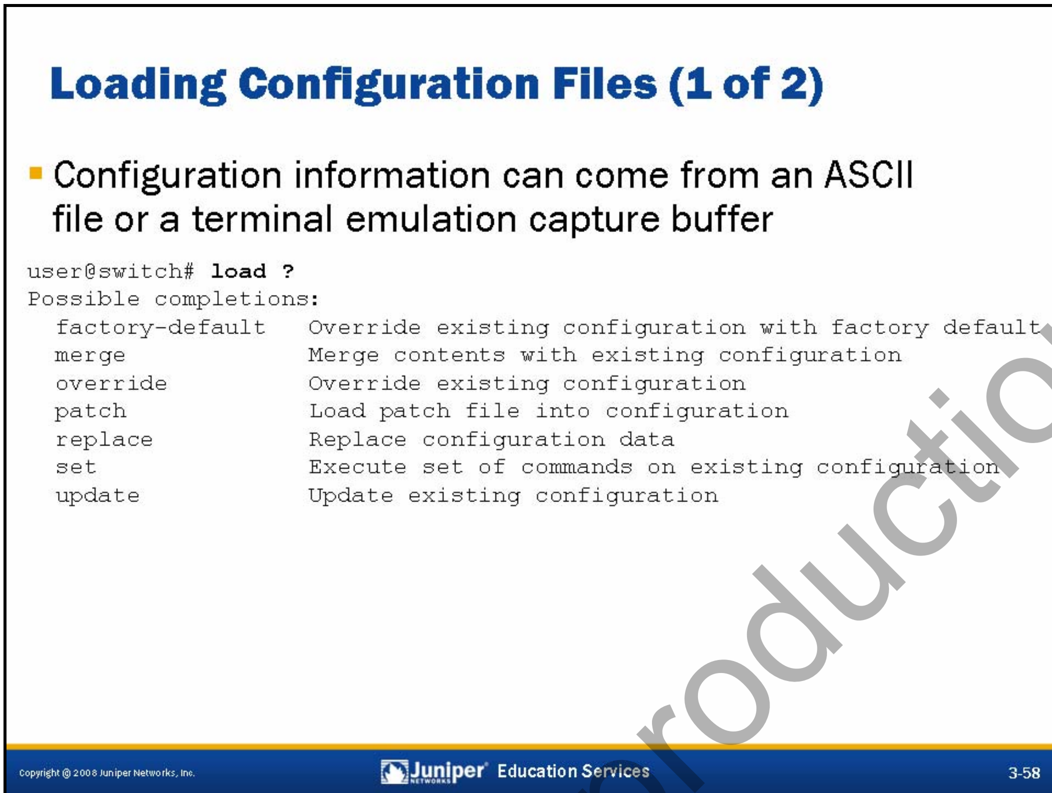
```
[edit]
user@switch# load replace terminal
[Type ^D at a new line to end input]
protocols {
replace: ospf {
        area 0.0.0.0 {
            interface ge-0/0/0.0;
            interface ge-0/0/1.0;
            interface ge-0/0/2.0;
        }
    }
}
load complete
```

Piping output to **`display set`** is supported. This feature converts a configuration into the actual **`set`** statements used to create the configuration; this option is intended to simplify the editing of configuration data being cut and pasted between switches:

```
[edit protocols ospf]
user@switch# # show | display set
set protocols ospf area 0.0.0.0 interface ge-0/0/0.0
set protocols ospf area 0.0.0.0 interface ge-0/0/1.0
set protocols ospf area 0.0.0.0 interface ge-0/0/2.0
```

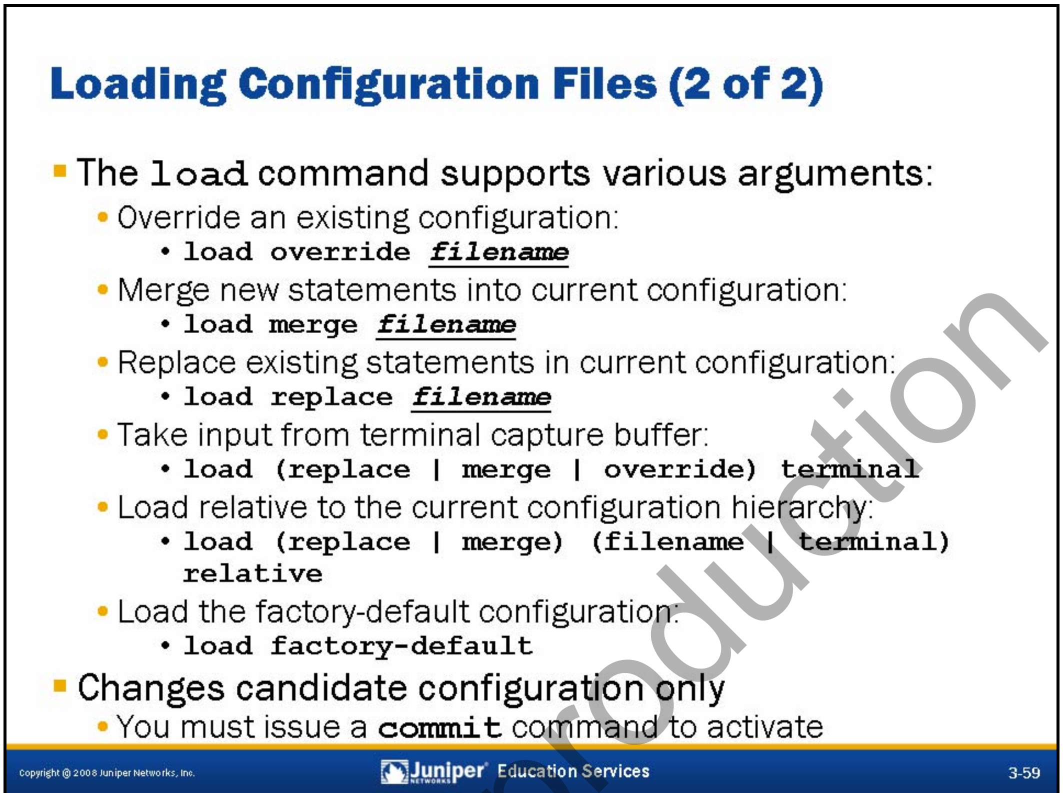
You can configure either a periodic or commit-driven upload of the switch's configuration to a particular host using FTP. A typical configuration is shown:

```
[edit system archival]
user@switch# show
configuration {
    transfer-on-commit;
    archive-sites {
        "ftp://lab:lab123@10.250.0.254";
    }
}
```



Loading a Configuration

You can use the configuration-mode **`load`** command to load a complete or partial configuration from a local file, from a file on a remote machine, or from a terminal emulation program's capture buffer. The **`load`** command supports several arguments that determine the specifics of the operation.



Load Options

The following list provides details about the arguments to the **load** command:

- **override**: Completely overwrites the current configuration with the configuration being loaded. You must perform override operations at the root of the configuration hierarchy.
- **merge**: Combines the current configuration with the configuration being loaded.
- **replace**: Looks for a replace tag in the configuration being loaded. Existing statements of the same name are replaced with the those in the loaded configuration for stanzas marked with the `replace` tag.
- **terminal**: Uses the text you type at the terminal as input to the configuration. Type Ctrl+d to end terminal input. This option is usually used in conjunction with a terminal emulation program's copy/paste functionality to copy and paste configuration data from one system to another.

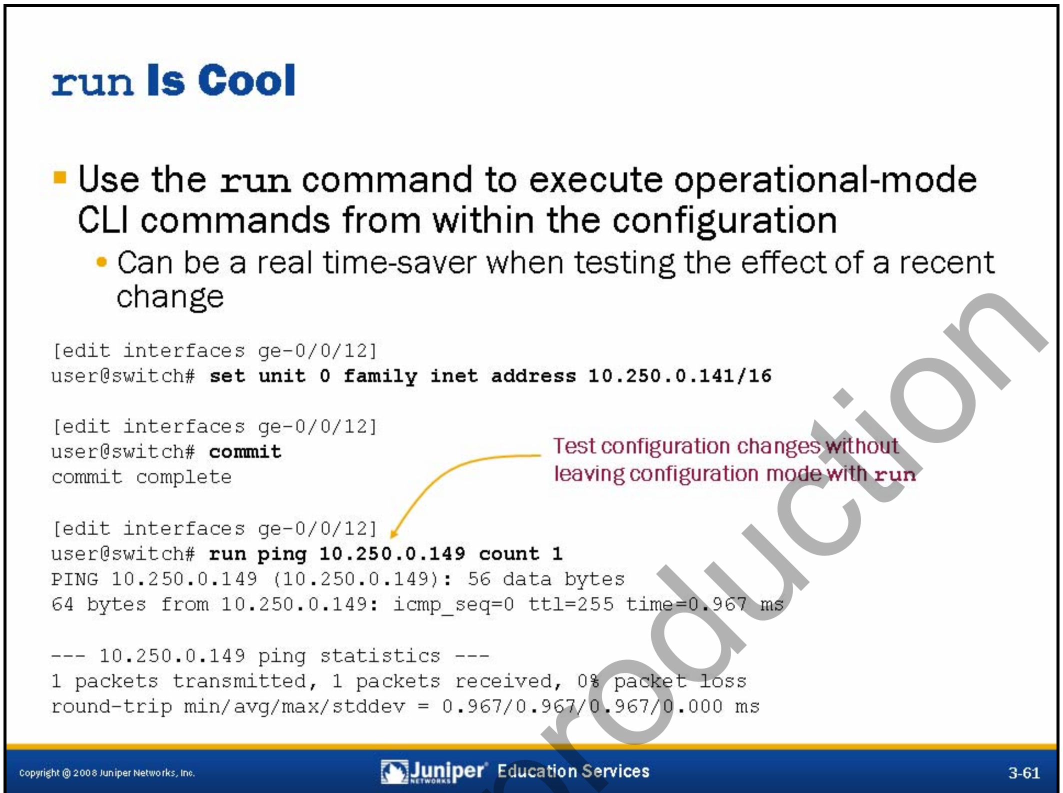
Continued on next page.

Load Options (contd.)

- **`relative`**: Normally, a **`load merge`** or **`load replace`** operation requires that the data being loaded contain a full path to the related configuration hierarchy. The **`relative`** option negates this need by telling the switch to assume that the data being loaded should be added *relative* to the current configuration hierarchy.
- **`factory-default`**: Replaces the full current configuration with the factory-default configuration.

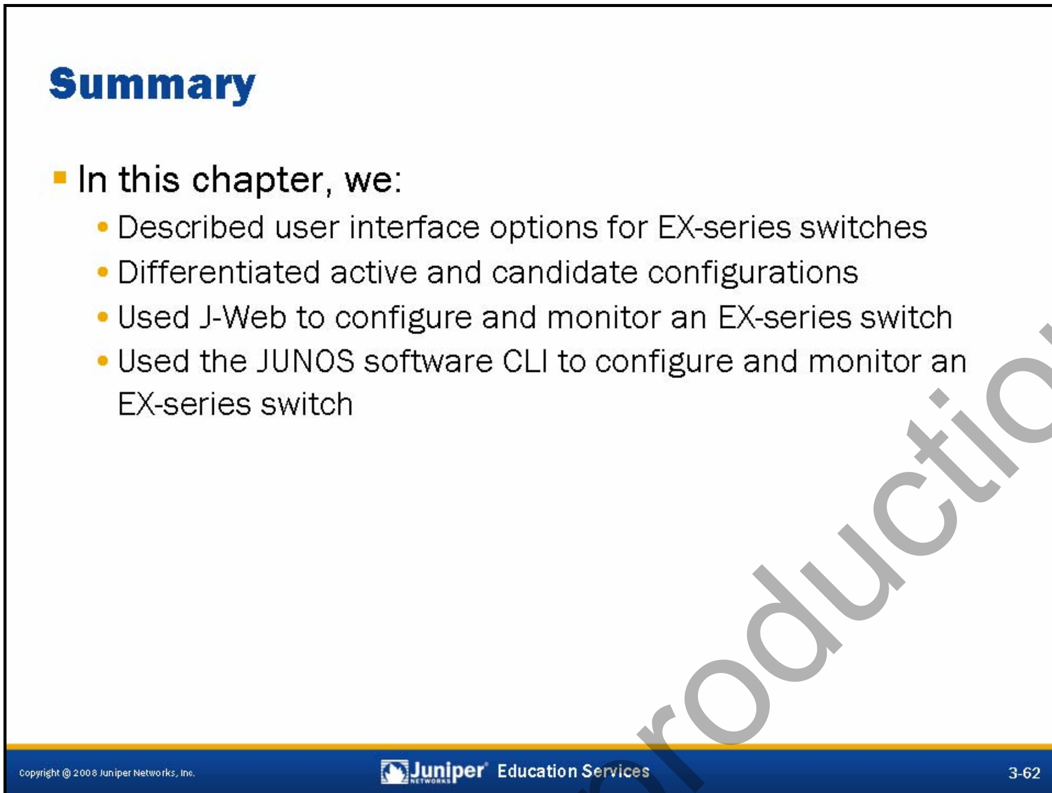
Changes Candidate Configuration Only

In all cases, after the **`load`** operation is complete, you must issue a **`commit`** to activate the changes made to the configuration.



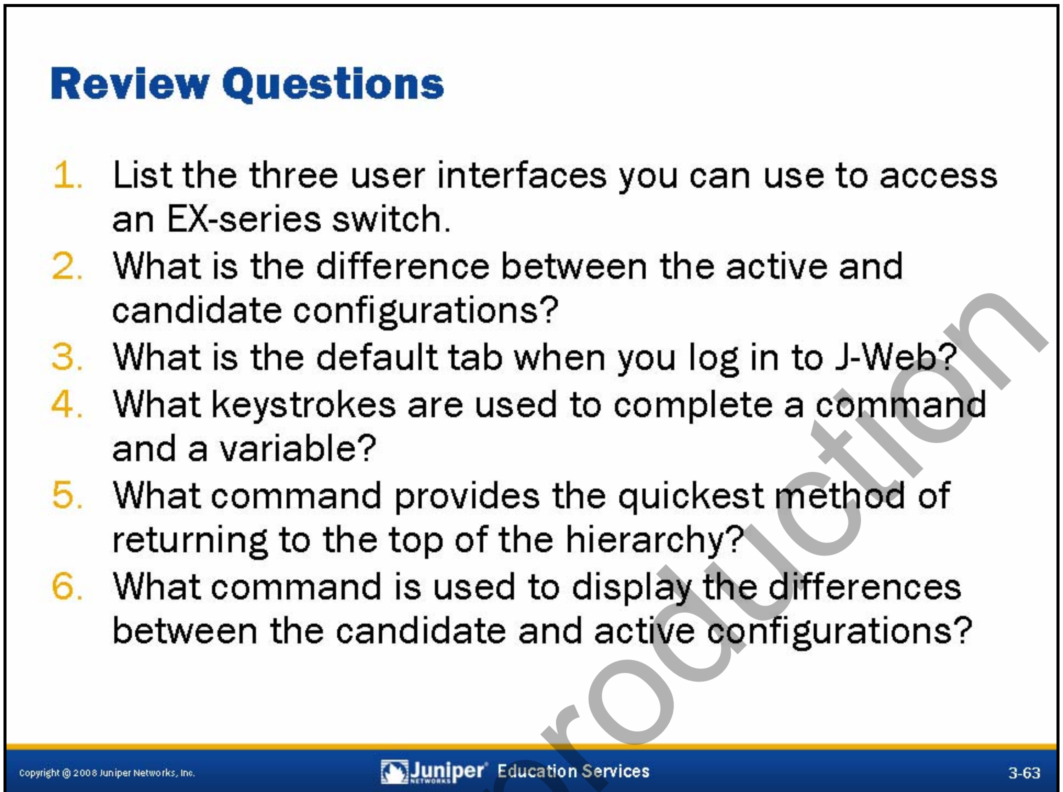
Run with It

The **run** command allows you to execute operational-mode commands while in configuration mode. It is similar to the **do** command on other vendors' equipment. This extremely handy time-saver works for all operational-mode commands and is supported at all configuration hierarchies. In the example on the slide, we are editing the configuration for the router's ge-0/0/12 interface. After assigning what we hope to be the correct IP address, we commit the change and invoke the **run** command to execute a quick ping test.



This Chapter Discussed:

- User interface options;
- User authentication and authorization;
- Active and candidate configurations;
- Using J-Web GUI to configure and monitor an EX-series switch; and
- Using the CLI to configure and monitor an EX-series switch.



Review Questions

1.

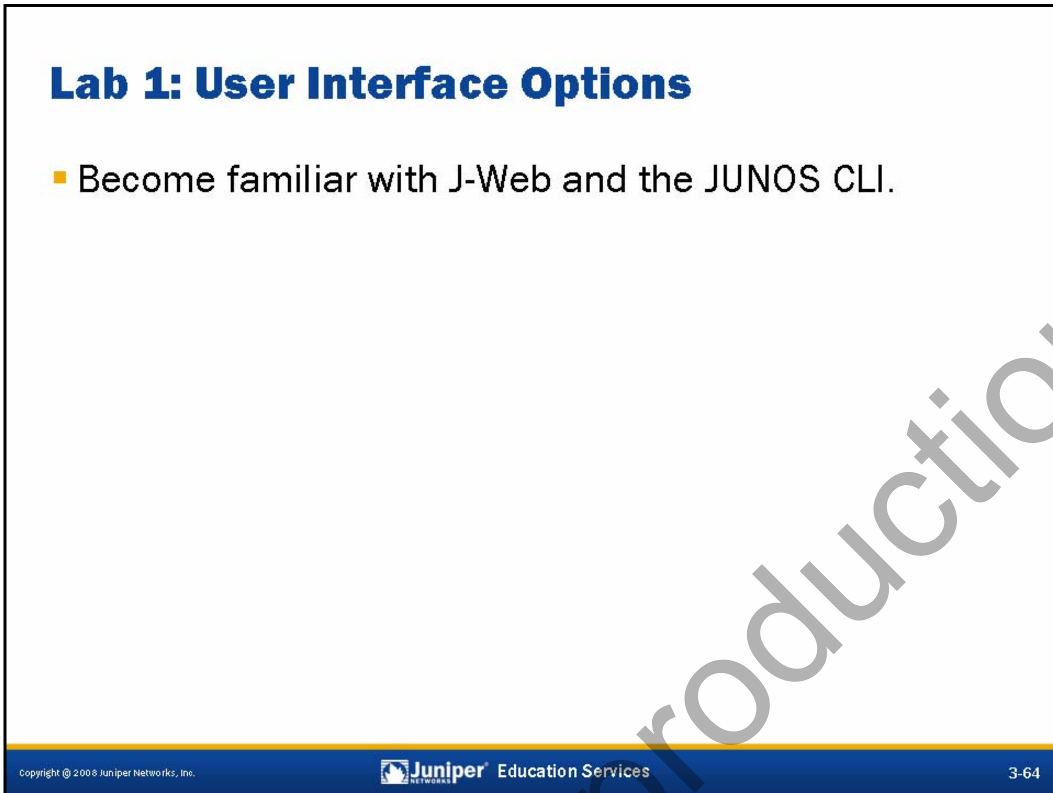
2.

3.

4.

5.

6.

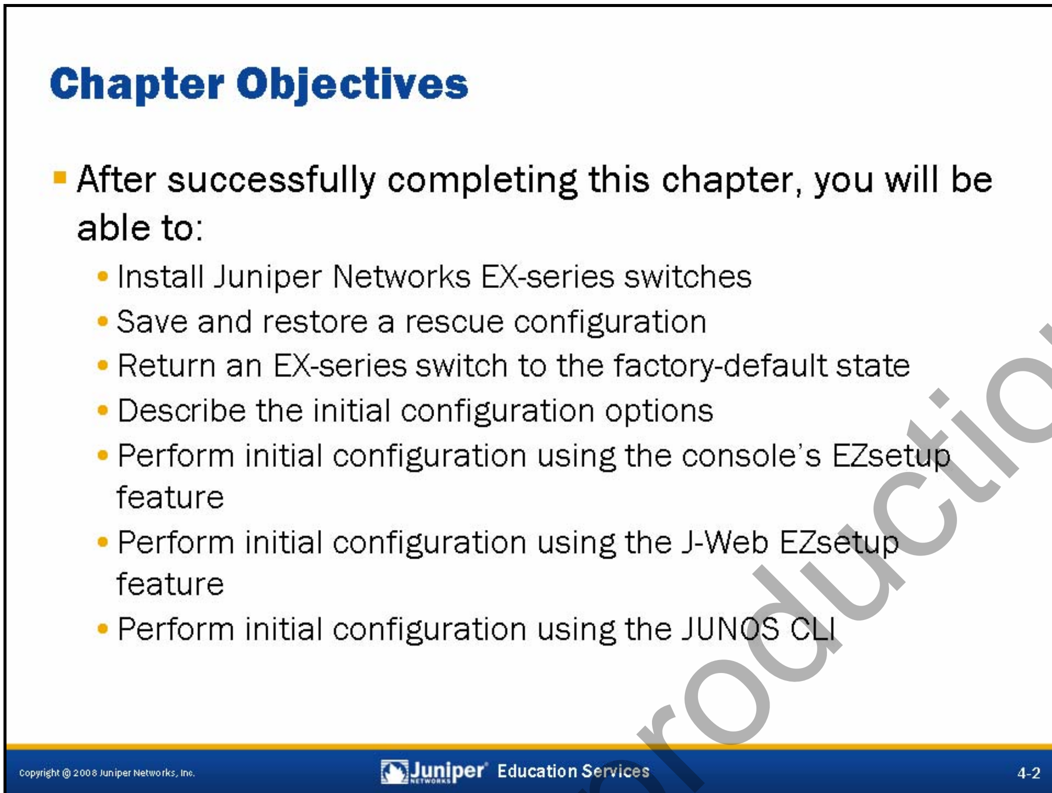


Lab 1: User Interface Options

The slide provides the objective for this lab.

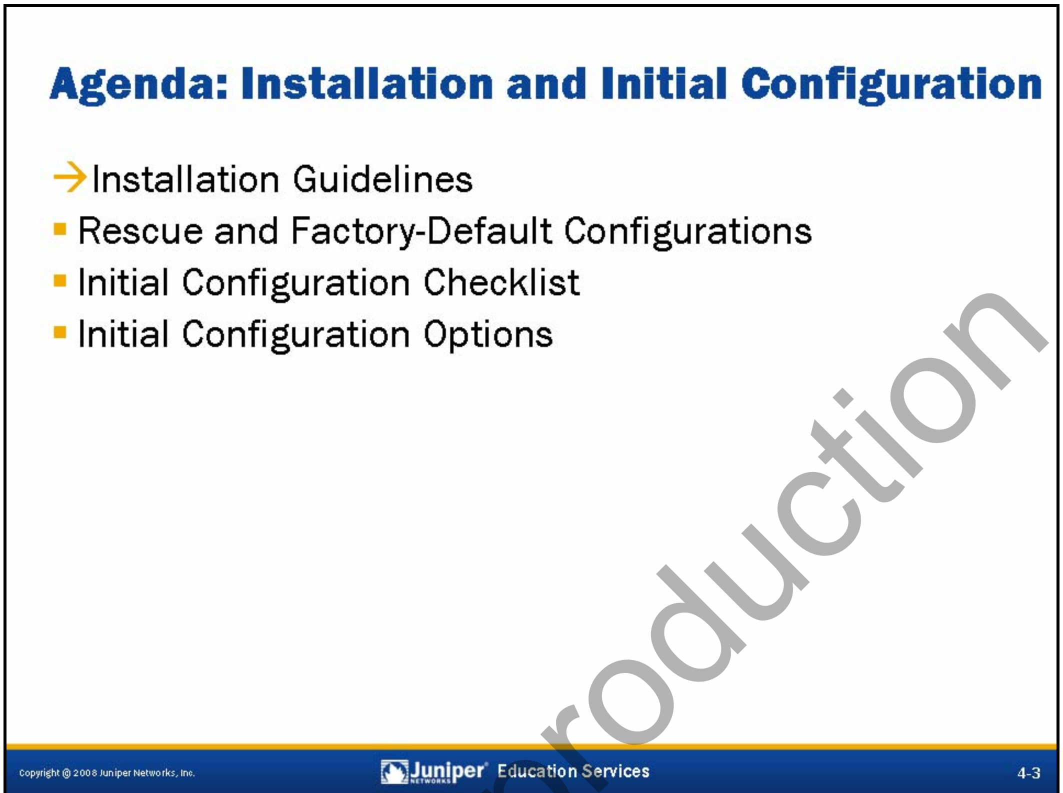
Operating Juniper Networks Switches in the Enterprise

Chapter 4: Installation and Initial Configuration



This Chapter Discusses:

- The general process and guidelines for installing Juniper Networks EX-series switches;
- Creating, saving and loading a rescue configuration file;
- Returning the switch to its factory-default configuration;
- Information required for initial configuration; and
- Performing initial configuration on the switch using the console's EZsetup option, J-Web EZsetup, and the command-line interface (CLI).



Installation Guidelines

The slide lists the topics we cover in this chapter. We discuss the highlighted topic first.

General Installation Guidelines

- Follow documented safety guidelines
- All EX-series enterprise switches:
  - Rack, desk, or wall mountable
    - 1 RU, weight might vary with platform
    - Multiple chassis for use as a Virtual Chassis system require rack mount
    - Place heavier devices at bottom of the rack
    - Center mount is preferred
- Attach network and console cables
- Attach power cables
- Use the hardware installation training resources:
  - Technical publications: http://www.juniper.net/techpubs/

 Juniper Education Services 4-4

Safety Guidelines

Be sure to read and follow applicable safety guidelines when installing a Juniper Networks EX-series enterprise switch. You can find these guidelines in the accompanying documentation that is included with the switch or online at http://www.juniper.net/techpubs/.

Switch Installation

Juniper Networks EX-series enterprise switches are rack, desk, or wall mountable. If you are installing multiple EX-series switches to function as a Virtual Chassis system, you must install the switches in a rack. One pair of mounting brackets and rubber feet are supplied with the EX-series switch. You must order wall mount kits separately. All EX-series enterprise switches are 1 rack unit (RU) in size but vary in weight. If multiple switches are being installed in one rack, install the first device at the bottom and proceed upward in the rack. Install heavier switches in the lower part of the rack.

Continued on next page.

Connecting Cables

You can connect the switch to the console using the provided console cable. Use a standard RJ-45 Ethernet cable, no crossover necessary, for connecting to the network ports.

Attaching Power

EX-series switches include an appropriate AC power cord for your geographic region.

Installation Resources

Resources are available with detailed installation procedures. You can find this documentation at http://www.juniper.net/techpubs/.

Power On and Power Off

- JUNOS software is a multitasking environment
  - A graceful shutdown of the operating system ensures file system integrity
    - Use the J-Web `Maintain > Reboot` page or the **`request system halt`** CLI command to gracefully halt JUNOS software
    - Power is maintained to the system; reboot with console activity
- Rebooting the system with the LCD menu:
  - System reboot option under Maintenance mode
- Automatic power-on feature

 Juniper Education Services 4-6

Gracefully Shutting Down the JUNOS Software

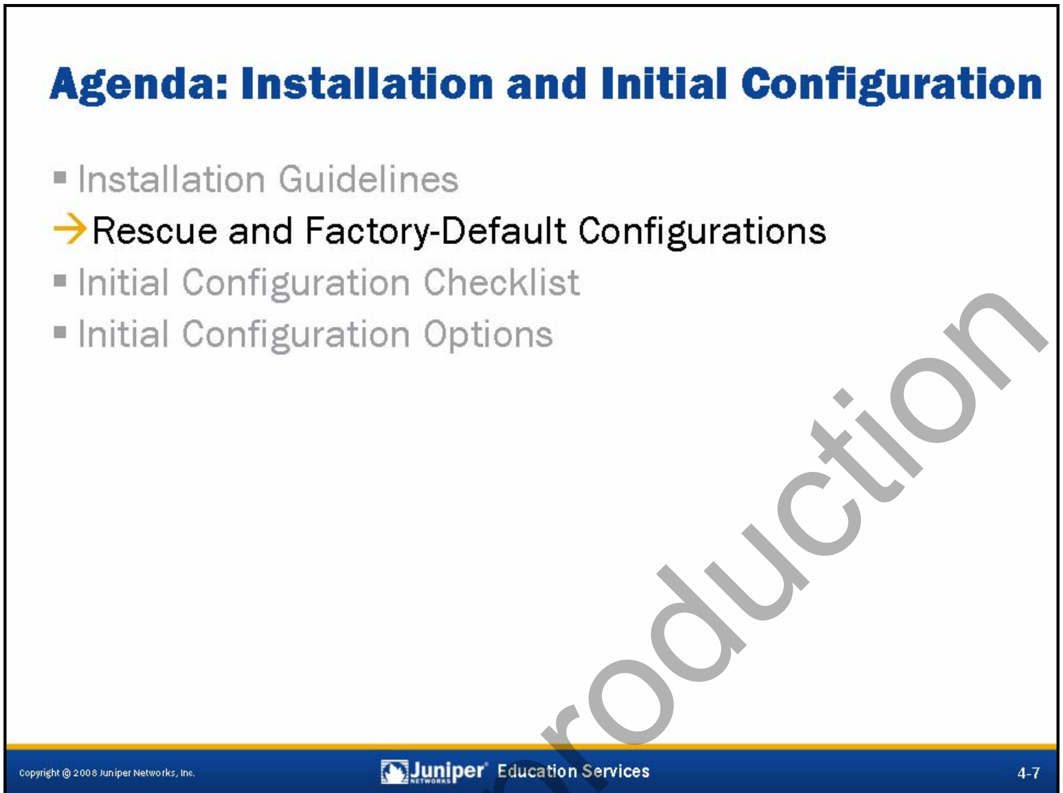
The JUNOS software is a multitasking environment. To ensure file system integrity, you should always gracefully shut down the switch. Although unlikely, failure to gracefully shut down the switch could possibly leave it unable to boot.

Rebooting the System with the LCD Menu

Press the LCD menu button to enter Maintenance mode. Choose the *System Reboot* option to reboot the EX-series switch.

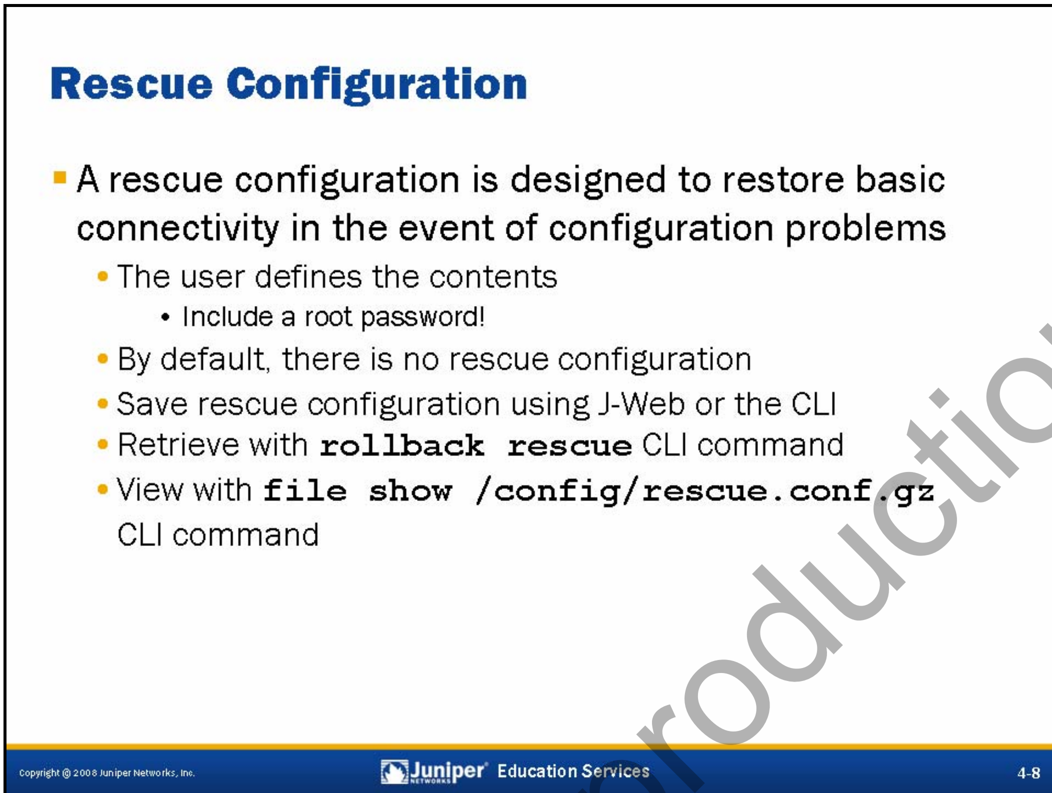
Automatic Power On

If power to an operating switch is interrupted, the switch automatically powers on upon power restoration. The switch does not require any intervention in this situation.



Rescue and Factory-Default Configurations

The slide highlights the topic we discuss next.



What Is a Rescue Configuration?

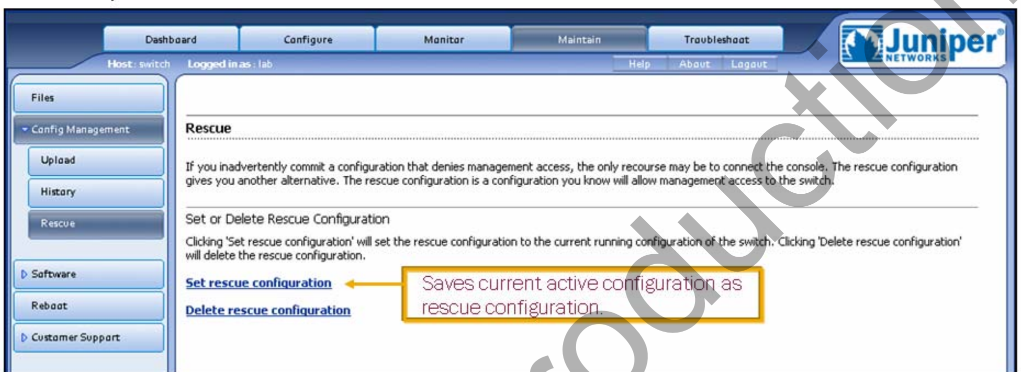
A rescue configuration is a user-defined, known-good configuration that you can quickly activate in the event that the active configuration is deleted or misconfigured in such a way that network connectivity to the switch is lost. We recommend that the rescue configuration contain the minimum elements necessary to restore network connectivity to the switch. For added security, the rescue configuration must include a root password.

By default, no rescue configuration is defined. You can save the current active configuration as the rescue configuration using J-Web or the CLI.

Once saved, you can activate the rescue configuration by entering the **`rollback rescue`** configuration-mode command.

Saving a Rescue Configuration

- Two methods of saving rescue configuration:
  - **`request system configuration rescue save`** CLI command
  - J-Web `Maintain > Config Management > Rescue` option

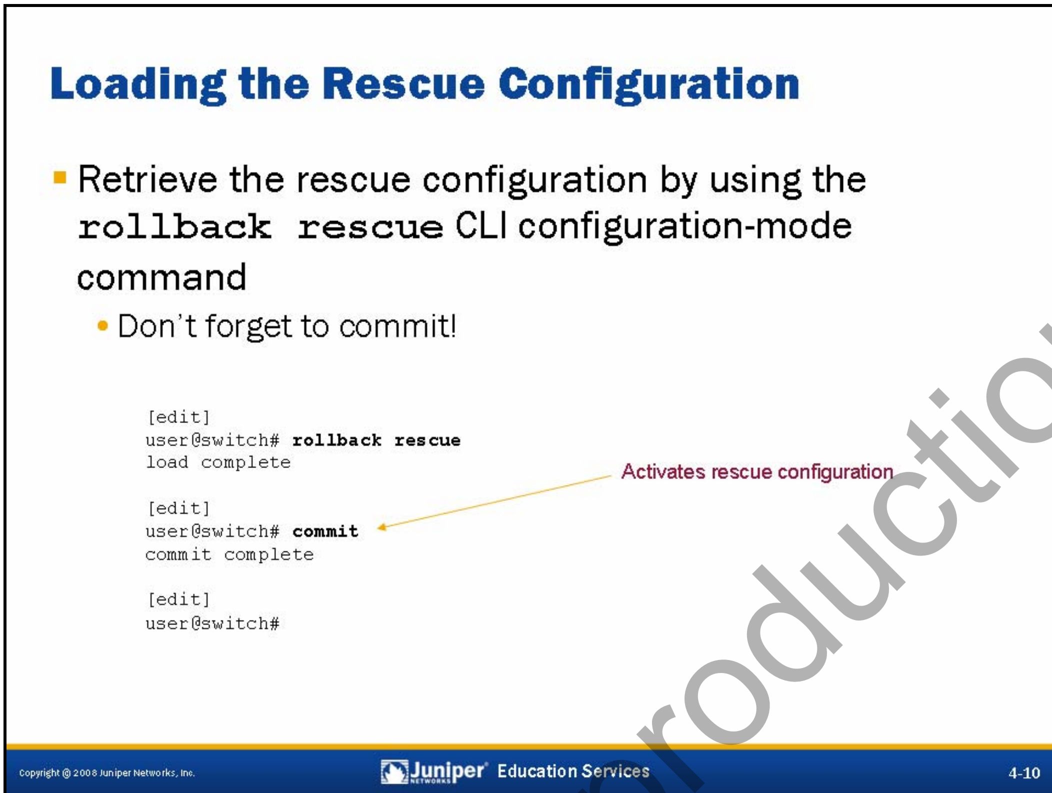


Juniper Education Services
4-9

Saving a Rescue Configuration Using J-Web or the CLI

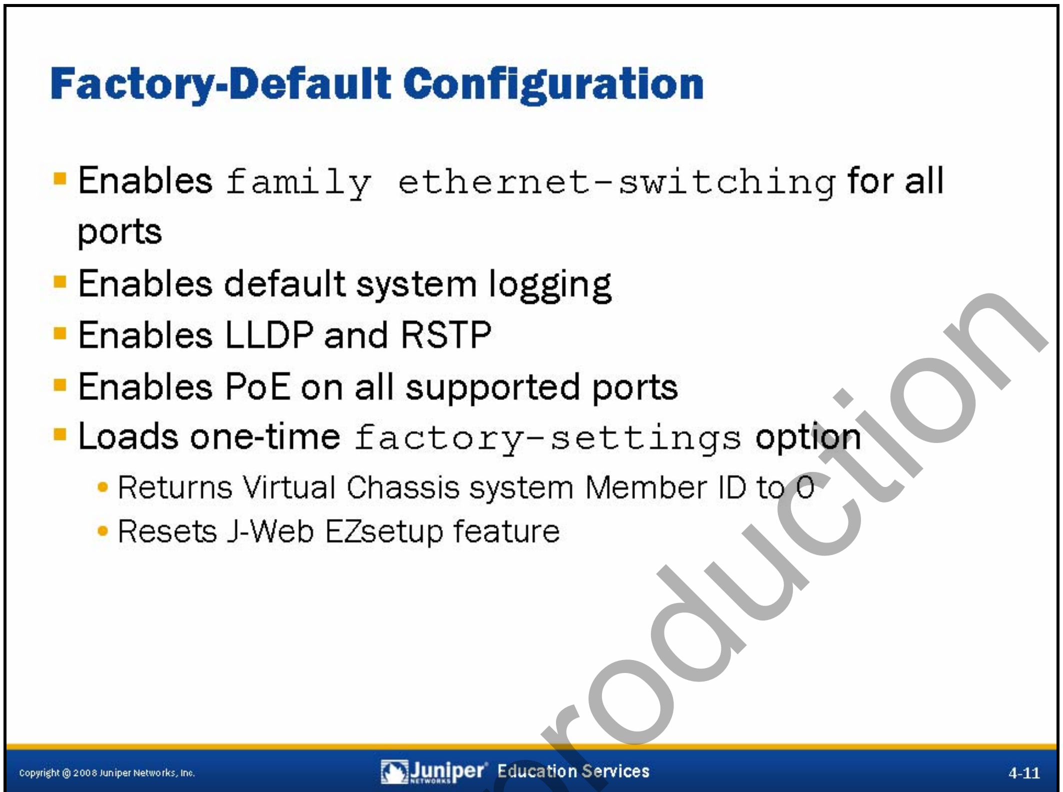
The J-Web `Maintain > Config Management > Rescue` option allows you to view, save, or delete the rescue configuration. The `Set rescue configuration` link sets the rescue configuration to the currently active configuration. The `Delete rescue configuration` link removes any rescue configuration previously set. The `View rescue configuration` link allows you to view the contents of the rescue configuration. It appears only if a rescue configuration is set.

You can also set or delete the rescue configuration from the CLI. The **`request system configuration rescue save`** command sets the rescue configuration to the currently active configuration, and the **`request system configuration rescue delete`** command deletes any rescue configuration previously set. The **`file show /config/rescue.conf.gz`** command allows you to see the contents of the rescue configuration file.



Using `rollback rescue`

The configuration-mode **`rollback`** command also accepts a **`rescue`** argument. Using **`rollback rescue`** overwrites the candidate configuration with the rescue configuration. As always, you must use the **`commit`** command to activate the candidate configuration.



Default Switching Mode

The factory-default configuration enables all switch ports for switching by configuring `family ethernet-switching` under each interface unit.

System Logging

The factory-default configuration enables the same syslog options as other JUNOS software-based products. JUNOS software creates a log named `messages` that logs *any* facility at the *notice* severity level. It also logs the *authorization* facility at the *info* severity level. The software creates an `interactive-commands` log that lists commands input into the switch. A third syslog option is enabled that notifies any user logged in to the switch of any messages received with an *emergency* severity level.

Switching Protocols

The switch's factory-default configuration enables the Link Layer Discovery Protocol (LLDP) and the Rapid Spanning Tree Protocol (RSTP) for all switch ports. We discuss these protocols further in subsequent chapters.

Continued on next page.

Power over Ethernet

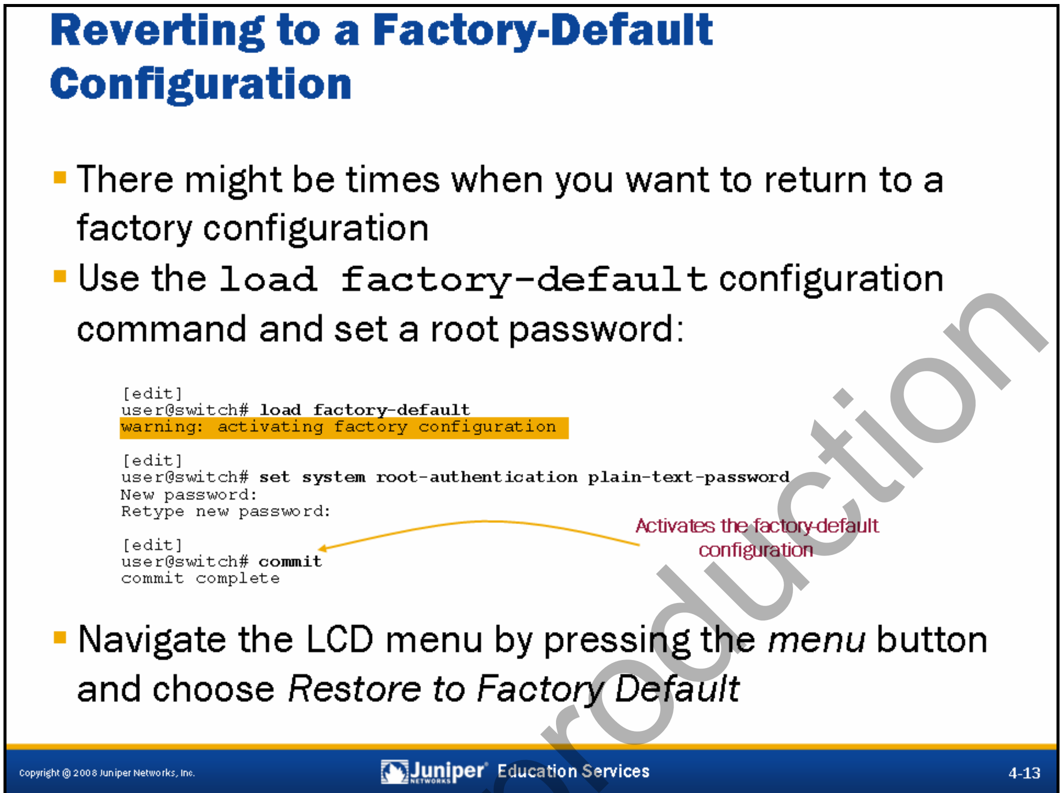
The factory-default configuration also enables Power over Ethernet (PoE) on the maximum number of supported switch ports. The maximum number of switch ports that can support PoE is dependant upon the model of EX-series switch.

Automated Factory-Default Features

If you view the factory-default configuration before committing any changes, you will notice `factory-settings` options under the `system commit` hierarchy:

```
[edit]
user@switch# show system commit
factory-settings {
    reset-chassis-lcd-menu;
    reset-virtual-chassis-configuration;
}
```

These options are removed once you commit changes to the factory-default configuration. The `reset-chassis-lcd-menu` option initializes the LCD menu for the J-Web EZsetup feature. The `reset-virtual-chassis-configuration` option resets the Virtual Chassis system member ID to 0. We cover Virtual Chassis technology in a subsequent chapter.



Returning to a Factory-Default Configuration

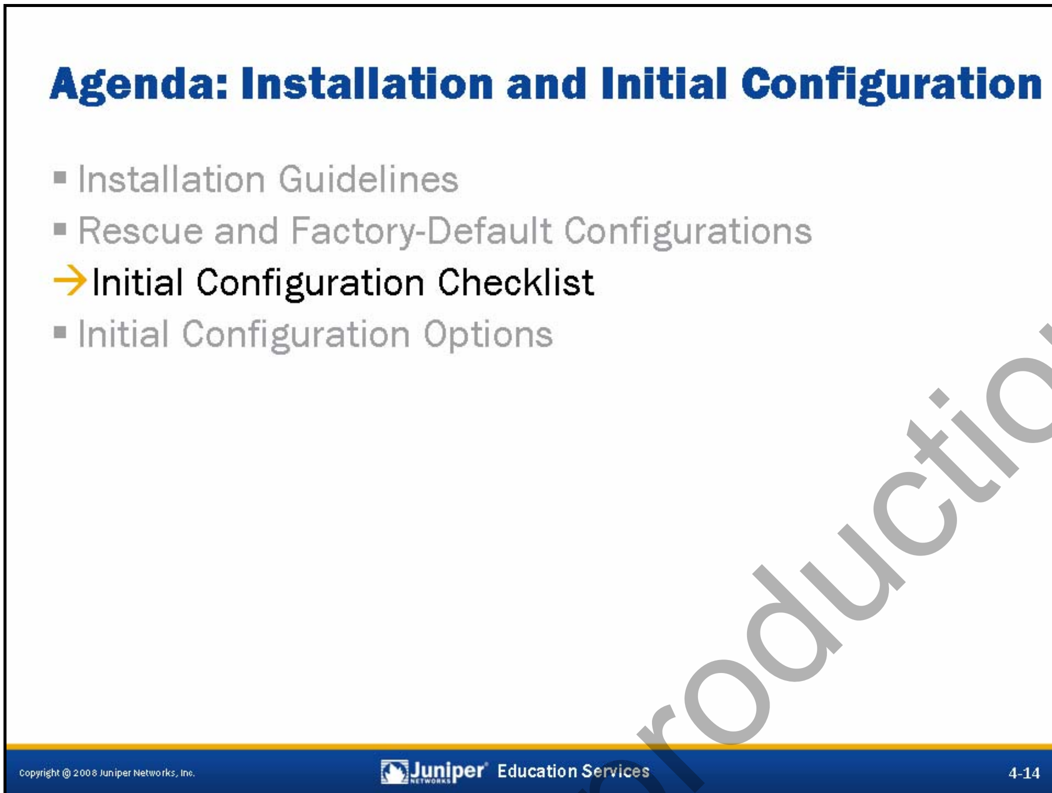
Under certain conditions, you might want to return the switch to its factory-default configuration. For example, you might want to reactivate EZsetup or simply clear the configuration to prepare the switch for redeployment in a new role.

Using the CLI

The CLI's configuration mode allows you to overwrite the candidate configuration with the factory-default configuration by using the **`load factory-default`** command. JUNOS software does not allow you to save the configuration until you configure root authentication information. Do not forget to issue a **`commit`** to activate your changes.

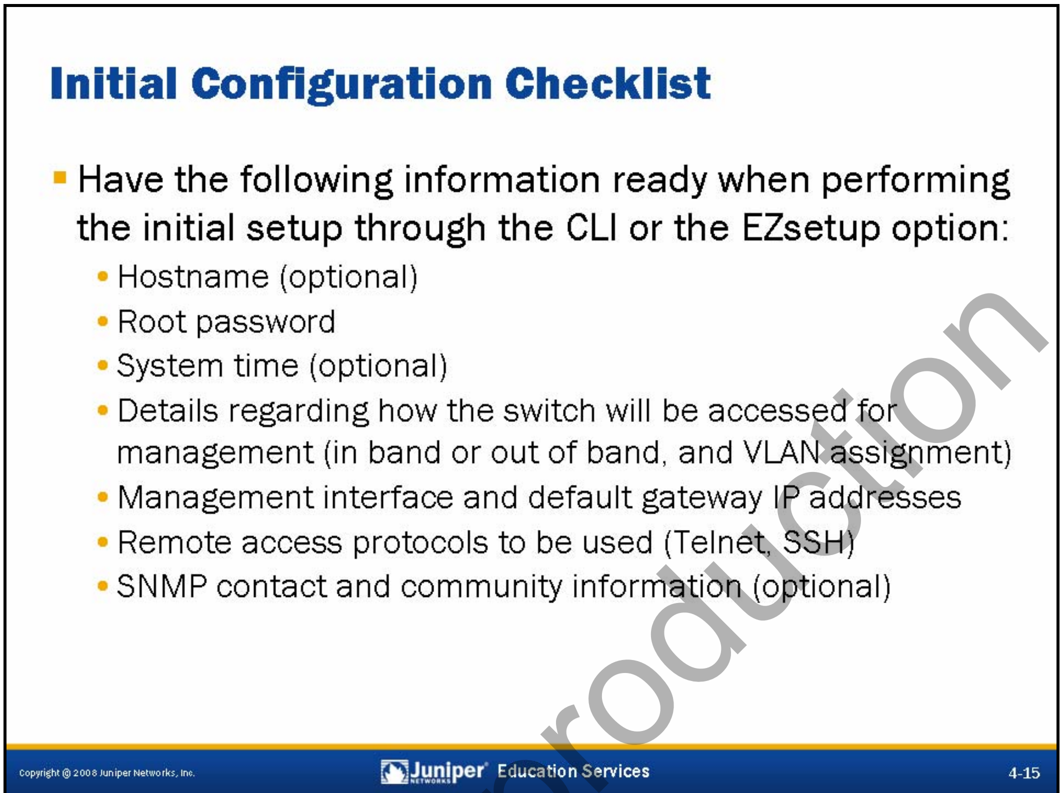
Using the LCD Menu

Press the *Menu* button next to the top right of the LCD to put the LCD in *Navigation Mode*. Use this same button to select options in the menu. The bottom button allows you to scroll through the menu. Choose *Restore to Factory Default*; the EX-series switch will load and commit the factory-default configuration.



Configuration Checklist

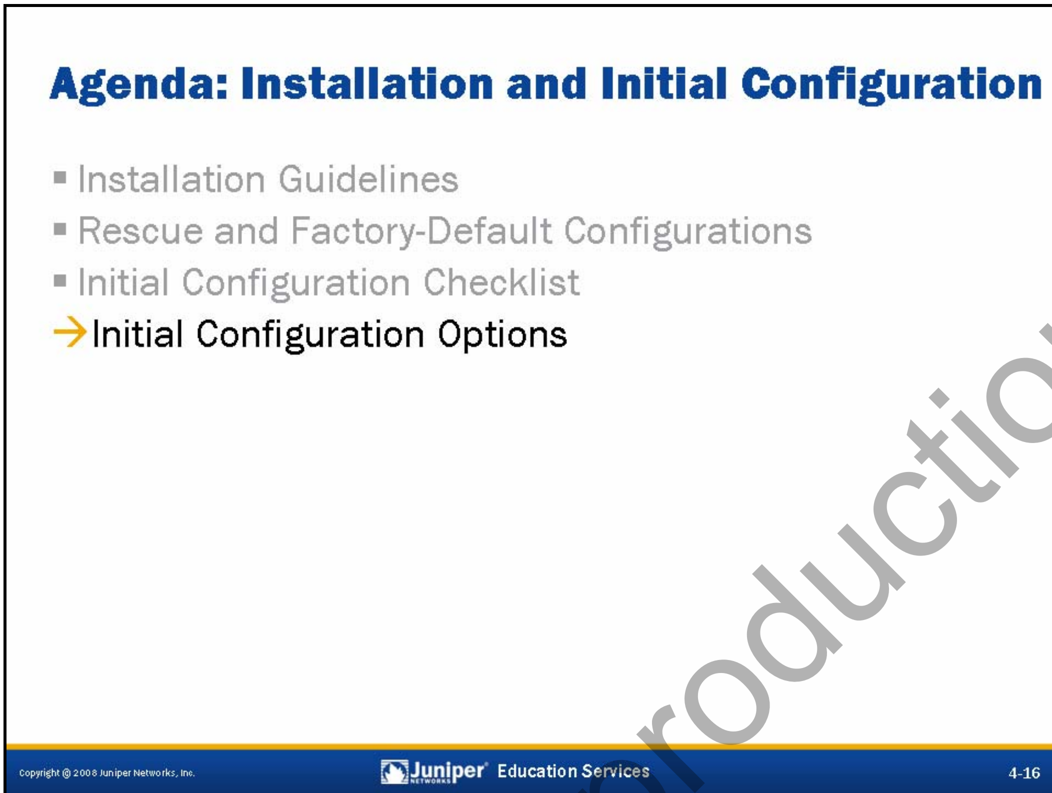
The slide highlights the topic we discuss next.



Initial Configuration

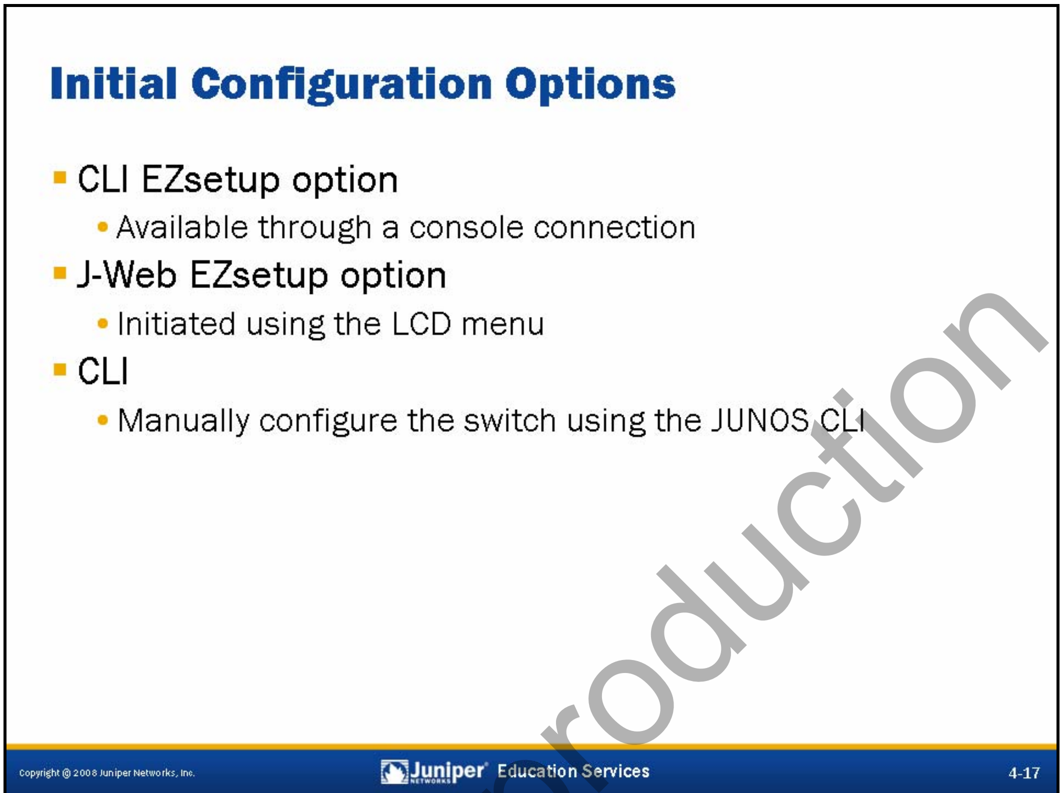
When you receive a Juniper Networks EX-series switch, the JUNOS software is preinstalled. Once you power on the switch, it is ready to attempt EZsetup or it is ready for manual configuration. You can configure the switch from a console connected to the switch's console port or using J-Web from a management host directly attached to the ge-0/0/0 or me0 interface initiated from the LCD menu. We recommend that you configure the following items at installation time:

- Hostname of the switch;
- Root password (By default, only the root user can access the switch.);
- Time of day/Network Time Protocol (NTP) server;
- Switch management details (You will be prompted to choose if you want to manage the switch in band using the default VLAN, in band by creating a new management VLAN, or out of band using the me0 management Ethernet interface.);
- Management interface IP address and gateway IP address for management network;
- System services for remote access (Telnet, SSH, and HTTP/HTTPS); and
- SNMP contact and community name.



Initial Configuration Options

The slide highlights the topic we discuss next.



CLI EZsetup

When an EX-series switch is loaded with the factory-default configuration, you can initiate the EZsetup wizard from the shell prompt by typing **`ezsetup`**. You must use a console connection for this option.

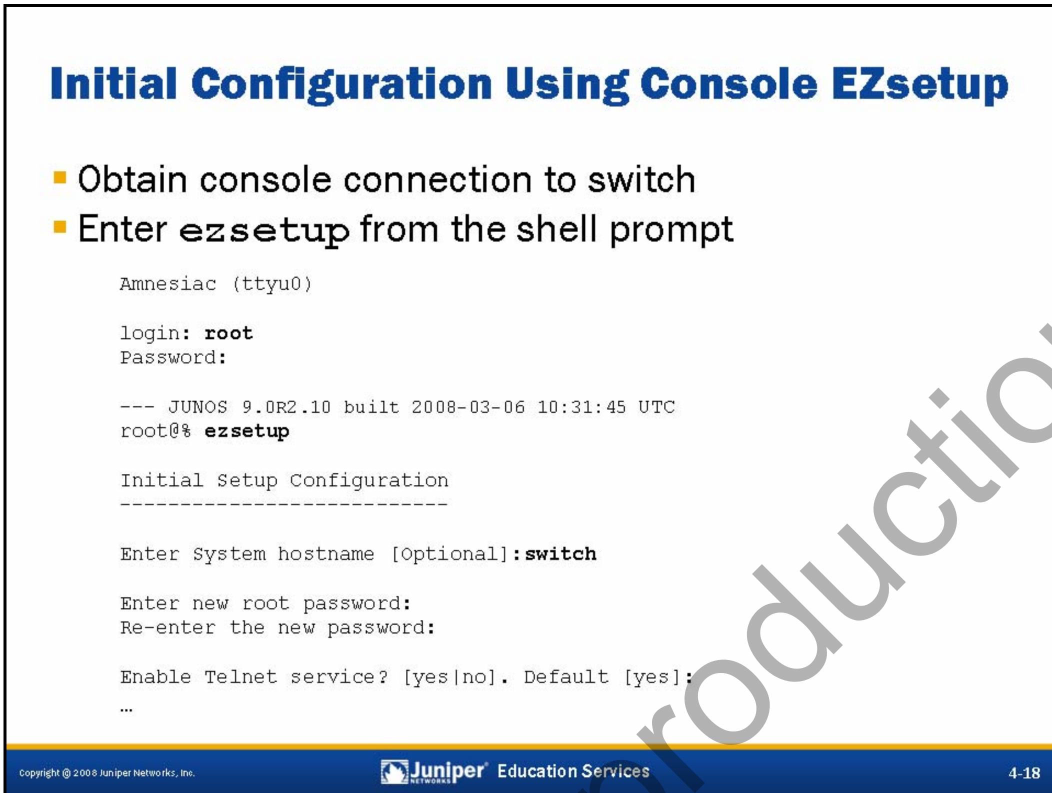
J-Web EZsetup

You initiate the J-Web EZsetup Initial Configuration Wizard through the LCD menu by choosing *Enter EZsetup*. You must use a DHCP-enabled device connected to either port ge-0/0/0 or me0 and use a Web browser for this option.

Manual CLI Configuration

You can also perform the initial configuration manually with the JUNOS CLI. You must initiate a console connection for this option because Telnet, SSH, and IP addressing are not preconfigured on the switch.

We discuss these initial configuration options in more detail on the next series of slides.



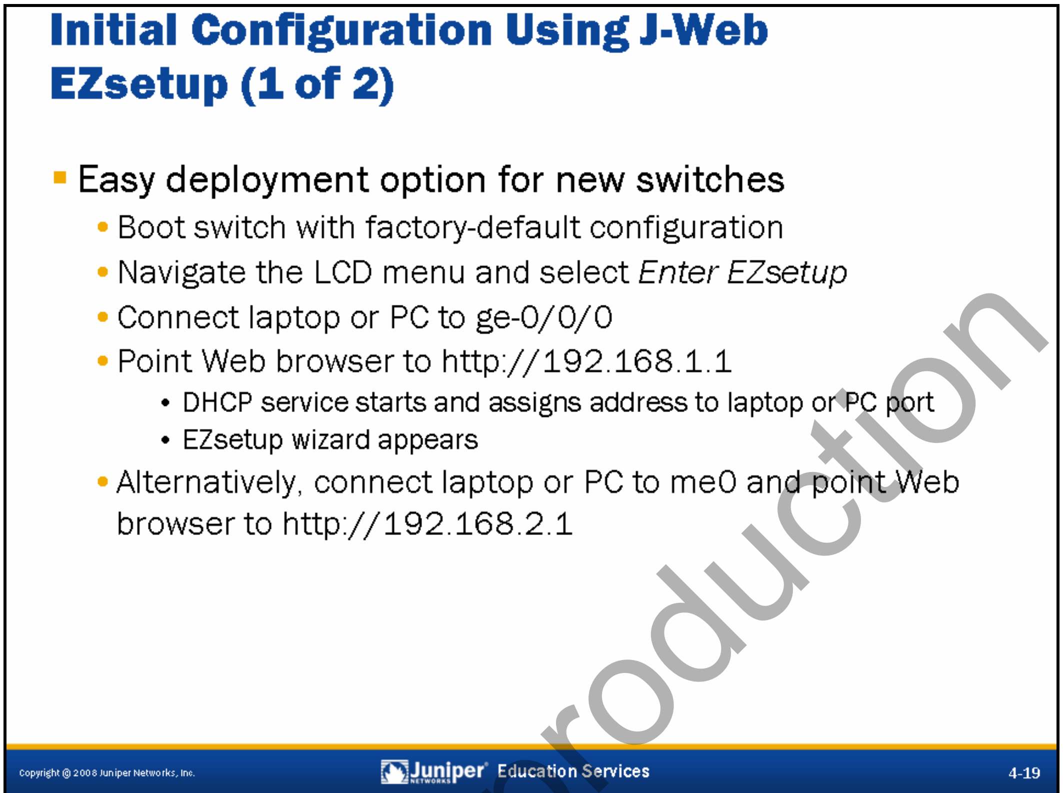
Console Connection

To use the CLI EZsetup initial configuration wizard, you must establish a console connection to the switch. Use the supplied console RS-232 connector with a standard RJ-45 Ethernet cable to connect your laptop or PC to the switch. Use a standard terminal emulation client using a 9600 Baud rate with an 8/N/1 data bits/parity/stop bits setting and VT100 terminal type. The terminal type is configurable once a session is established, but the other terminal settings are not configurable.

EZsetup Initial Configuration Wizard

Once you establish a console session with the switch, JUNOS software prompts you for a username. The factory-default username is `root` and is granted all privileges for switch access. No password is needed when the switch is in its factory-default state. However, the EZsetup wizard and any subsequent commits require the root password to be defined.

After logging in, you are presented with a UNIX shell prompt (`%`). At this prompt, type **`ezsetup`** and press Enter to begin the EZsetup wizard. The EZsetup wizard asks a series of questions based on the aforementioned initial configuration checklist. When completed, the EZsetup wizard overrides the factory-default configuration and activates your new configuration based on the parameters you entered. At this point, you can log in as the root user and configure secondary configuration options such as new user accounts, protocols, and interface properties. After the initial setup, you can invoke the console EZsetup feature by first placing the switch in a factory-default state using the LCD menu.



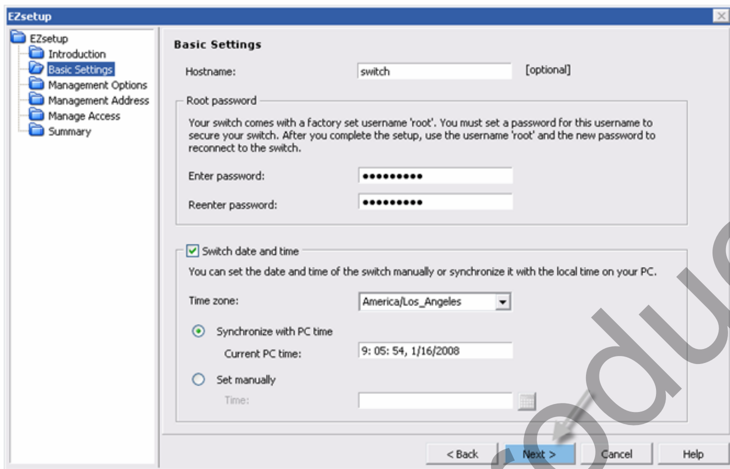
Initial Configuration Using J-Web

Thanks to the built-in initial configuration wizard, using the J-Web interface for initial configuration is extremely easy. With the switch in its factory-default state, choose the *Enter EZsetup* option using the LCD menu on the front of the switch chassis. Then, simply use an RJ-45 cable to directly connect a DHCP-configured management host to either the ge-0/0/0 or the me0 management Ethernet interface. The switch configures the ge-0/0/0 interface with an IP address of 192.168.1.1 and the me0 interface with an IP address of 192.168.2.1. It also acts as a DHCP server on these interfaces, assigning IP addresses in the 192.168.1.0/24 and 192.168.2.0/24 networks, respectively.

Point the Web browser on your management host to http://192.168.1.1 or http://192.168.2.1, depending on to which interface you are connected; you are automatically directed to the J-Web EZsetup initial configuration wizard.

Initial Configuration Using J-Web EZsetup (2 of 2)

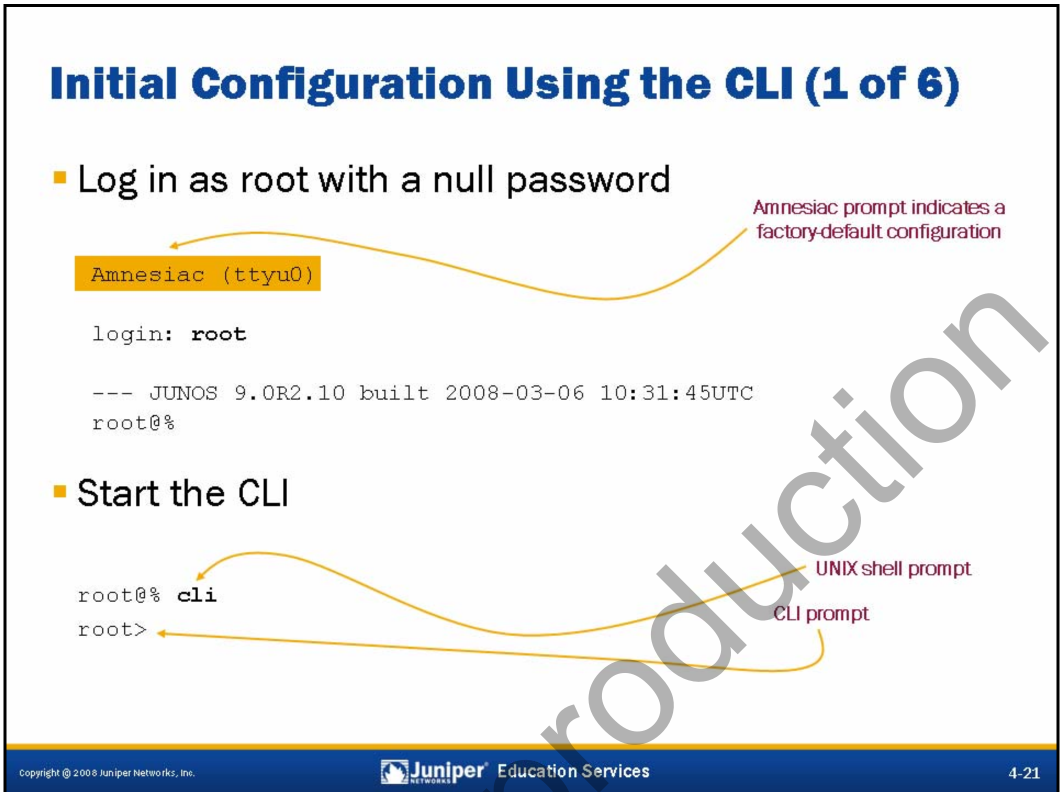
- EZsetup in J-Web automates initial management configuration options



 Juniper Education Services 4-20

Automating Initial Management Configuration with J-Web EZsetup

The J-Web EZsetup feature walks you through a series of options designed to get your switch up and running. Using the aforementioned initial configuration checklist, complete each page and click `Next`. Once you click `Finish` on the last screen, EZsetup delivers your new configuration to the switch, and it is then ready for deployment or for more advanced configuration. The J-Web EZsetup feature uses a 10-minute time limit. If you do not complete the setup within 10 minutes, switch access is revoked and the configuration reverts to the factory-default configuration.



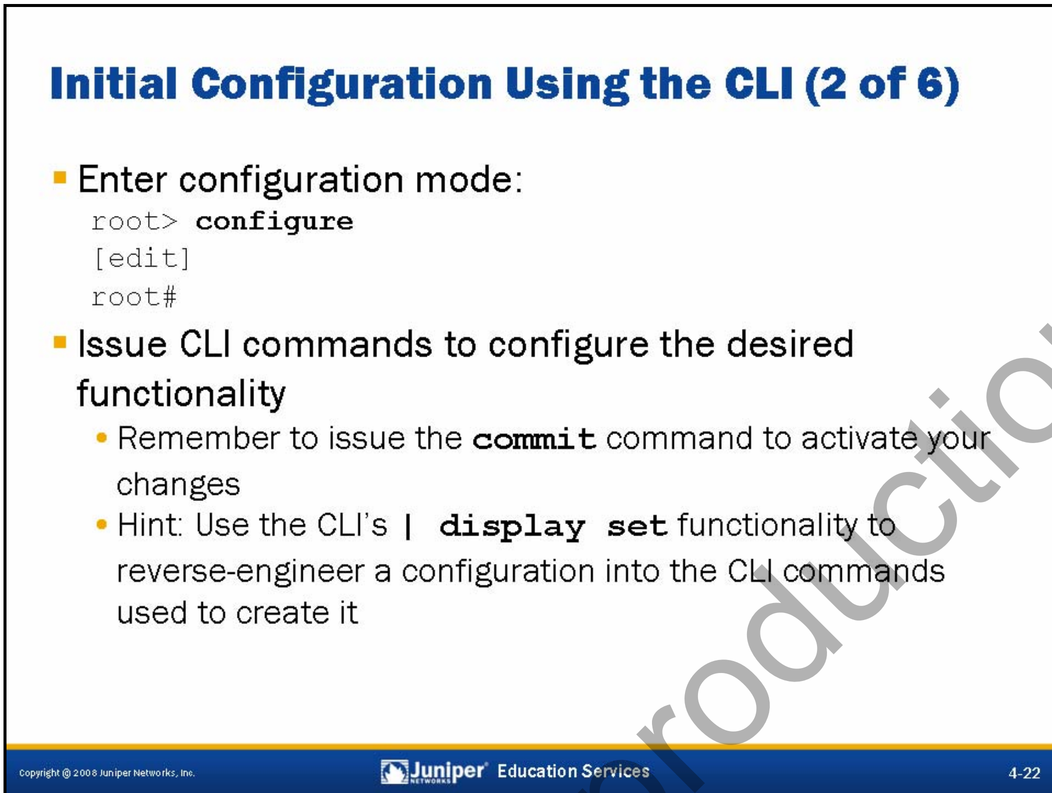
Logging In as Root

Remember when you receive a EX-series platform from the factory, the root password is not set. To log in to the switch's CLI for the first time, you must log in through the console port using the root username with no password. If you already performed the initial switch configuration with EZsetup or J-Web, you use the root password you defined at that time.

The console login normally displays the switch's configured hostname. When no hostname is configured, such as is the case with a factory-default configuration, `Amnesiac` is displayed in place of the hostname.

Starting the CLI

When you log in as the root user, you are placed at the UNIX shell. You must start the CLI by typing the **`cli`** command. When you exit the CLI, you return to the UNIX shell. For security reasons, make sure you also log out of the shell using the **`exit`** command.



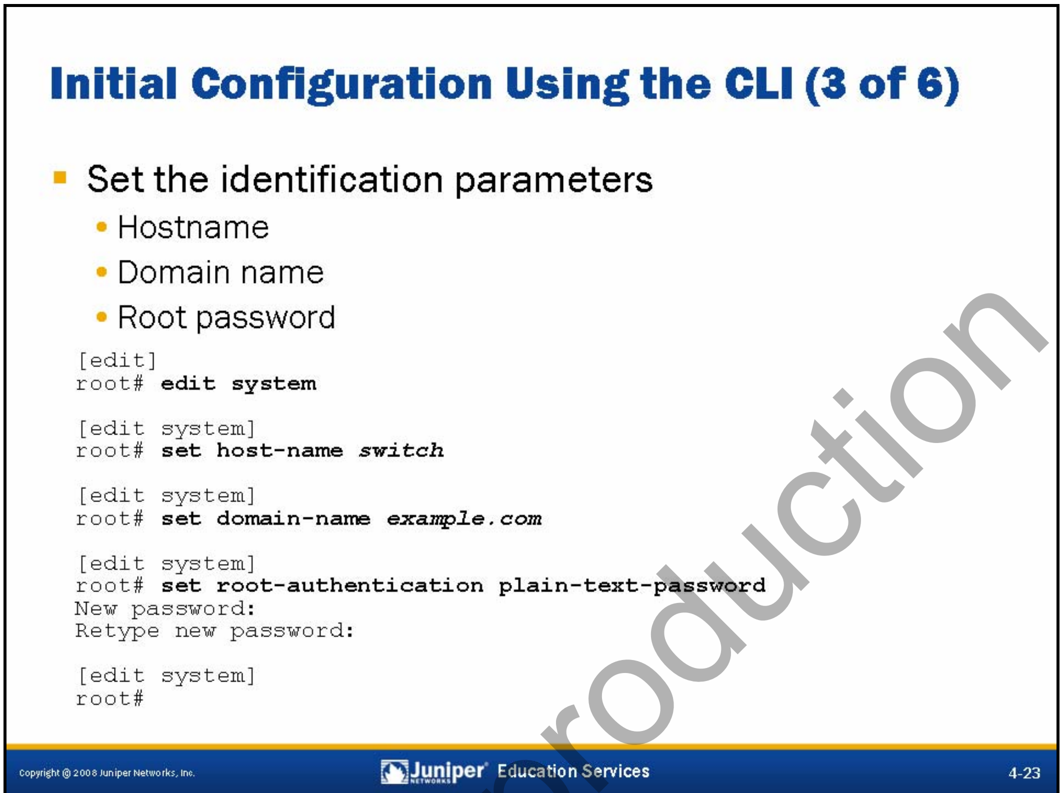
Entering Configuration Mode

After starting the CLI, you enter operational mode. You can make changes to the configuration only in configuration mode. Enter configuration mode by entering the command **`configure`** at the operational-mode prompt, as shown on the slide.

Issuing Commands

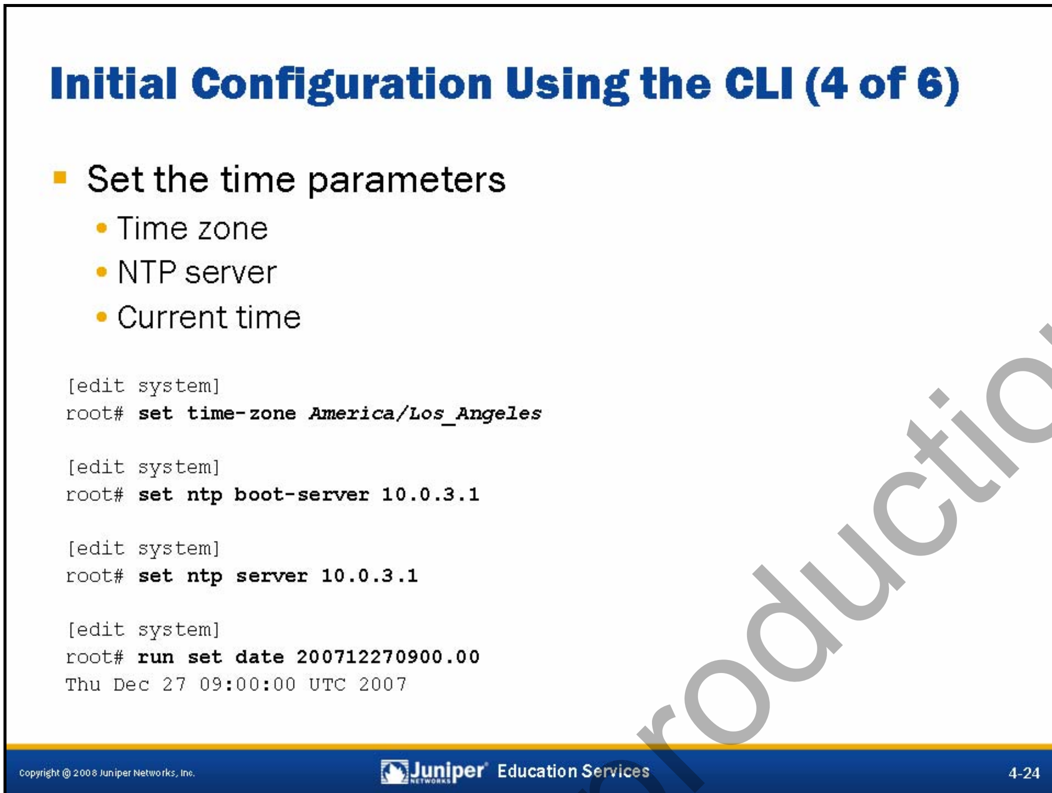
Once in configuration mode, issue **`set`** commands to configure the functionality you want. Remember that your changes do not take effect until you issue a **`commit`** command. To help learn CLI configuration syntax, you might try displaying a configuration with the results piped to the `display set` functionality as shown:

```
[edit]
root# show interfaces me0
unit 0 {
      family inet {
          address 192.168.2.2/24;
      }
}

[edit]
root# show interfaces me0 | display set
set interfaces me0 unit 0 family inet address 192.168.2.2/24
```



Identification Parameters

This slide shows how to use the CLI to configure the switch's hostname, domain name, and a root password. Notice that the example shows these configuration parameters entered at the `[edit system]` hierarchy rather than at the top level of the hierarchy, which makes for less typing.



Time Parameters

This slide shows how to use the CLI to configure the time settings. You can configure the switch with current date and time information where it maintains its own time or, preferably, for Network Time Protocol (NTP) synchronization. When defining NTP server parameters, specify an NTP boot server to function as an NTP reference device upon booting and an NTP server for continuous time synchronization while the switch is running. In many cases, the NTP boot server and NTP server are the same device.

Initial Configuration Using the CLI (5 of 6)

- Set the network parameters
  - DNS name servers
  - Domain search and name
  - Default gateway
  - me0 address

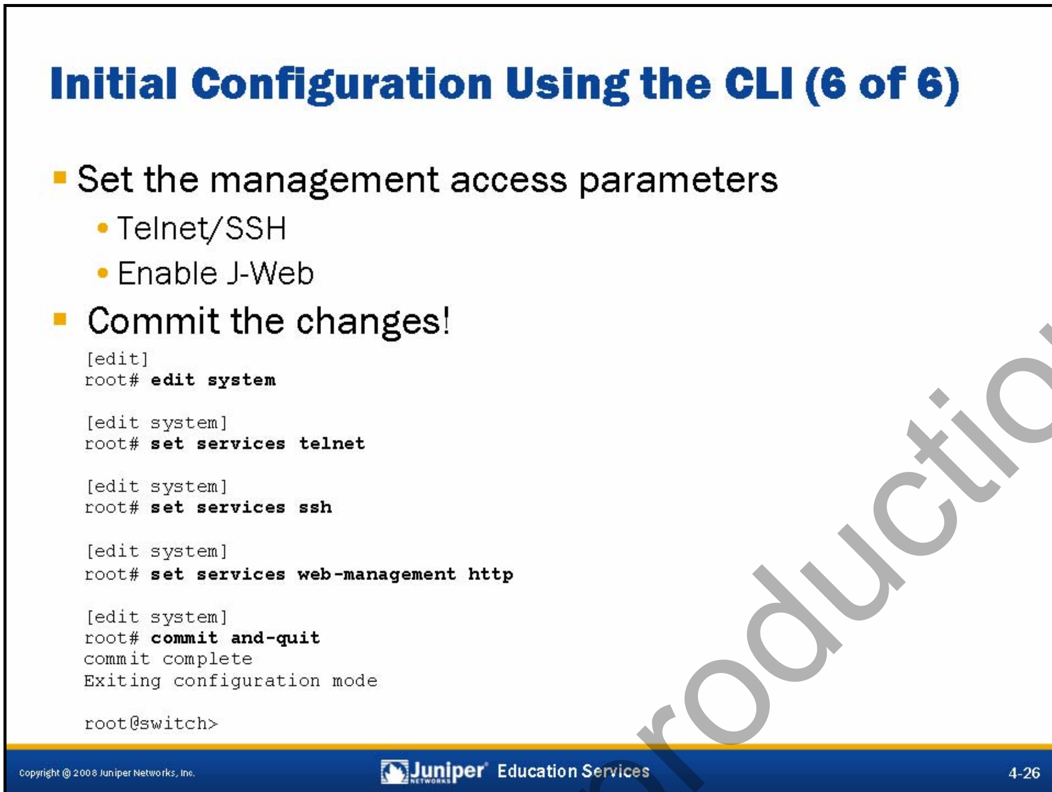
```
[edit system]
root# set name-server 10.0.2.1

[edit system]
root# set domain-search example.com

[edit system]
root# set domain-name example.com

[edit system]
root# top

[edit]
root# set routing-options static route 0.0.0.0/0 next-hop 10.0.1.254

[edit]
root# set interfaces me0 unit 0 family inet address 10.0.1.1/24
```

 Juniper Education Services 4-25

Network Parameters

This slide shows how to use the CLI to configure a DNS server, a domain name and a domain search list. JUNOS software appends the domain name entered to hostnames that are not fully qualified. JUNOS software uses the domain search list to set an order in which clients append domain names when searching for the IP address of a host. These statements are optional.

Defining a static route of 0.0.0.0/0 sets a default route to which the switch sends packets in the event that the destination is a remote host. The me0 management Ethernet port provides out-of-band management access for the switch. Alternatively, configure the virtual management Ethernet (vme) port for out-of-band management access to a Virtual Chassis device. We cover Virtual Chassis technology in more detail in a subsequent chapter.



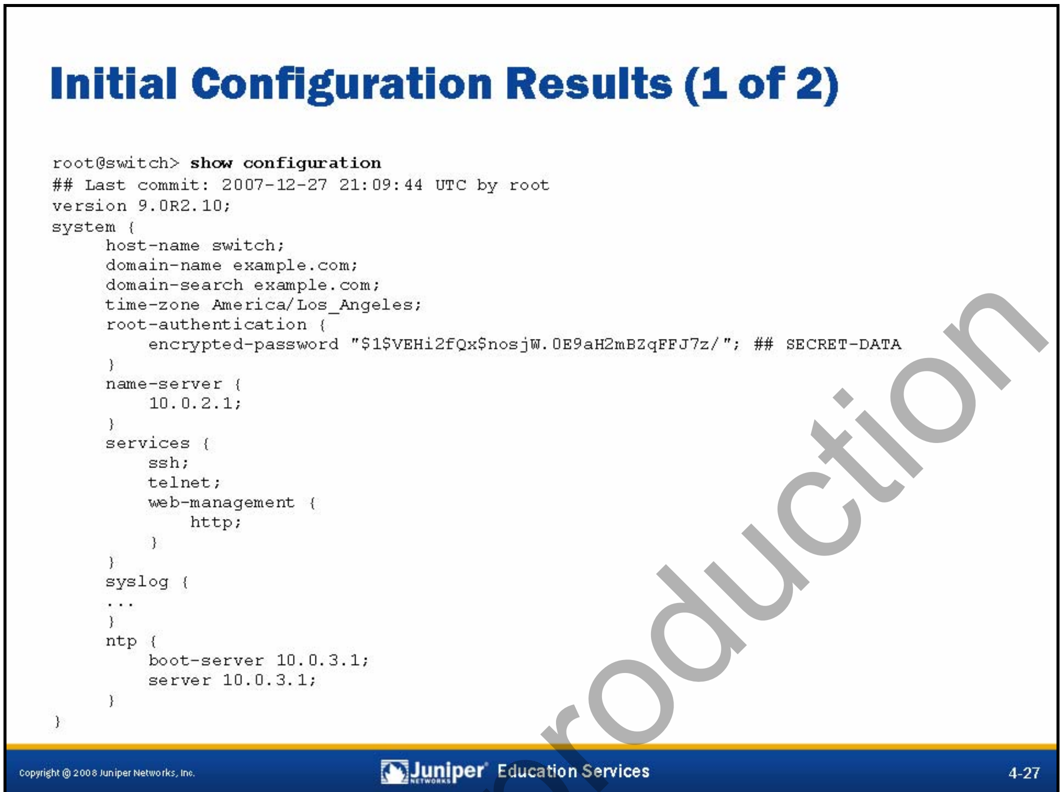
Management Access Parameters

This slide shows how to use the CLI to enable SSH, Telnet, and HTTP access to the switch. When connecting to the switch using one of these access protocols, use the same user logins defined under the `[edit system login]` hierarchy.

Applying Your Configuration

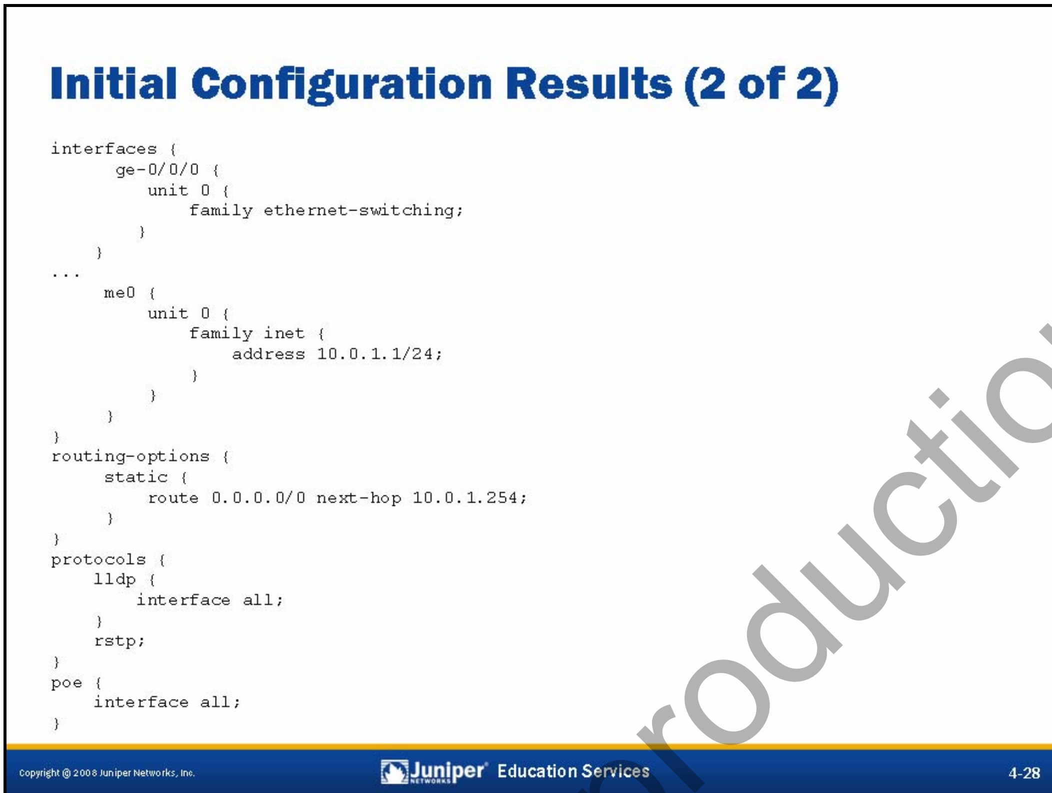
Once you complete your initial configuration, use the **`commit`** command to apply your changes. You can include the **`and-quit`** option, as shown, to return to operational mode.

Note that the EZsetup and J-Web initial configuration wizards automatically commit the new configuration upon completion.



Displaying the Initial Configuration: Part 1

The screen capture on the slide uses the operational-mode **`show configuration`** command to display the hierarchical configuration file created by our initial configuration **`set`** statements. The `syslog` hierarchy included in the factory-default configuration is suppressed for brevity.



Displaying the Initial Configuration: Part 2

This slide displays the remaining hierarchies created by our initial configuration **`set`** statements combined with the factory-default configuration. The `interfaces` hierarchy included in the factory-default configuration is suppressed for brevity.

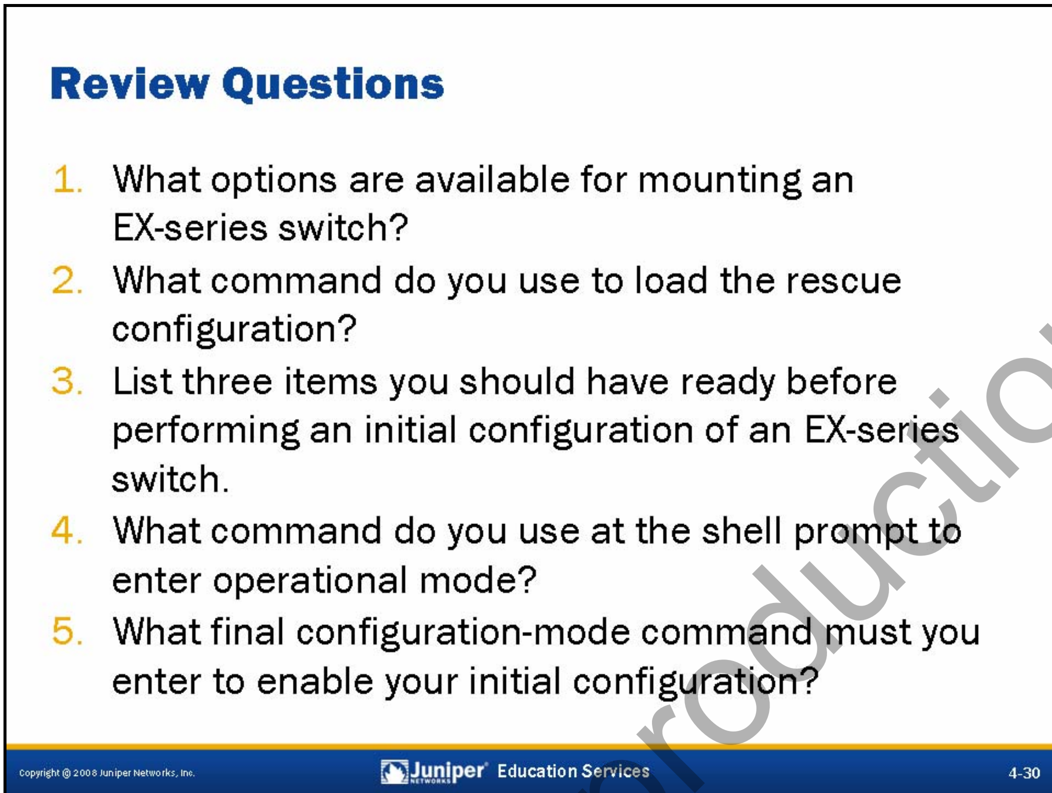
Summary

- In this chapter, we:
  - Described installation of Juniper Networks EX-series switches
  - Saved and restored a rescue configuration
  - Returned an EX-series switch to the factory-default state
  - Described the initial configuration options
  - Performed initial configuration using the console's EZsetup option
  - Performed initial configuration using J-Web and the LCD menu
  - Performed initial configuration using the CLI

 Juniper Education Services 4-29

This Chapter Discussed:

- The general process and guidelines for installing Juniper Networks EX-series switches;
- Loading a configuration file, and saving and restoring rescue configurations;
- Returning the switch to its factory-default configuration;
- Options for initial configuration of the switch;
- Performing an initial configuration using the console's EZsetup option;
- Performing an initial configuration using J-Web and the LCD menu and;
- Performing an initial configuration manually using the CLI.



Review Questions

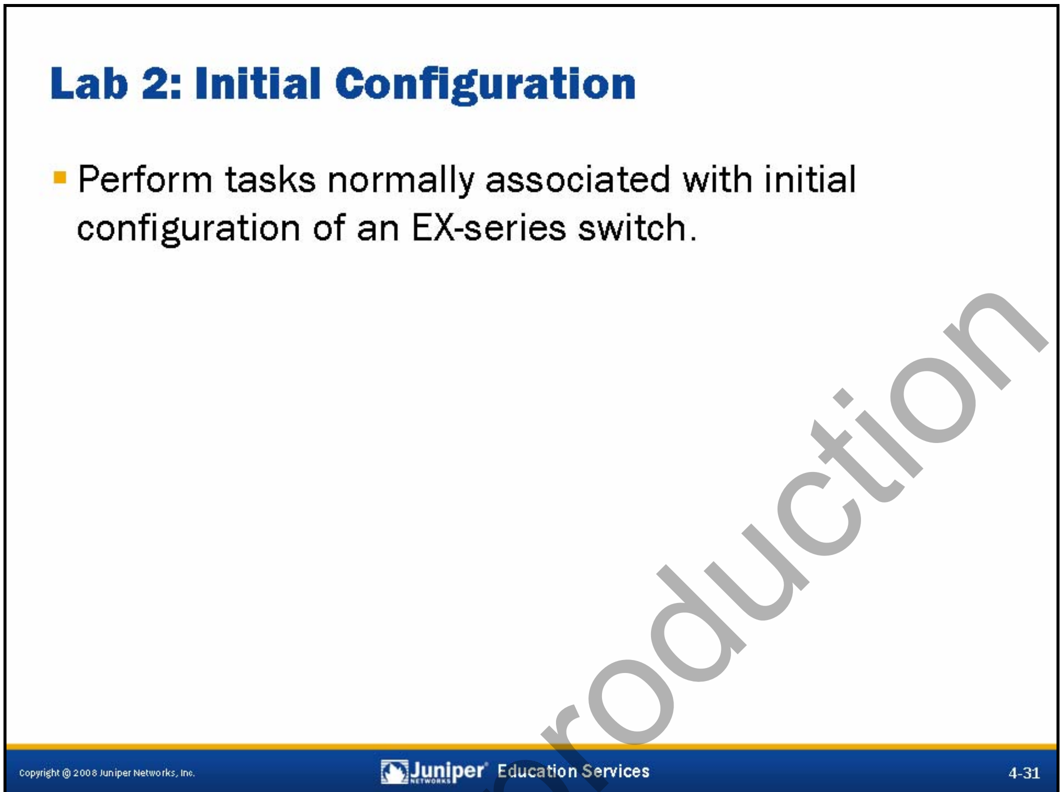
1.

2.

3.

4.

5.

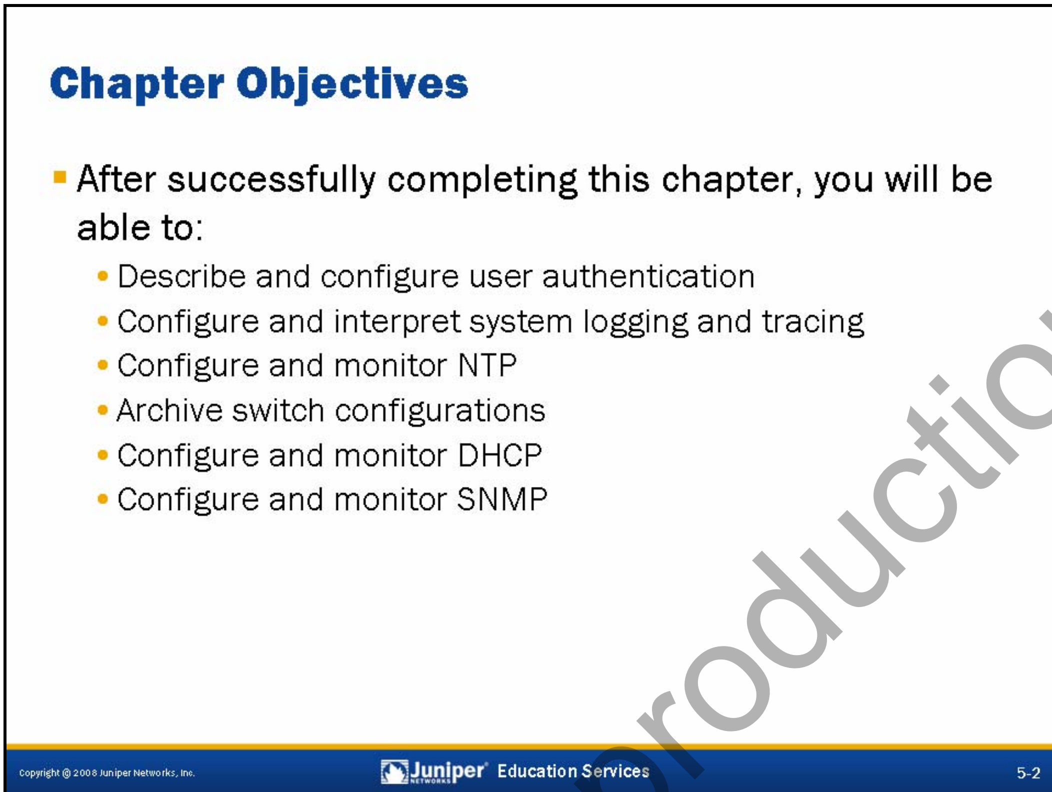


Lab 2: Initial Configuration

The slide provides the objective for this lab.

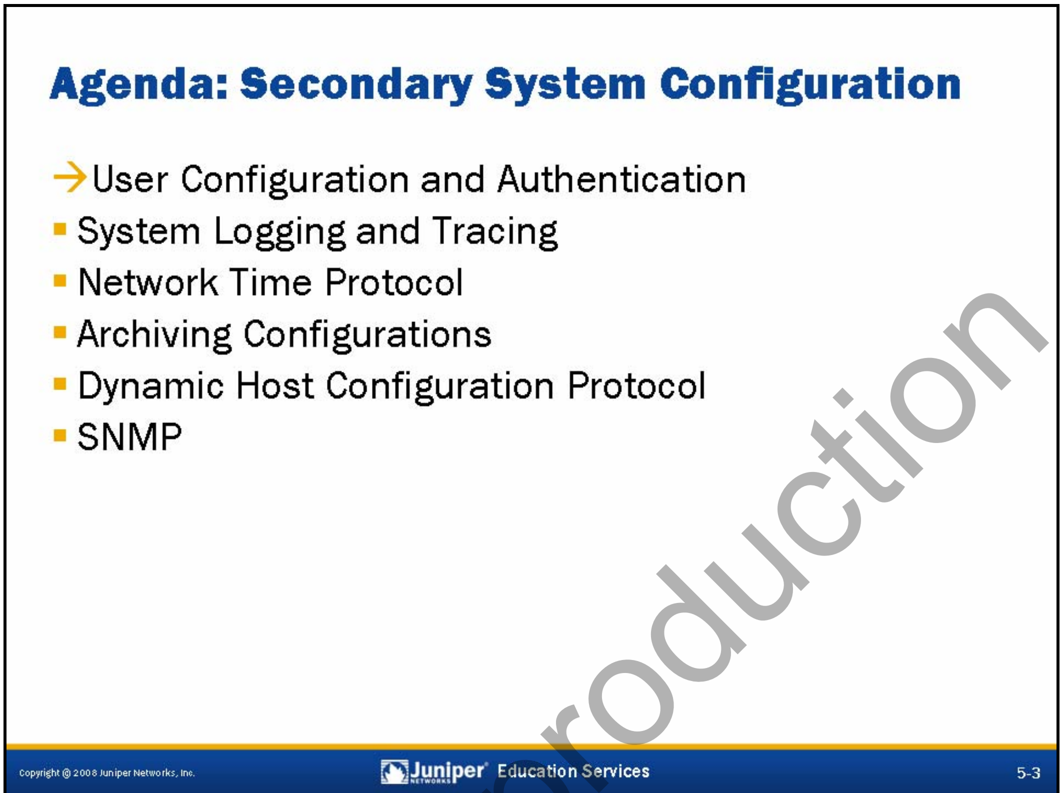
Operating Juniper Networks Switches in the Enterprise

Chapter 5: Secondary System Configuration



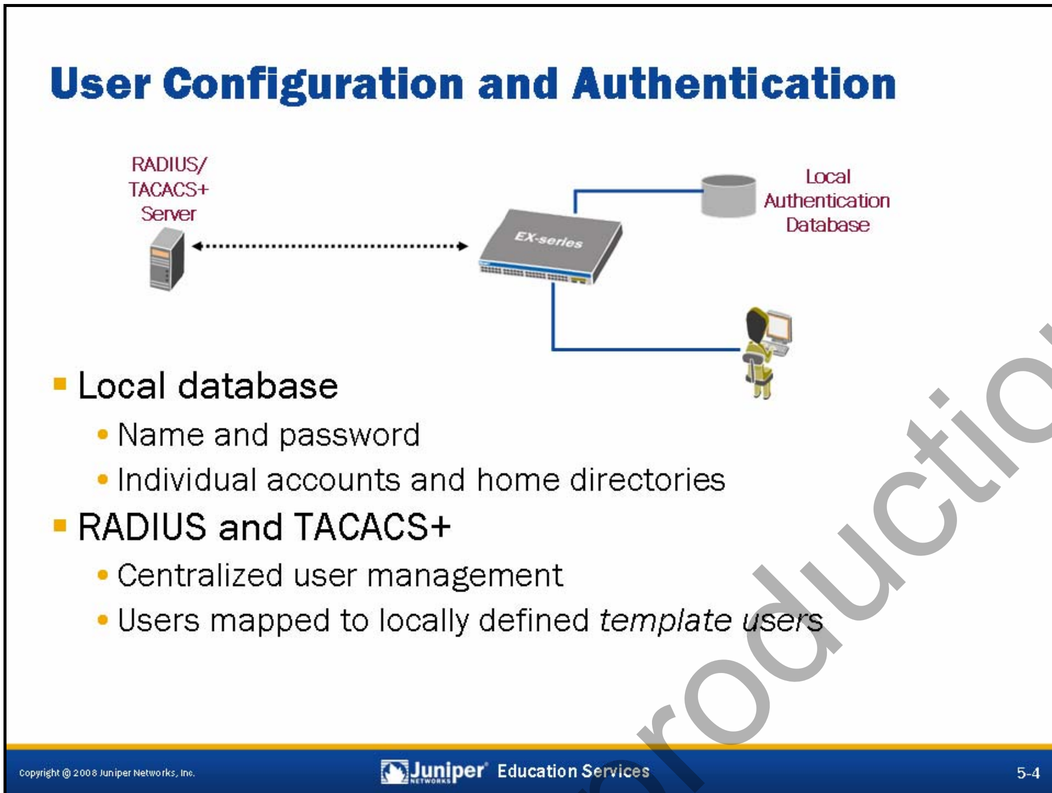
This Chapter Discusses:

- User authentication methods and configuration;
- Configuring and interpreting system logging and tracing;
- Network Time Protocol (NTP) configuration and operation;
- Archiving switch configurations on remote devices;
- Configuring and monitoring the Dynamic Host Configuration Protocol (DHCP); and
- Configuring and monitoring the SNMP.



User Configuration and Authentication

The slide lists the topics we cover in this chapter. We discuss the highlighted topic first.



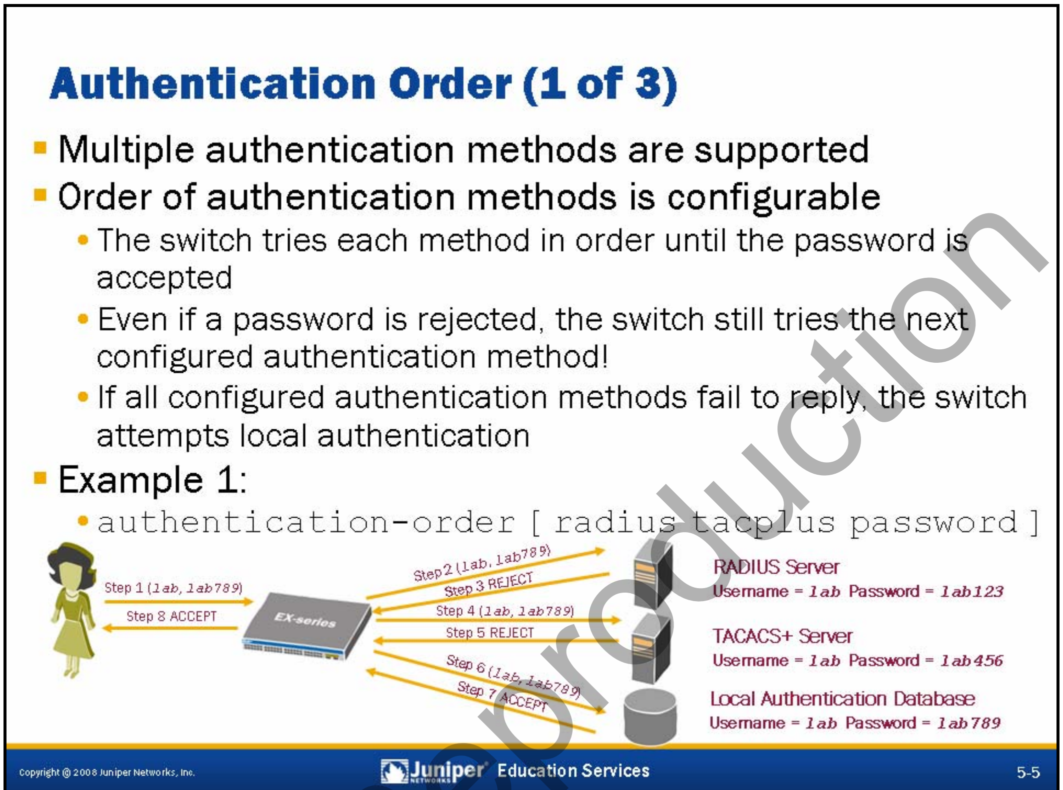
Local Password Authentication

With local password authentication, you can configure usernames and passwords individually for each user to log in to the switch. JUNOS software enforces the following password restrictions:

- The password must be at least 6 characters.
- You can include most character classes in a password (alphabetic, numeric, and special characters), except control characters.
- Valid passwords must contain at least one change of case or character class.

RADIUS and TACACS+

RADIUS and TACACS+ are authentication methods used for validating users who attempt to access the switch. They are both distributed client/server systems. The RADIUS and TACACS+ clients run on the Juniper Networks EX-series switch; the server runs on a host connected to a remote network. Both protocols allow for user authentication. A locally defined user account determines authorization. Multiple RADIUS or TACACS+ authenticated users can be mapped to a locally defined user account. These local accounts are referred to as *template users* and avoid the need for each RADIUS or TACACS+ user to also have a locally defined user account. With the appropriate Juniper Networks extensions loaded on the server, both RADIUS and TACACS+ can override these template user authorization parameters by passing extended regular expressions. Coverage of regular expressions is outside the scope of this class.



Multiple Authentication Methods

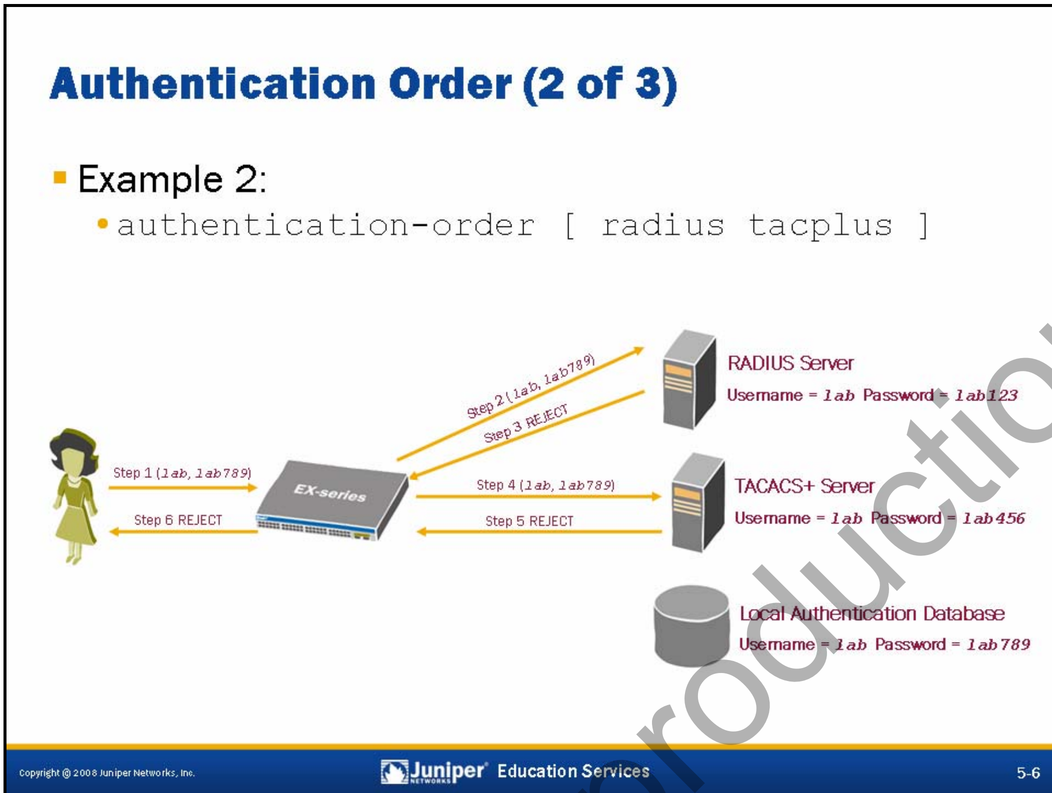
You can configure the switch to be both a RADIUS and TACACS+ client, and you can prioritize the order in which the software tries one or more of the three different authentication methods.

Authentication Order

For each login attempt, JUNOS software tries the authentication methods in order, until the password is accepted. The next method in the authentication order is consulted if the previous authentication method failed to reply or if the method rejected the login attempt. If no reply (accept or reject) is received from any of the listed authentication methods, JUNOS software consults local authentication as a last resort.

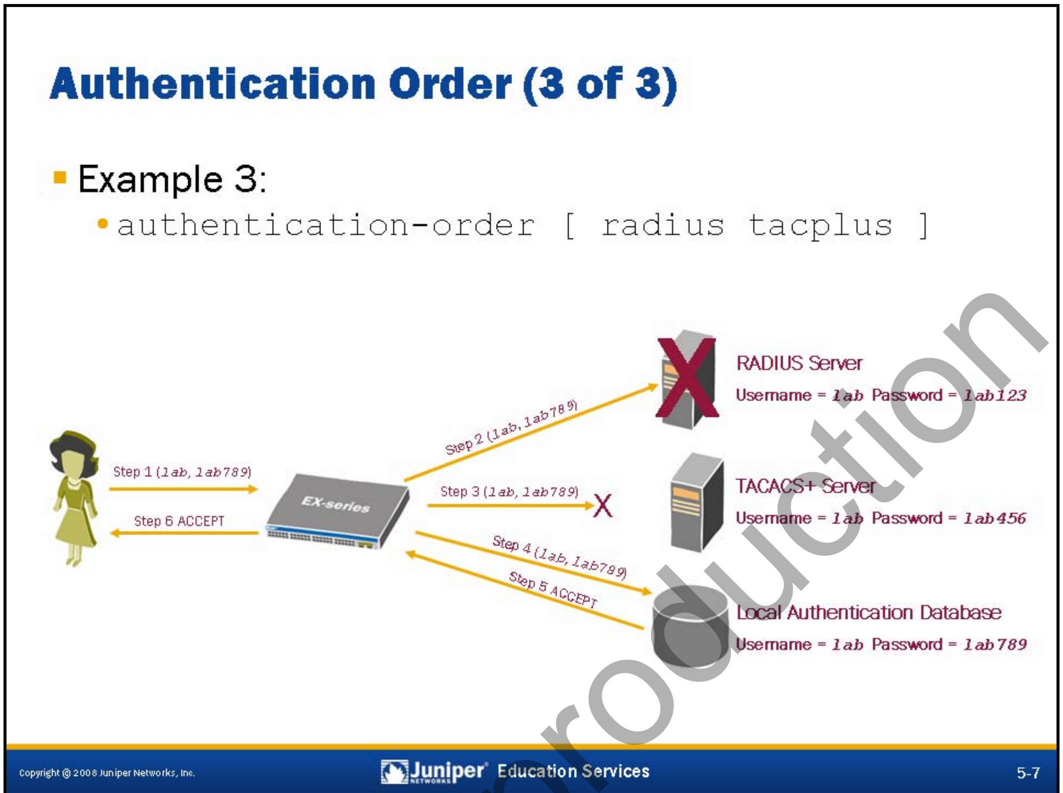
Example 1

In the example shown on the slide, we configured `authentication-order [ radius tacplus password ]`. We enter a username of *lab* and a password of *lab789*. We are successfully authenticated because each configured authentication method is attempted until the password is accepted by the local authentication database.



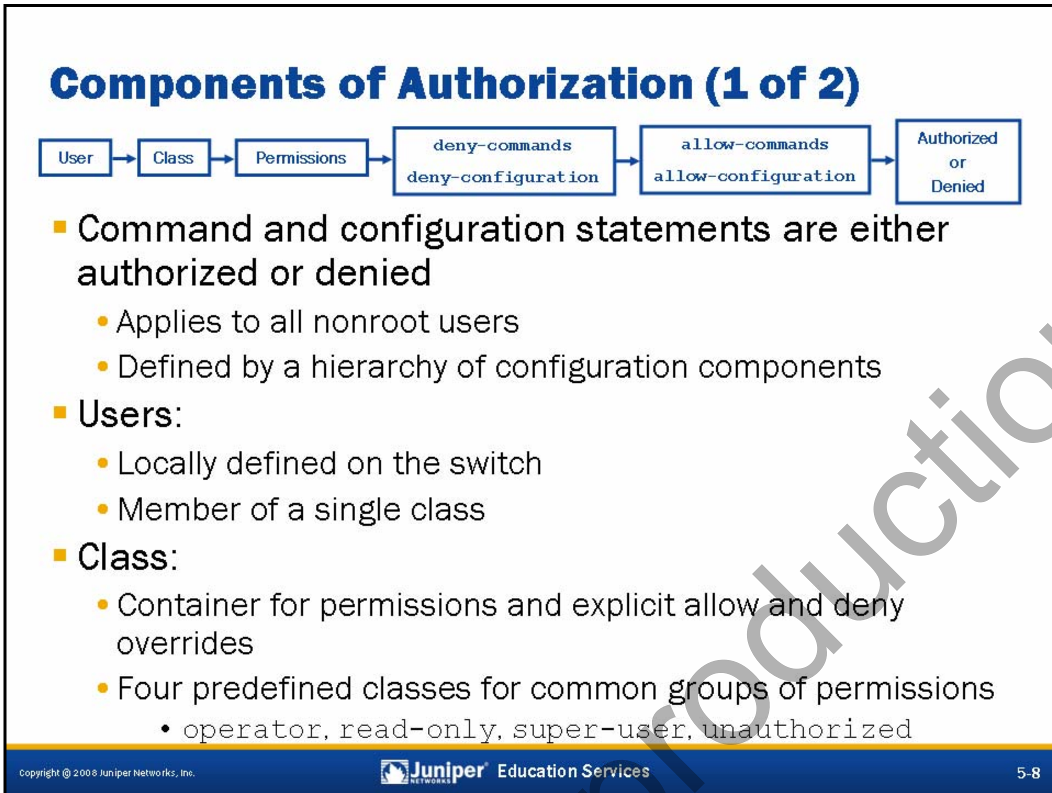
Example 2

In this example, we configured `authentication-order [ radius tacplus ]`. We enter a username of *lab* and a password of *lab789*. JUNOS software tries the password against the RADIUS server, which rejects it. It then tries it against the TACACS+ server, which also rejects it. JUNOS software does not consult local authentication because it is not listed in the authentication order, and at least one of the configured authentication methods did respond. The password is rejected.



Example 3

In this example, `authentication-order [ radius tacplus ]` is still configured. We enter a username of *`lab`* and a password of *`lab789`*. JUNOS software tries the password against the RADIUS server, which is down. The switch receives no response, and after a timeout period, tries the TACACS+ server. A temporary network problem causes the TACACS+ server to be unreachable. After a timeout period, local authentication is consulted and the password is accepted. JUNOS software consults local authentication because none of the configured authentication methods responded.



Authorization Overview

Each command or configuration statement is subject to authorization. The switch applies authorization to all nonroot users, and you cannot disable this feature. Authorization applies to both the J-Web interface and the command-line interface (CLI). A configured hierarchy of authorization components, as shown by the graphic on the slide, defines whether or not a command is authorized.

Users

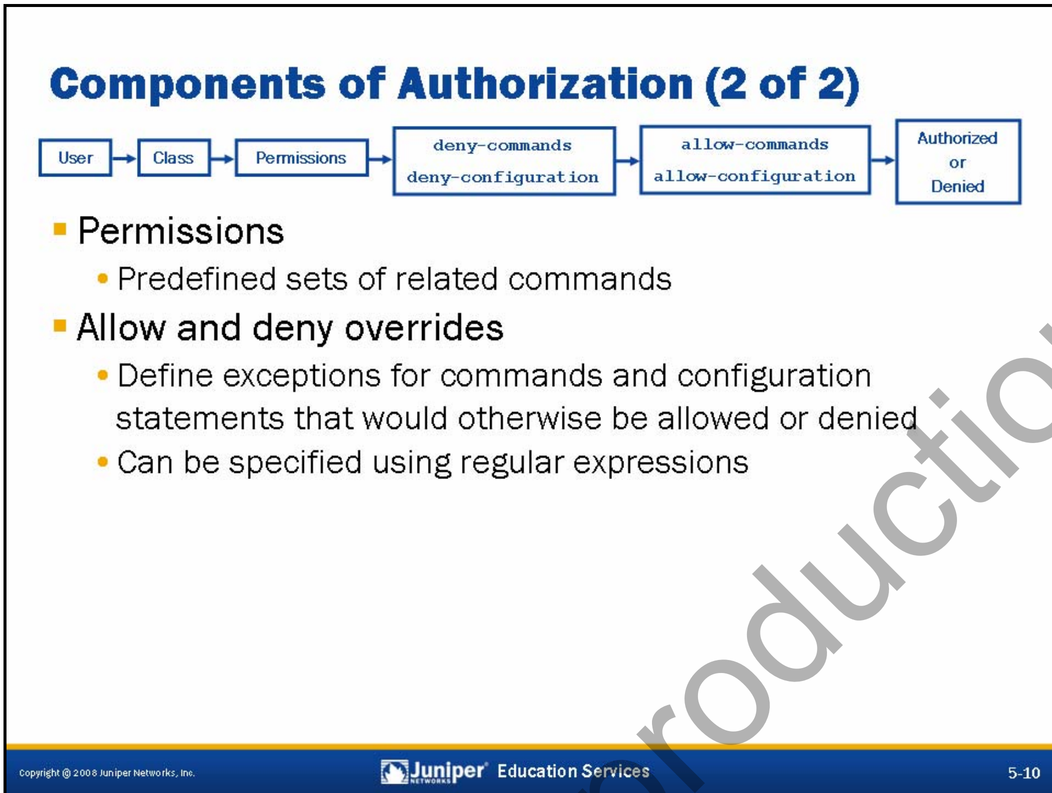
At the highest level, the configuration of user accounts on the switch defines authorization parameters. Multiple remotely authenticated users can be mapped to a locally defined template user. Users are members of a single login class.

Continued on next page.

Class

A login class is a named container that groups together a set of one or more permission flags. Login classes can also specify that the permission flags should be overridden for certain commands. You can configure custom login classes, but there are four predefined login classes that exist to handle most situations. These classes and associated permission flags are the following:

- `super-user`: All permissions;
- `operator`: Clear, network, reset, trace, and view permissions;
- `read-only`: View permissions; and
- `unauthorized`: No permissions.



Permissions

The following predefined permission flags group together the authorization of related commands:

- `access`: Allows the viewing of network access configuration;
- `access-control`: Allows the modifying of network access configuration;
- `admin`: Allows the viewing of user accounts;
- `admin-control`: Allows the modifying of user accounts;
- `all`: Enables all permission bits to be turned on;
- `clear`: Allows the clearing of learned network information;
- `configure`: Allows the entering of configuration mode;
- `control`: Allows the modifying of any configuration values;
- `field`: Is reserved for field (debug) support;
- `firewall`: Allows the viewing of firewall configuration;
- `firewall-control`: Allows the modifying of firewall configuration;
- `floppy`: Allows the reading and writing of information to the floppy drive;
- `flow-tap`: Allows the viewing of flow-tap configuration;
- `flow-tap-control`: Allows the modifying of flow-tap configuration;
- `flow-tap-operation`: Enables the tapping of flows;

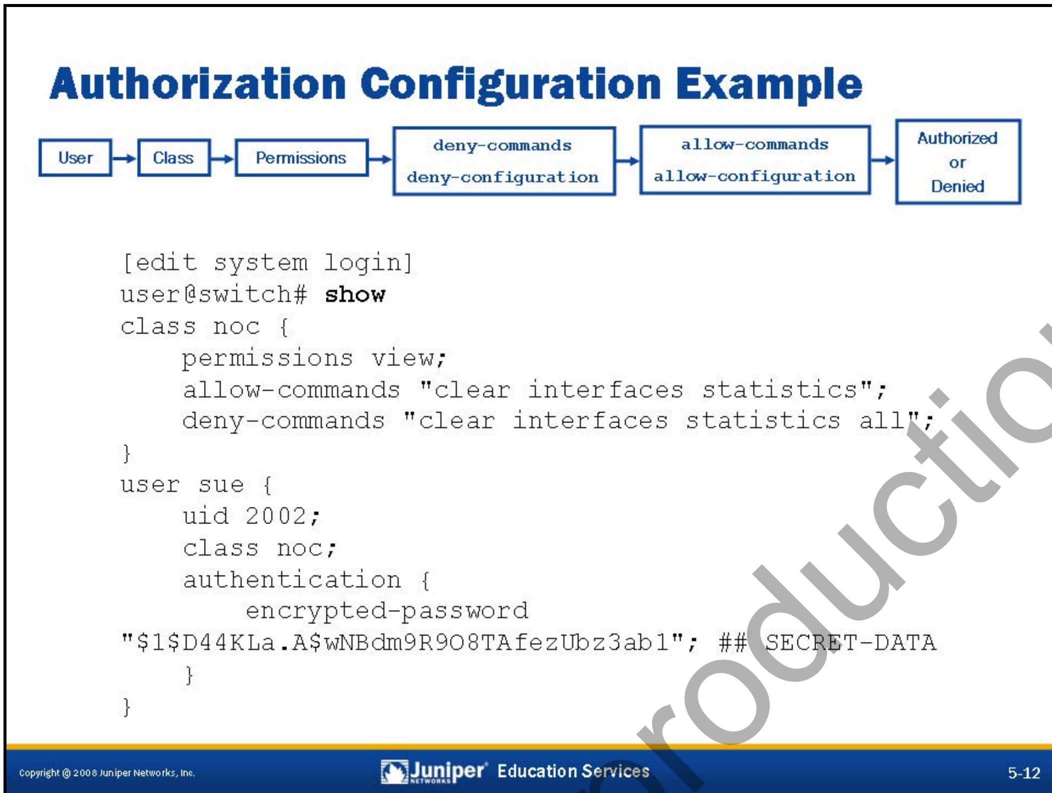
Continued on next page.

Permissions (contd.)

- `idp-profiler-operation`: Enables IDP profiler;
- `interface`: Allows the viewing of interface configuration;
- `interface-control`: Allows the modifying of interface configuration;
- `maintenance`: Allows system maintenance, including starting a local shell on the switch and becoming the superuser in the shell, and can halt and reboot the switch;
- `network`: Allows network access;
- `reset`: Allows the resetting and restarting of interfaces and processes;
- `rollback`: Allows the ability to roll back for depth greater than zero;
- `routing`: Allows the viewing of routing configuration;
- `routing-control`: Allows the modifying of routing configuration;
- `secret`: Allows the viewing of secret configuration;
- `secret-control`: Allows the modifying of secret configuration;
- `security`: Allows the viewing of security configuration;
- `security-control`: Allows the modifying of security configuration;
- `shell`: Allows the starting of a local shell;
- `snmp`: Allows the viewing of SNMP configuration;
- `snmp-control`: Allows the modifying of SNMP configuration;
- `system`: Allows the viewing of system configuration;
- `system-control`: Allows the modifying of system configuration;
- `trace`: Allows the viewing of trace file settings;
- `trace-control`: Allows the modifying of trace file settings;
- `view`: Allows the viewing of current values and statistics; and
- `view-configuration`: Allows the viewing of all configuration (not including secrets).

Allow and Deny Overrides

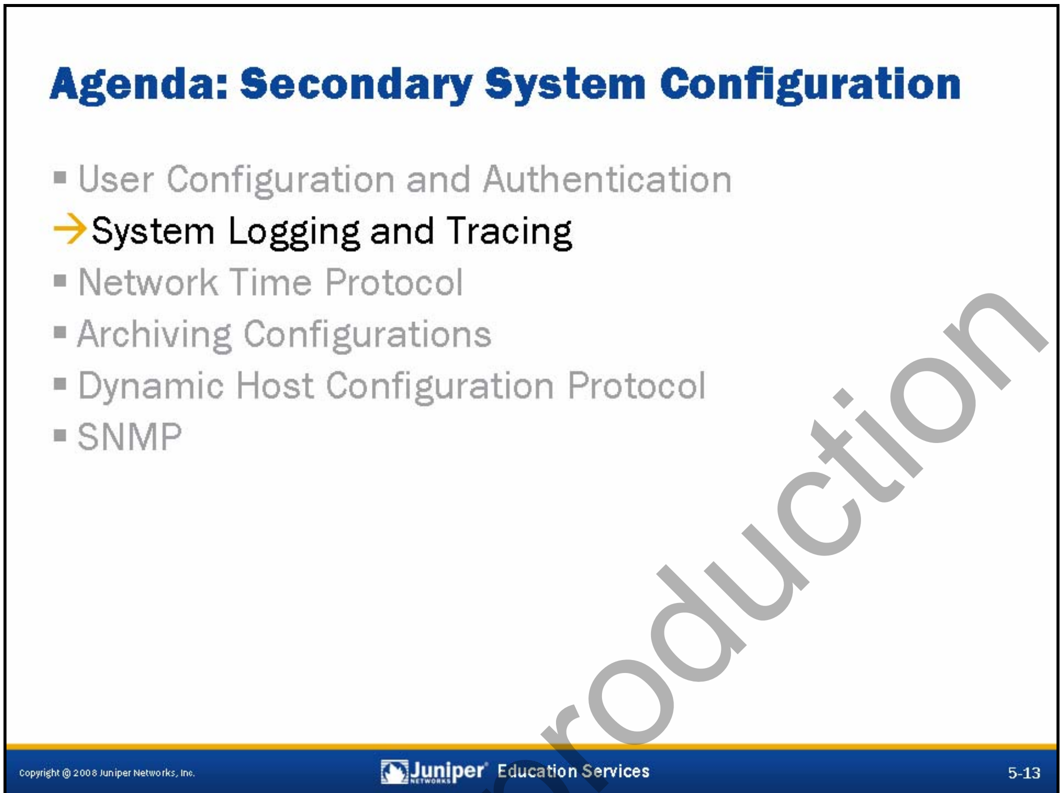
You can use the **`deny-commands`**, **`allow-commands`**, **`deny-configuration`**, and **`allow-configuration`** statements to define regular expressions that match operational commands or configuration statements. Matches are explicitly allowed or denied, regardless of whether you set the corresponding permission flags. You apply the **`deny-`** statements before the corresponding **`allow-`** statements, resulting in the authorization of commands that match both.



Authorization Example

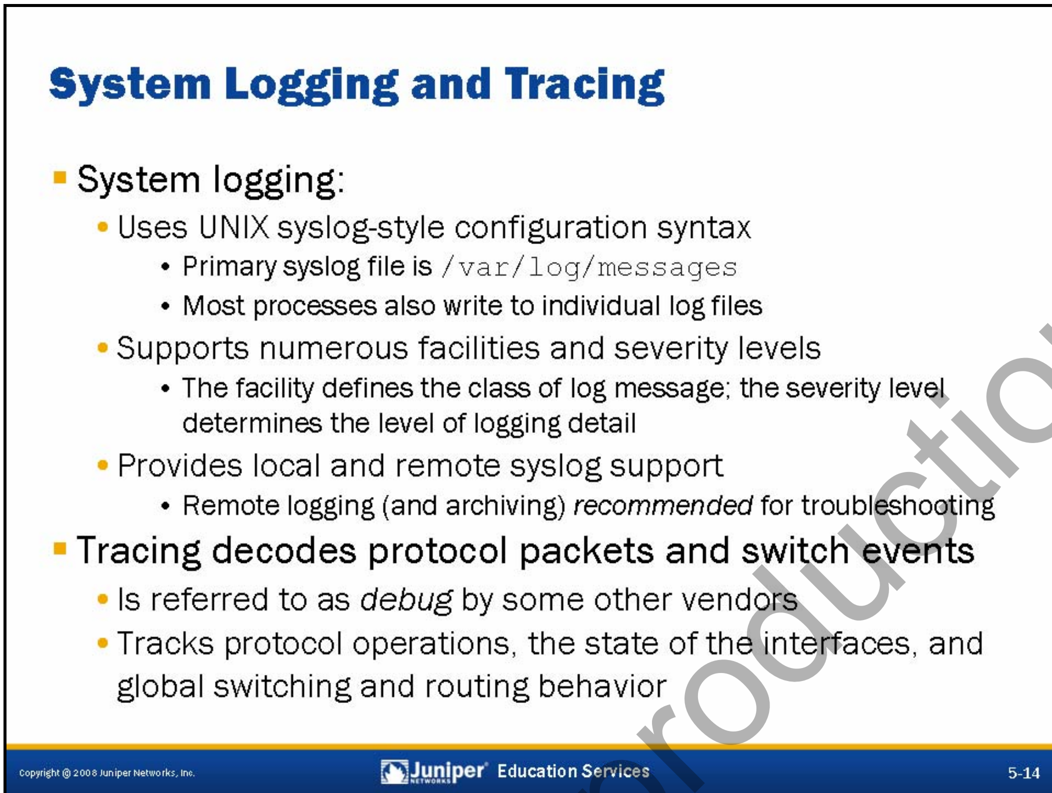
The configuration example on the slide shows how the various authorization components are configured:

- User *sue* is a member of the *noc* class.
- The *noc* class has `view` permissions.
- In addition, the *noc* class can clear statistics on individual interfaces using the **`clear interfaces statistics`** ***`interface-name`*** command.
- However, the *noc* class is denied the ability to clear the statistics of all interfaces at once with the **`clear interfaces statistics all`** command.



System Logging and Tracing

The slide highlights the topic we discuss next.



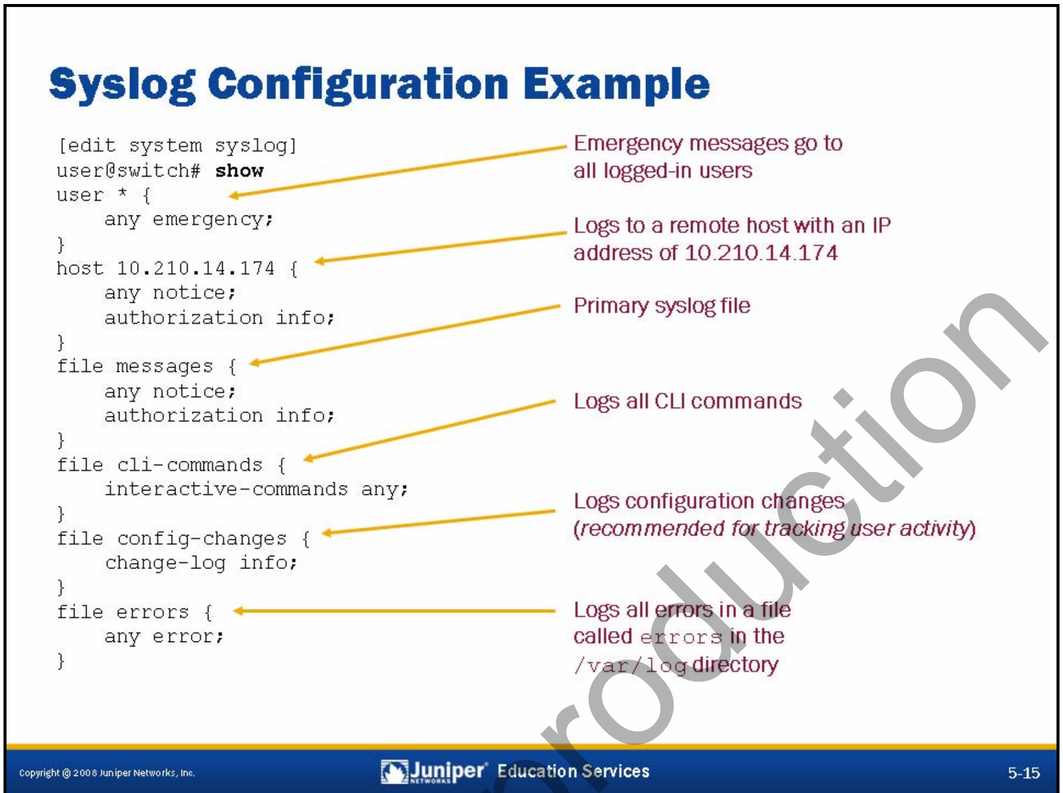
System Logging

System logging (syslog) operations use a UNIX syslog-style mechanism to record system-wide, high-level operations, such as interfaces going up or down or users logging in to or out of the switch. Configure these operations by using the **`syslog`** statement at the `[edit system]` hierarchy level and the **`options`** statement at the `[edit routing-options]` hierarchy level.

JUNOS software places the results of tracing and logging operations in files that are stored in the `/var/log` directory on the switch. Use the **`show log`** ***file-name*** command to display the contents of these files.

Tracing Operations

Tracing operations allow you to monitor the operation of protocols by decoding the protocol packets that are sent and received. Tracing is also available for other switch processes. In many ways tracing is synonymous with the debug function on equipment made by other vendors. Note that because of the design of EX-series platforms, you can enable reasonably detailed tracing in a production network without a negative impact on overall performance or packet forwarding.



System Logging Options Example

The slide shows various syslog configuration examples. General syslog configuration options include the following:

- **`host`** ***`name`*** or ***`IP address`***: Sends syslog messages to the configured host;
- **`archive`**: Configures how to archive system logging files (default is to keep 10 archive files with a maximum size of 128 K each);
- **`console`**: Configures the types of syslog messages to log to the system console;
- **`facility`**: Displays the class of log messages;
- **`severity`**: Displays the severity level of log messages;
- **`file`** ***`filename`***: Configures the name of the log file; and
- **`files`** ***`number`***: Displays the maximum number of system log files.

Interpreting Syslog Messages

- Standard log entries consist of the following fields:
  - Timestamp, platform name, software process name or PID, a message code, and the message text
  - Use **explicit-priority** to include a numeric priority value

```
Jan 12 17:08:00  switch mgd[2253]: %INTERACT-6-UI_CMDLINE_READ_LINE: User
'user', command 'show version '
```

- Use **help syslog ?** to help interpret message codes

```
user@switch> help syslog UI_CMDLINE_READ_LINE
Name:          UI_CMDLINE_READ_LINE
Message:       User '<username>', command '<command>'
Help:          User entered command at CLI prompt
Description:   The indicated user typed the indicated command at the CLI
prompt
               and pressed the Enter key, sending the command string to the
               management process (mgd).
Type:          Event: This message reports an event, not an error
Severity:      info
```

 Juniper Education Services 5-16

Interpreting System Log Entries

When using the standard syslog format, each log entry written to the messages file consists of the following fields:

- `timestamp`: Indicates when the message was logged.
- `name`: Displays the configured system name.
- `Process name` or `PID`: Displays the name of the process (or the process ID when a name is not available) that generated the log entry.
- `message-code`: Provides a code that identifies the general nature and purpose of the message. In the example shown, the message code is `UI_CMDLINE_READ_LINE`.
- `message-text`: Provides additional information related to the message code.

When you add the **`explicit-priority`** statement, JUNOS software alters the syslog message format to include a numeric priority value. In this situation, the value 0 indicates the most significant and urgent messages (emergency), and 7 indicates debug-level messages.

Continued on next page.

Interpreting Message Codes

Consult the *System Log Messages Reference* documentation for a full description of the various message codes and their meanings. Or, better yet, use the CLI's **`help`** function to obtain this information. The example on the slide shows the operator obtaining help on the meaning of the `UI_CMDLINE_READ_LINE` message code. Based on the output, it becomes relatively clear that the message code shows a command that a user entered at the CLI prompt.

Traceoptions Overview

- Tracing is the JUNOS software equivalent of *debug*
  - Can enable tracing on a production network
  - Requires configuration
  - Can trace multiple options (flags) to a single file
- Generic traceoptions configuration syntax:

The protocol or function being traced

Where to write the trace results

```
[edit protocols protocol-name]
user@switch# show
    traceoptions {
        file filename [replace] [size size] [files number] [no-
  stamp] [world-readable] [no-world-readable];
        flag flag [flag-modifier] [disable];
    }
```

Flags identify what aspects of the protocol are traced and at what level of detail

 Juniper Education Services 5-18

Hear Tracing, Think Debug

Tracing is the JUNOS software term for what other vendors sometimes call *debug*. In most cases when you enable tracing (through configuration), you create a trace file that is used to store decoded protocol information. You analyze these files using standard CLI log file syntax such as **`show log`** ***`logfile-name`***. Because of the design of EX-series switching platforms, you can enable detailed tracing in a production network without significantly impacting performance. Even so, you should always remember to turn off tracing once you have completed your testing to avoid unnecessary resource consumption.

Generic Tracing Configuration

The slide shows a generic tracing stanza, which, if applied to the `[edit ethernet switching-options]` portion of the configuration hierarchy, would result in the tracing of switching events. Ethernet switching traceoptions track general switching operations and record them in the specified log file. To trace the operations for an individual protocol, configure traceoptions under the desired protocol hierarchy.

Continued on next page.

Generic Tracing Configuration (contd.)

Configuration options for tracing are the following:

- **file *filename***: Specifies the name of the file in which to store information.
- **size *size***: Specifies the maximum size of each trace file, in kilobytes (KB), megabytes (MB), or gigabytes (GB). When a trace file named *trace-file* reaches this size, it is renamed *trace-file.0*. When the trace file again reaches its maximum size, *trace-file.0* is renamed *trace-file.1*, and *trace-file* is renamed *trace-file.0*. This renaming scheme continues until the maximum number of trace files is reached. The software then overwrites the oldest trace file. If you specify a maximum file size, you also must specify a maximum number of trace files with the files option. The default size is 128 KB.
- **files *number***: Specifies the maximum number of trace files. When a trace file named *trace-file* reaches its maximum size, it is renamed *trace-file.0*, then *trace-file.1*, and so on, until the maximum number of trace files is reached. The software then overwrites the oldest trace file. The default is ten files.
- **no-stamp**: Prevents timestamp information from being placed at the beginning of each line in the trace file. By default, if you omit this option, timestamp information is placed at the beginning of each line of the tracing output.
- **replace**: Replaces an existing trace file if one exists. By default, if you omit this option, tracing output is appended to an existing trace file.
- **readable**: Allows any user to view the file.
- **no-world-readable**: Allows only the user who configured the file to view it. This is the default setting.

Including the **traceoptions** statement at the `[edit interfaces` *`interface-name`*`]` hierarchy level allows you to trace the operations of individual switch interfaces. You can also trace the operations of the interface process, which is the device-control process (dcd).

When tracing a specific interface, the specification of a trace file is not supported. The JUNOS software kernel does the logging in this case, so the tracing information is placed in the system's `messages` file. In contrast, global interface tracing supports an archive file; by default, `/var/log/dcd` is used for global interface tracing.

Traceoptions Example

- Include the **traceoptions** statement at the [edit protocols *protocol-name*] hierarchy level
  - Useful for troubleshooting
  - Traceoptions also available for other hierarchies

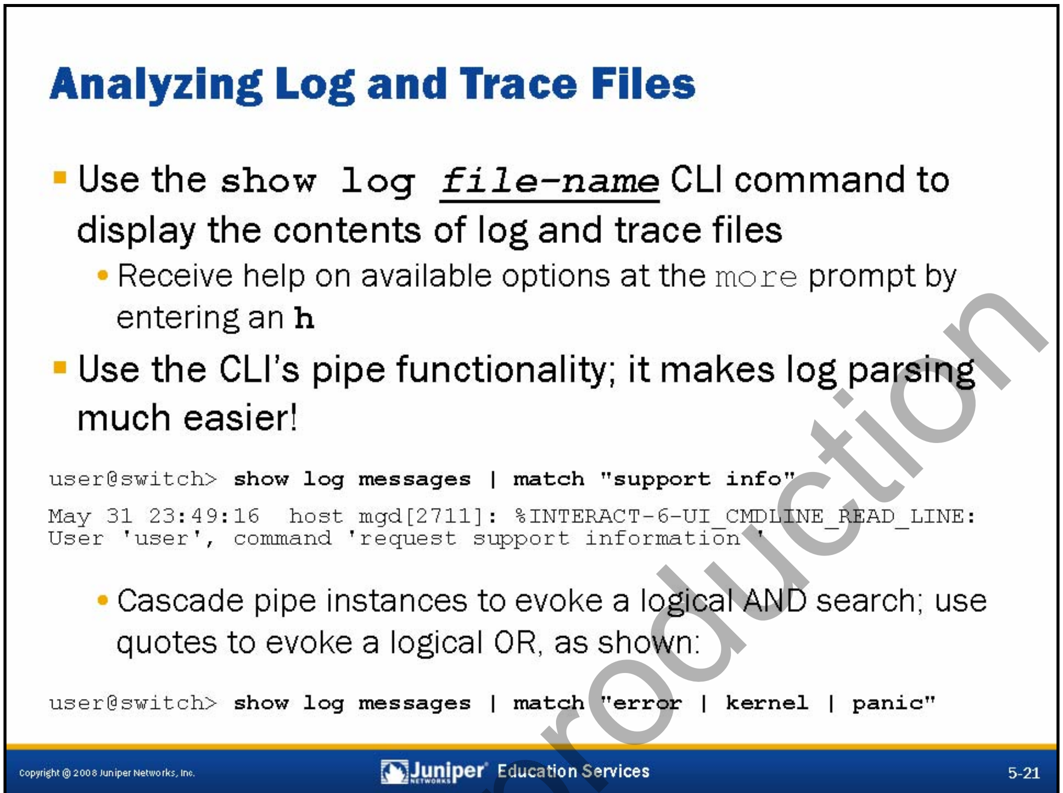
```
[edit protocols rstp]
user@switch# show
traceoptions {
    file rstp-trace;
    flag bpdu;
}
user@switch> show log rstp-trace
Jan 13 09:23:38 trace_on: Tracing to "/var/log/rstp-trace" started
Jan 13 09:23:38.549565 MSG: Port ge-0/0/15.0: Validating received bpdu ...
Jan 13 09:23:38.566047 MSG: Port ge-0/0/15.0: Extracting Info from BPDU ...
Jan 13 09:23:38.566105 MSG: Port ge-0/0/15.0: Received bpdu successfully
  validated...
Jan 13 09:23:38.566248 MSG: BPDU received successfully
```

 Juniper Education Services 5-20

Traceoptions Example

Trace the operations of a specific protocol by including the **traceoptions** statement at the [edit protocols *protocol-name*] hierarchy. In most cases you will want to be a bit selective in what you trace because selecting the **all** keyword will likely numb your mind with trivial details. The sample Rapid Spanning Tree Protocol (RSTP) stanza on the slide reflects a typical tracing configuration that provides details about bridged protocol data unit (BPDU) events. In many cases you will want to use the **detail** switch to a given protocol flag for the added information often needed in troubleshooting scenarios.

The slide shows a sampling of the results obtained with the tracing configuration. As with any log file, simply enter a **show log *trace-file-name*** command to view the decoded protocol entries. The sample trace output reflects the receipt of a BPDU message on the ge-0/0/15.0 interface and the successful validation of the BPDU.



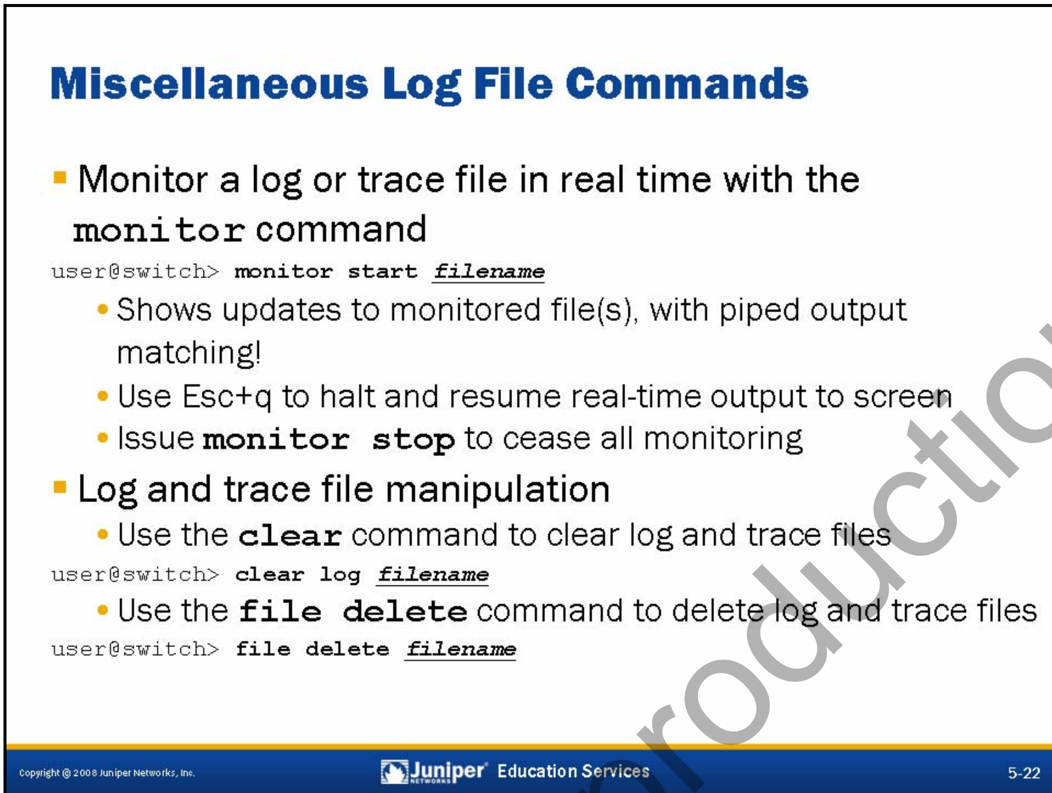
Viewing Logs and Traces

By default, JUNOS software stores log and trace files in `/var/log`. To view stored log files, use the **`show log`** command. Recall that the CLI automatically pauses when there is more then one screen's worth of information, and that at this `more` prompt, you can enter a forward slash (`/`) character to conduct a forward search. As a hint, enter **h** when at a `more` prompt to view the context help screen of available commands, shown in the following example:

```
---(Help for CLI automore)---
    Clear all match and except strings:                c or C
    Display all line matching a regexp:                m or M <string>
    Display all lines except those matching a regexp:  e or E <string>
    Display this help text:                            h
    Don't hold in automore at bottom of output:        N
    Hold in automore at bottom of output:              H
    Move down half display:                            TAB, d, or ^D
    Move down one line:         Enter, j, ^N, ^X, ^Z, or Down-Arrow
   . . .
```

CLI's Pipe Functionality

Being able to cascade multiple instances of the CLI's pipe functionality is a real benefit when you must search a long file for associated entries. You can also search for multiple criteria in a logical OR fashion as shown by the example on the slide that searches for lines that include any of the words `error`, `kernel`, or `panic`.



Monitoring Logs and Trace Files

Use the **`monitor start`** CLI command to view real-time log information. You can monitor several log files at one time. The messages from each log are identified by *`filename`*, where *`filename`* is the name of the file from which entries are being displayed. JUNOS software displays this line initially and when the CLI switches between log files.

Use Esc+q to enable and disable syslog output to the screen; use the **`monitor stop`** command to cease all monitoring. Note that you can use the CLI's **`match`** functionality to monitor a file in real time while displaying only entries that match your search criteria. To use this functionality, use a command in the following format:

```
user@switch> monitor start messages | match fail
```

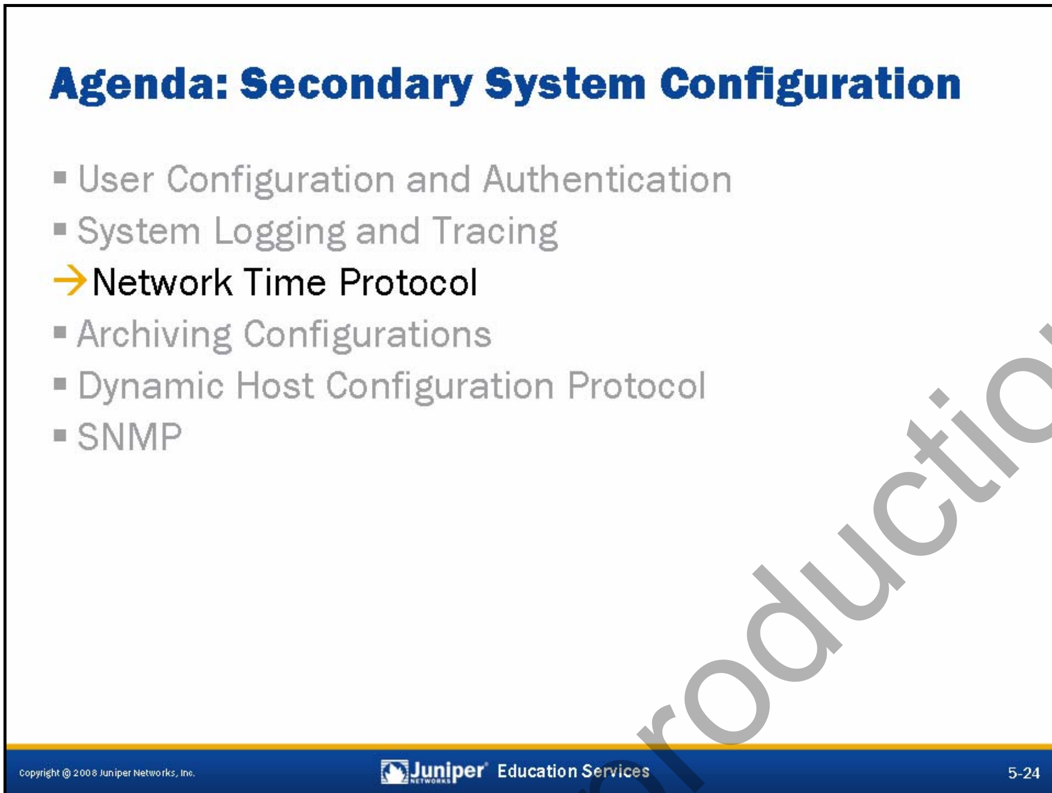
If you do not delete or disable all trace flags, tracing continues in the background and the output continues to be written to the specified file. The file remains on the switch's compact flash drive until it is either deleted manually or overwritten according to the `traceoptions` file parameters. To disable all tracing at a particular hierarchy, issue a **`delete traceoptions`** command at that hierarchy and commit the change.

Continued on next page.

Log and Trace File Manipulation

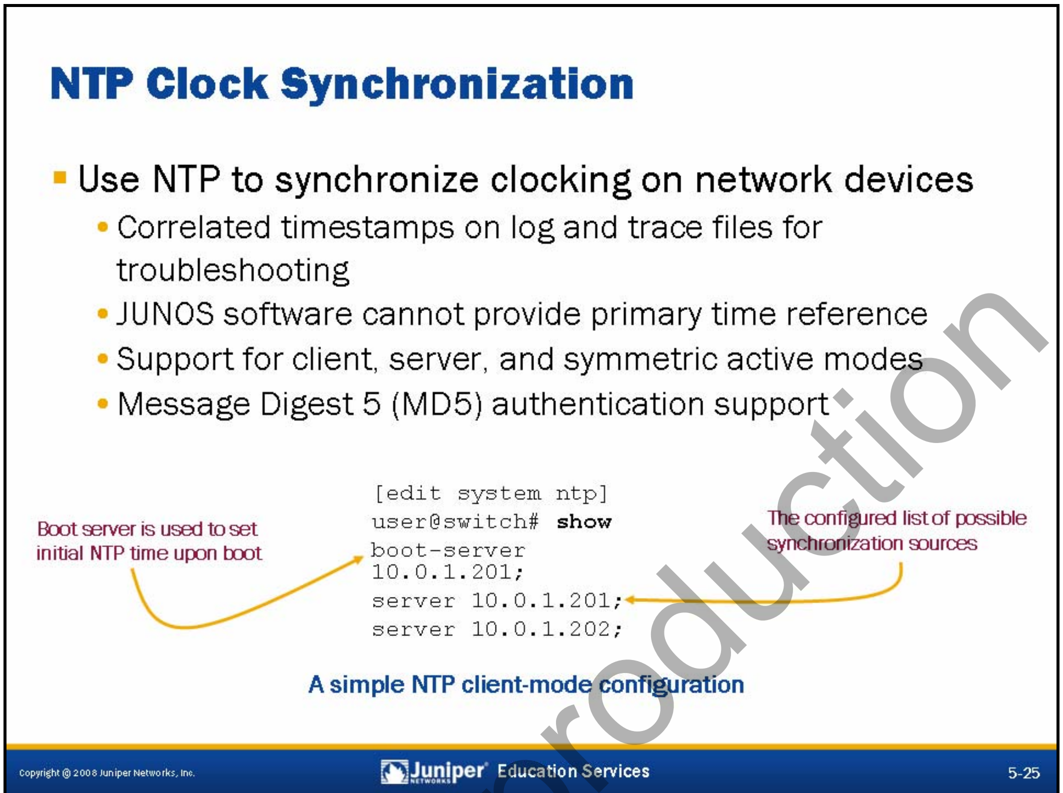
To truncate files used for logging, use the **`clear log`** ***`filename`*** command.

To delete a file, use the **`file delete`** command. If you want, you can also use wildcards with the file command's **`delete`**, **`compare`**, **`copy`**, **`list`**, and **`rename`** operations.



Network Time Protocol

The slide highlights the topic we discuss next.



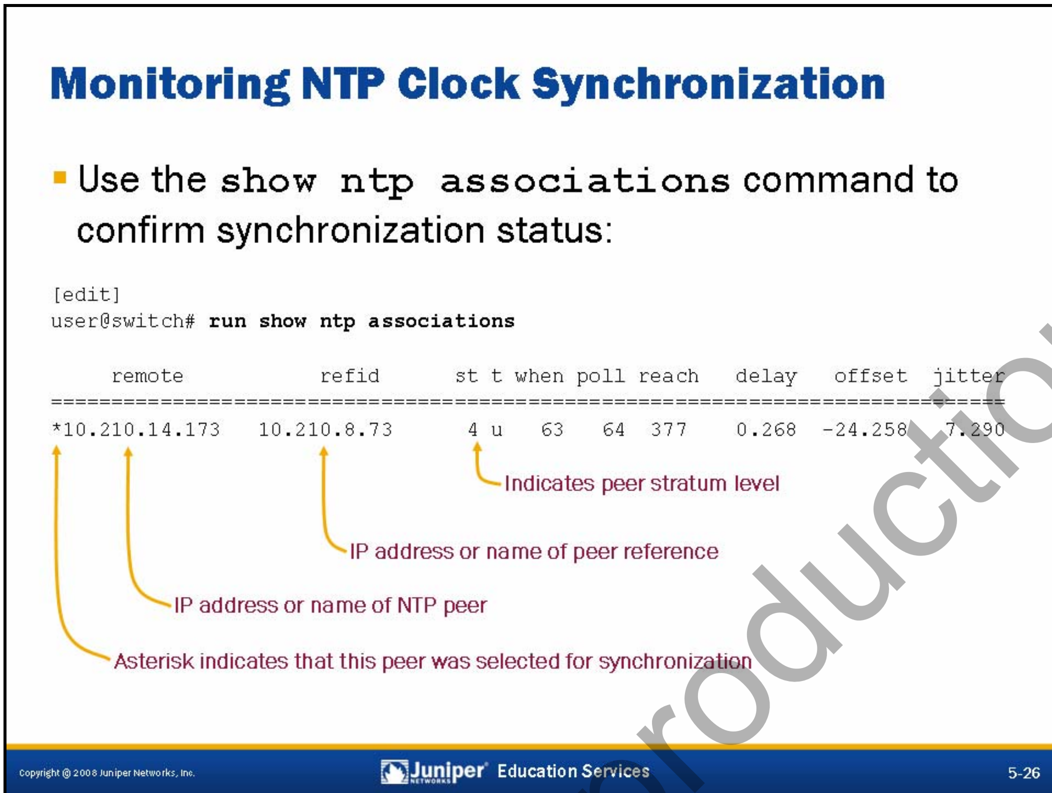
What Time Is It?

Use the Network Time Protocol (NTP) to synchronize network devices to a common, and preferably accurate, time source. By synchronizing all switches, timestamps on log messages are both accurate and meaningful.

NTP is based on a series of timing hierarchies, with a Stratum 1 (atomic) timing source at the very top. While accuracy is desirable, there is no need to synchronize to a Stratum 1 reference to benefit from synchronizing to the time of day. JUNOS software cannot provide its own timing source because the definition of a local, undisciplined clock source (for example, the local crystal oscillator) is not supported. If needed, obtain a commodity UNIX box of some type configured to provide a timing reference based on its local clock. Any synchronization, even if based on an inaccurate local clock, is better than none.

JUNOS software supports client, server, and symmetric modes of NTP operation, and can also support broadcast and authentication. We recommend that authentication be used to ensure that an attacker cannot compromise a system's synchronization.

The slide provides a typical NTP-related configuration stanza. Two machines can synchronize only when their current clocks are relatively close. A boot server is used to set a switch's clock at boot time to ensure that it is close enough to later synchronize to the configured time server. Issue the operational-mode **`set date ntp`** ***`address`*** command as a substitute for a boot server.

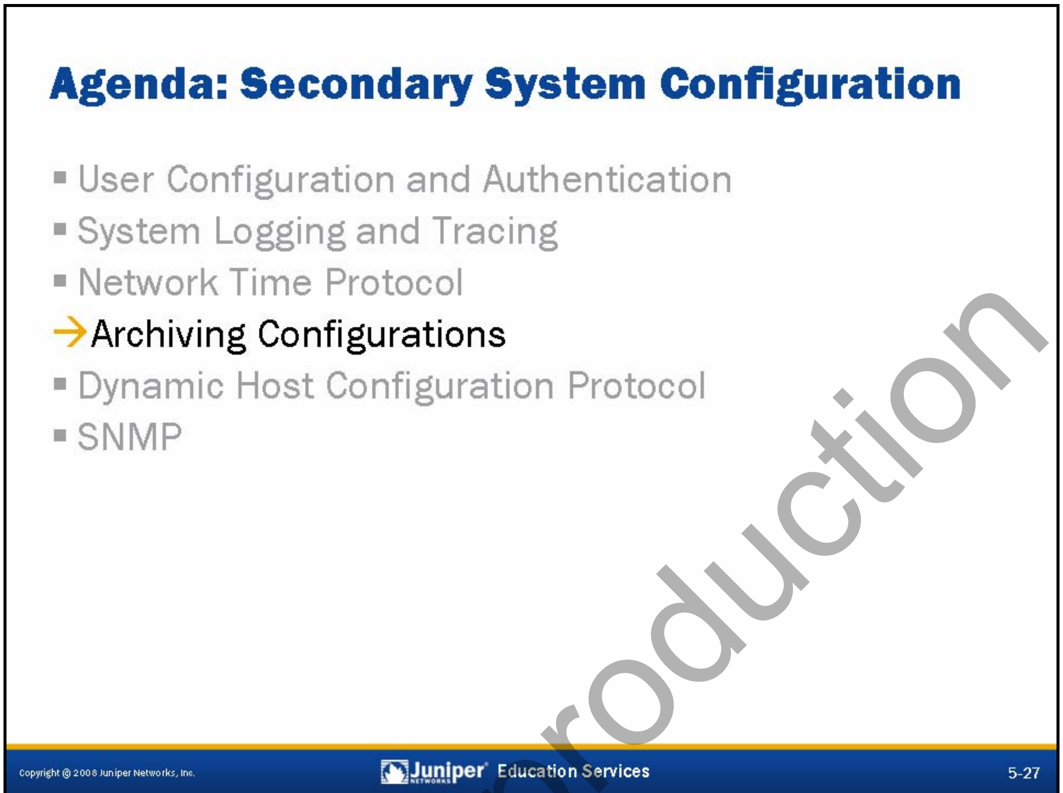


Monitoring NTP

Use the **show ntp associations** command to display synchronization status. The address column shows the hostname or IP address of the remote NTP peer. The symbol next to the hostname or IP address gives the status of the peer in the clock selection process. The possible symbols include the following:

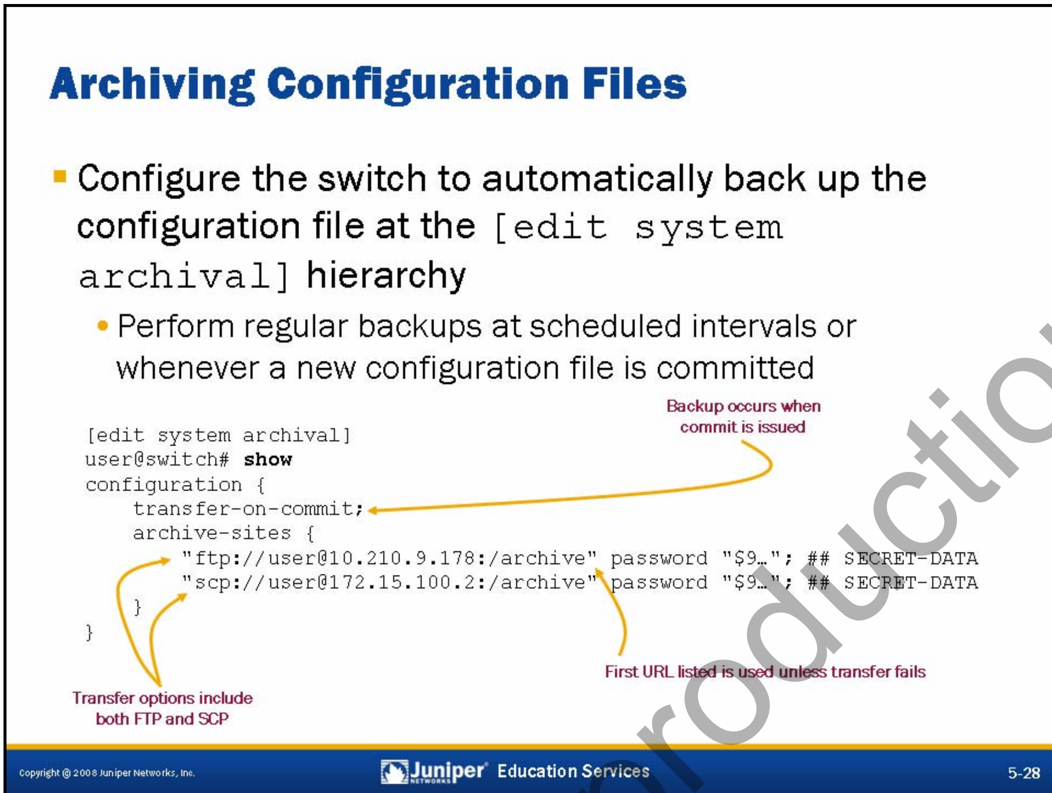
- Space: Discarded because of a high stratum value or failed sanity check;
- x: Designated *falseticker* by the intersection algorithm;
- . (period): Culled from the end of the candidate list;
- – (hyphen): Discarded by the clustering algorithm;
- + (plus): Included in the final selection set;
- # (pound): Selected for synchronization, but the distance exceeds the maximum;
- * (asterisk): Selected for synchronization; and
- o: Selected for synchronization, but the packets-per-second (pps) signal is in use.

You can view further synchronization details with the **show ntp status** command.



Archiving Configurations

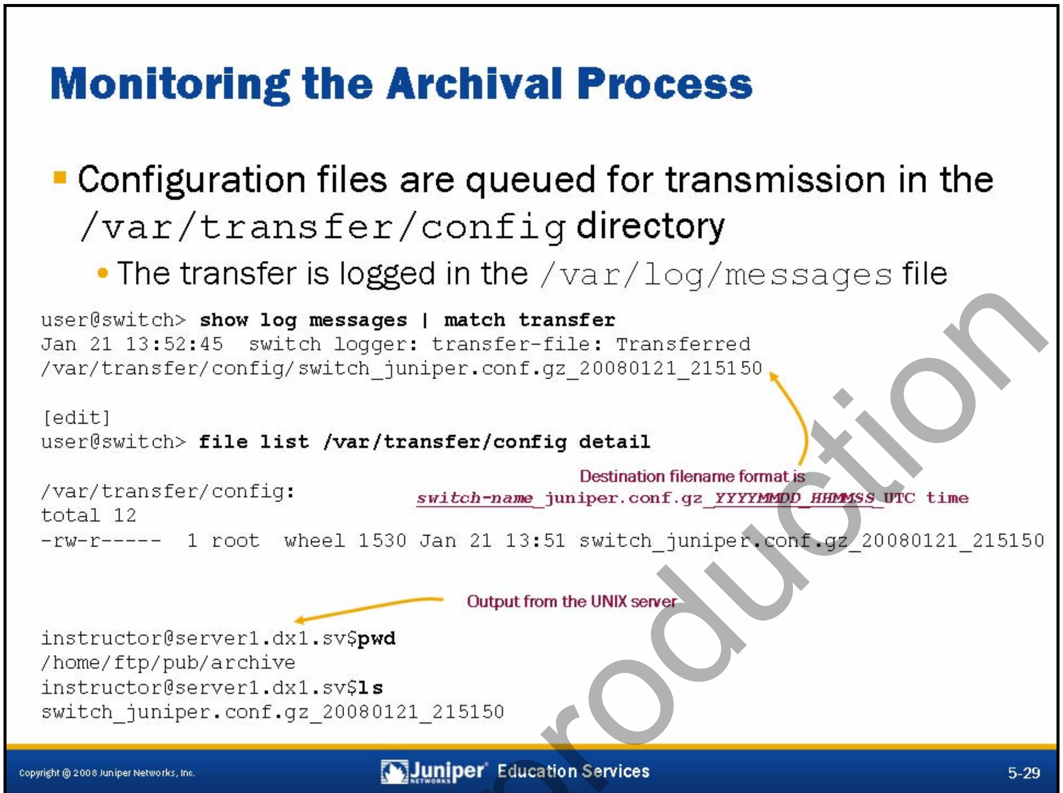
The slide highlights the topic we discuss next.



Automated Configuration Backup

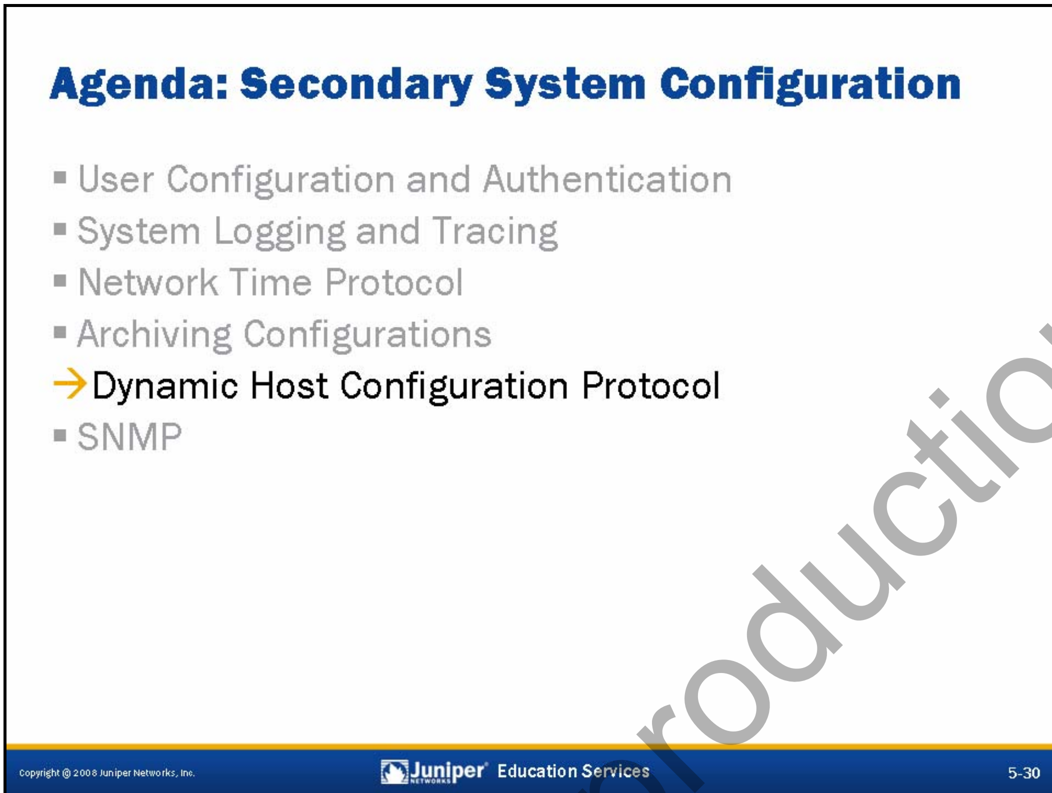
Certain failures might render the storage device, which holds the configuration files, unusable. In the event of such a disaster, it might be helpful to have the most recent configuration file stored on a separate device, such as an FTP or SCP server. To automatically back up a switch's configuration file to a remote system, configure the necessary configuration archival parameters at the `[edit system archival]` hierarchy level. When you configure the switch to transfer its configuration files, you specify an archive site, in the form of a URL, to which the files are transferred. If you specify more than one archive site, the switch attempts to transfer to the first archive site in the list, moving to the next site only if the transfer fails.

Backups can occur at regular intervals with the use of the **`transfer-interval`** statement. The frequency at which the file transfer occurs can be from 15 to 2880 minutes, and you can define this frequency. Alternatively, the configuration file can be transferred every time a new configuration becomes active with the use of the **`transfer-on-commit`** statement.



How It Works

Upon entering a **`commit`** command or reaching the specified time interval, the switch copies the configuration file into the `/var/transfer/config` directory and an FTP or SCP session is opened with the remote storage device. Once the configuration file is transferred to the remote storage device, a system log message is generated, confirming success or failure of the transfer. The destination filename format, as shown on the slide, cannot be altered by configuration.



Dynamic Host Configuration Protocol

The slide highlights the topic we discuss next.

DHCP Concepts and Terminology

- DHCP transfers host-specific configuration details from a designated DHCP server to DHCP clients while managing the allocation of IP addresses on a LAN
  - Scalable method of managing LAN resources
  - Follows client/server model
  - Based on BOOTP
- EX-series enterprise switches support two modes:
  - DHCP server
    - Configured under `[system services dhcp]` hierarchy
  - DHCP relay agent
    - Configured under `[forwarding-options helper bootp]`
  - Both options cannot be configured at the same time

 Juniper Education Services 5-31

Understanding DHCP

The Dynamic Host Configuration Protocol (DHCP) serves multiple purposes. The first function of DHCP is to serve as a framework for relaying configuration details from a designated server to individual clients (such as PCs) within a TCP/IP network. The second function of DHCP deals with the allocation of IP addresses for the requesting clients. The use of DHCP services within a network to allocate and relay client configuration details can significantly reduce the administrative overhead required when managing LAN resources. DHCP follows a client/server model and requires communication exchanges between both the client and server using packets that are based on the Bootstrap Protocol (BOOTP). RFC 2131 defines DHCP.

The DHCP server stores the administratively defined configuration details to be used on the requesting clients. Each DHCP server receiving the DHCPDISCOVER broadcast message sends a DHCPOFFER message to the client, which offers an IP address for a set period of time, known as the lease period.

Once the client sends the DHCPREQUEST message, the selected DHCP server sends a DHCPACK acknowledgment that includes configuration information, such as the IP address, subnet mask, default gateway, and the negotiated lease details. The DHCP server manages the assigned IP addresses along with any reported address conflicts. Once the negotiated lease period expires, the DHCP server renews the address assignment.

Continued on next page.

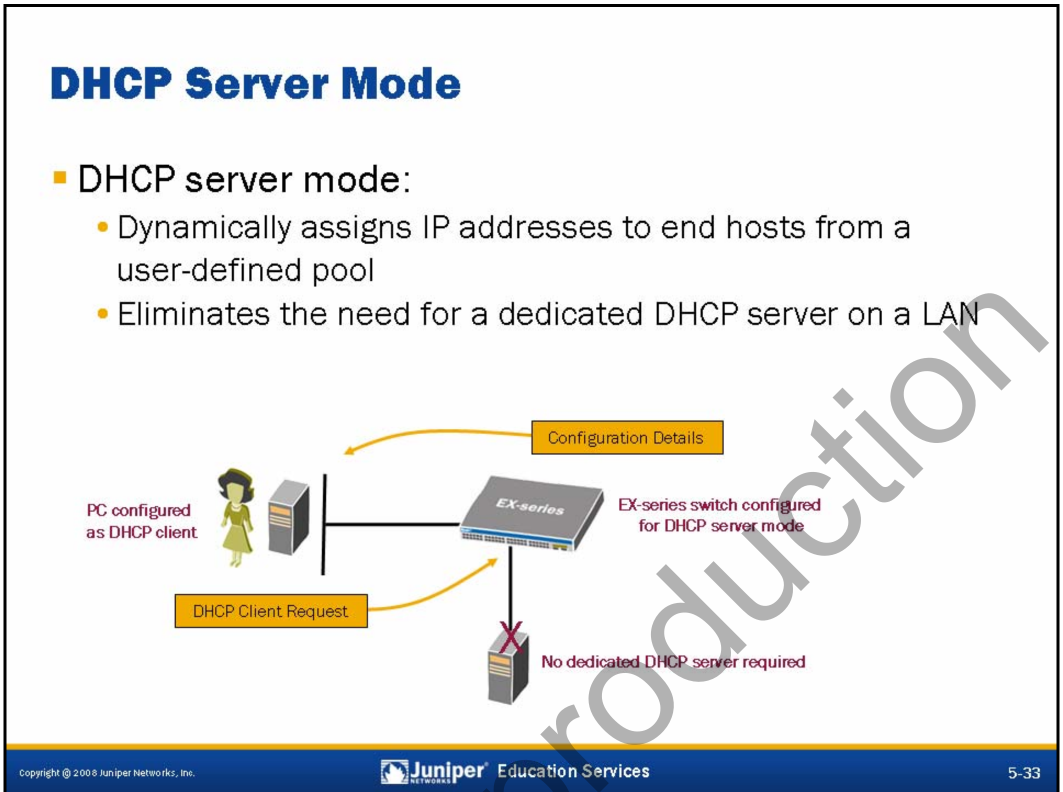
DHCP Server Mode and DHCP Relay Mode

EX-series platforms can operate in either DHCP server mode or DHCP relay mode.

DHCP server mode enables the switch to function as a DHCP server. This feature eliminates the need for a dedicated DHCP server on the LAN. You can enable DHCP server mode using the `[system services dhcp]` hierarchy level. This functionality is also employed automatically in the J-Web EZsetup wizard to provide initial HTTP access to the switch.

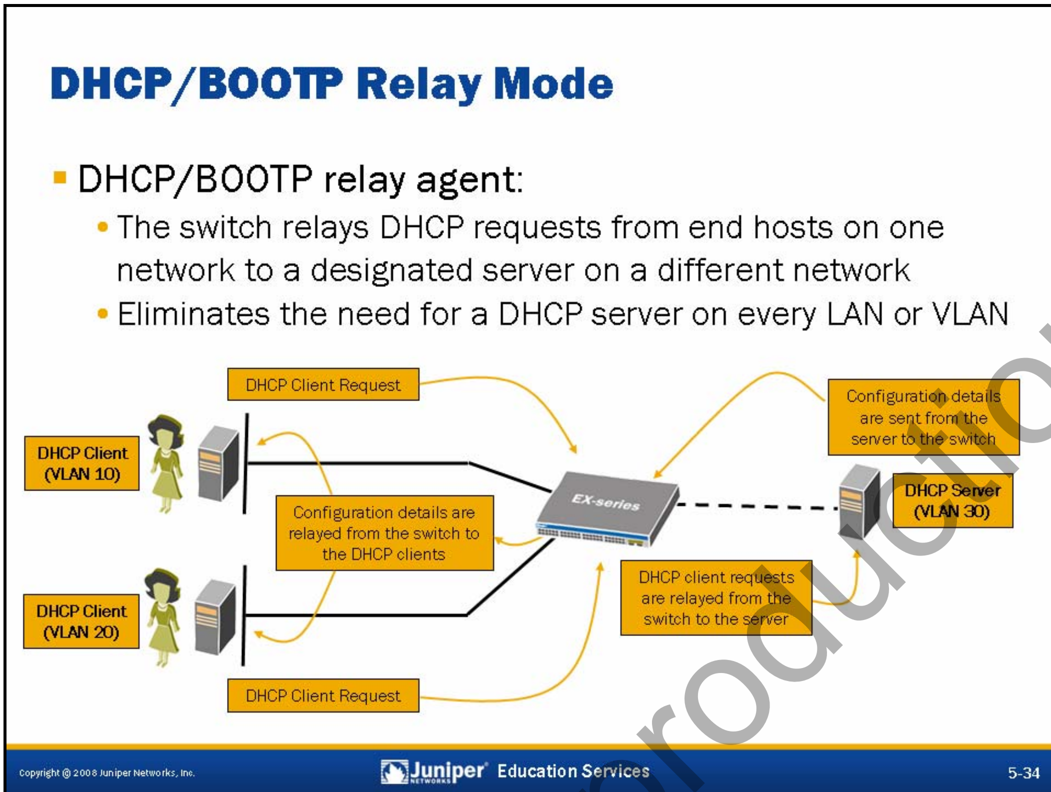
You can configure EX-series switches to function as DHCP/BOOTP relay agents. This feature allows DHCP/BOOTP requests to be sent from a client on one network to a DHCP server on a different network. The advantage provided by this feature is the elimination of a dedicated DHCP server on each network. You configure this feature at the `[forwarding-options helpers bootp]` hierarchy level.

Because DHCP/BOOTP messages are sent as a broadcast and are not directed to a specific server, switch, or router, EX-series switches cannot function as both a DHCP server and a DHCP/BOOTP relay agent at the same time. JUNOS software generates a commit error if both options are configured at the same time, and the commit will not succeed until one of the options is removed.



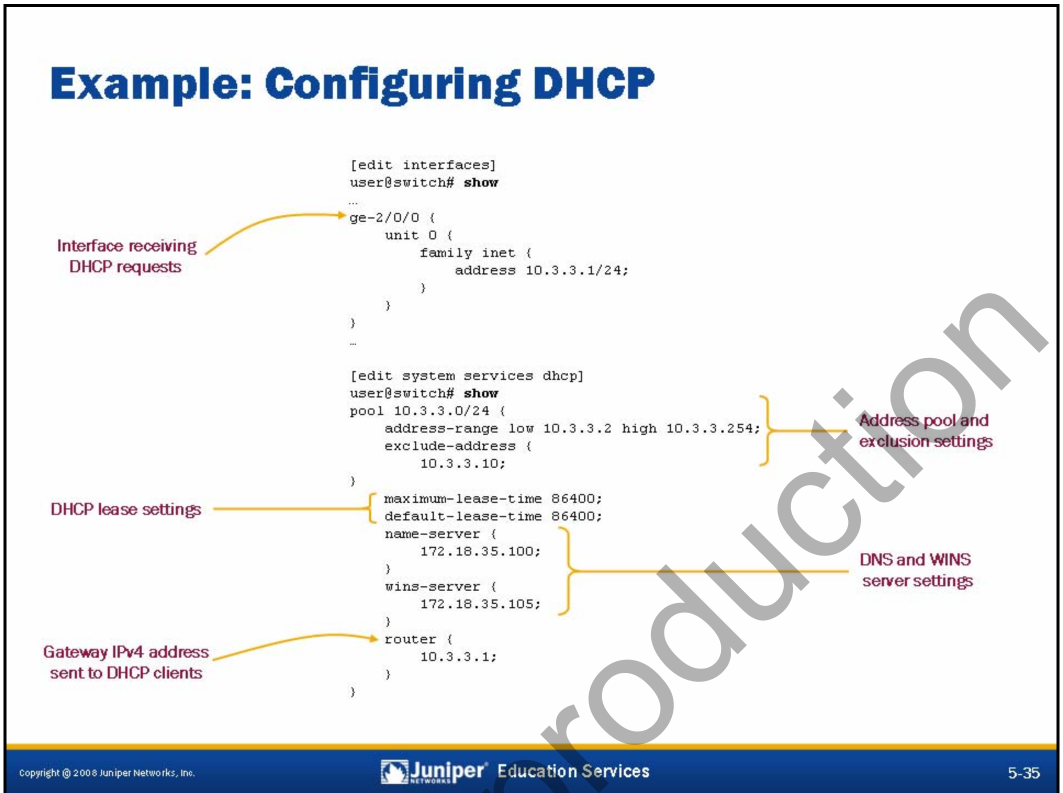
DHCP Server Mode

This slide illustrates the advantage of configuring an EX-series switch in DHCP server mode as well as the basic operation involved between the DHCP client and an EX-series switch functioning as the DHCP server.



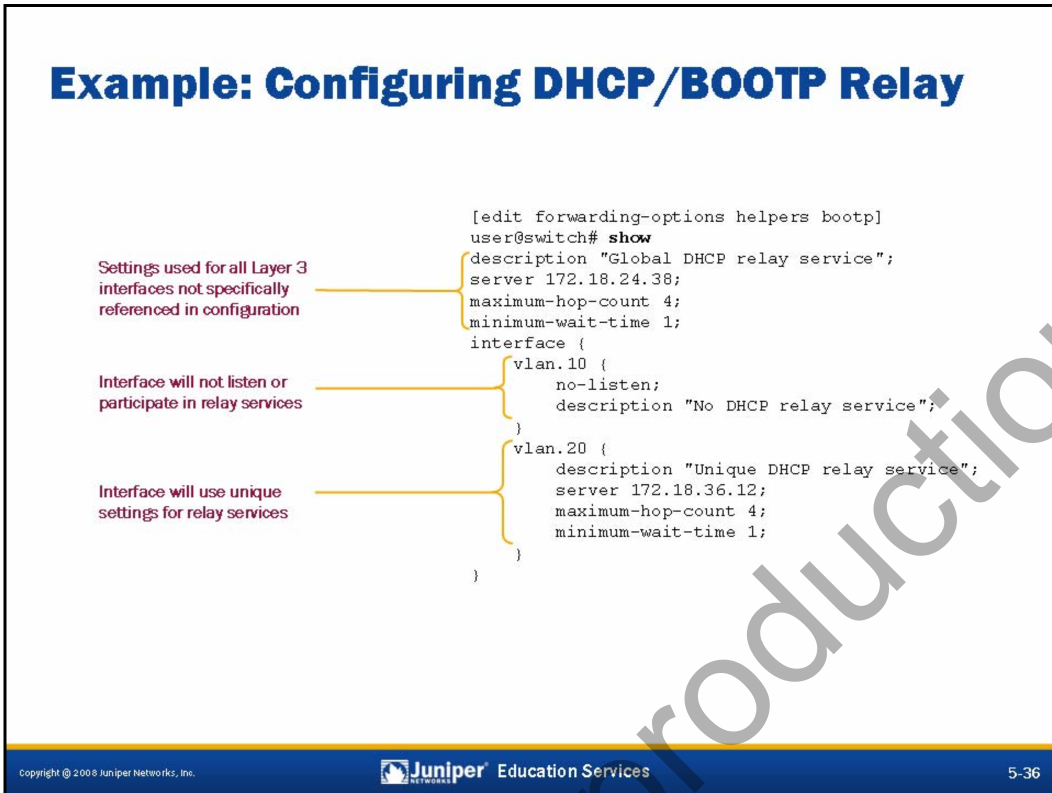
DHCP/BOOTP Relay Agent

DHCP requests sent from a client to a server are normally restricted to the same physical segment, LAN, or virtual LAN (VLAN) on which the client resides. In the event that the server and client are on different LANs or VLANs, a relay agent is needed. The slide illustrates the basic process involved when a relay agent is required to pass the DHCP/BOOTP requests between a client and a server. The main advantage of this feature is that a single DHCP server can serve clients on remote LANs or VLANs, eliminating the need for a dedicated DHCP server in each LAN or VLAN environment.



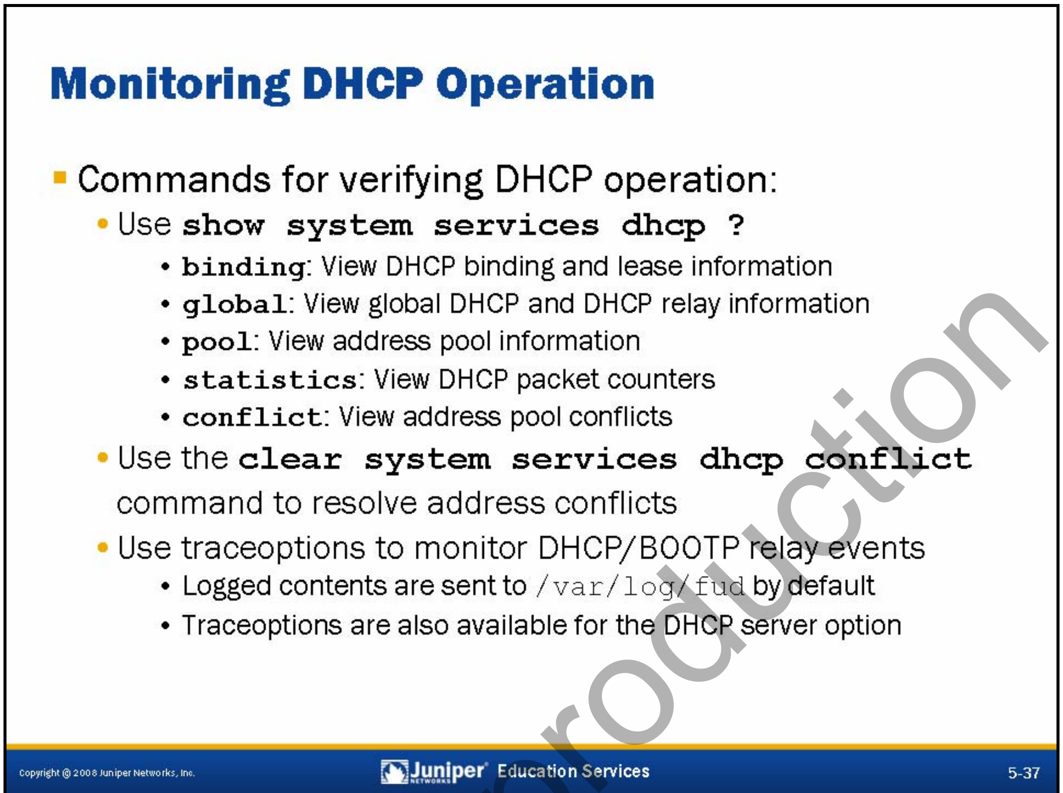
Example: Configuring DHCP

The slide shows a DHCP configuration example using some common DHCP configuration options.



Sample DHCP/BOOTP Relay Configuration

The slide shows a sample DHCP/BOOTP configuration using some common DHCP/BOOTP configuration options. This example defines global parameters that will be used for all interfaces not specifically referenced in the configuration. The example also references two distinct interfaces and their associated parameters.



Monitoring DHCP Operations

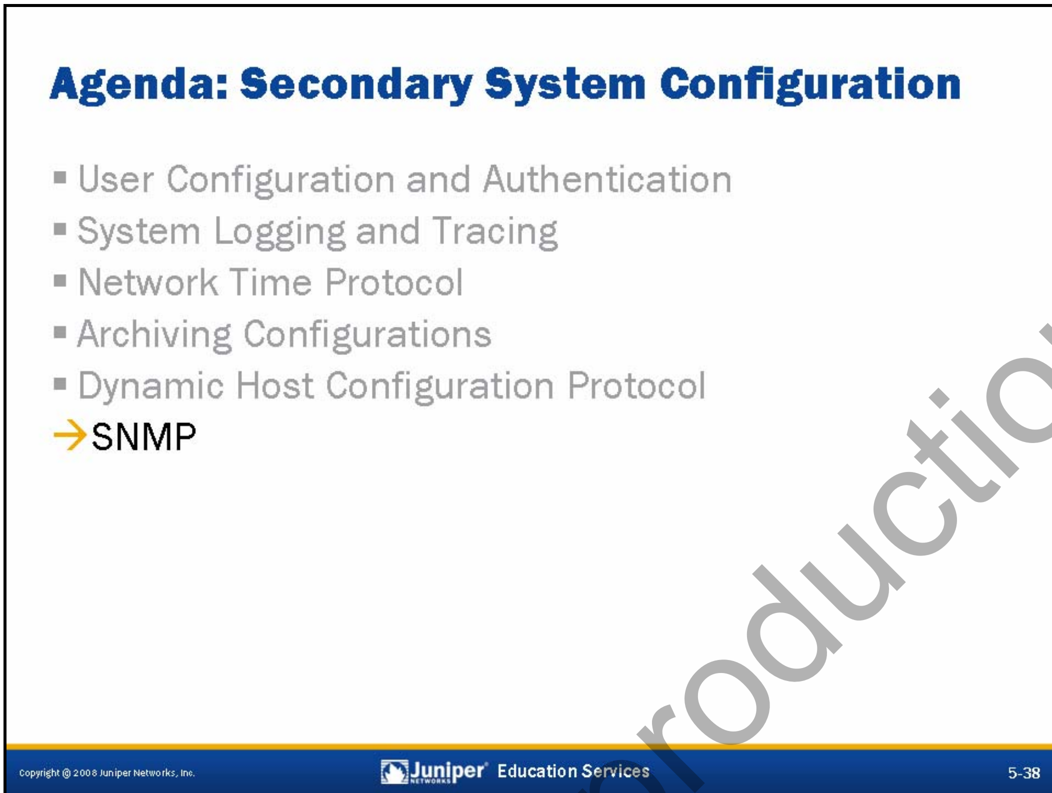
The slide highlights some commonly used outputs to monitor DHCP operation. Use the **show system services dhcp pool** command to view DHCP address pool details. The **show system services dhcp binding** command outputs DHCP binding and lease information. Use the **show system services dhcp statistics** command to display statistics for the various DHCP and BOOTP packets that are sent and received from the switch. The **show system services dhcp conflict** command identifies address conflicts, if any exist.

Use the **clear system services dhcp conflict** command to clear existing address conflicts. Use the **address** knob when clearing a specific address conflict.

To monitor DHCP/BOOTP relay events, configure traceoptions at the `[edit forwarding-options helpers]` hierarchy level. Ensure that the **bootp** flag option is selected along with the logging level you want.

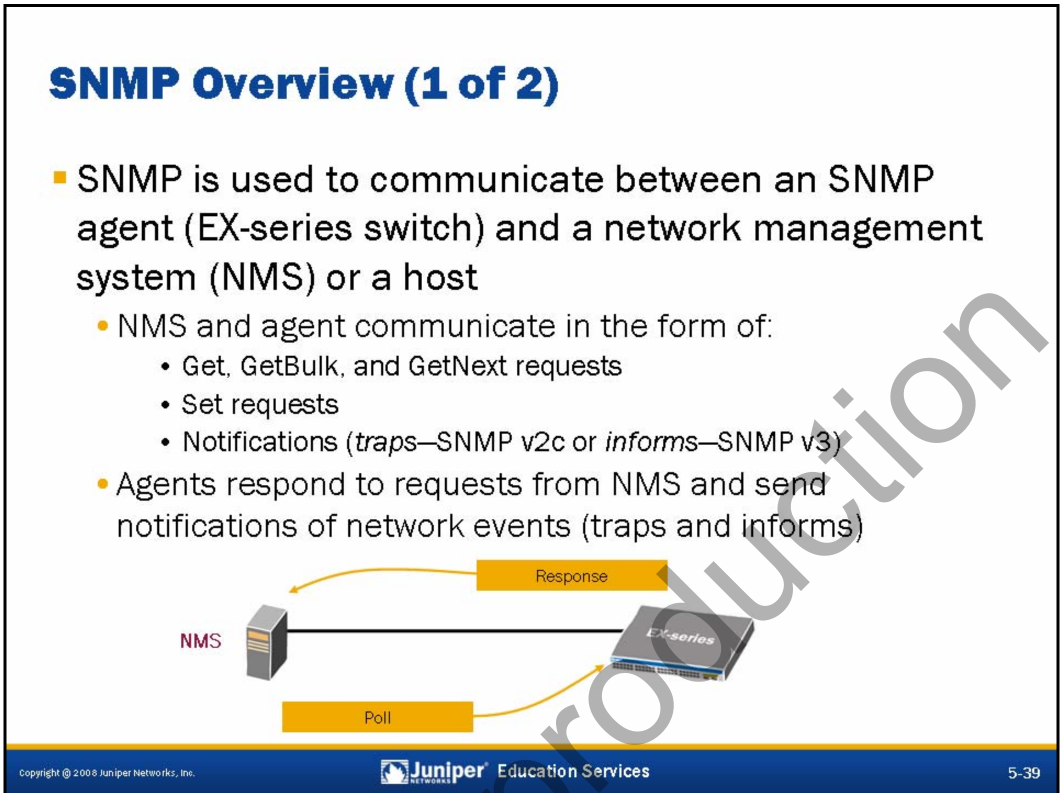
By default, the switch sends all logged events for DHCP/BOOTP relay to the `/var/log/fud` log file. To view the logged events, use the **show log fud** command.

You can also configure traceoptions for the DHCP server option at the `[edit system services dhcp]` hierarchy level. If no filename is specified, events are logged to the `/var/log/dhcpd` log file.



Simple Network Management Protocol

The slide highlights the topic we discuss next.

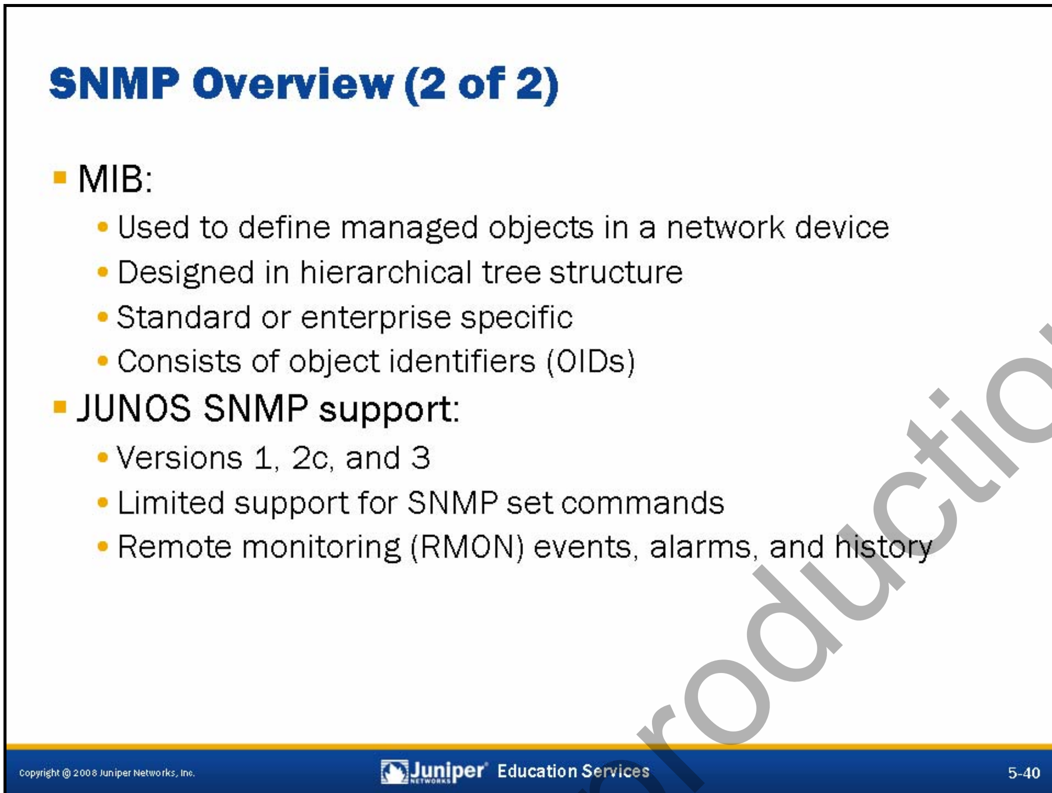


SNMP Operation

A network device, such as the EX-series switch, acts as an SNMP agent. An SNMP agent exchanges network management information with SNMP manager software running on a network management system (NMS) or host. The agent responds to requests for information and actions from the manager. An agent communicates with the SNMP manager using the following message types:

- Get, Getbulk, or Getnext requests: The SNMP manager requests information from an SNMP agent. The agent responds with a Get response message.
- Set requests: The SNMP manager changes the value of a Management Information Base (MIB) object controlled by the agent. The agent returns the status in a Set response message.
- Notifications: The SNMP agent sends traps to notify the manager of significant events regarding the network device. SNMP version 3 uses *informs* to notify the manager of significant events. Informs increase SNMP reliability by requiring the receiver to acknowledge the receipt of an inform notification.

By polling managed network devices, the NMS collects information about network resources. An SNMP agent can also notify the NMS of events and resource constraints through the use of SNMP traps.

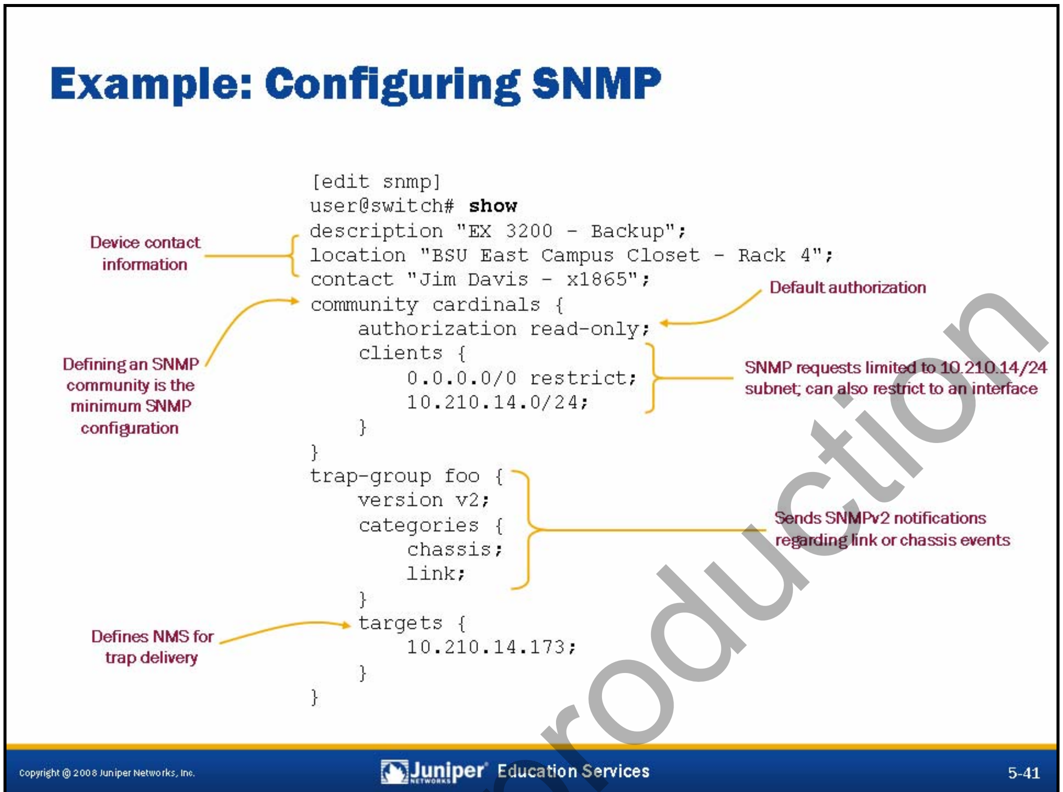


Here Come the MIBs

A MIB is a collection of objects maintained by the SNMP agent in a hierarchical fashion. The SNMP manager views or changes objects within the MIB structure. MIBs can be defined at the enterprise level to provide enterprise-specific information about the managed network device, or MIBs can be standardized to provide common information across multiple vendor network devices. NMS devices poll object identifiers (OIDs) to retrieve management information. An OID is considered a leaf in the tree-like hierarchy of a MIB. The Internet Engineering Task Force (IETF) provides standard MIBs you can download at http://www.ietf.org. You can download Juniper Networks enterprise MIBs at http://www.juniper.net/techpubs.

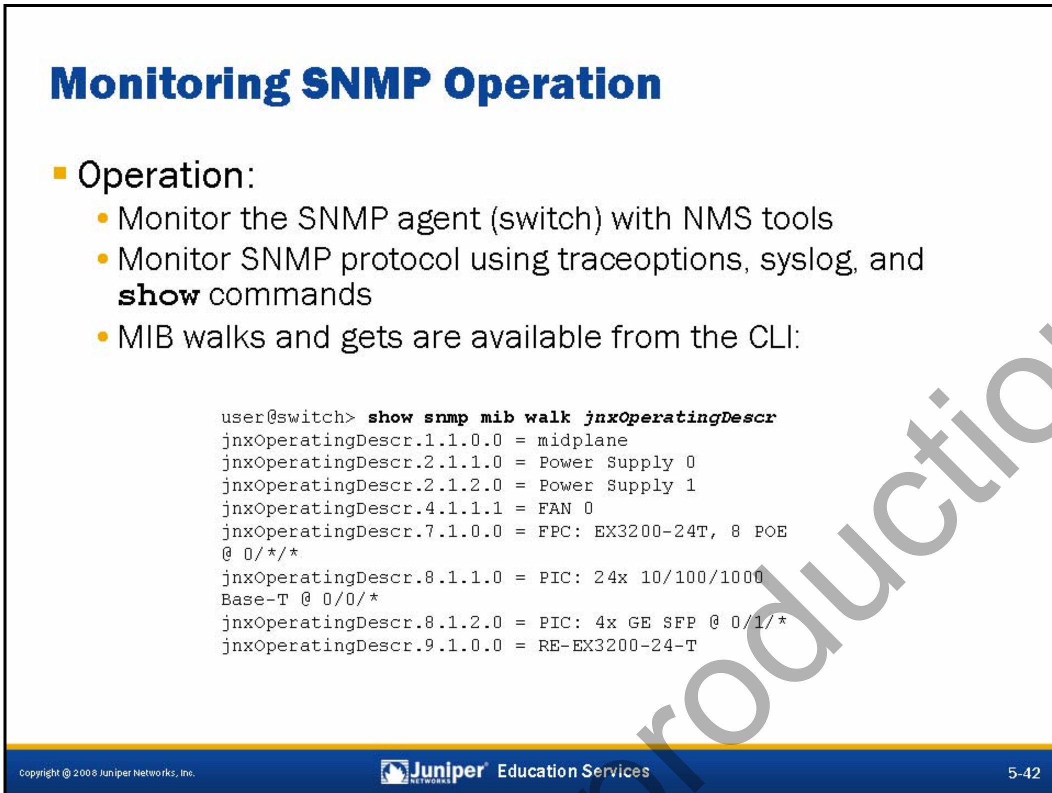
JUNOS SNMP Support

JUNOS software provides support for SNMP versions 1, 2c, and 3. Version 1 is the initial implementation of SNMP that defines the architecture and framework for SNMP. Version 2 added support for community strings, which act as passwords determining access to SNMP agent MIBs. SNMPv3 is the most up-to-date version and provides enhanced security features including the definition of a user-based security model (USM) and a view-based access control model (VACM). SNMPv3 provides message integrity, authentication, and encryption, which is a superior security model over SNMPv2c, which uses plain text community strings. JUNOS software also provides support for remote monitoring (RMON) events, alarms, and history.



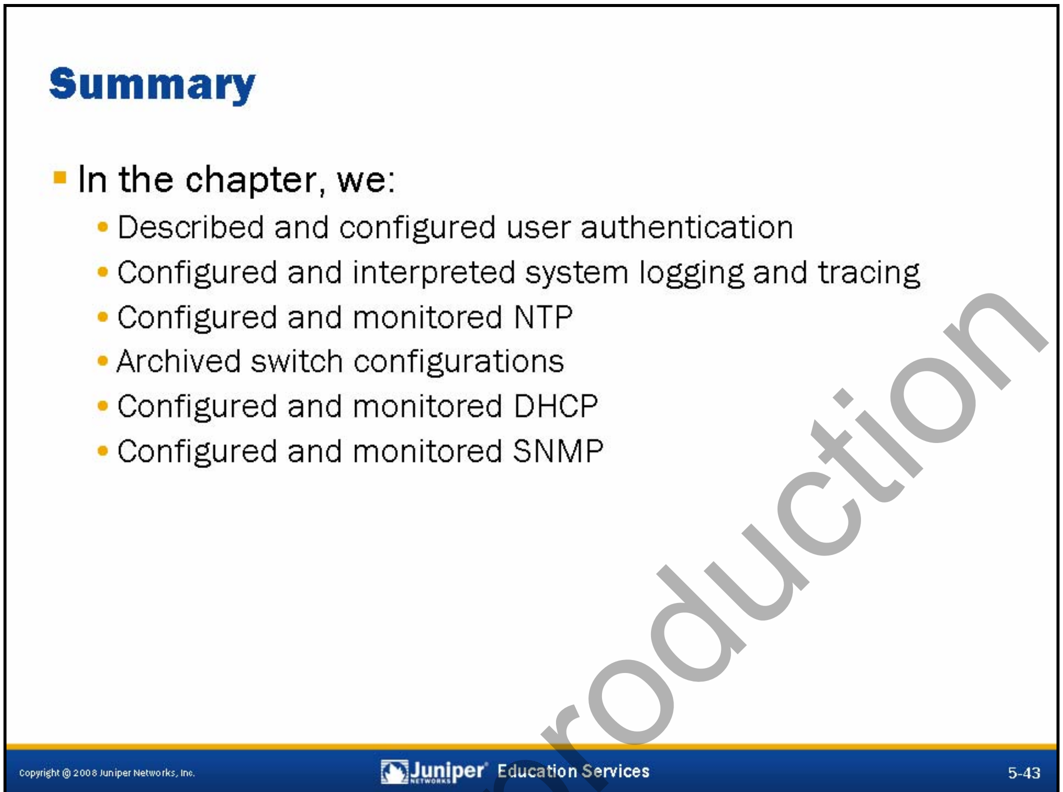
Sample SNMP Configuration

The slide shows a sample SNMP configuration using some common SNMP configuration options. When configuring contact information, you must be as specific as possible. This information comes in handy when trying to resolve issues with a network device. The example restricts SNMP access to the 10.210.14.0/24 network with read-only authorization. The example also shows the configuration of an SNMP trap group, necessary for the delivery of SNMP traps to an NMS.



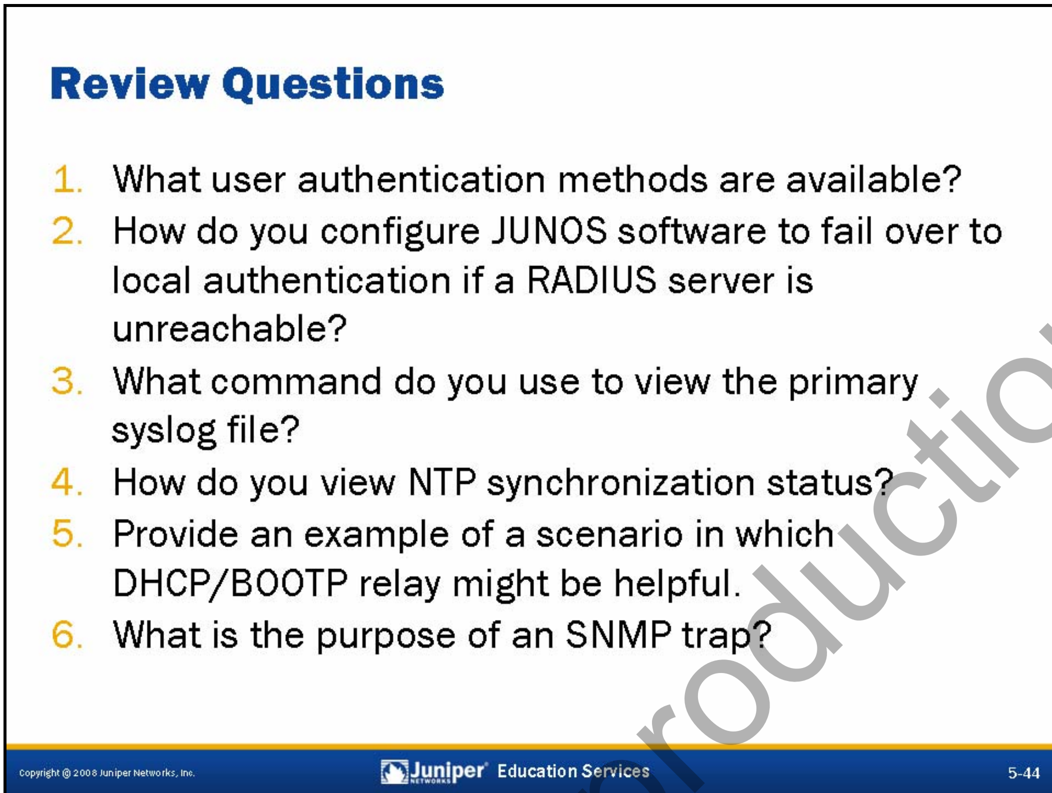
Monitoring SNMP Operation

An NMS or host provides the interface for most SNMP monitoring. To monitor SNMP operation from an EX-series switch, you can use traceoptions, system logging, and various **show snmp** commands. When a trap condition occurs, some traps are logged if the system logging is configured with the appropriate facility and severity levels, regardless of whether a trap group is configured. The sample **show** command output on the slide illustrates that you can also issue standard SNMP manager commands to view agent OID values. You can specify the OIDs in ASCII text format or dotted-decimal notation.



This Chapter Discussed:

- User authentication methods and configuration;
- Configuring and interpreting system logging and tracing;
- NTP configuration and operation;
- Archiving switch configurations on remote devices;
- Configuring and monitoring the DHCP; and
- Configuring and monitoring the SNMP.



Review Questions

1.

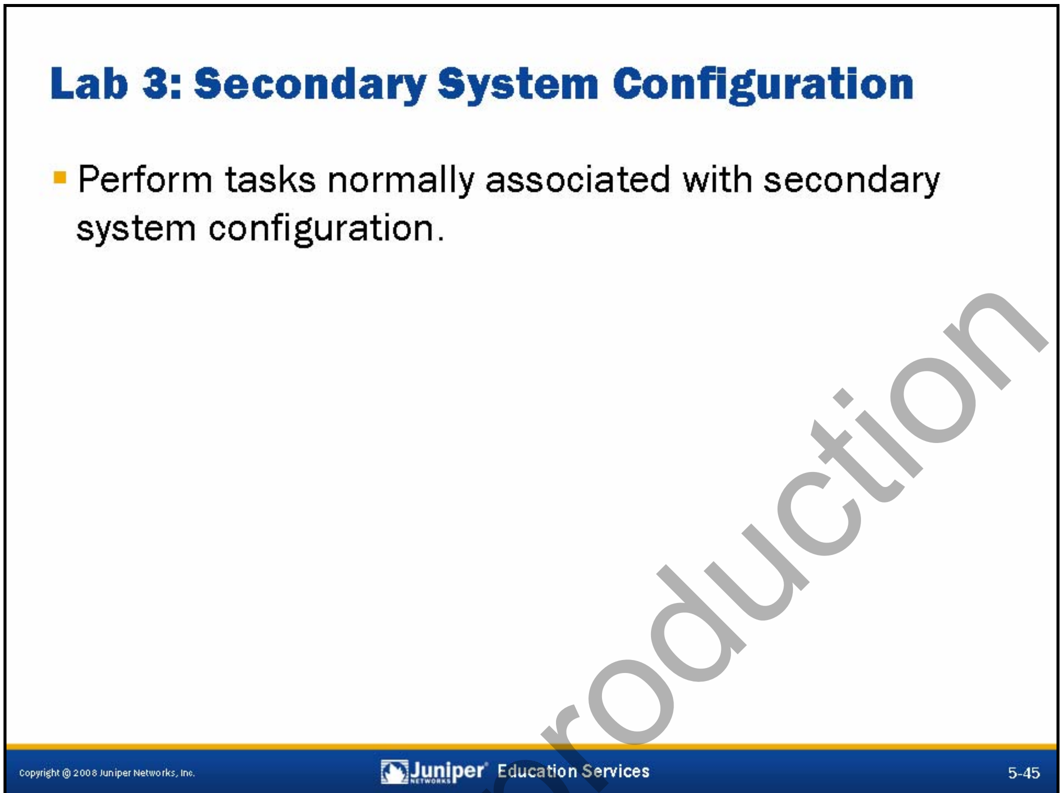
2.

3.

4.

5.

6.

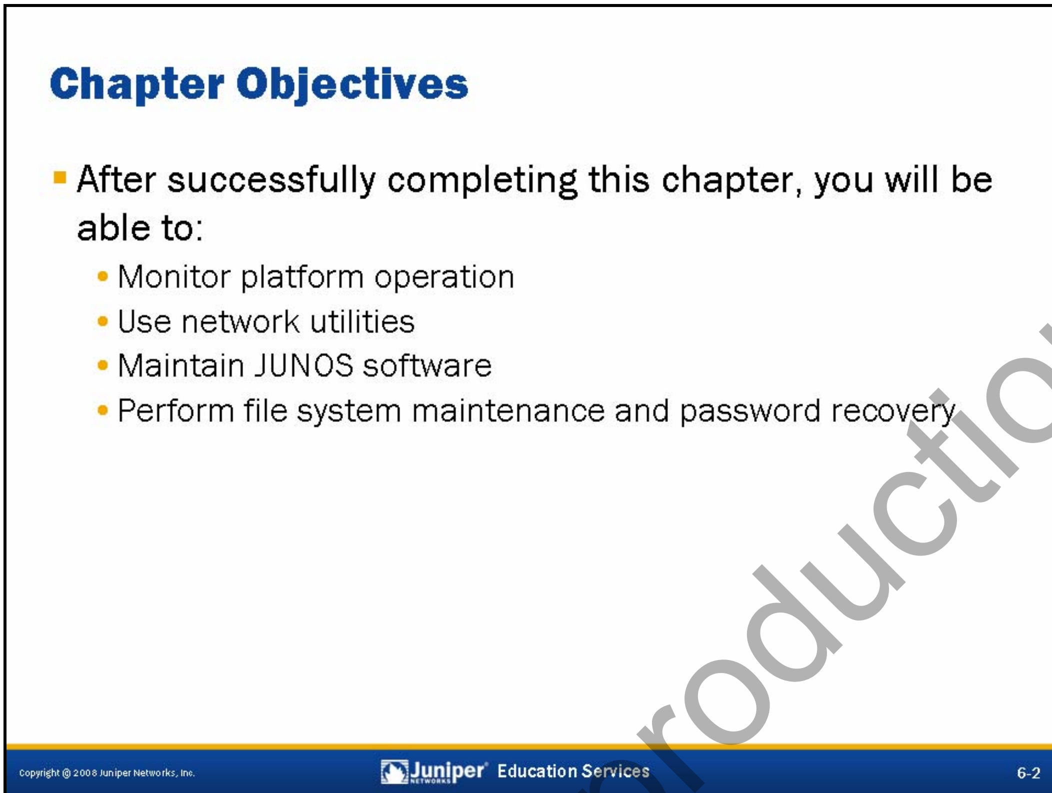


Lab 3: Secondary System Configuration

The slide provides the objective for this lab.

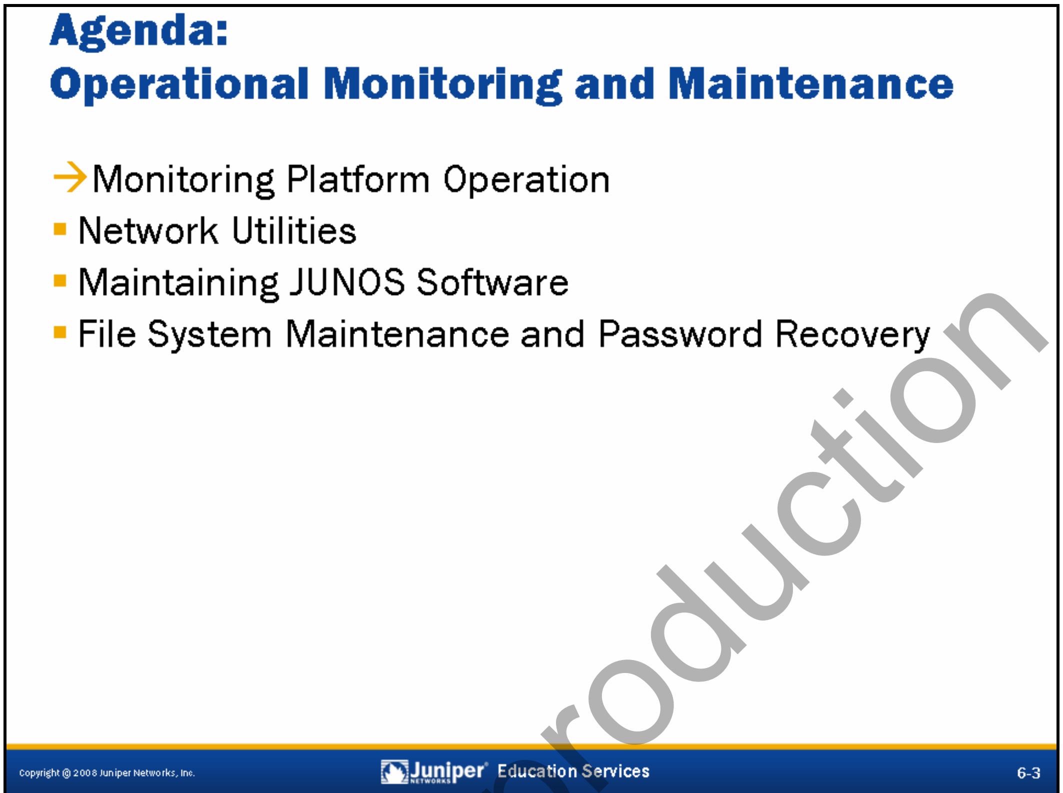
Operating Juniper Networks Switches in the Enterprise

Chapter 6: Operational Monitoring and Maintenance



This Chapter Discusses:

- Monitoring platform operation;
- Using network utilities;
- Maintaining JUNOS software; and
- Performing file system maintenance and password recovery.



Monitoring Platform Operation

The slide lists the topics we cover in this chapter. We discuss the highlighted topic first.

EX-series Front Panel LEDs (1 of 2)

- Visual Indicators summarize platform status

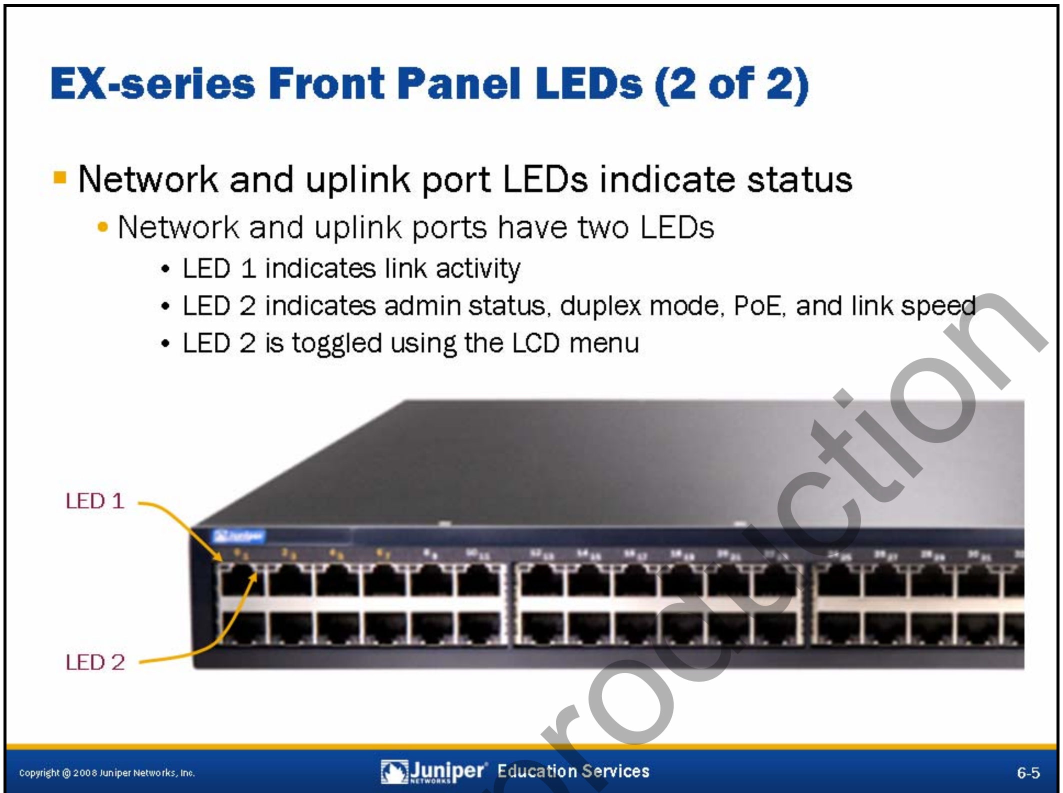
Front Panel LEDs	On Steadily	Blinking
ALM (Alarm)	If the LED is lit red, an alarm is present.	Not Applicable—If ALM LED is unlit, no alarm is present.
SYS (System)	If the SYS LED is lit steadily green, JUNOS software is loaded on the switch.	If the SYS LED is blinking green, the switch is booting JUNOS software.
MST (Master)	If the MST LED is lit steadily green, the switch is the master of the Virtual Chassis configuration. (This LED is always lit steadily on EX 3200 models).	If the MST LED is blinking green, the switch is the backup of the Virtual Chassis configuration. MST LED remains unlit on member (*line card*) switches.

 Juniper Education Services 6-4

EX-Series Status Summary: Part 1

The front panel LEDs on EX-series platforms provide a summary of the switch's status. The LEDs are located on the far right side of the panel, next to the LCD. These LEDs include the following indicators:

- *Alarm LED*: This LED lights steadily red when a system alarm is present and remains unlit in the absence of alarms.
- *System LED*: This LED blinks green while the JUNOS kernel is booting and lights steadily green after the bootup process is complete.
- *Master LED*: This LED lights steadily green when the switch is acting as the master in a Virtual Chassis configuration. It blinks green when the switch is acting as the backup in a Virtual Chassis configuration. On an EX 3200 switch this LED remains lit steadily green at all times.



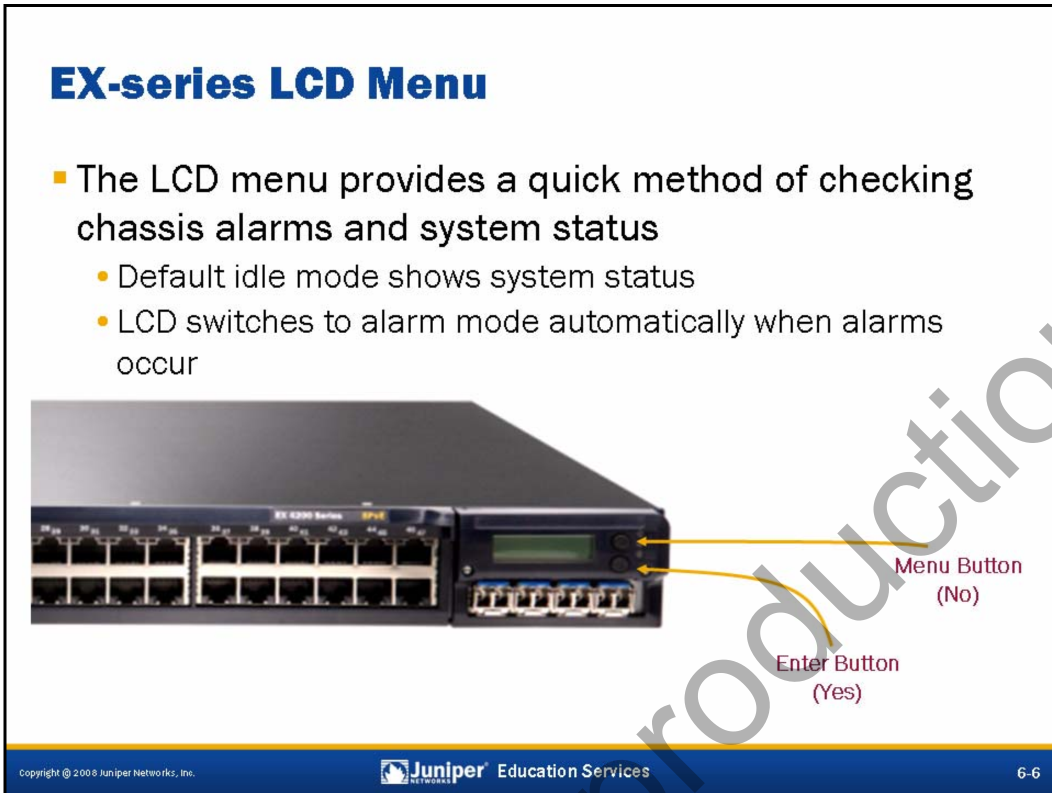
EX-series Status Summary: Part 2

The EX-series front panel provides two LEDs for each network port and uplink port. LED 1, the LED on the left side of the port, blinks green when the link is up and active. It lights steadily green when the link is up but there is no current link activity.

LED 2, the LED on the right side of the port, can indicate multiple settings. Pressing the LCD Enter button toggles LED 2 between administrative status (ADM), duplex mode (DPX), PoE status (POE), and speed (SPD). The following table lists these states:

LED 2 Port Status

ADM	DPX	POE	SPD
Green: Port is enabled. *Unlit*: Port is disabled.	*Green*: Port is full duplex. *Unlit*: Port is half duplex. (Always lit green for SFP and XFP uplink ports because uplink ports are always set to full duplex.)	*Green*: PoE is enabled. *Amber*: PoE negotiation failure. *Unlit*: PoE is not enabled. (Always unlit for SFP and XFP uplink ports because they do not support PoE.)	1/2/3 blinks per second for 10/100/1000 Mbps, respectively. (Lit green steadily for SFP/XFP uplink ports at full speed; unlit for SFP ports set at 10/100 Mbps)



Monitoring with the LCD Menu

The LCD menu on both the EX 3200 and EX 4200 devices provides a quick, menu-driven interface for monitoring and performing tasks on the switch. There are two navigation buttons: the bottom button is used to make or confirm selections, and the top menu button switches between the following two main menus:

- *Maintenance menu*: This menu contains the following options:
  - *System reboot*: This option reboots the switch.
  - *Factory default*: This option resets the switch configuration to its factory-default state.
  - *Enter EZsetup*: This option starts the J-Web EZsetup feature if the switch is in a factory-default state. (This option shows on the LCD menu only after the switch is put in a factory-default state.)
- *Status menu*: This menu contains the following options:
  - *VCP status*: This option displays the status of Virtual Chassis ports.
  - *PWR status*: This option displays the status of power supplies.
  - *FANS/TEMP status*: This option displays the environmental state.

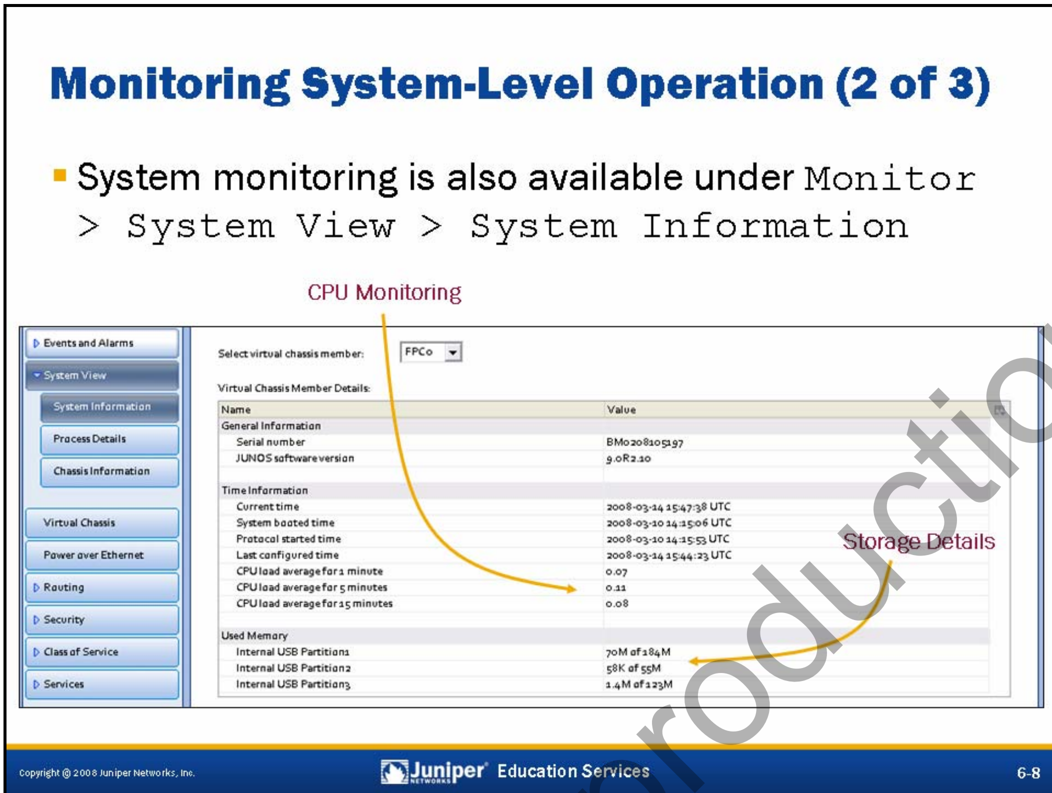
By default, the LCD rests in idle mode. If an alarm is triggered, the LCD switches to alarm mode and displays errors. You can also view the output of the LCD with the command-line interface (CLI) using the **`show chassis lcd`** command.

Monitoring Overall System Operation: Part 1

The J-Web `Dashboard` tab provides an overview of the switch's operational state. The `Dashboard` tab includes the following information:

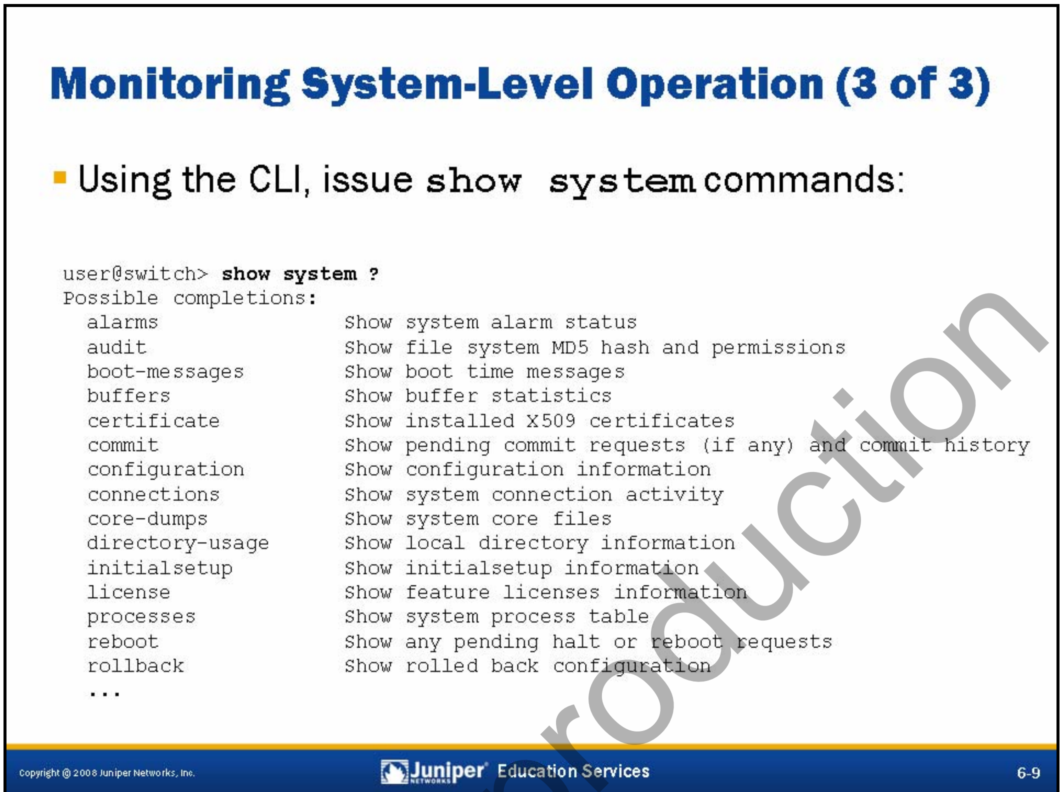
- `System Information`: This section shows the switch's hostname, model number, inventory details, software version, the last time the system was booted, and the last time the system was configured.
- `Health Status`: This section displays the current memory utilization, system temperature, CPU load, and fan status.
- `Capacity Utilization`: This section gives the total number of utilized versus supported interface ports, MAC table entries, and VLANs.
- `Alarms`: This section displays red or yellow alarms, if any alarms are present.

The `Dashboard` tab also provides a status of all the interfaces and a graphical depiction of the current LCD contents.



Monitoring Overall System Operation: Part 2

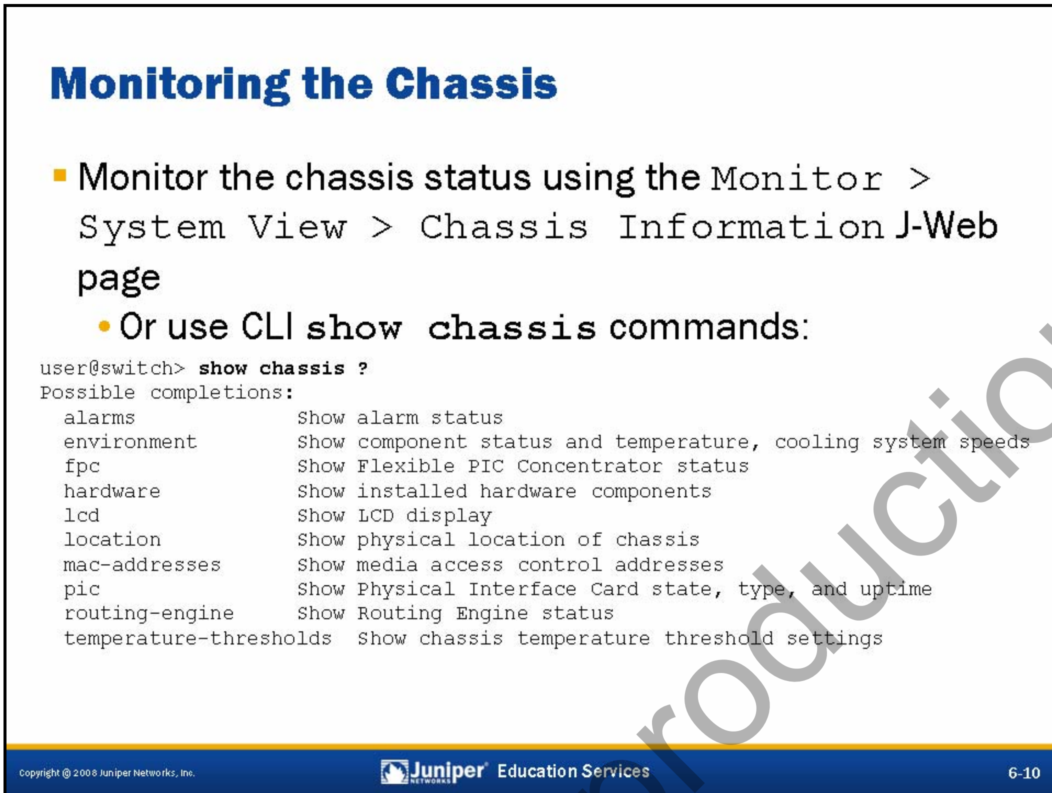
The `Monitor > System View` page provides more system information, including the system serial number, software version, system uptime information, compact-flash device usage, and user details.



Monitoring Overall System Operation: Part 3

You can obtain most system information using **show system *argument*** commands. The following arguments are some of the most common:

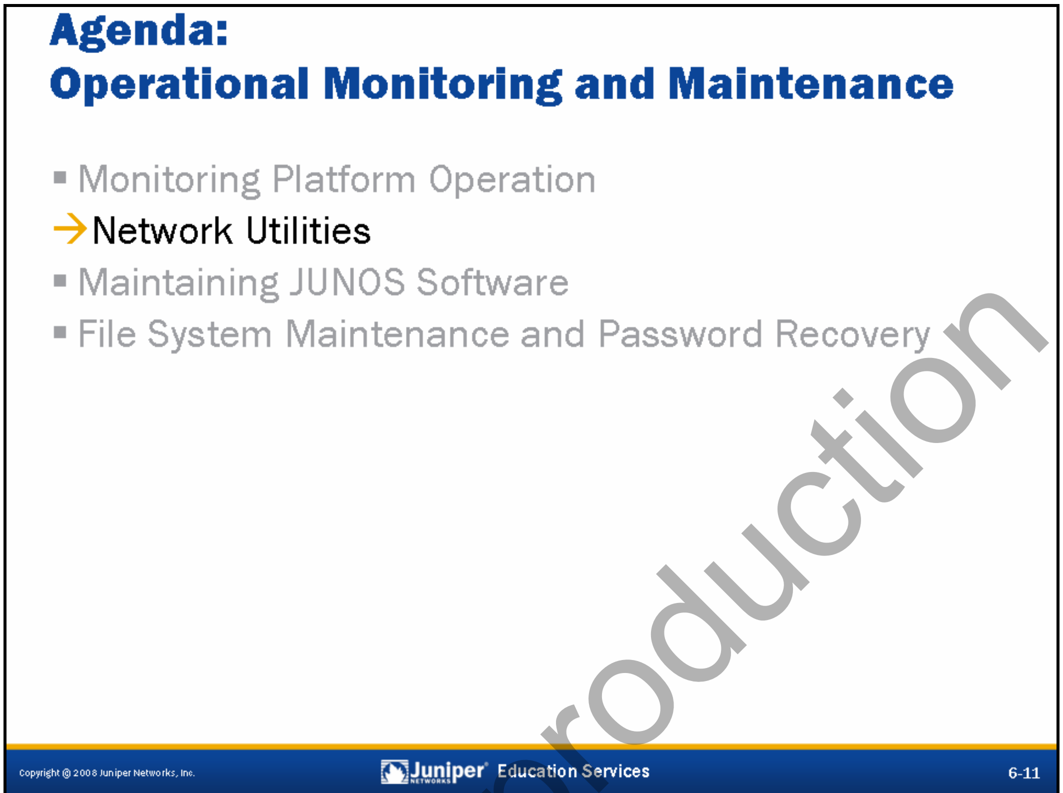
- **alarms**: This argument displays current system alarms.
- **boot-messages**: This argument displays the messages seen during the last system boot.
- **connections**: This argument displays the status of local TCP and UDP connections.
- **statistics**: This argument provides options for viewing various protocol statistics.
- **storage**: This argument displays the status of the file system storage space.



Monitoring the Chassis

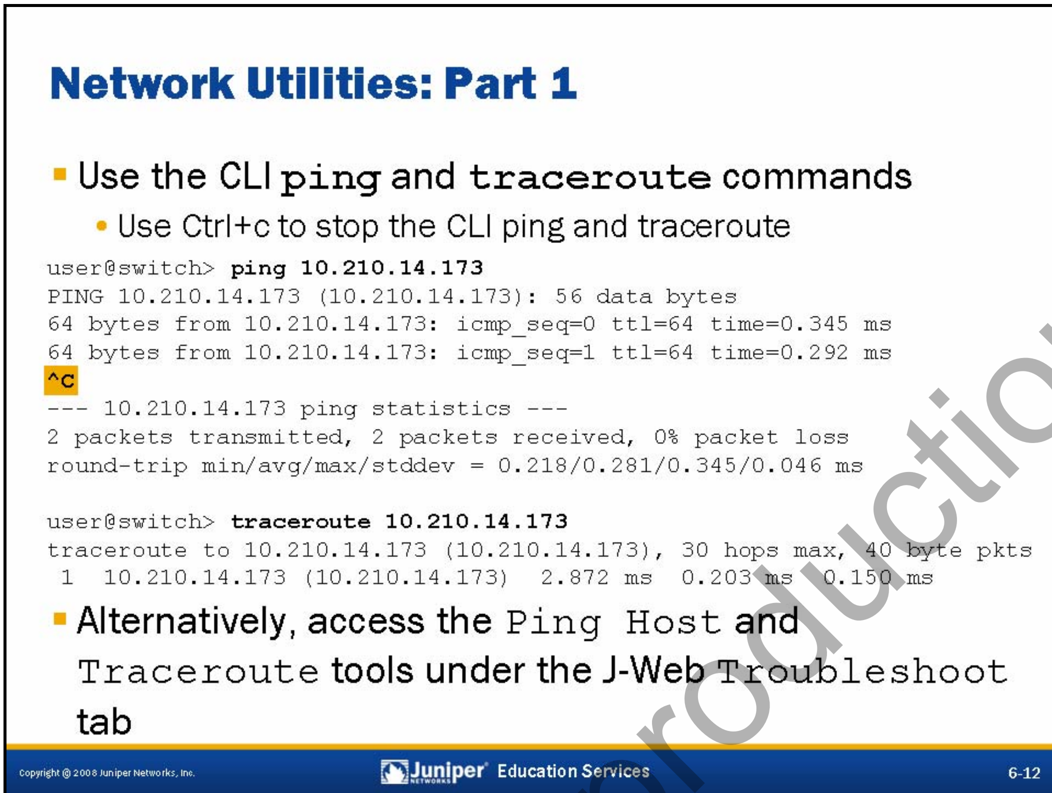
The J-Web `Monitor > System View > Chassis Information` page provides a convenient summary of the chassis environment. You can check the status of multiple switches configured as a Virtual Chassis system by using the drop-down box. For system components, a yellow alarm occurs when the temperature reaches 80 degrees Centigrade (176 degrees Fahrenheit), and a red alarm occurs when system components reach 95 degrees Centigrade (203 degrees Fahrenheit). This threshold is lowered by 10 degrees Centigrade when the switch detects a faulty cooling fan.

Chassis environmental information is also available using the CLI **`show chassis environment`** command.



Network Utilities

The slide highlights the topic we discuss next.



Ping and Traceroute Utilities

The JUNOS CLI provides ping and traceroute utilities. You can use these tools to determine general network reachability and the path that packets take to reach a destination. You can use various arguments with the **`ping`** and **`traceroute`** commands, such as source IP address and packet size, to further assist in problem isolation. These tools are also available through J-Web under the `Troubleshoot` tab.

Network Utilities: Part 2

- Use the CLI `monitor traffic` command to decode packets, or access the packet capture utility under the J-Web `Troubleshoot` tab
  - Displays traffic only originating or terminating on the switch
    - Use the **interface** ***interface-name*** option to capture local traffic from a specific interface
    - The best way to perform analysis of Layer 2 header information in JUNOS software is using the **layer2-headers** option
    - Use the **no-resolve** knob to avoid DNS reverse-lookup delays
    - Use **matching** option to filter packets
    - Packet capture can be saved for packet analysis (hidden **write-file** and **read-file** options)

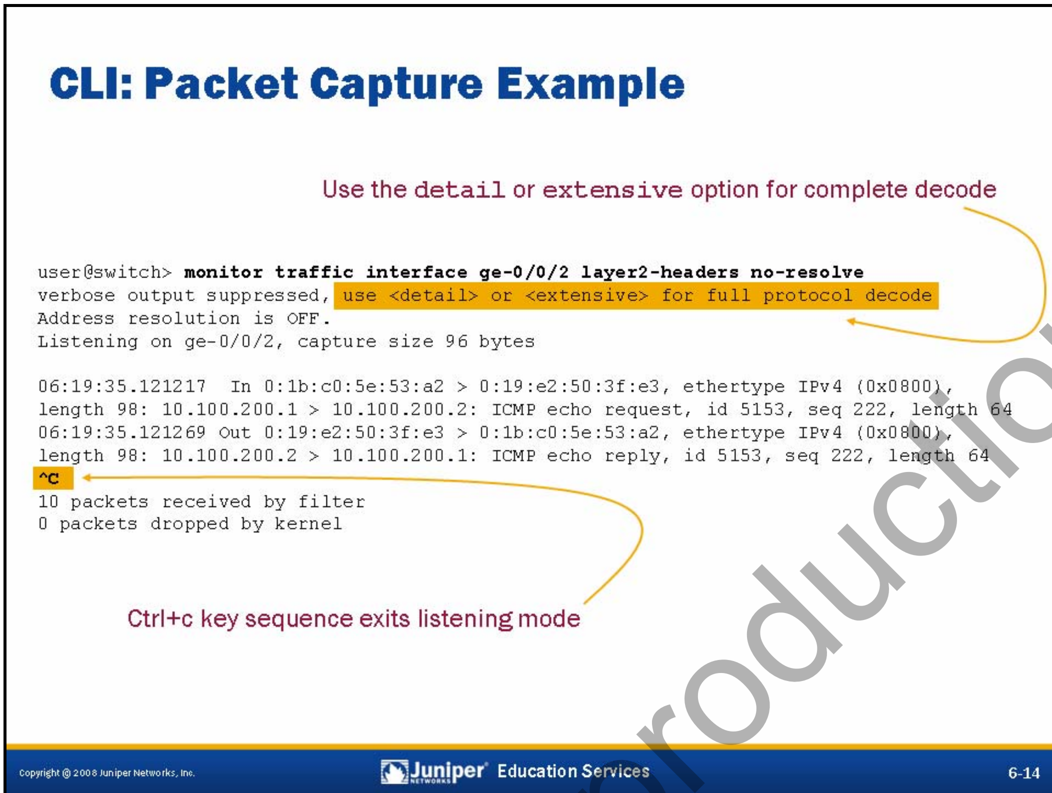
```
user@switch> monitor traffic interface ge-0/0/0 layer2-headers no-resolve
```

 Juniper Education Services 6-13

Monitoring Traffic

The CLI's **monitor traffic** command and J-Web's `Troubleshoot > Packet capture` utility provide access to the tcpdump tool. This tool monitors traffic that originates or terminates on the local Routing Engine (RE). If you do not specify an interface, the me0 management interface will be monitored. This capability provides a way to monitor and diagnose problems at Layer 2 using the **layer2-headers** argument. You can match on packet fields using the **matching** option and save packet captures for analysis from a third-party packet decoder such as Ethereal or Wireshark using the **write-file** option.

The **write-file** option is hidden and should be used with caution. If used improperly, this command option could fill the storage space on the switch.



Packet Capture Example

This slide provides an example of the CLI **monitor traffic** command. Note that to stop a packet capture, use the Ctrl+c keyboard sequence.

Network Utilities: Part 3

- Access Telnet, SSH, and FTP client commands from the CLI

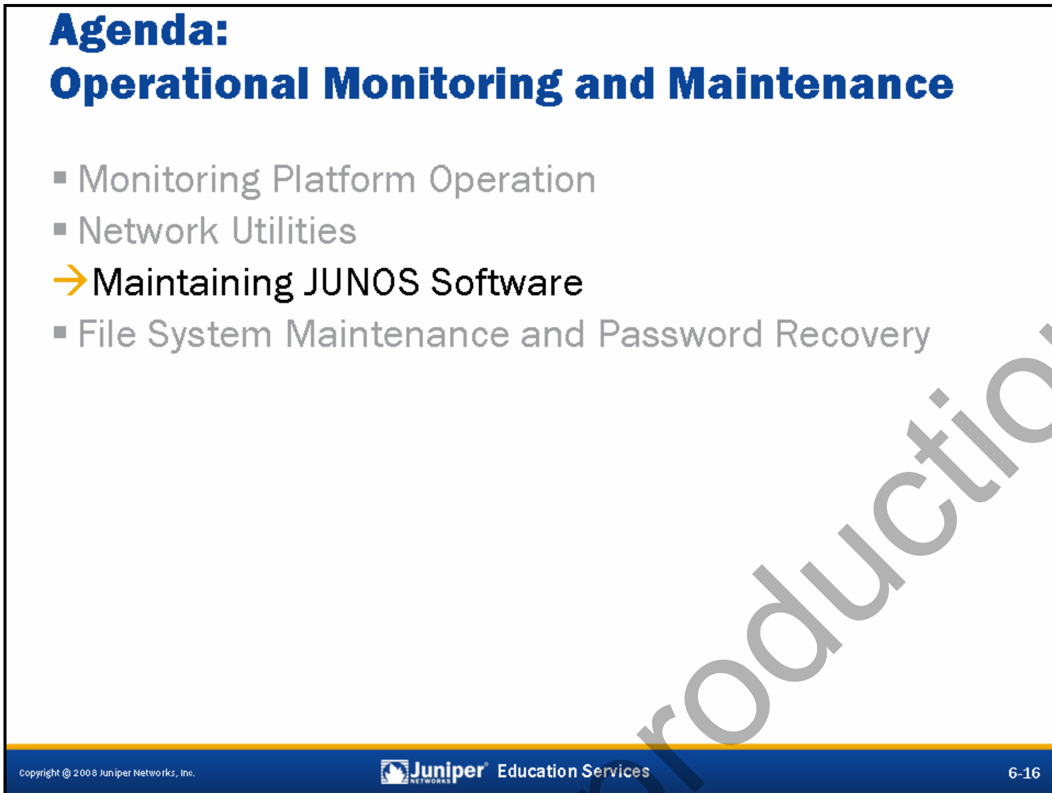
```
user@switch> telnet ?
Possible completions:
  <host>                 Hostname or address or remote host
  8bit                   Use 8-bit data path
  bypass-routing         Bypass routing table, use specified interface
  inet                   Force telnet to IPv4 destination
  interface              Name of interface for outgoing traffic
  no-resolve             Don't attempt to print addresses symbolically
  port                   Port number or service name on remote host
  source                 Source address to use in telnet connection

user@switch> telnet 127.0.0.1
Trying 127.0.0.1...
Connected to 127.0.0.1.
Escape character is '^]'.

switch (ttyp0)

login: user
Password:
. . .
```


Juniper Education Services

6-15

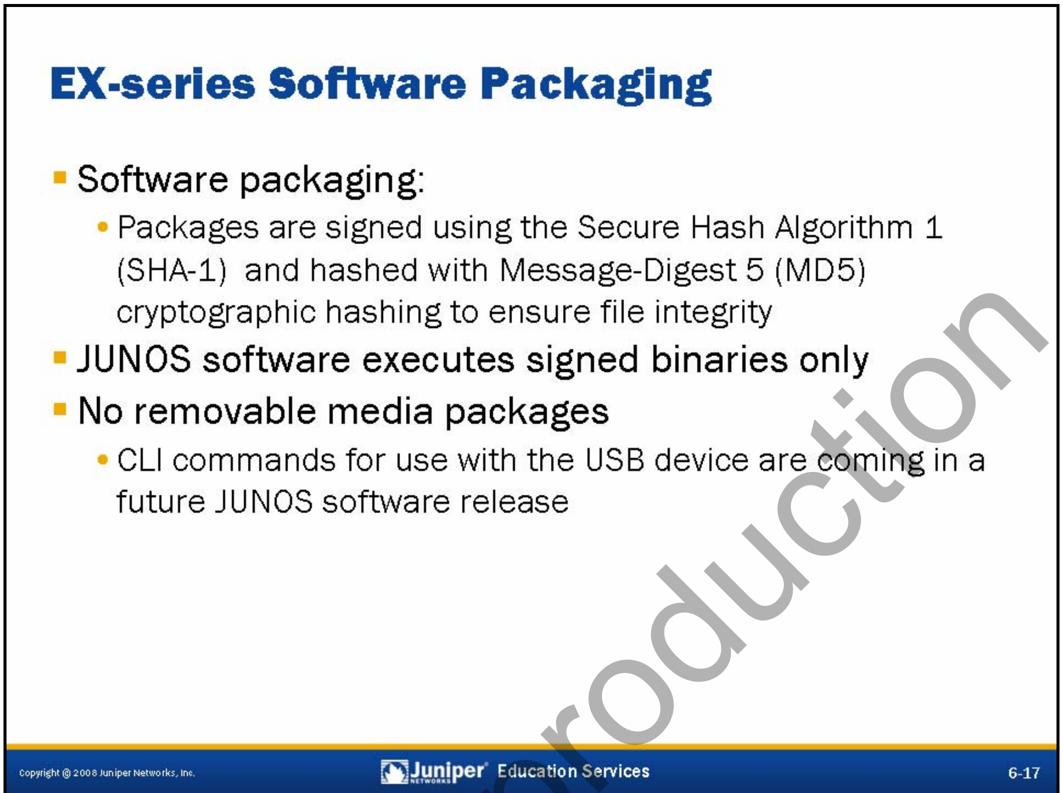
Network Utilities

The CLI supports powerful Telnet, SSH and FTP clients. These clients support various arguments that tailor their specific operations.



Maintaining JUNOS Software

The slide highlights the topic we discuss next.



EX-series Software Packaging

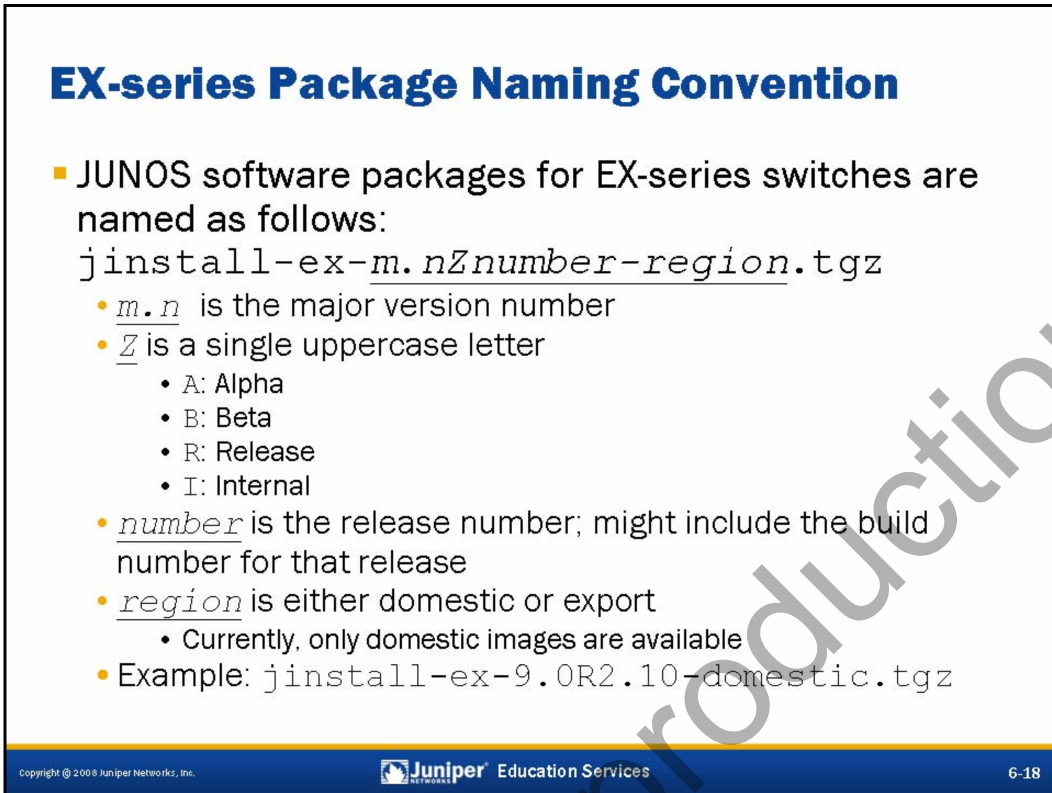
Although JUNOS software for EX-series switches is built from the same code base as M-series, J-series, T-series, and MX-series software, it is packaged differently. You can install the JUNOS software package for EX-series switches only on an EX-series switch.

Signed Binaries

Juniper Networks EX-series switches run binaries supplied only by Juniper Networks. Each JUNOS software image includes a digitally signed manifest of executables, which are registered with the system only if the signature can be validated. JUNOS software does not execute any binary without a registered fingerprint. This feature is designed to protect the system against unauthorized software and activity that might compromise the integrity of your switch.

No Removable Media Packages

JUNOS software for EX-series switches comprises several different components, and you can see these components listed with the CLI **`show version detail`** command. However, there is no separate removable media package for individual components or for universal serial bus (USB) compact-flash device use. We will be adding commands for using the USB storage device in upcoming releases of JUNOS software.

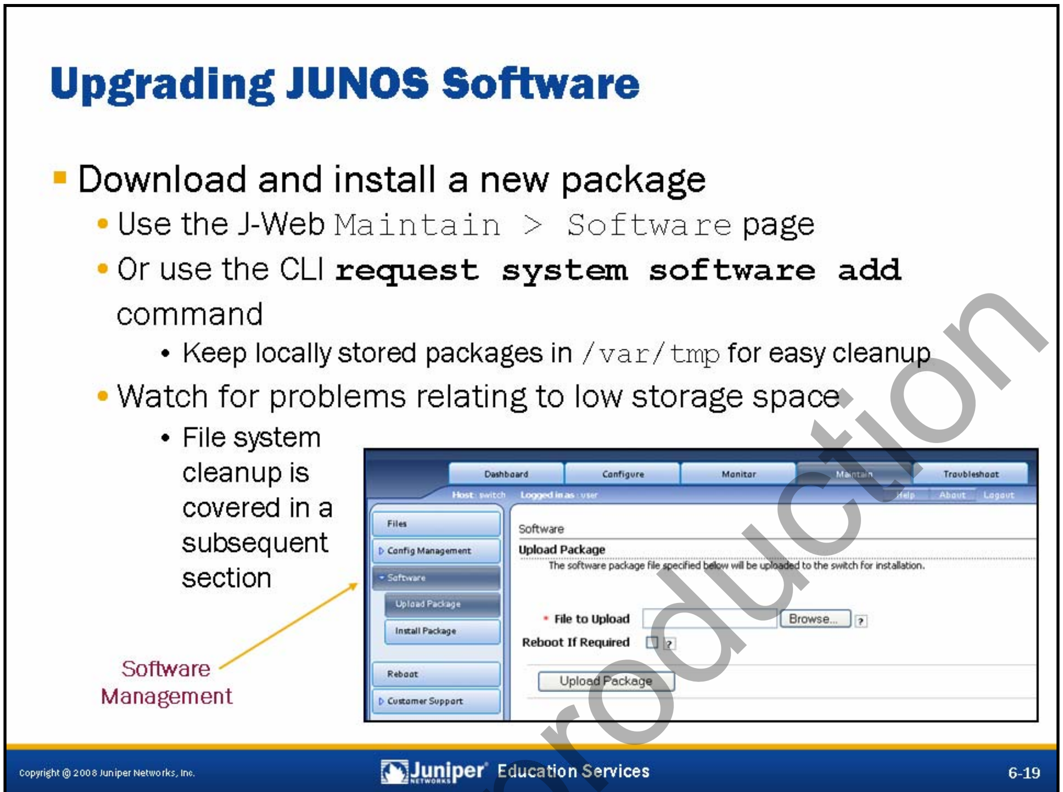


Package Naming

A JUNOS software package has a name in the format *package-m.nZnumber-region*.tgz:

- *package* is a description of the software contents. This description is jinstall-ex for all EX-series software images.
- *m.n* are two integers that represent the software release number.
- *Z* is a capital letter that indicates the type of software release. In most cases, it is an `R` to indicate that this is released software. If you are involved in testing prereleased software, this letter might be an `A` (for alpha-level software), `B` (for beta-level software), or `I` (for internal, test, or experimental versions of software).
- *number* represents the version of the software release and includes the internal build number for that version. For example, `jinstall-ex-9.0R2.10-domestic.tgz` indicates a JUNOS software bundle associated with version 9.0, release 2, build 10.
- *region* will be either `domestic` or `export`. Currently, only domestic images are available for the EX-series switches. Domestic versions support strong encryption, whereas export versions do not.

Again, ensure that you always load EX-series bundles on EX-series platforms only.



Installing Software

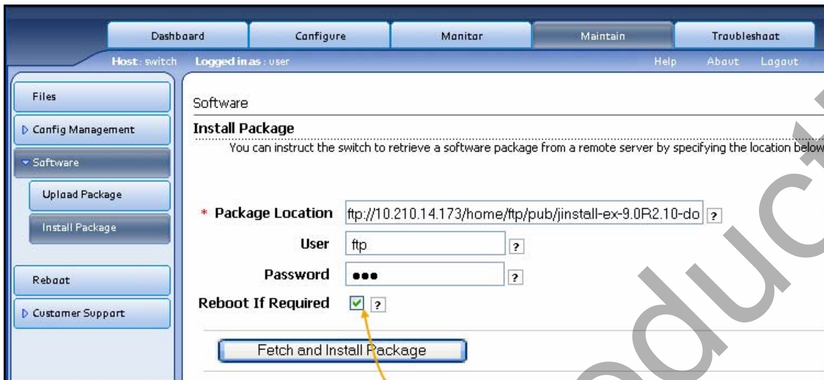
You can upgrade JUNOS software from either J-Web or the CLI. You should store JUNOS bundles in the switch's `/var/tmp` directory. Temporary software files stored in this directory are automatically deleted after an upgrade is completed to conserve space.

Although the compact-flash cards are 1 gigabyte, it is a good practice to check available storage capacity before downloading a new JUNOS software bundle. You can view compact-flash device storage details at the J-Web `Maintain > Files` page or with the CLI **`show system storage`** command. Note from the following output that the compact-flash drive has multiple partitions:

```
user@switch> show system storage
Filesystem              Size        Used       Avail  Capacity   Mounted on
/dev/da0s2a              86M         70M        9.4M       88%  /
...
/dev/da0s3e              31M         58K         29M        0%  /config
/dev/da0s2f              67M        8.5M         53M       14%  /var
/dev/da0s3d             154M       10.0K        141M        0%  /var/tmp
procfs                  4.0K        4.0K          0B      100%  /proc
```

Upgrade Example (1 of 2)

- Use the J-Web `Maintain > Software > Install Package` page to install a package from a remote host
  - An FTP-based URL is shown in this example:

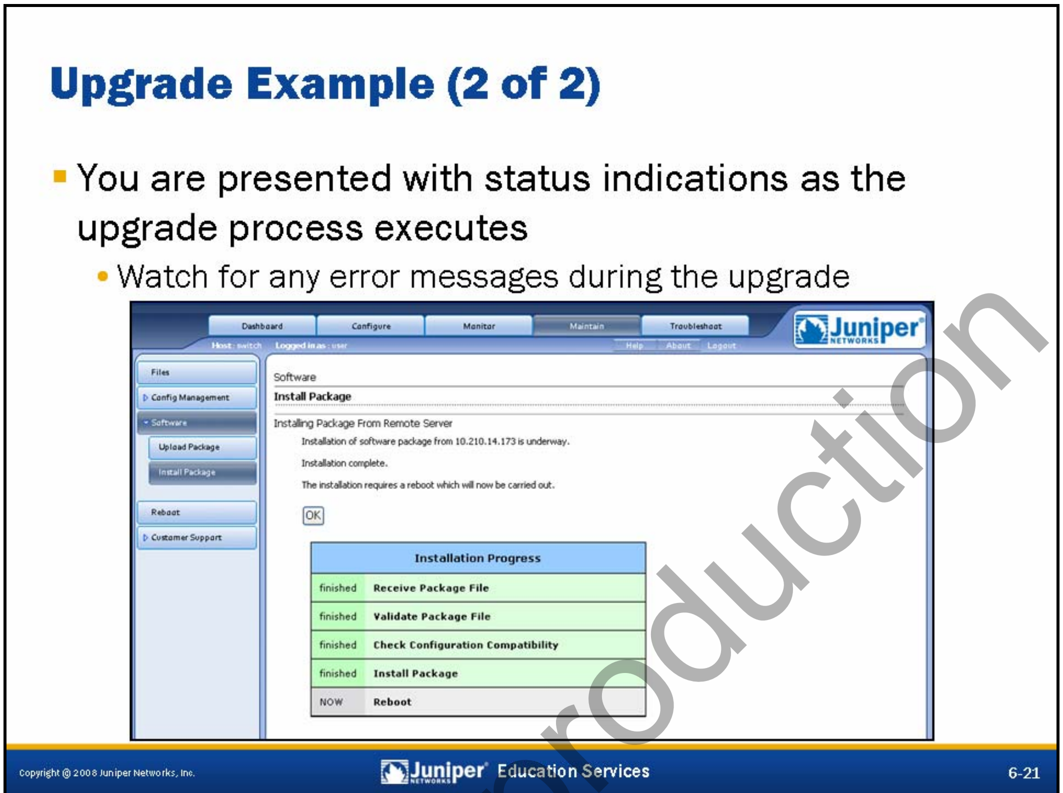


A reboot is required to activate new software

Installing Software from a Remote Server with J-Web

Use the J-Web `Maintain > Software > Install Package` page to specify a remote URL that contains a JUNOS software bundle to download and install. To activate the new software you must reboot the switch. You can perform this reboot directly from the `Maintain > Software > Install Package` page by using the `Reboot If Required` check box, or you can reboot later using the `Maintain > Reboot` page. Alternatively, you can remotely install software by using the CLI **`request system software add`** command.

You can also use the J-Web `Maintain > Software > Upload Package` page to copy software directly from your PC to the switch.

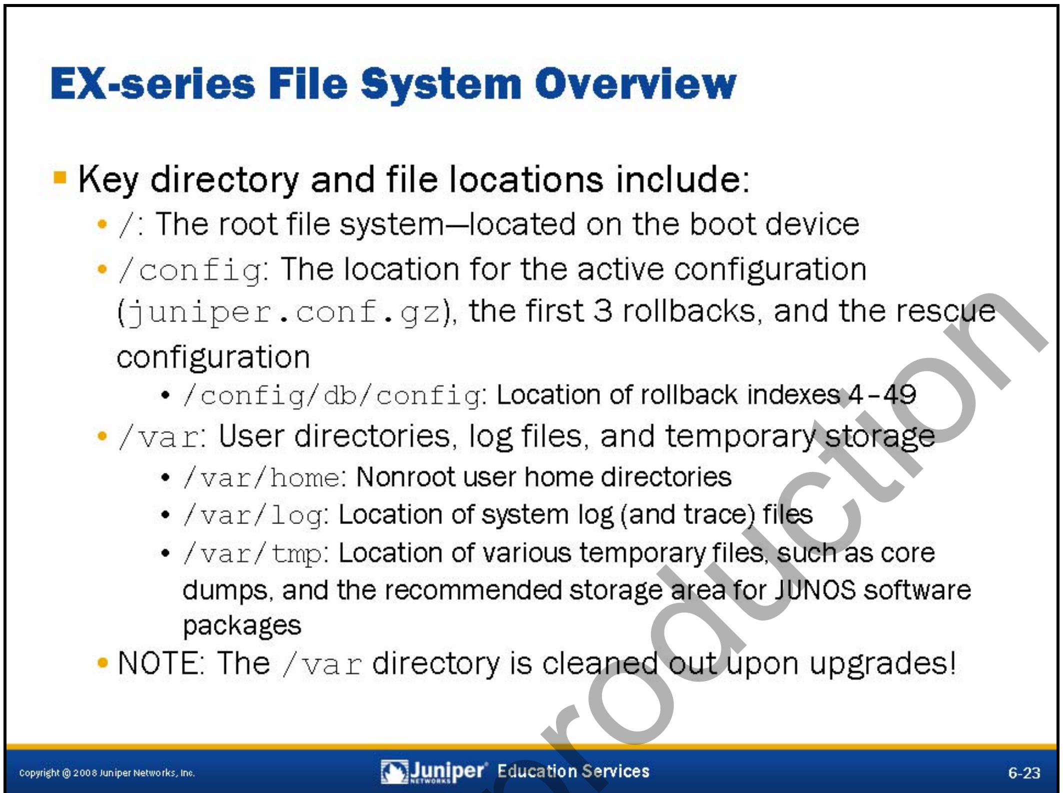


J-Web Software Upgrade Status

Once the JUNOS software is installed, you are notified that the switch is rebooting to complete the installation. Use a console connection to view details of the upgrade process. Watch for any error messages indicating a problem with the upgrade.

File System Maintenance and Password Recovery

The slide highlights the topic we discuss next.



Overview of the JUNOS Software File System

The following list shows the key directories and file locations:

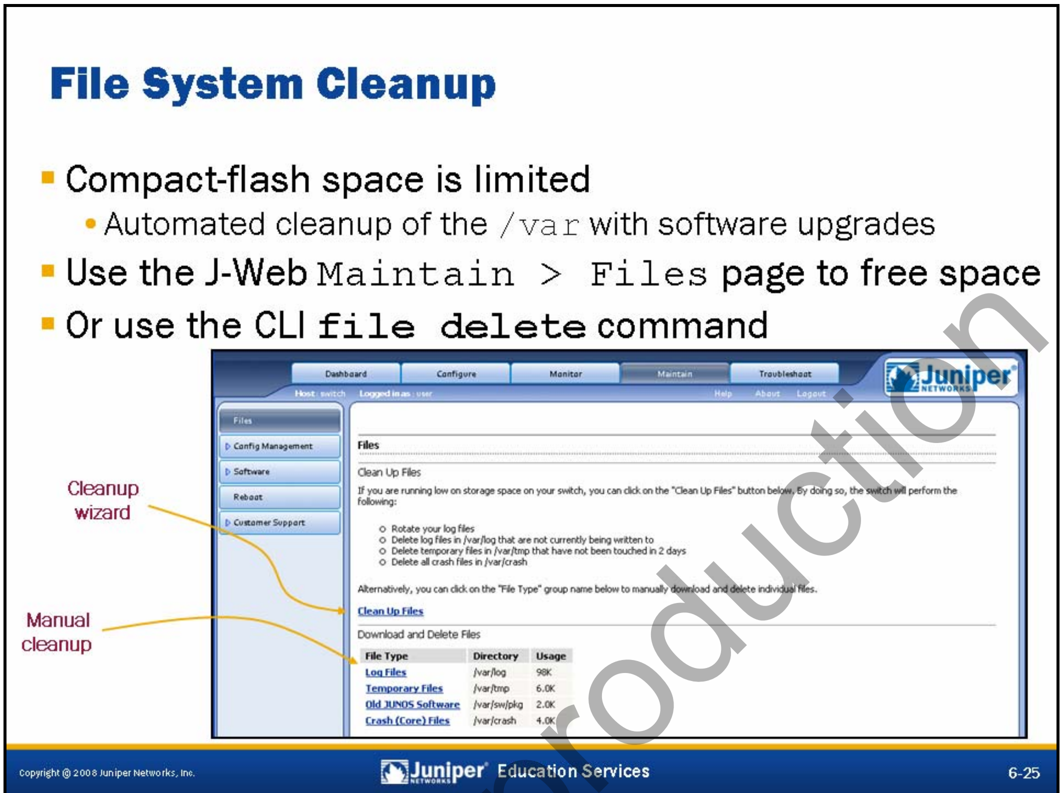
- `/`: This is the root file system located on the switch's boot device, which is normally the primary compact-flash drive.
- `/config`: This directory is located on the boot device and contains the current operational switch configuration and the last three committed configurations, as well as the rescue configuration, if one is saved.
  - `/config/db/config`: This subdirectory contains up to 46 additional previous versions of committed configurations, which are stored in the files `juniper.conf.4.gz` through `juniper.conf.49.gz`.

Continued on next page.

Overview of the JUNOS Software File System (contd.)

- `/var`: This directory is also located on the boot device. This file system contains the following subdirectories:
  - `/var/home`: This subdirectory contains users' home directories, which are created when you create user access accounts. For users using SSH authentication, a `.ssh` directory, which contains their SSH key, is placed in their home directory. When users save or load configuration files, those files are loaded from their home directory unless the users specify a full path name.
  - `/var/log`: This subdirectory contains system log and tracing files.
  - `/var/tmp`: This subdirectory contains daemon core files (if present) and is the recommended location for storing temporary files.

Note that to keep space free on the file system, JUNOS software deletes the contents of the `/var` directory upon a successful software upgrade, which makes it especially important to back up wanted files and logs externally.



Limited Space

The compact-flash drive used for primary storage on EX-series switches, although 1 gigabyte in size, is somewhat limited in comparison to the hard drives found on M-series and T-series routers. Although the switch will continue to forward traffic if the compact-flash drive becomes full, you will lose log messages and be unable to modify the configuration until space is freed. You can monitor usage in the `Used Memory` section of the J-Web `Monitor > System View > System Information` page or by using the CLI **`show system storage`** command.

File System Cleanup

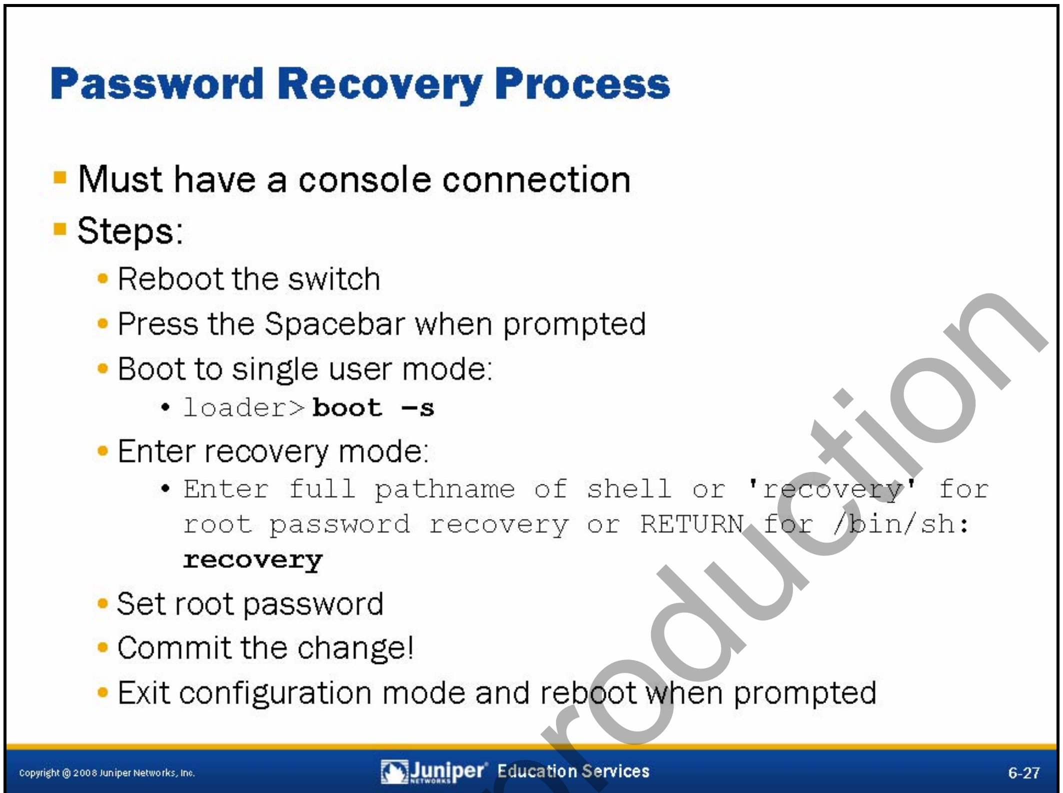
JUNOS software for EX-series devices automatically removes unused software files following a successful upgrade. Should you need to free additional space from the compact-flash drive, you can use the J-Web `Clean Up Files` wizard found on the `Maintain > Files` page. You can also manually remove files using the other links on the `Maintain > Files` page.

Continued on next page.

File System Cleanup Using the CLI

You can also manually free storage space using the CLI. Use the **file delete *file-name*** command to remove unnecessary files. Like many operational-mode commands, wildcard characters are supported as shown in the following example. Do not forget to include the directory path!

```
user@switch> file list /var/tmp

/var/tmp:
.snap/
sampled.pkts
shm_ipc_cs2*
test
test1

user@switch> file delete /var/tmp/te*

user@switch> file list /var/tmp

/var/tmp:
.snap/
sampled.pkts
shm_ipc_cs2*
```



Password Recovery Requires Console Connection

If you become locked out of the switch, you can recover the root password. As a security precaution, the recovery can be performed using only the console connection.

Password Recovery Steps

The following steps list the process for recovering the root password.

1. Obtain console access and reboot the system. Watch as the system boots, and press the Spacebar when prompted during the boot loader process. When the system presents a `loader>` prompt, enter **`boot -s`** to boot into single-user mode as shown:

```
FreeBSD/PowerPC U-Boot bootstrap loader, Revision 1.0
(marcelm@apg-bbuild01.juniper.net, Tue Oct  2 19:15:34 PDT 2007)
Memory: 512MB
Loading /boot/defaults/loader.conf
/kernel data=0x951858+0x5d53c syms=[0x4+0x811d0+0x4+0x85ee0]

Hit [Enter] to boot immediately, or space bar for command prompt.
<user presses Spacebar>
```

Continued on next page.

Password Recovery Steps (contd)

```
Type '?' for a list of commands, 'help' for more detailed help.
loader> boot -s
Kernel entry at 0x1000100 ...
GDB: no debug ports present
KDB: debugger backends: ddb
KDB: current backend: ddb
Copyright (c) 1996-2008, Juniper Networks, Inc.
...
```

2. The system performs a single-user boot-up process and prompts you to run the recovery script, enter a shell pathname, or press Enter for a default shell. Enter **recovery** at this point.

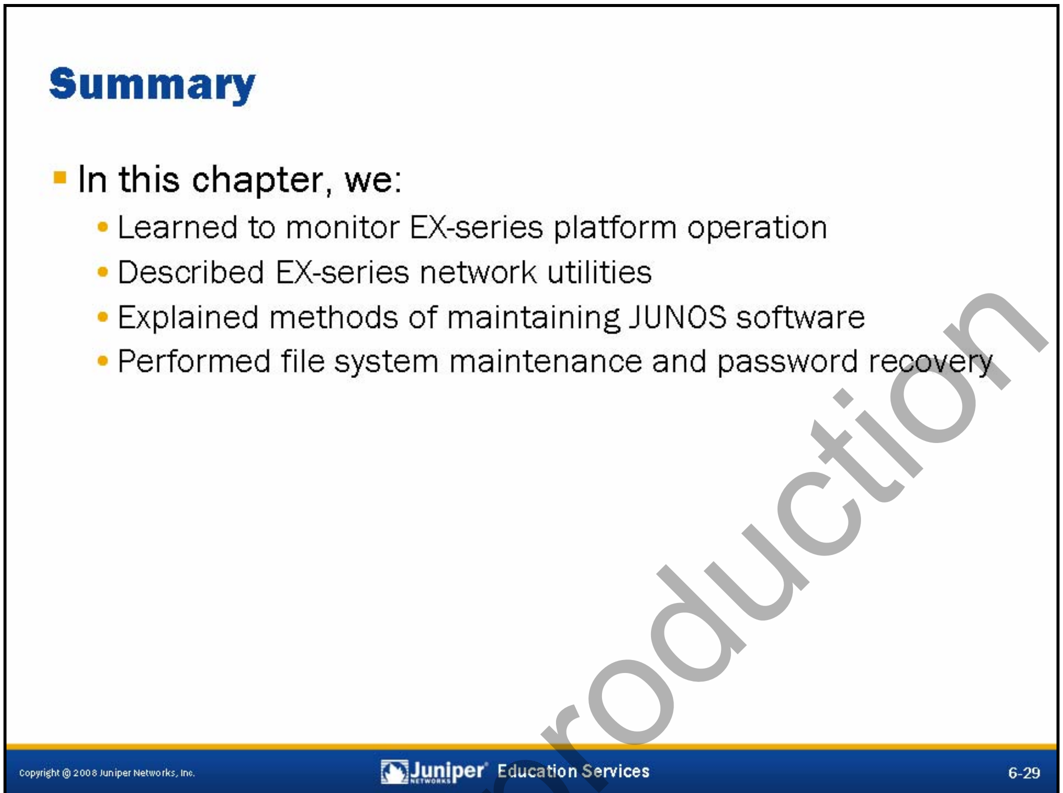
```
Mounted jbase package on /dev/md0...
System watchdog timer disabled
Enter full pathname of shell or 'recovery' for root password recovery or RETURN
for /bin/sh: recovery
```

3. After a series of messages, the CLI starts and you are presented with an operational mode command prompt. At this point, you can enter configuration mode and reset the root password. Do not forget to commit your configuration.

```
Performing filesystem consistency checks ...
/dev/da0s1a: FILE SYSTEM CLEAN; SKIPPING CHECKS
...TRIMMED...
Starting CLI ...
root> configure
Entering configuration mode

[edit]
root# set system root-authentication plain-text-password
New password:
Retype new password:

[edit]
root# commit
commit complete
```

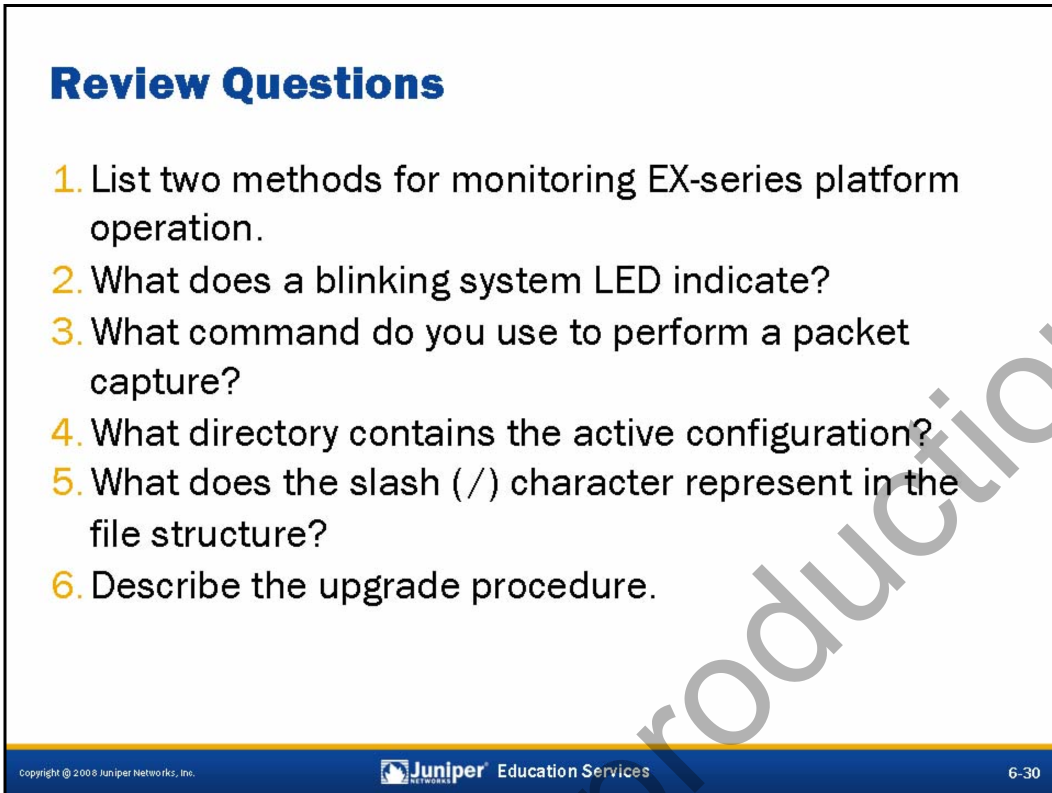
4. To complete the recovery, exit configuration mode. You are then prompted to reboot the switch. Choose yes to reboot the system. Once the reboot is complete you can access the switch with the new root password.

```
[edit]
root@s1# exit
Exiting configuration mode

root@s1> exit

Reboot the system? [y/n] y
```



This Chapter Discussed:

- EX-series platform operation;
- EX-series network utilities;
- Methods of maintaining JUNOS software; and
- File system maintenance and password recovery.



Review Questions

1.

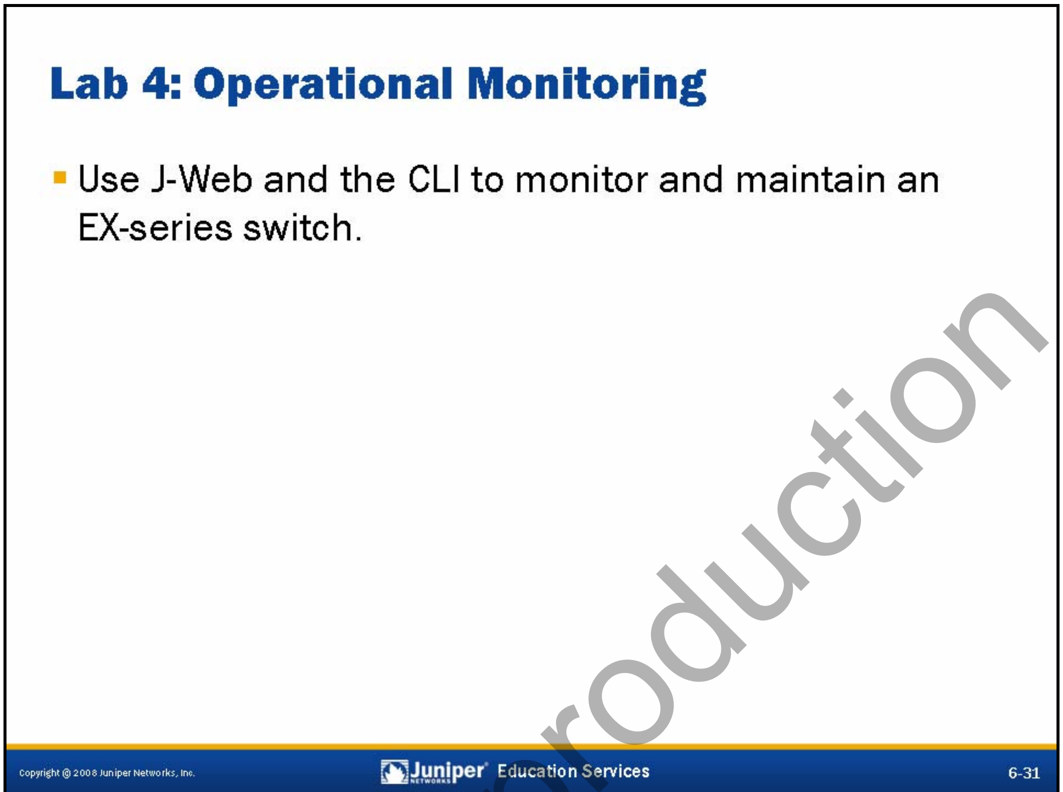
2.

3.

4.

5.

6.

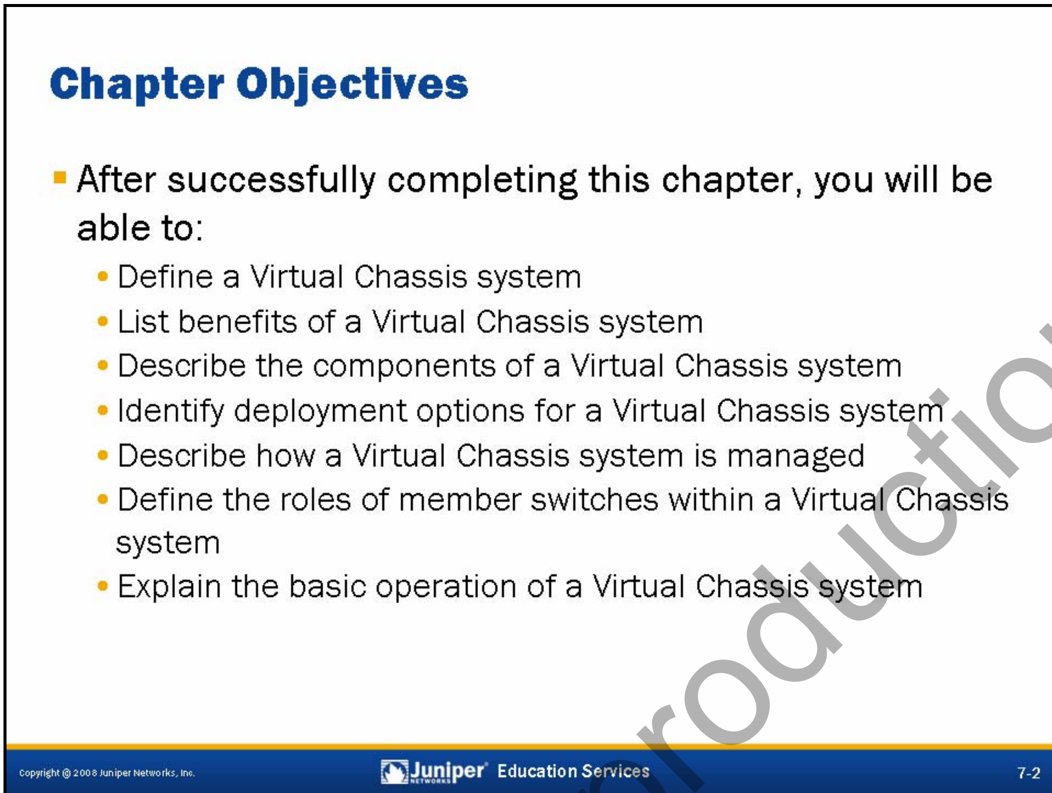


Lab 4: Operational Monitoring

The slide provides the objective for this lab.

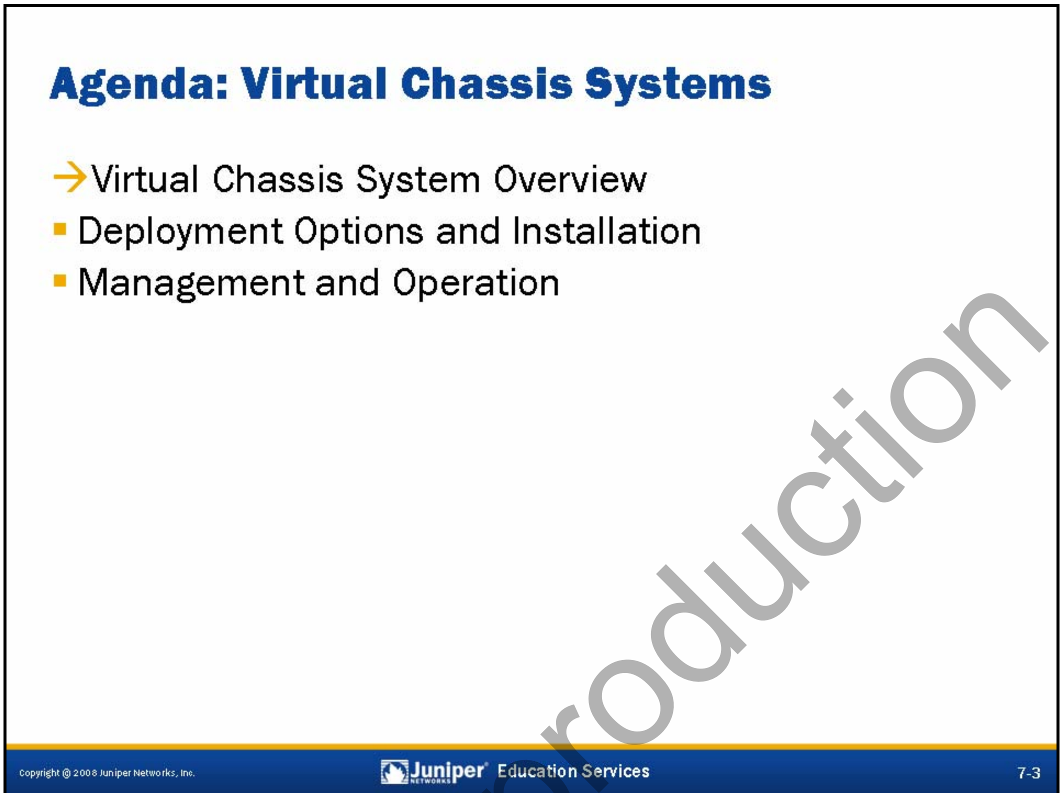
Operating Juniper Networks Switches in the Enterprise

Chapter 7: Virtual Chassis Systems



This Chapter Discusses:

- Benefits of using a Virtual Chassis system;
- Components of a Virtual Chassis system;
- Virtual Chassis deployment options;
- Management of a Virtual Chassis system;
- Roles and responsibilities of member switches; and
- Basic operation of a Virtual Chassis system.



Virtual Chassis System Overview

The slide lists the topics we cover in this chapter. We discuss the highlighted topic first.

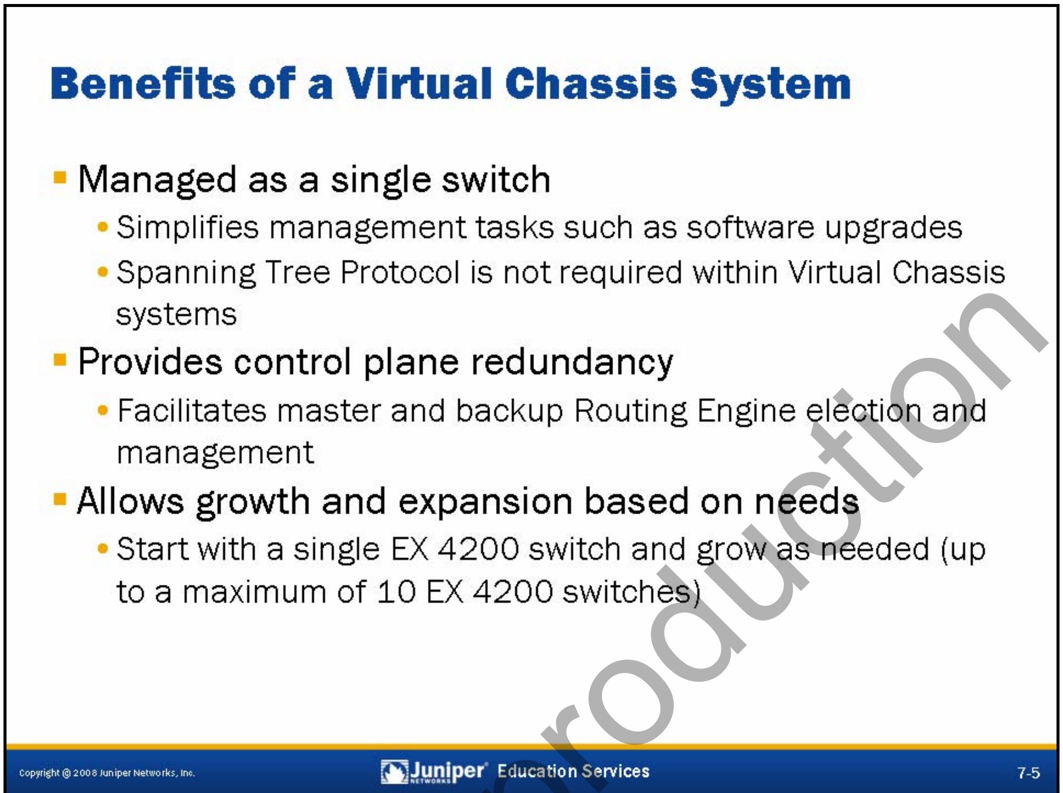
What Is a Virtual Chassis System?

- A collection of interconnected EX 4200 switches that are represented and managed as a single entity
  - Consists of 1 to 10 EX 4200 switches
  - Provides a scaling solution within a switching environment
    - Allows up to 480 10/100/1000 Ethernet ports

Juniper Education Services
7-4

Definition of a Virtual Chassis System

Simply put, a Virtual Chassis system is a collection of interconnected EX 4200 switches that are managed as a single switch. A Virtual Chassis system consists of one to ten EX 4200 switches known as *member switches*. The member switches work together to provide higher port density than they would as single switches. Collectively, the member switches can offer up to 480 10/100/1000 Ethernet ports. In addition to the fixed Ethernet ports, a Virtual Chassis system can also offer up to 40 1-Gigabit Ethernet (SFP) uplink ports or 20 10-Gigabit Ethernet (XFP) uplink ports.



Managed as a Single Switch

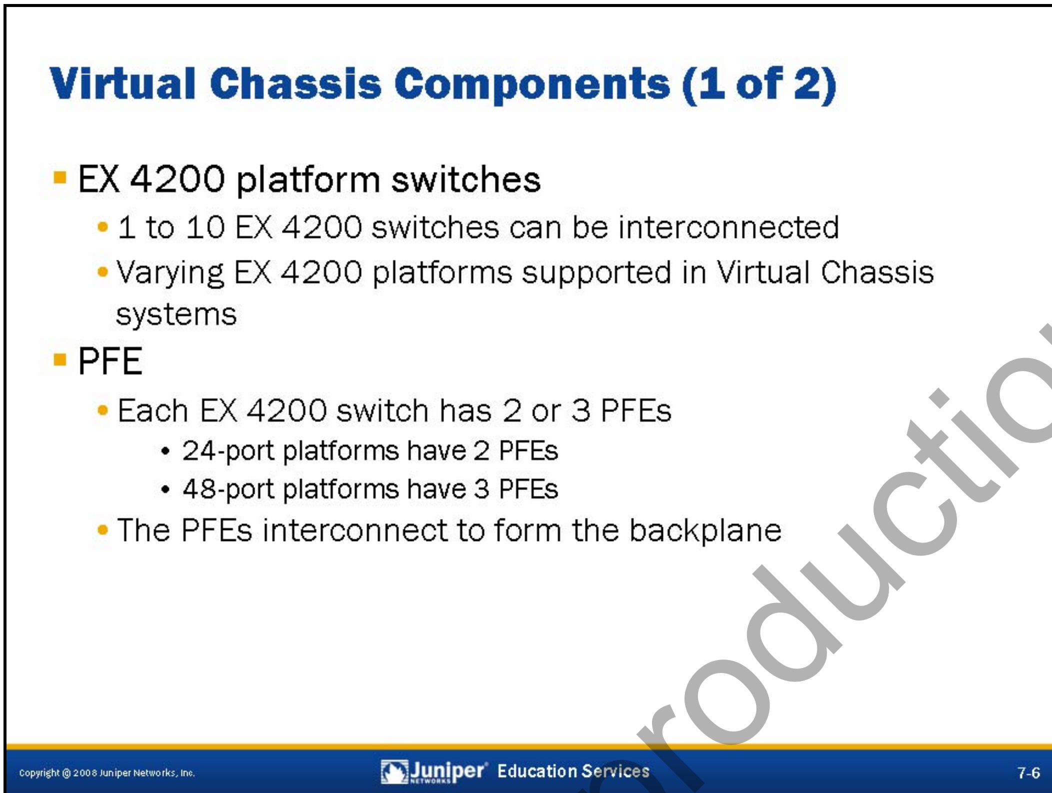
You can connect EX 4200 switches together to form a Virtual Chassis system, which you then manage as a single device. Comparatively speaking, managing a Virtual Chassis system is much simpler than managing up to ten individual switches. For example, when upgrading the software on a Virtual Chassis system, only the master switch must have the software upgraded. However, if all members function as standalone switches, all individual members must have the software upgraded separately. Also, in a Virtual Chassis scenario, there is no need to run the Spanning Tree Protocol (STP) between the individual members because in all functional aspects, a Virtual Chassis system is a single device.

Control Plane Redundancy

In a Virtual Chassis configuration, one of the member switches is elected as the master switch and a second member switch is chosen as the backup switch. This design approach provides control plane redundancy and is a requirement in many enterprise environments.

Expand as Needed

The Virtual Chassis system offers *add-as-you-grow* flexibility. A Virtual Chassis system can start with a single EX 4200 switch and grow, based on customer needs, to as many as ten EX 4200 switches. This ability to grow and expand within and across wiring closets is a key advantage in many enterprise environments.

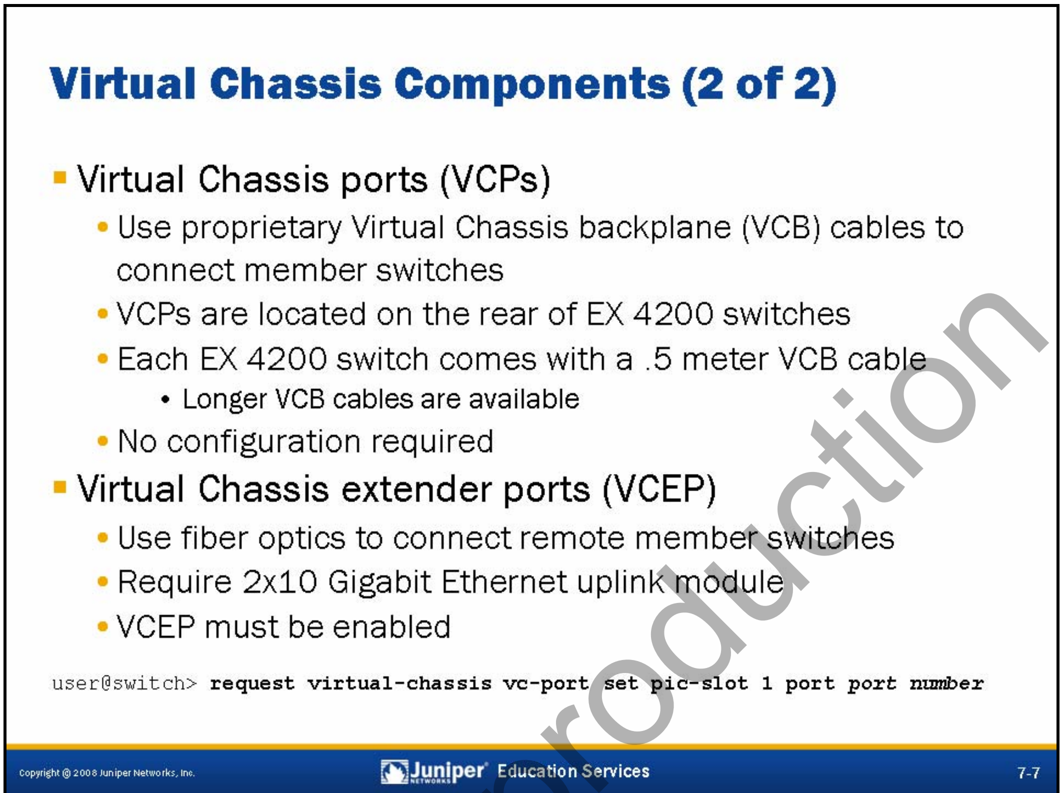


EX 4200 Platforms

You can interconnect one to ten EX 4200 switches to form a Virtual Chassis system. A Virtual Chassis system can consist of any combination of model numbers within the EX 4200 family of switches.

PFE

Each EX 4200 switch has two or three Packet Forwarding Engines (PFEs) depending on the platform. All PFEs are interconnected, either through internal connections or through the Virtual Chassis ports (VCPs) on the rear of the EX 4200 platform. Collectively, the PFEs and their connections constitute the Virtual Chassis backplane.

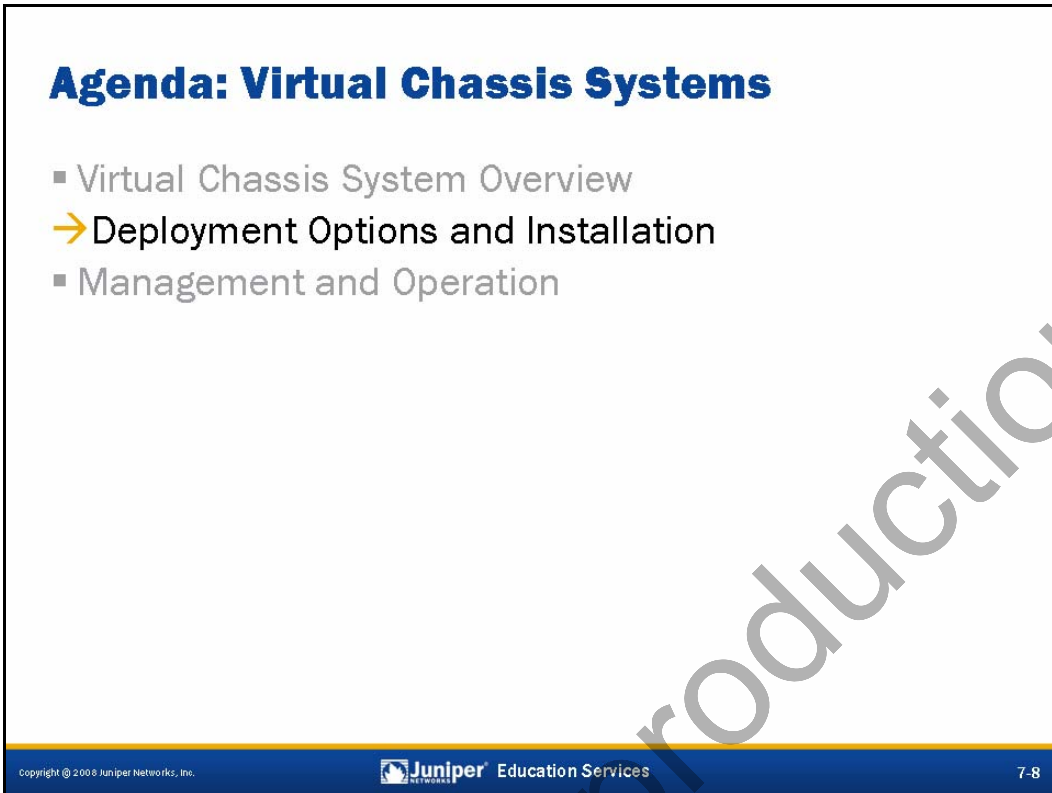


Virtual Chassis Ports

Each EX 4200 switch has two dedicated Virtual Chassis ports (VCPs) on its backplane. These ports can be used to interconnect two to ten EX 4200 switches as a Virtual Chassis system, which functions as a single network entity. The software automatically configures the VCP interfaces. The VCP interfaces are called vcp-0 and vcp-1. The interfaces for these dedicated Virtual Chassis ports are not configurable and support a speed of 64 Gbps full-duplex. A single Virtual Chassis backplane (VCB) cable is included with each EX 4200 switch. The length of this proprietary cable is .5 meters. If the deployment scenario requires a VCB cable longer than .5 meters, the cable must be ordered separately.

Virtual Chassis Extender Ports

It is possible to interconnect EX 4200 switches across wider distances by using the EX-UM-2XFP uplink module ports and fiber optics. To use an EX-UM-2XFP uplink module port as a Virtual Chassis port, explicit configuration for the uplink module ports is required. The example on the slide shows a typical configuration when using the xe-0/1/0 uplink port as a Virtual Chassis extender port (VCEP).



Deployment Options and Installation

The slide highlights the topic we discuss next.

Deployment Options

- Some common deployment options include:
  - Single-rack application
    - Generally spans less than 5 meters
  - Top-of-rack application
    - Can span up to 15 meters
  - Remote wiring closet application
    - Uses VCEPs to interconnect switches in remote wiring closets; can span up to 500 meters

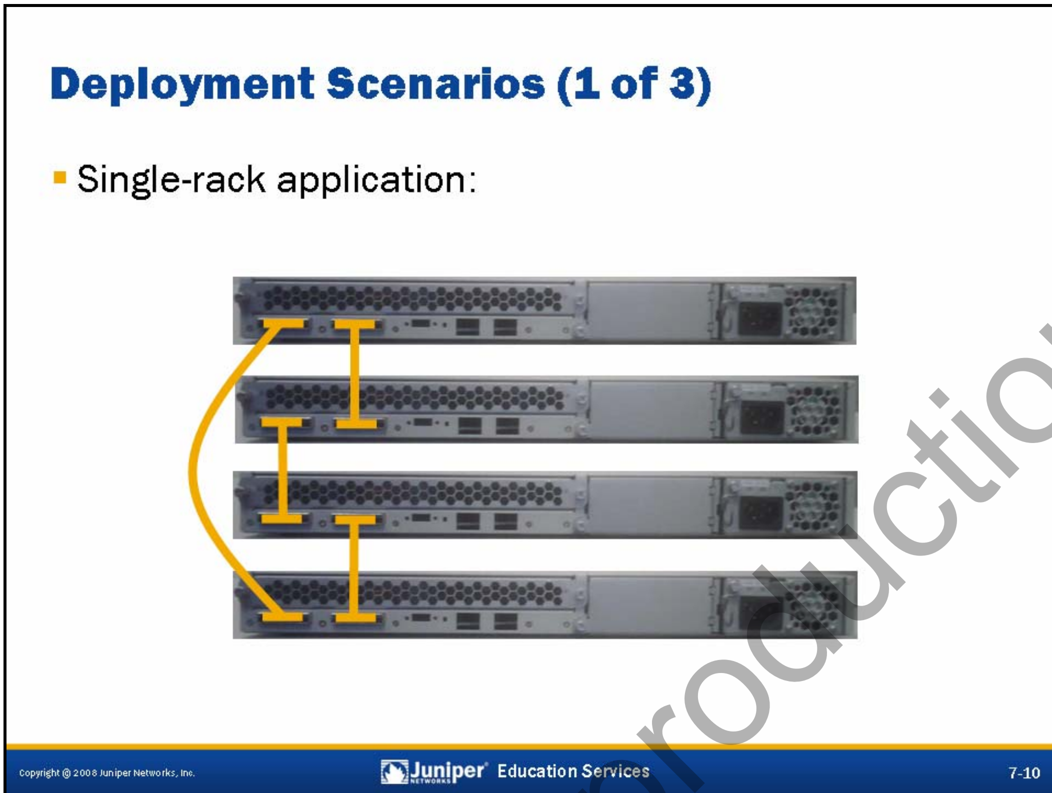
 Juniper Education Services 7-9

Common Deployment Options

The slide highlights some common deployment options. The first option listed is the single-rack application. In this deployment option, all member switches participating in the Virtual Chassis system reside in the same rack. The distance between the top and bottom switches in the single rack application is generally less than 5 meters. Because the distance between the top and bottom switch is less than 5 meters in this deployment option, the default VCB cables (.5 meters) should be adequate.

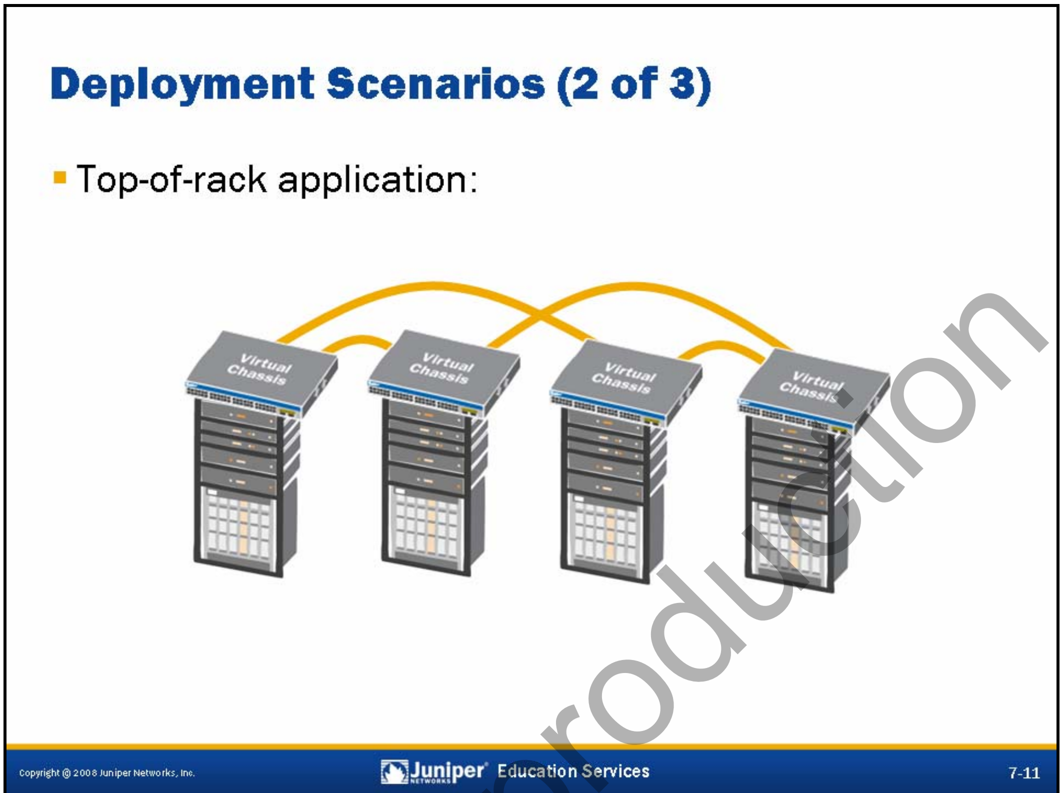
In the top-of-rack application, the switches reside in multiple racks. This deployment option can span up to 15 meters and generally requires the use of VCB cables with lengths greater than .5 meters.

The remote wiring closet application makes use of the EX-UM-2XFP uplink module ports, which must be configured as VCEPs. A Virtual Chassis ring, which connects remote wiring closets with VCEPs, can span a total distance of up to 500 meters.



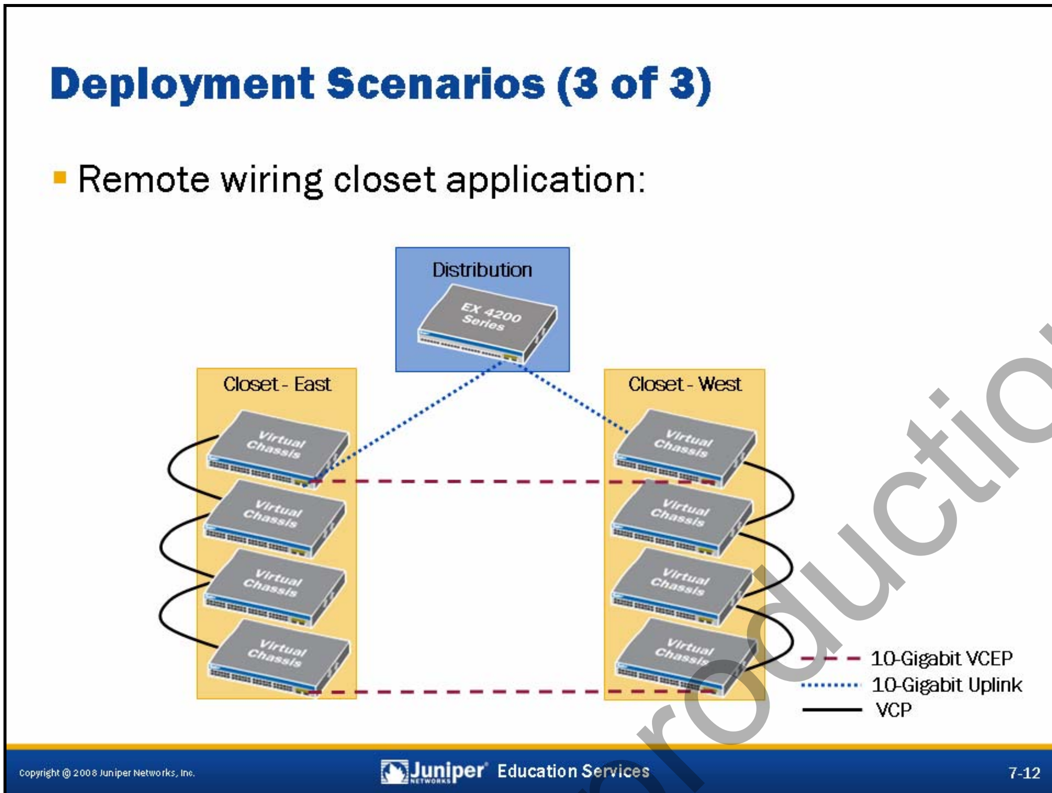
Single-Rack Application

The slide illustrates a common cabling scenario for the single-rack deployment option. In this example, a complete ring is formed by connecting the top and bottom switches together with the VCPs.



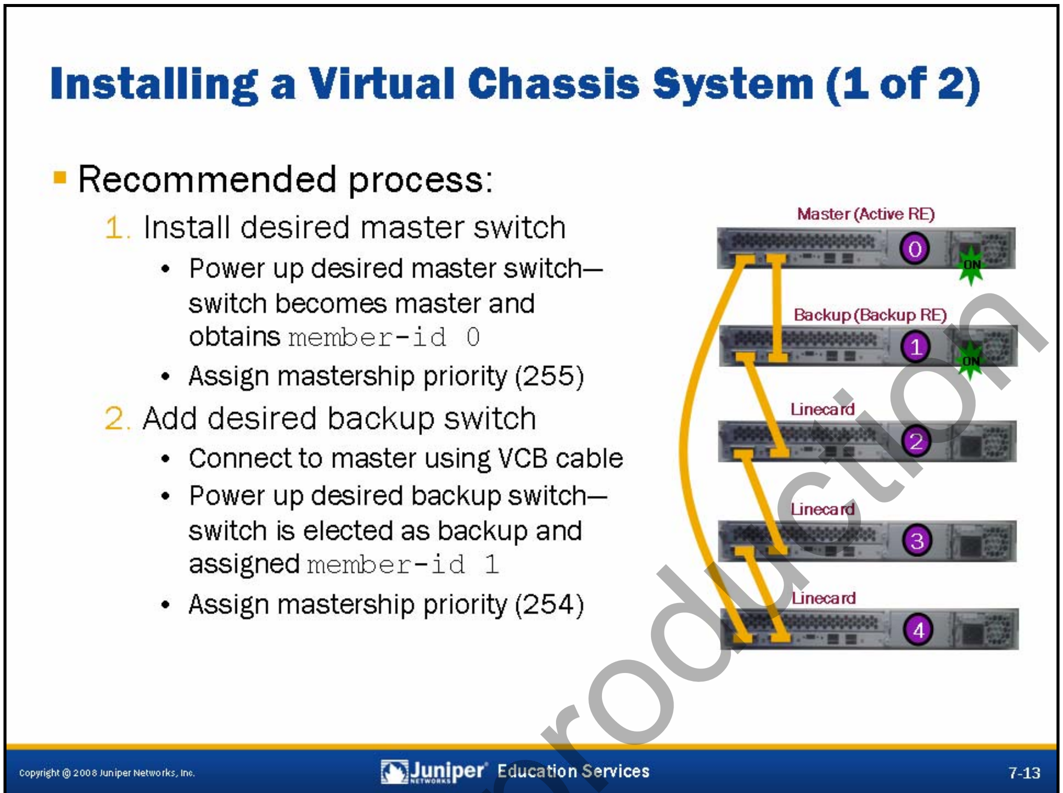
Top-of-Rack Application

This slide illustrates the top-of-rack deployment option. This example also forms a complete ring and ensures no single point of failure.



Remote Wiring Closet Application

The example on the slide uses both VCPs and VCEPs. The remote closets are connected with the VCEPs to form a single Virtual Chassis system. The switches within the individual closets are connected with the VCPs. The Virtual Chassis system, which consists of devices in two distinct locations, then connects to a distribution switch by way of multiple 10-gigabit uplink connections.

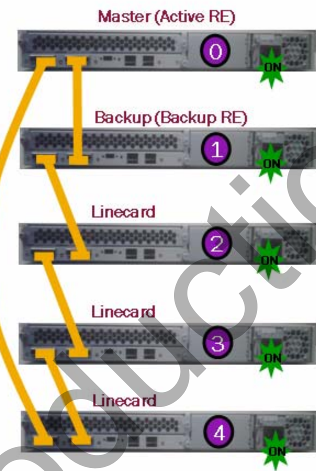


Virtual Chassis System Installation Process: Part 1

This slide and the next slide outline the process for installing a Virtual Chassis system. Although there might be other methods for installing a Virtual Chassis system, this process outlines the recommended steps that have been tested and proven successful.

Installing a Virtual Chassis System (2 of 2)

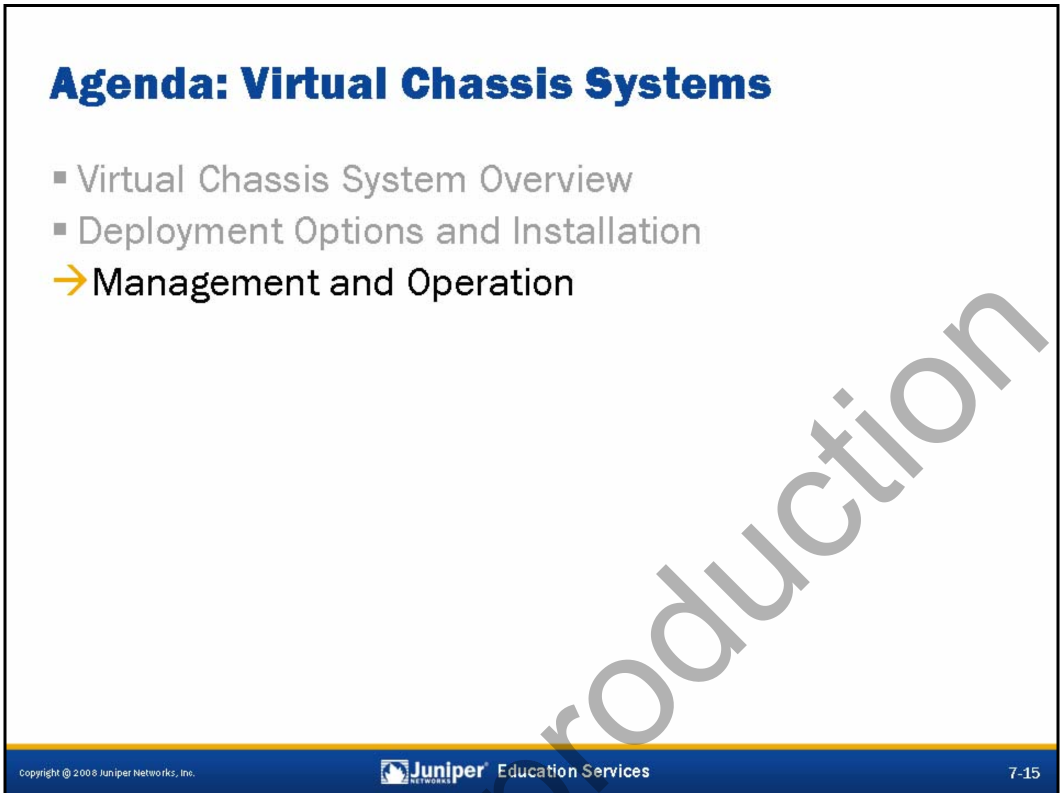
- Recommended process (contd.):
  3. Add *line card* switch
     - Connect to switch above with VCB cable
     - Power up third switch—switch becomes *line card* and is assigned `member-id 2`
     - Assign desired mastership priority
  4. Repeat Step 3 to add subsequent *line card* switches
     - Last *line card* switch completes loop by connecting with master



 Juniper Education Services 7-14

Virtual Chassis System Installation Process: Part 2

The slide shows the remainder of the recommended steps for installing a Virtual Chassis system.



Management and Operation

The slide highlights the topic we discuss next.

Connectivity

- Single management interface
  - Individual management Ethernet ports on member switches are tied to a special management VLAN associated with a Layer 3 virtual management interface (vme)
- Single management IP address
  - The Virtual Chassis system is managed as a single network element; therefore, it has only one management IP address
- Single virtual console
  - Connection to a console on any member switch in a Virtual Chassis system is redirected to the VC master by virtual console software running on all member switches

 Juniper Education Services 7-16

Single Management Interface

The management Ethernet ports on the individual member switches are automatically associated with a management VLAN. This management VLAN uses a Layer 3 virtual management interface that facilitates communication through the Virtual Chassis system to the master switch even if the master switch's management Ethernet port is inaccessible.

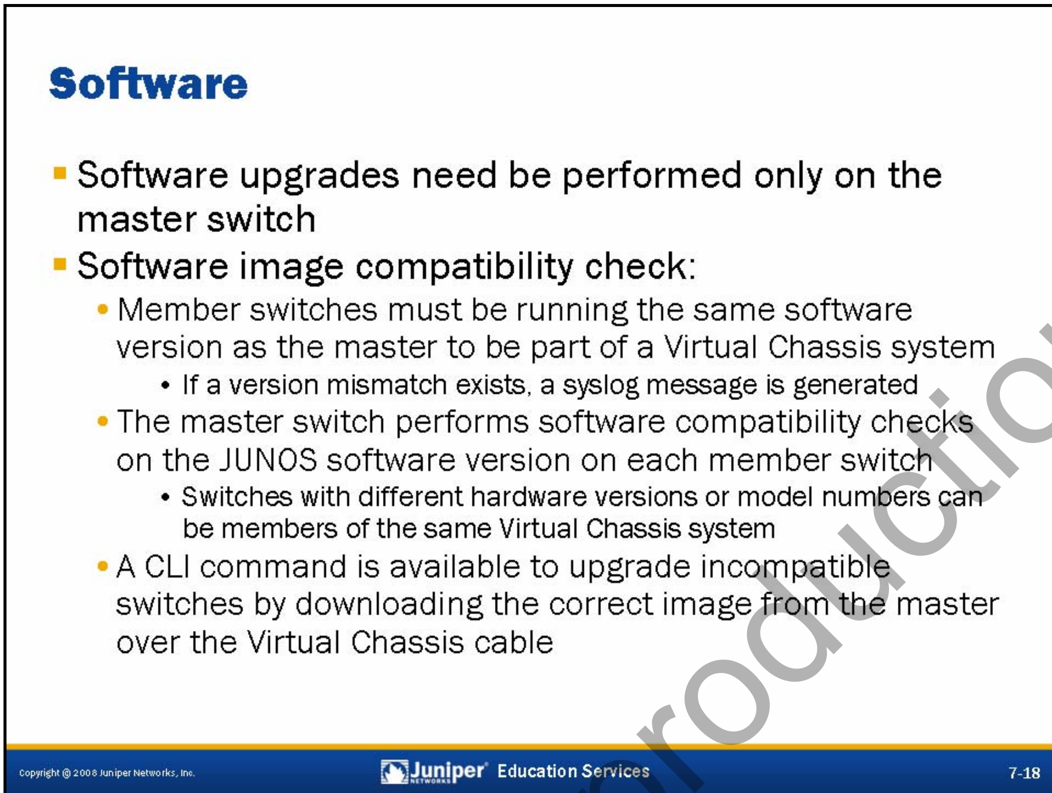
Single Management IP Address

When you set up the master switch, you specify an IP address for the virtual management Ethernet interface (vme). This single IP address allows you to configure and monitor all members of the Virtual Chassis system remotely through Telnet or SSH.

Continued on next page.

Single Virtual Console

All member switches participating in a Virtual Chassis system run virtual console software. This software redirects all console connections to the master switch. The exception is, of course, any console connection made with the master switch itself. The ability to redirect management connections to the master switch simplifies Virtual Chassis management tasks. Generally speaking, you can obtain all status-related information for the individual switches participating in a Virtual Chassis system through the master switch. It is, however, possible to establish individual virtual terminal (vty) connections from the master switch to individual member switches, if needed.



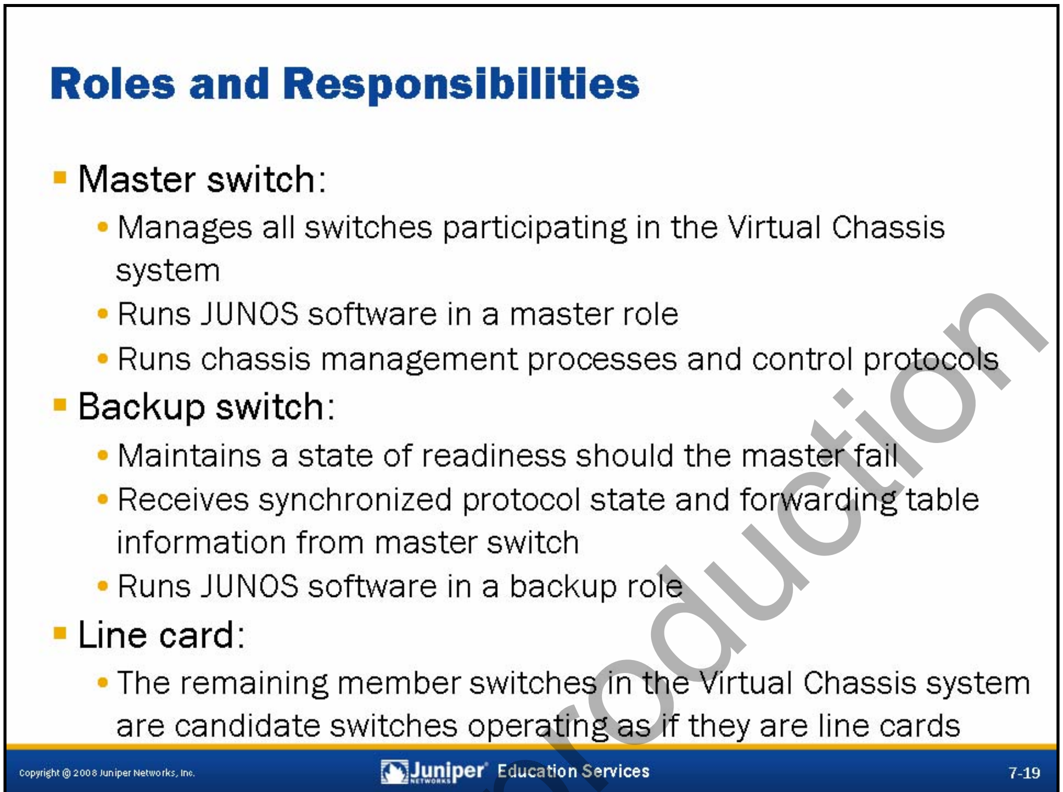
Software Upgrades

You perform software upgrades within a Virtual Chassis system on the master switch. All other member switches are automatically upgraded.

Software Compatibility Check

To connect switches as a Virtual Chassis system, the switches must be running the same software version. The master switch checks the hardware version, the JUNOS software version, and other component versions running in a switch that is physically interconnected to its VCP or VCEP. If a software version mismatch exists, a syslog message is generated to notify the user. Different hardware models can be members of the same Virtual Chassis system. However, the master switch will not assign a member ID to a switch running a different software version. A switch running a version of software different from the master switch is not allowed to join the Virtual Chassis system. In this situation, the switch must be upgraded. It is possible to upgrade individual switches from the master switch through the Virtual Chassis cable. The following command is used to upgrade an individual member within a Virtual Chassis system:

```
user@switch> request system software add member ?
Possible completions:
  <member>              Install package on VC Member (0..9)
```



Master Switch

The master switch represents all member switches interconnected within the Virtual Chassis configuration. Any configuration parameters assigned to this switch apply to all members of the Virtual Chassis system. We highly recommend that all changes made on the master switch are replicated to the backup switch through the use of the **`commit synchronize`** command. The master switch also manages the individual member switches, runs JUNOS software in a master role, and runs the chassis management processes and control protocols. The forwarding table is built on the master switch and then replicated to all PFEs within the Virtual Chassis system.

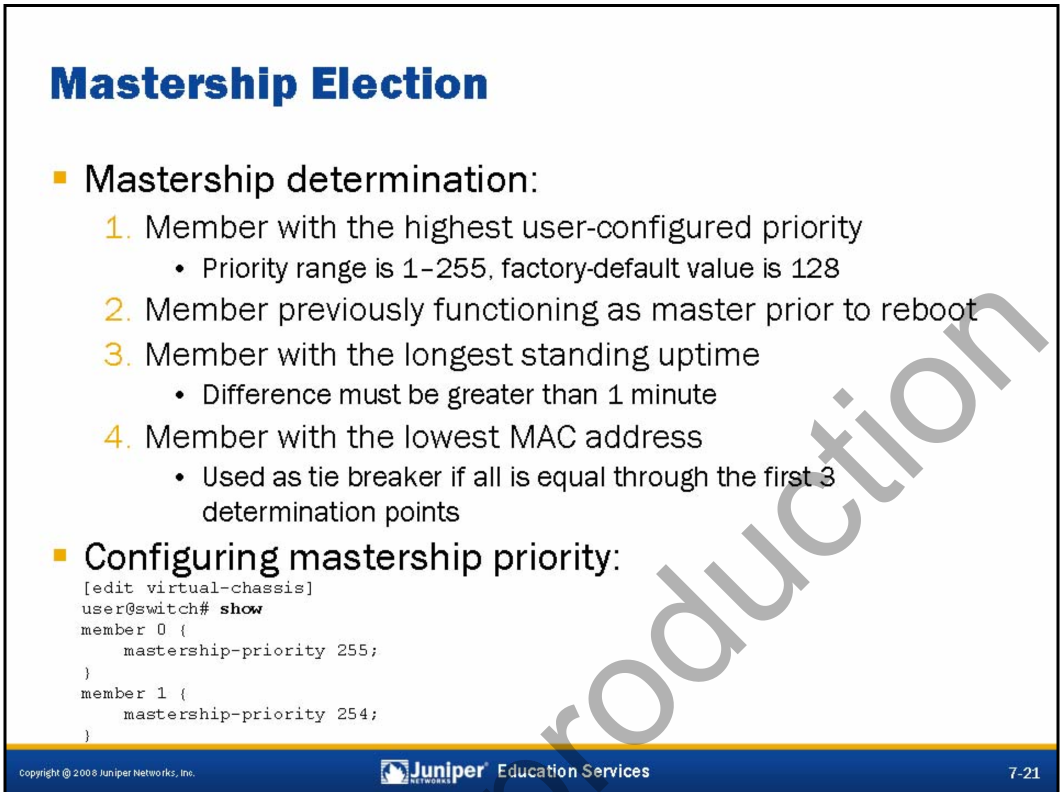
Backup Switch

The backup switch maintains a state of readiness to take over as master should the active master fail. The backup switch synchronizes protocol state and forwarding tables with the master switch so that it is prepared to preserve routing information and maintain network connectivity without disruption in case the master switch is unavailable. The switch designated as backup also runs JUNOS software in a backup role.

Continued on next page.

Line Card

A line card switch, which is any member switch other than the master or backup, programs its own local hardware. It does not run the chassis management process or control protocols. A line card switch is responsible only for its local interfaces within a chassis.

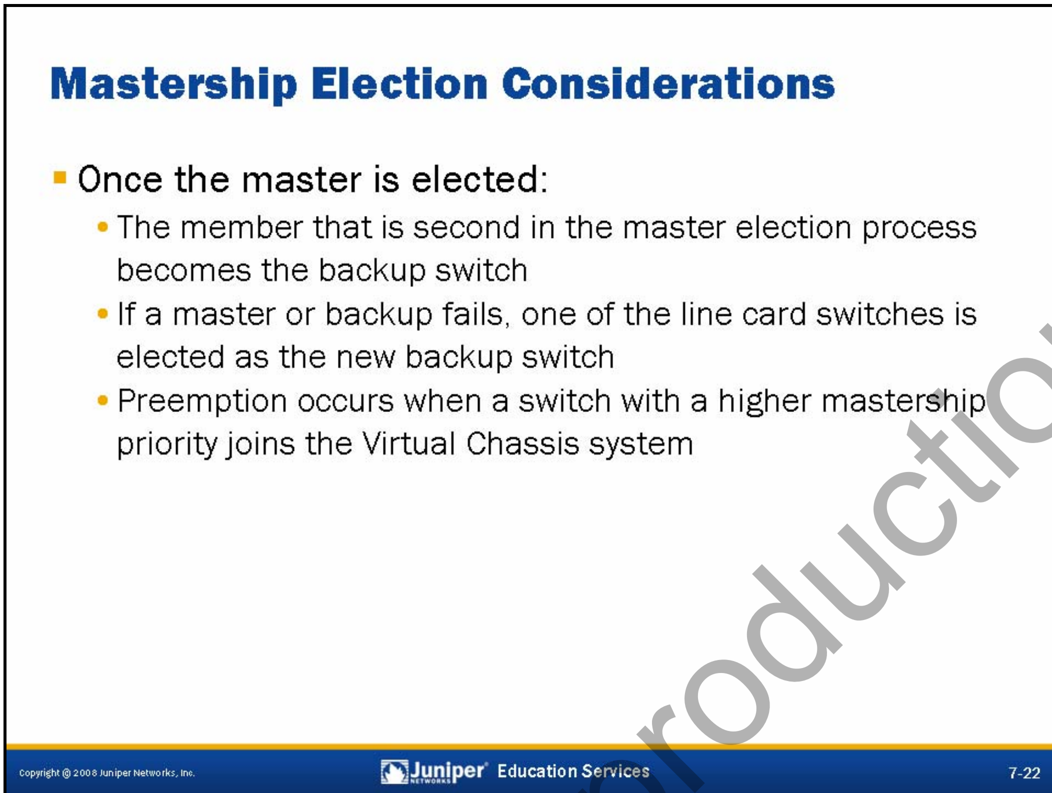


Mastership Determination

The steps shown on the slide outline the mastership election process.

Configuring Mastership Priority

The configuration example shown on the slide illustrates how to configure mastership priority.



Mastership Election Considerations

The slide identifies some considerations pertaining to the election process.

Member ID Assignment

- Member ID assignment and considerations:
  - Master switch typically assumes member ID 0
  - Master switch assigns unique member IDs (1–9) to each member switch
  - Member IDs are assigned in ascending order based on the sequence in which member switches were added to the Virtual Chassis system
  - Member ID is preserved across reboot within a Virtual Chassis system
  - Member ID serves as slot number for interface naming
  - Member ID can be manually configured through CLI

 Juniper Education Services 7-23

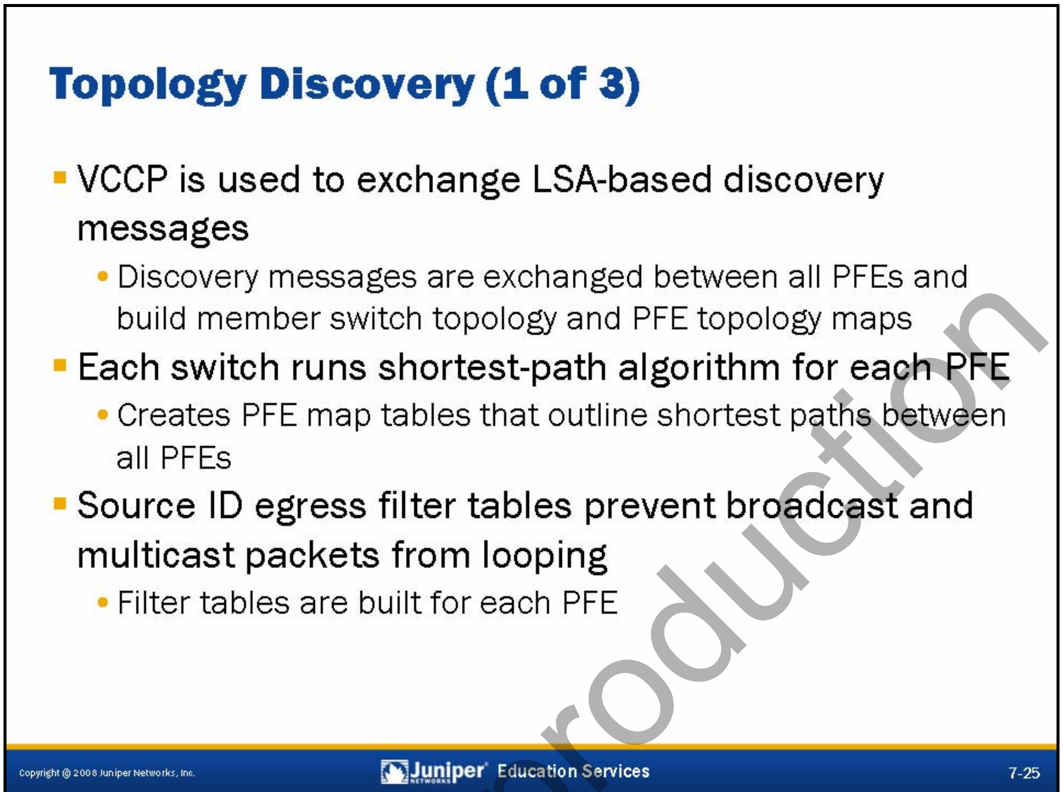
Member ID Assignment and Considerations

The master switch typically assumes a member ID of 0 because it is the first switch powered on. When the remainder of the member switches are interconnected and powered on, the master switch assigns a member ID from 1 through 9, making the complete member ID range 0–9. The master assigns each switch a member ID based the sequence that the switch was added to the Virtual Chassis system. The member ID associated with each member switch is preserved, for the sake of consistency, across reboots. This preservation is helpful because the member ID is also a key reference point when naming individual interfaces. The member ID serves the same purpose as a slot number when configuring interfaces. Although the member ID is initially assigned by the master switch, you can change the member ID values by using the CLI. Each LCD displays the member ID assigned to that switch. The sequence on the next page shows how to view the LCD information through the CLI as well as how to change the member ID:

Continued on next page.

Member ID Assignment and Considerations (contd.)

```
user@switch> show chassis lcd
FPM Display contents for slot: 9
    09:RE switch
    LED:SPD ALARM 00

user@switch> request virtual-chassis renumber member-id 9 new-member-id 8

To move configuration specific to member ID 9 to member ID 8, please
use the replace command. e.g. replace pattern ge-9/ with ge-8/

Do you want to continue ? [yes,no] (yes) yes


user@switch>
switch (ttyu0)

login: user
Password:

--- JUNOS 9.0R2.10 built 2008-03-06 10:31:45 UTC
user@switch> show chassis lcd
FPM Display contents for slot: 8
    08:RE switch
    LED:SPD ALARM 00
```



Discovery Mechanism

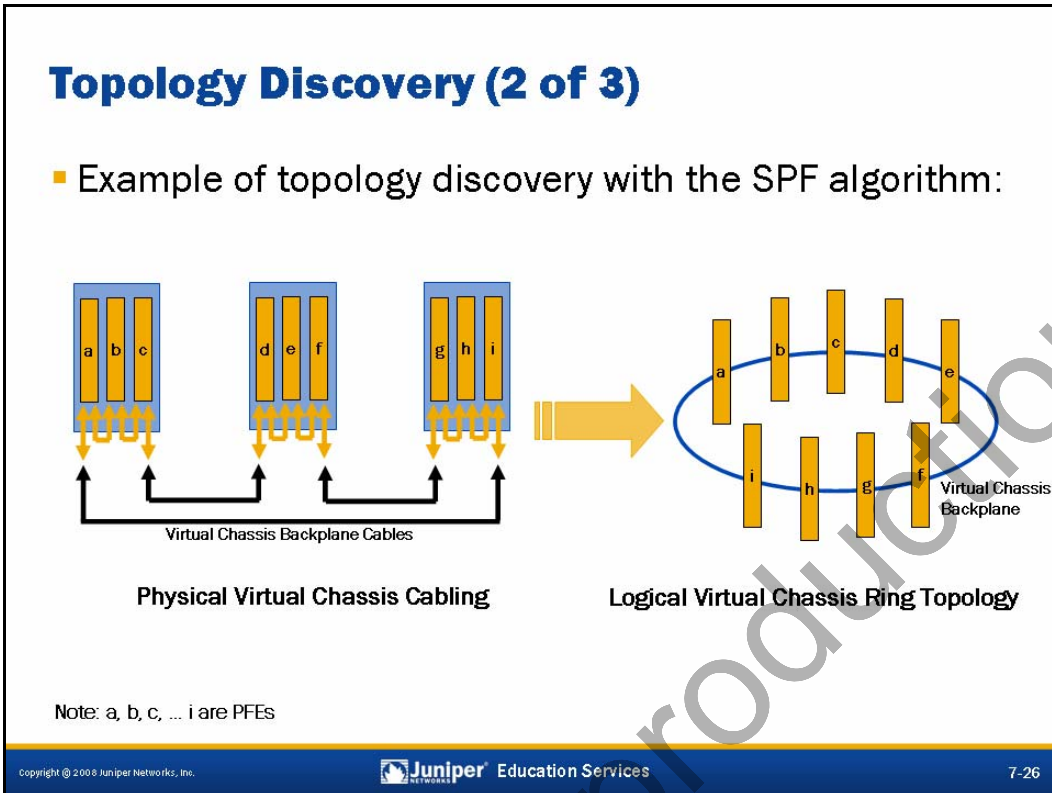
All switches participating in the Virtual Chassis system use the Virtual Chassis Control Protocol (VCCP) to exchange link-state advertisement (LSA) based discovery messages between all interconnected PFEs within a Virtual Chassis system. Based on these LSA-based discovery messages, each PFE builds a member switch topology in addition to a PFE topology map. These topology maps are used when determining the best paths between individual PFEs.

Identifying the Shortest Path

Once the PFE topology map is built, the individual switches run a shortest-path algorithm for each PFE. This algorithm is based on hop count and bandwidth. The result is a map table for each PFE that outlines the shortest path to all other PFEs within the Virtual Chassis system. In the event of a failure, a new SPF calculation is performed.

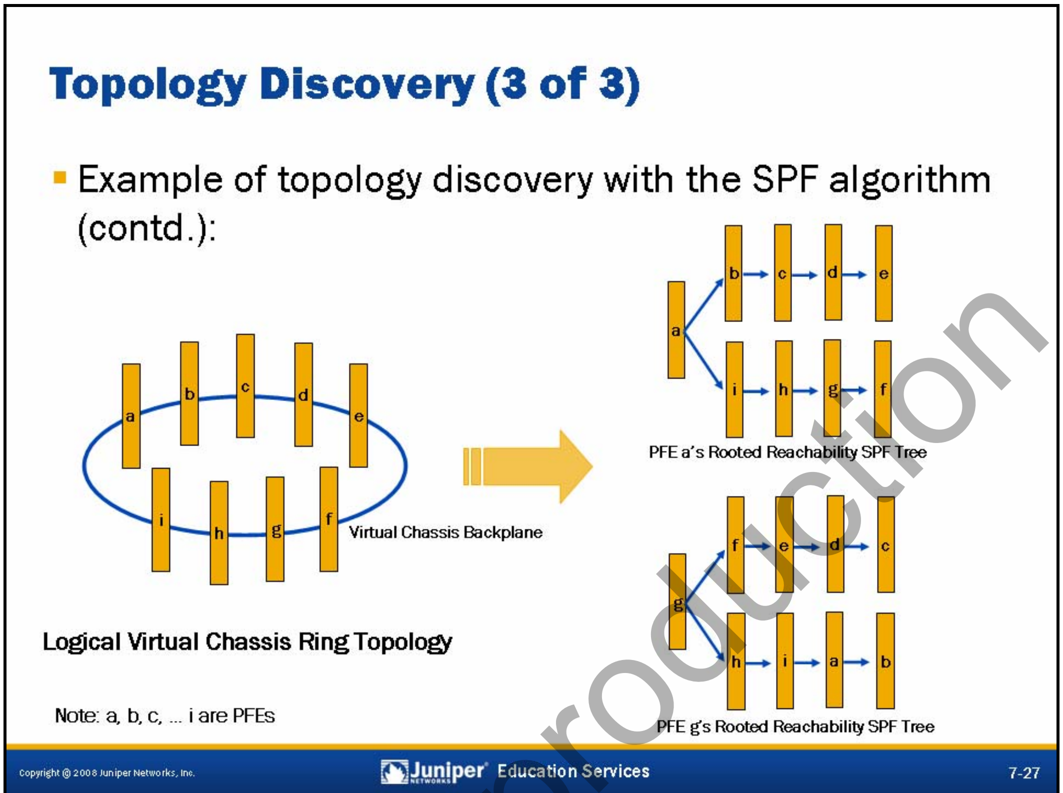
Preventing Loops

To prevent broadcast and multicast loops, each switch creates a unique source ID egress filter tables on each PFE. We provide an example of preventing loops on a subsequent page in this chapter.



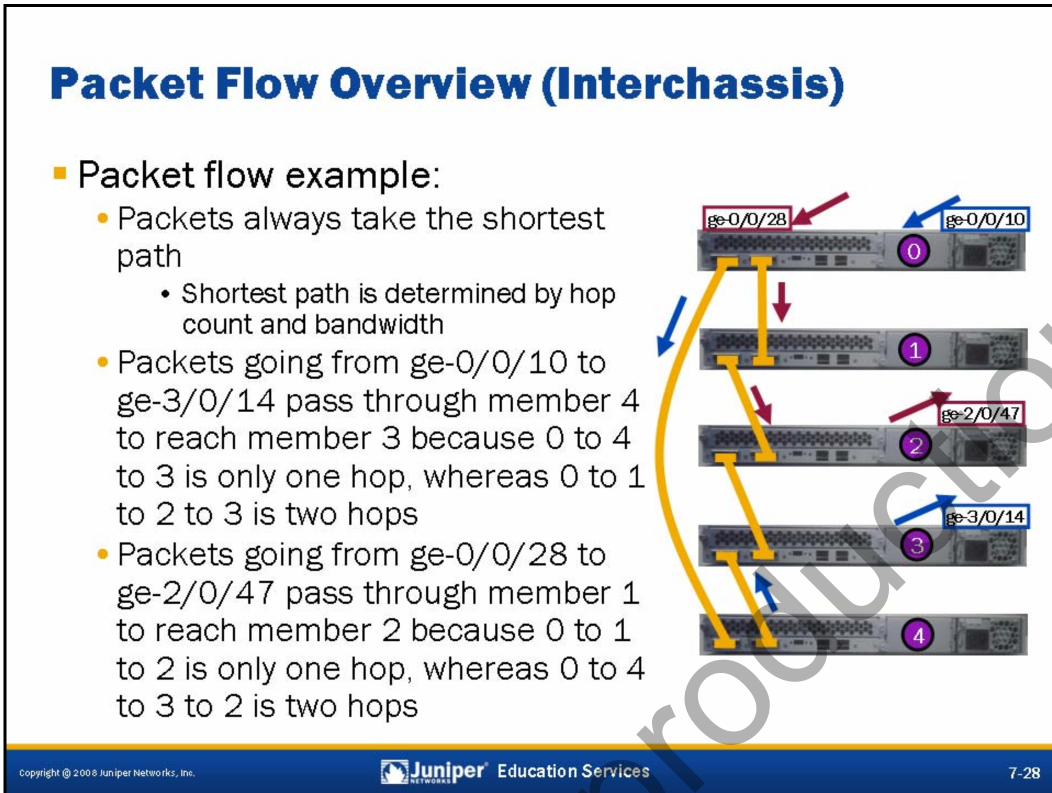
Example of Topology Discovery with the SPF Algorithm: Part 1

The slide provides a visual example of the physical cabling and logical ring topology of a Virtual Chassis system.



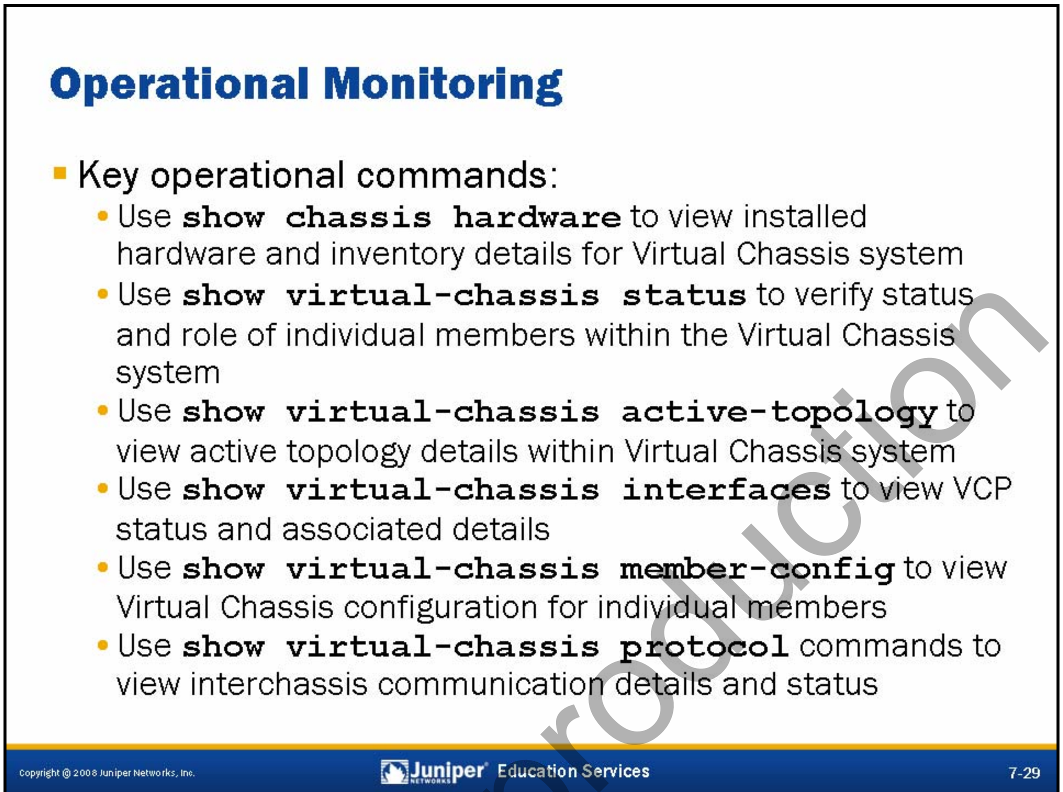
Example of Topology Discovery with the SPF Algorithm: Part 2

Using the SPF algorithm, each PFE builds its own shortest-path tree to all other PFEs, based on hop count. This process is automatic and is not configurable. The slide provides a visual example of this process.



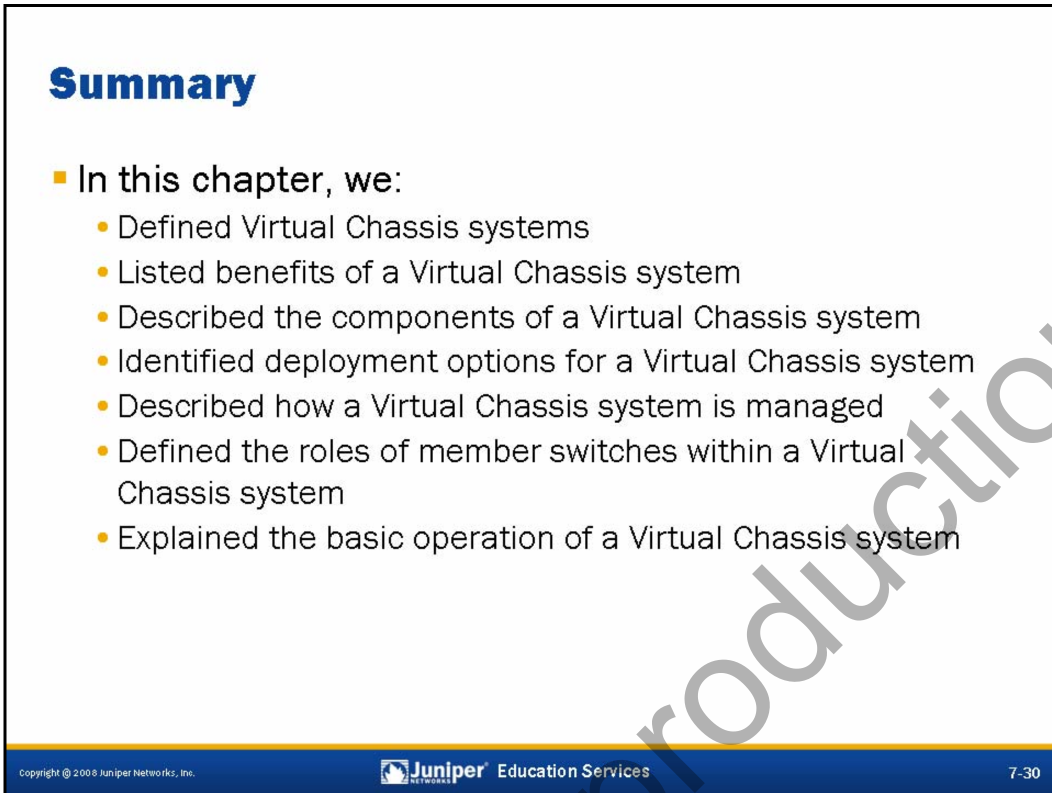
Interchassis Packet Flow Example

As packets flow from one physical chassis to another through a Virtual Chassis system, they always takes the shortest path, which is based on a combination of hop count and bandwidth. The first example on the slide shows a packet that enters the Virtual Chassis system on port ge-0/0/10, which is a fixed 10/100/1000 Ethernet port on the chassis with a member ID of 0. The packet is destined for an egress port of ge-3/0/14, which is a fixed 10/100/1000 Ethernet port on the chassis with the member ID of 3. Based on the physical topology, this packet passes through member switch 4 to member switch 3, which owns the egress port in question. In the second example, we see similar results in which the shortest path is selected once again.



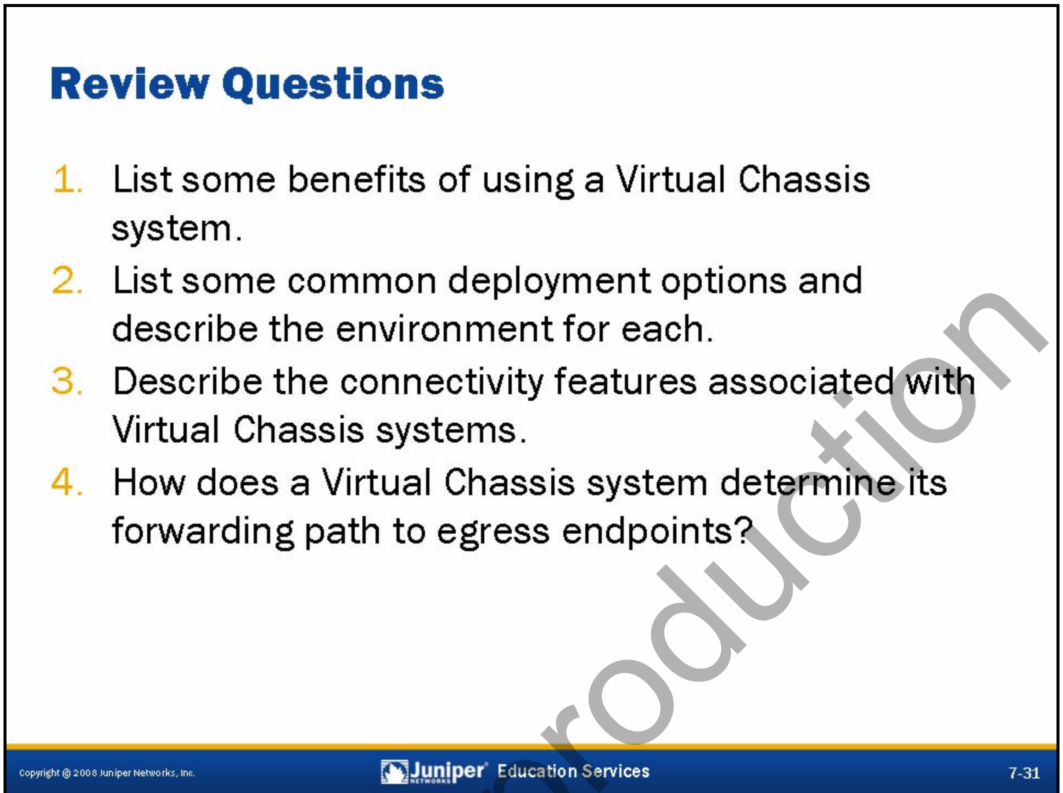
Key Operational Commands

This slide highlights some key operational-mode commands along with a short description of the content each command displays.



This Chapter Discussed:

- Benefits of using a Virtual Chassis system;
- Components of a Virtual Chassis system;
- Virtual Chassis deployment options;
- Management of a Virtual Chassis system;
- Roles and responsibilities of member switches; and
- Basic operation of a Virtual Chassis system.



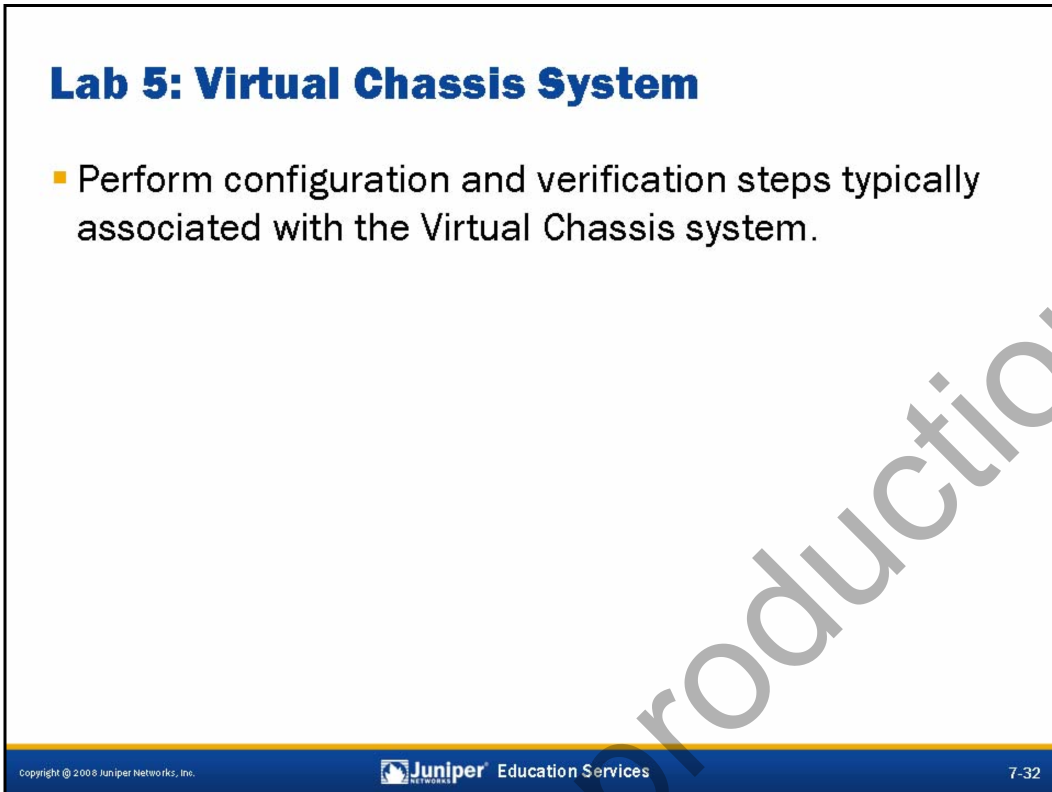
Review Questions

1.

2.

3.

4.

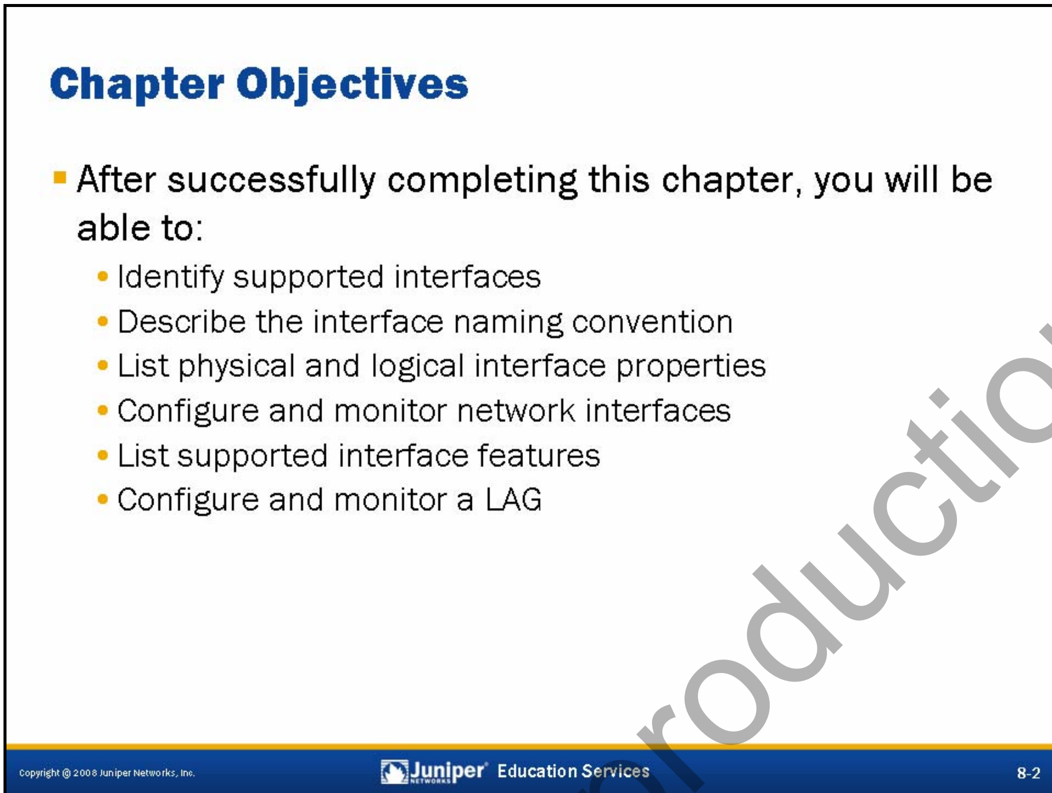


Lab 5: Virtual Chassis System

The slide provides the objective for this lab.

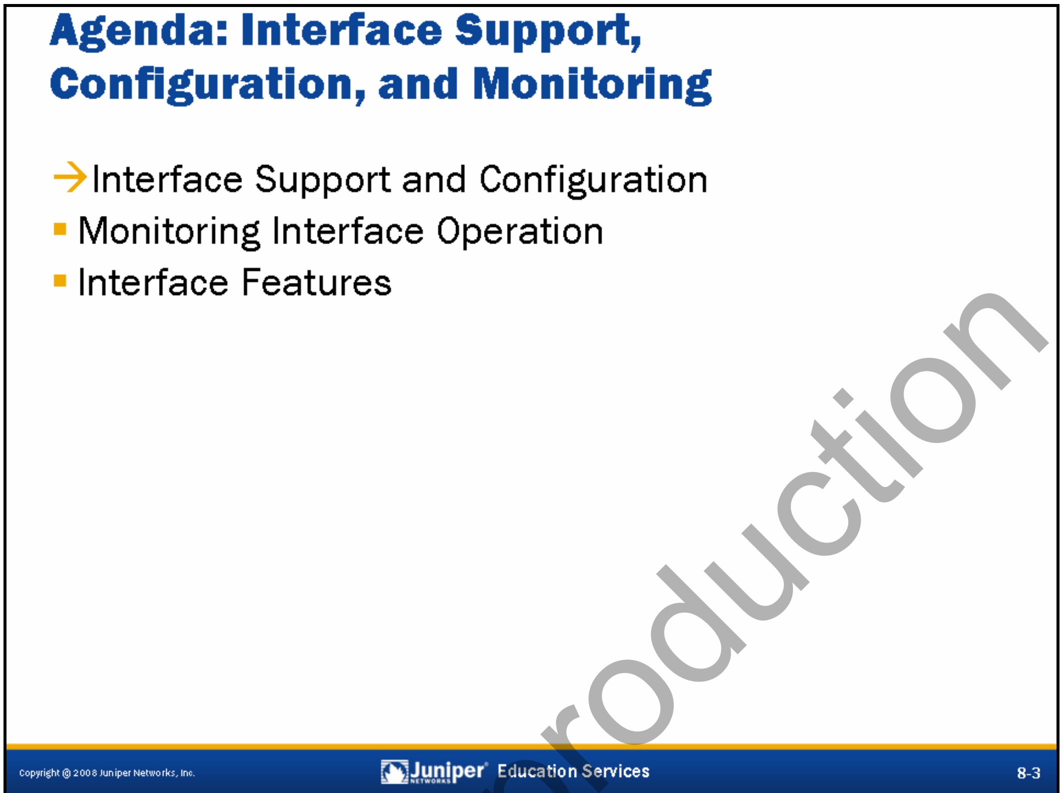
Operating Juniper Networks Switches in the Enterprise

Chapter 8: Interface Support, Configuration, and Monitoring



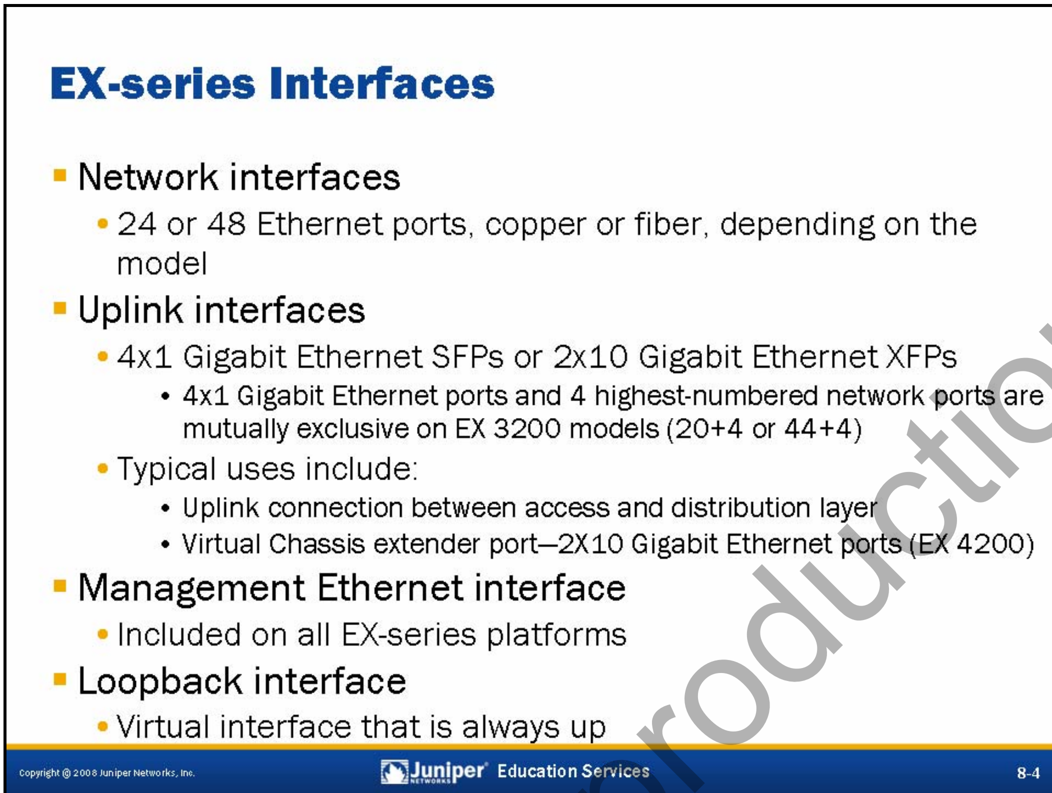
This Chapter Discusses:

- Supported interface types;
- Interface naming convention;
- Physical and logical interface properties;
- Configuration and monitoring details for network interfaces;
- Interface features; and
- Configuration and monitoring of a link aggregation group (LAG).



Interface Support and Configuration

The slide lists the topics we cover in this chapter. We discuss the highlighted topic first.



Network Interfaces

Network interfaces connect to and carry traffic over the network. EX-series switches provide either 24 or 48 Ethernet network ports, which use copper or fiber, depending on the switch model. These ports are used to interconnect devices such as personal computers, laptops, file servers, and printers on the network.

Uplink Interfaces

Uplink modules accommodate either four 1-Gigabit Ethernet small form-factor pluggable transceivers (SFPs) or two 10-Gigabit Ethernet small form-factor pluggable transceivers (XFPs). Both options are available for all EX 3200 and EX 4200 platforms. On EX 3200 fixed-configuration switches, the four 1-Gigabit Ethernet uplink ports and the four highest-numbered 10/100/1000 network ports are mutually exclusive and cannot be enabled and active at the same time. This design effectively produces a 20 + 4 or 44 + 4 network port offering for the EX 3200 models with the four 1-Gigabit Ethernet uplink module installed.

Uplink ports are commonly used to connect an access switch to a distribution switch, or to interconnect member switches of a Virtual Chassis system across multiple wiring closets.

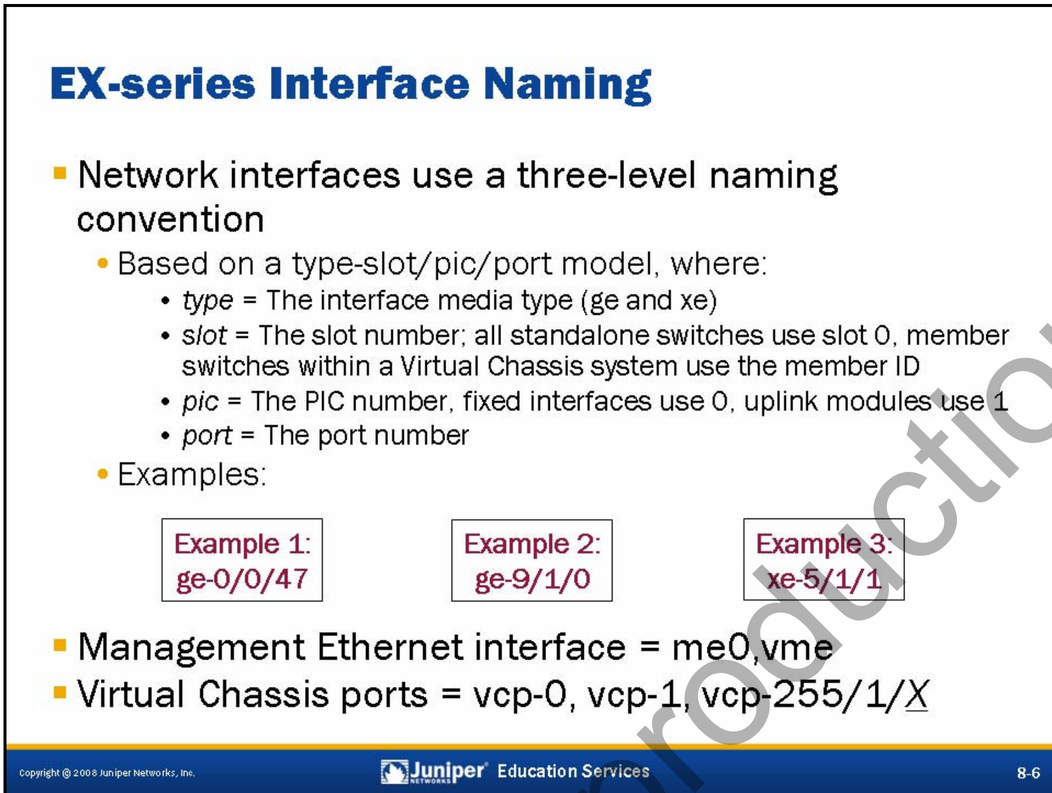
Continued on next page.

Management Ethernet Interface

All EX-series switches come standard with an out-of-band (OoB) management Ethernet interface (me0), which is located on the rear of the chassis. You can access the switch through the me0 interface over the network by using utilities such as SSH and Telnet. In the case of a Virtual Chassis configuration, the vme interface is used in place of the me0 interface. SNMP can use the management interface to gather statistics from the switch. To access the switch through the me0 or vme interface, you must configure it with a valid IP address.

Loopback Interface

The loopback interface is a software-only virtual interface that is always up. This interface provides a stable and consistent interface and IP address on the switch. The address associated with the loopback interface is commonly used by various protocols to avoid any impact if a physical interface goes down.



EX-series Interface Naming Convention

When configuring a network interface, follow the three-level naming convention outlined on the slide. The interface media type for the network interfaces is either ge- for 1-Gigabit Ethernet interfaces or xe- for the 10-Gigabit Ethernet interfaces. The slot number is always zero (0) for EX 3200 switches and stand-alone EX 4200 switches. When working with a Virtual Chassis system, the slot number matches the member ID assigned to the individual switch. The PIC level of the interface name is either 0, in the case of fixed Ethernet ports, or 1 when configuring a port associated with an uplink module. The port level indicates the individual port that is being configured. A hyphen (-) separates the interface media type from the slot number, and a slash (/) separates the slot number, PIC number, and port number for the configured interface. The examples provided on the slide use different media types and reflect various positioning scenarios.

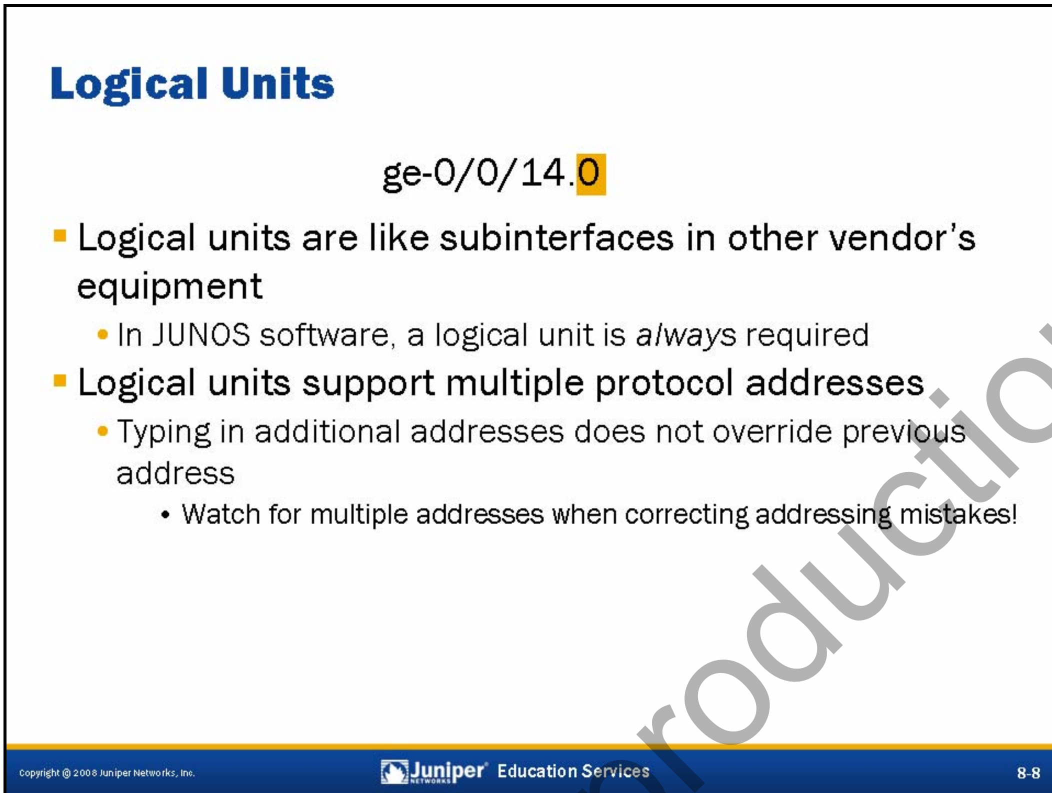
Management Ethernet Interface

All EX-series switches come standard with an OoB management Ethernet interface, which is located on the rear of the chassis. The management Ethernet interface assumes an interface name of me0. In a Virtual Chassis configuration the vme interface is configured in place of the me0 interface, even though the me0 port is still used for the physical connection.

Continued on next page.

Virtual Chassis Ports

Each EX 4200 switch has two dedicated Virtual Chassis ports (VCPs). The dedicated VCP interfaces are called vcp-0 and vcp-1. These ports are automatically configured by the software and are not user configurable. You can configure the two 10-Gigabit Ethernet uplink ports as Virtual Chassis extender ports (VCEPs). When these interfaces are configured as VCEP interfaces, they assume an interface name of vcp-255/1/*X* where *X* is equal to the actual port number (either 0 or 1).



Logical Interfaces

Each physical interface requires a logical unit. All physical network interfaces must use a logical unit value of zero. Layer 3 VLAN interfaces use multiple logical units. We cover Layer 3 VLAN interfaces in a subsequent chapter.

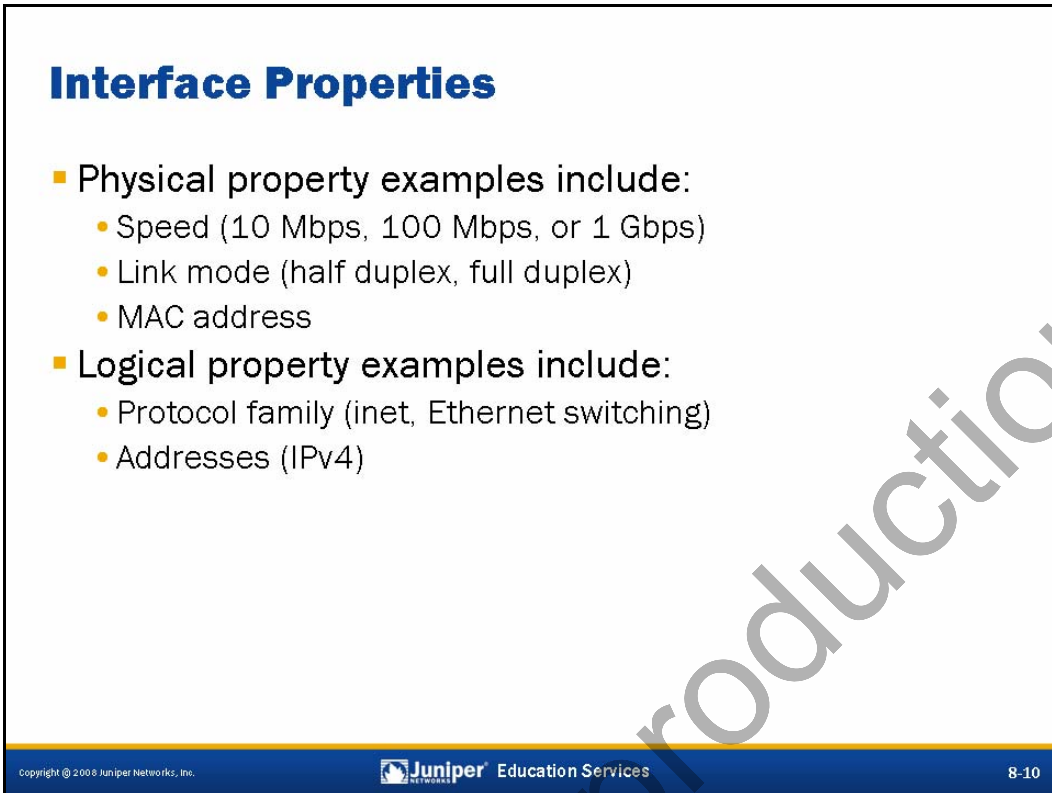
Continued on next page.

Multiple Addresses

A Juniper Networks EX-series platform can have more than one address on a single logical interface. Issuing a second **set** command does not overwrite the previous address but rather adds an additional address under the logical unit. Use of the CLI's **rename** command is an excellent way to correct addressing mistakes. An example is shown here:

```
[edit interfaces ge-0/0/10 unit 0]
user@switch# set family inet address 10.1.1.1

[edit interfaces ge-0/0/10 unit 0]
user@switch# show
family inet {
    address 10.1.1.1/32;
}

[edit interfaces ge-0/0/10 unit 0]
user@switch# rename family inet address 10.1.1.1/32 to address 10.1.1.1/24

[edit interfaces ge-0/0/10 unit 0]
user@switch# show
family inet {
    address 10.1.1.1/24;
}
```

Also note that JUNOS software forms interior gateway protocol (IGP) adjacencies over all subnets when the IGP is configured on a logical interface; this behavior is worth noting because some vendors form an adjacency over only the primary address of an interface.



Physical Properties

The following list provides details of some common physical interface properties:

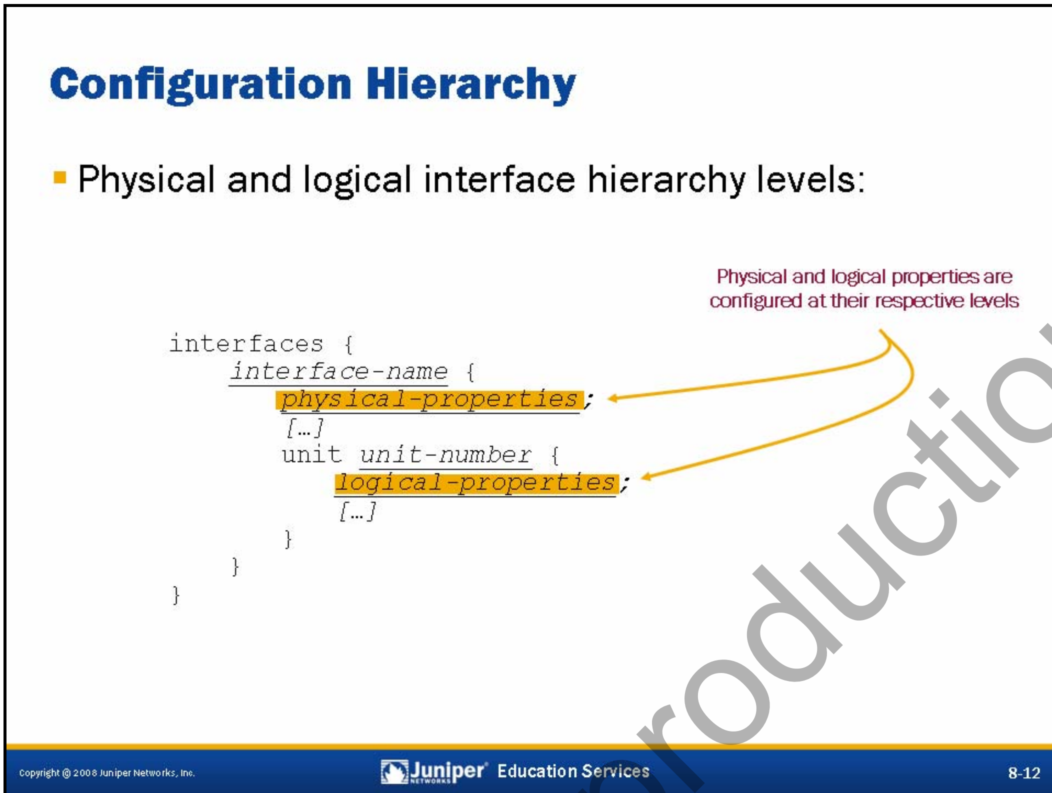
- `description`: Defines the physical interface description.
- `disable`: Disables the physical interface.
- `ether-options`: Sets Ethernet options such as link speed and duplex.
- `gratuitous-arp-reply`: Enables gratuitous Address Resolution Protocol (ARP) replies.
- `hold-time`: Defines the hold time for link up and link down events.
- `mac`: Sets the hardware MAC address for the interface.
- `mtu`: Defines the maximum transmit packet size (256–9216).
- `no-gratuitous-arp-reply`: Disables gratuitous ARP replies.
- `no-gratuitous-arp-request`: Ignores a gratuitous ARP request.
- `no-traps`: Disables SNMP notifications on state changes.
- `traceoptions`: Enables interface traceoptions.
- `traps`: Enables SNMP notifications on state changes.
- `unit`: Defines a logical interface.

Continued on next page.

Logical Properties

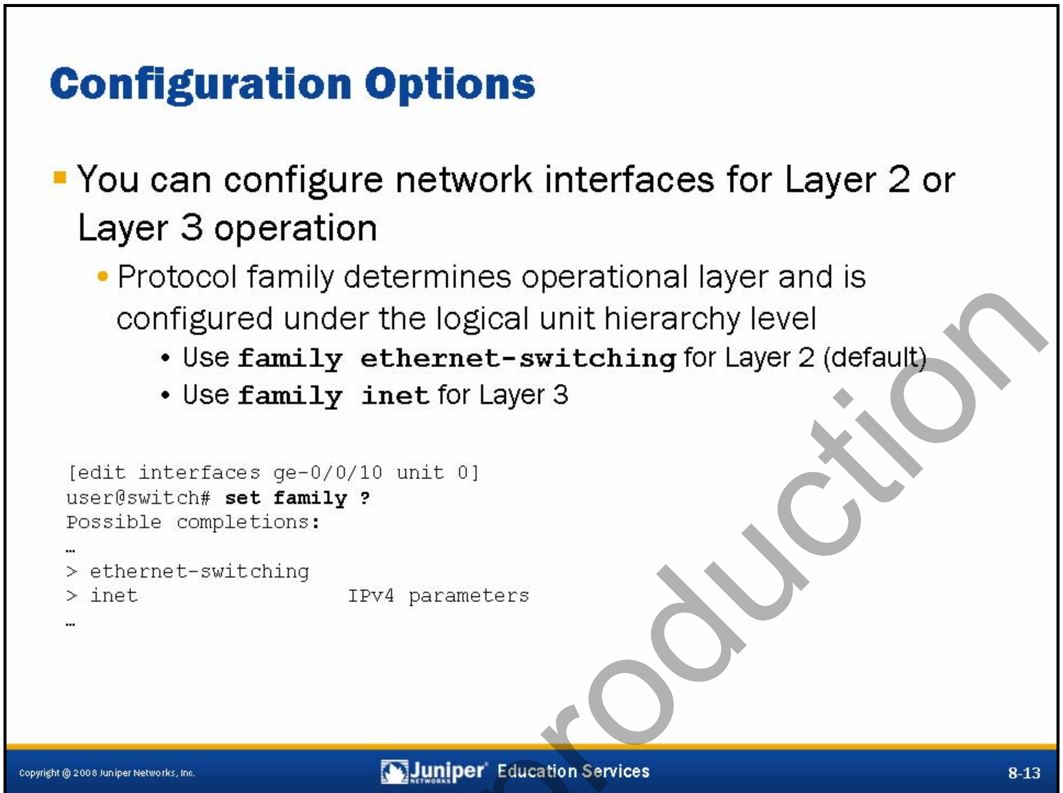
The following list provides details of some common logical interface properties:

- `accept-source-mac`: Enables Ethernet MAC address filtering.
- `bandwidth`: Defines the bandwidth for a logical unit (informational only).
- `description`: Defines the logical interface description.
- `disable`: Disables the logical interface.
- `family`: Sets the protocol family for interface.
- `no-traps`: Disables SNMP notifications on state changes.
- `traps`: Enables SNMP notifications on state changes.



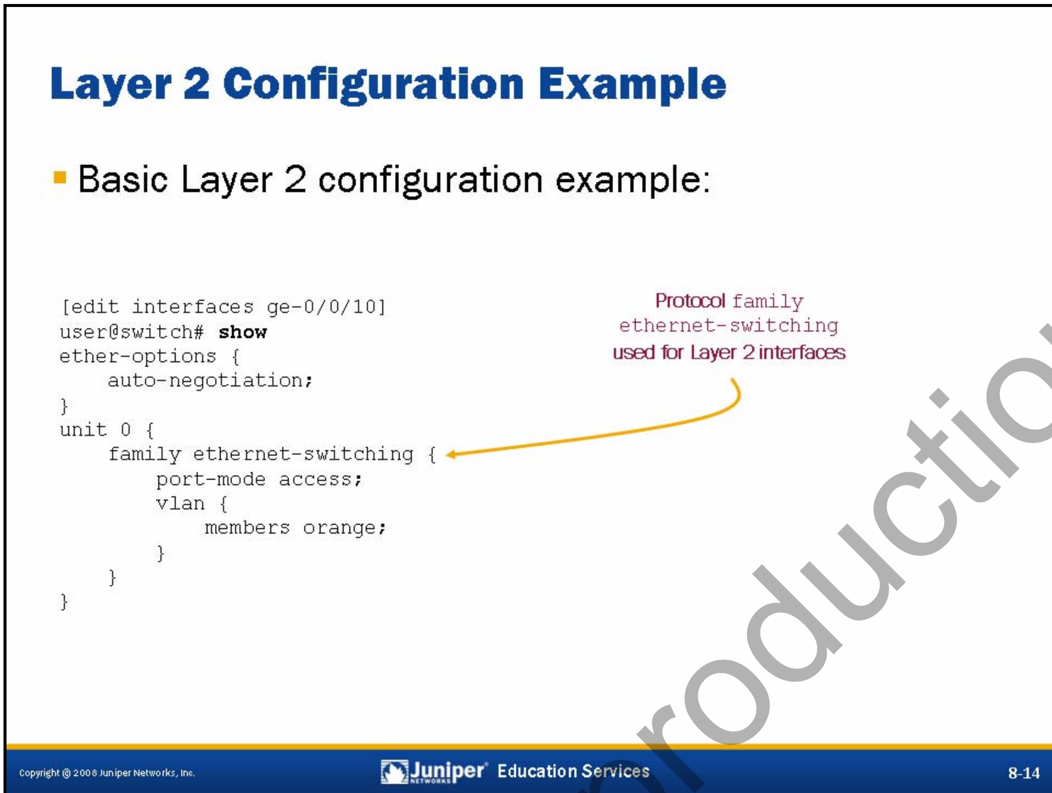
Organization of Interface Configuration

All interfaces have the same general configuration hierarchy organization. JUNOS software considers all properties defined directly under the interface name to be the physical properties of that interface. The unit number represents a particular logical interface or subinterface. JUNOS software considers all properties defined directly under the unit number to be the logical properties of each particular subinterface.



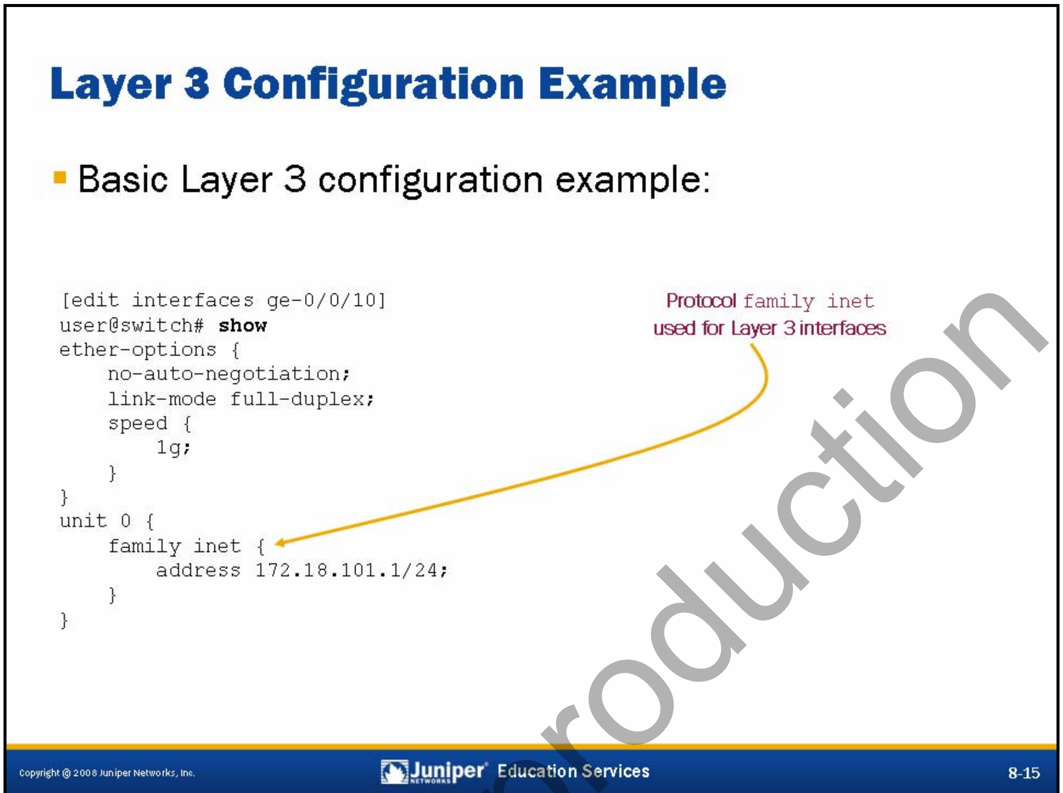
Interface Configuration Options

You can configure network interfaces for either Layer 2 or Layer 3 operation. The mode of operation for an interface is determined by the protocol family specified under the logical unit hierarchy level. Use **`family ethernet-switching`** for Layer 2 operation and **`family inet`** for Layer 3 operation.



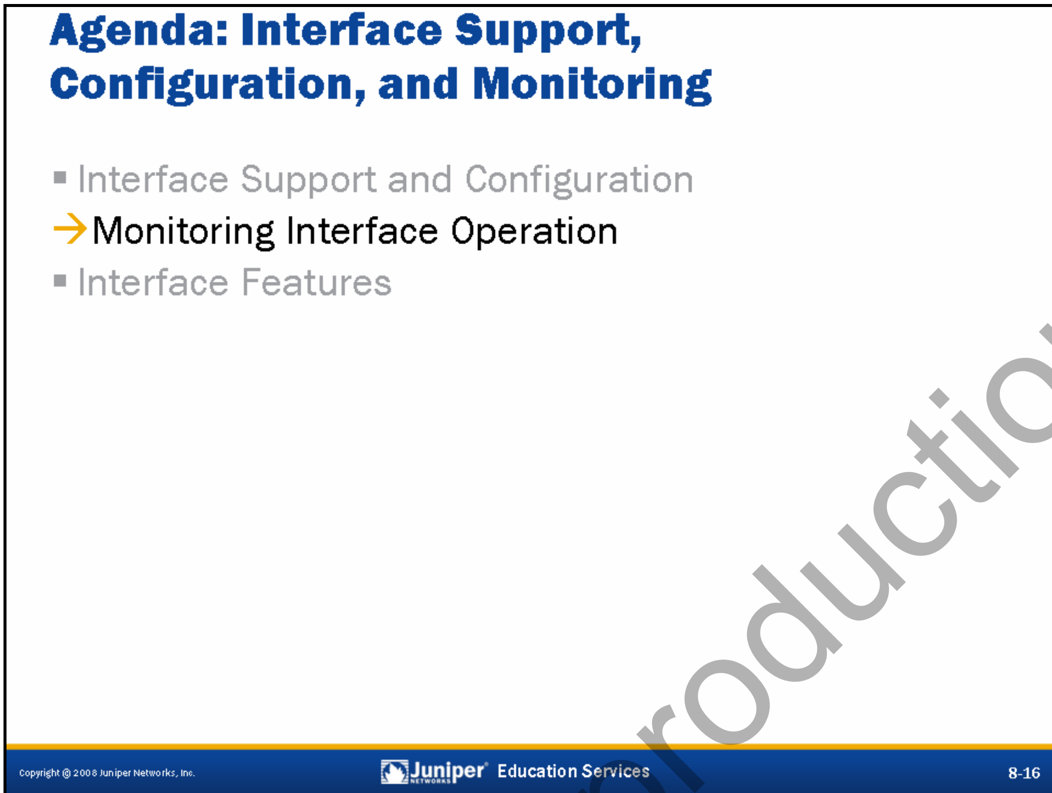
Layer 2 Configuration Example

The slide illustrates a typical Layer 2 interface configuration example. This example highlights the `family ethernet-switching` designation, which is required for an interface to operate in Layer 2 mode. This example displays the distinct physical and logical interface properties configured at their respective hierarchy levels.



Layer 3 Configuration Example

The slide illustrates a typical Layer 3 interface configuration example. This example highlights the `family inet` designation, which is required for an interface to operate in Layer 3 mode. This example also shows distinct physical and logical interface properties configured at their respective hierarchy levels.



Monitoring Interface Operation

The slide highlights the topic we discuss next.

Verifying Interface Status

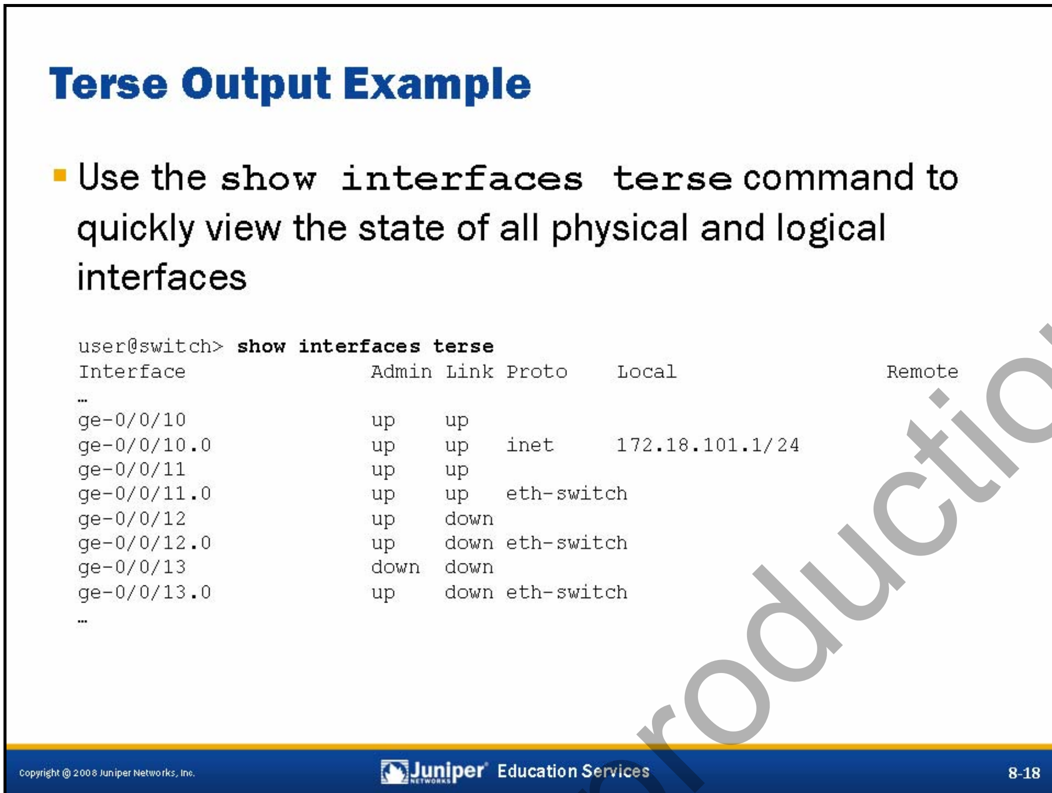
- Use the **`show interfaces`** command to verify interface status
  - Command options include **`terse`**, **`brief`**, **`detail`**, and **`extensive`**
    - Details displayed vary depending on the command option used
    - Use the interface name option to view details for a specific interface

```
user@switch> show interfaces ge-0/0/10 ?
Possible completions:
  <[Enter]>            Execute this command
  brief                Display brief output
  descriptions         Display interface description strings
  detail               Display detailed output
  extensive            Display extensive output
  media                Display media information
  snmp-index           SNMP index of interface
  statistics           Display statistics and detailed output
  terse                Display terse output
  |                    Pipe through a command
```

 Juniper Education Services 8-17

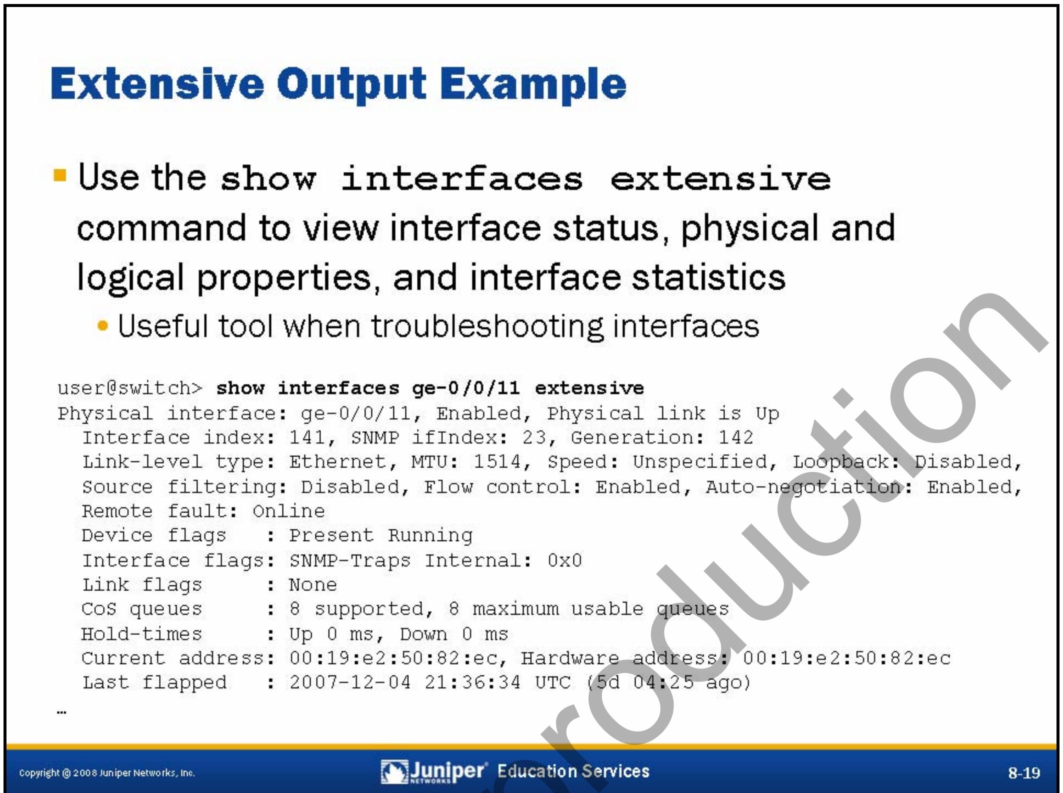
Interface Status Verification

You can use the **`show interfaces`** command to verify various details and status information for interfaces. A number of command options exist that determine the generated output for the **`show interfaces`** command. The example on the slide illustrates the use of the ***`interface-name`*** option, which filters the generated output and displays details only for the specified interface. If the ***`interface-name`*** option is excluded, the switch displays interface details for all installed interfaces.



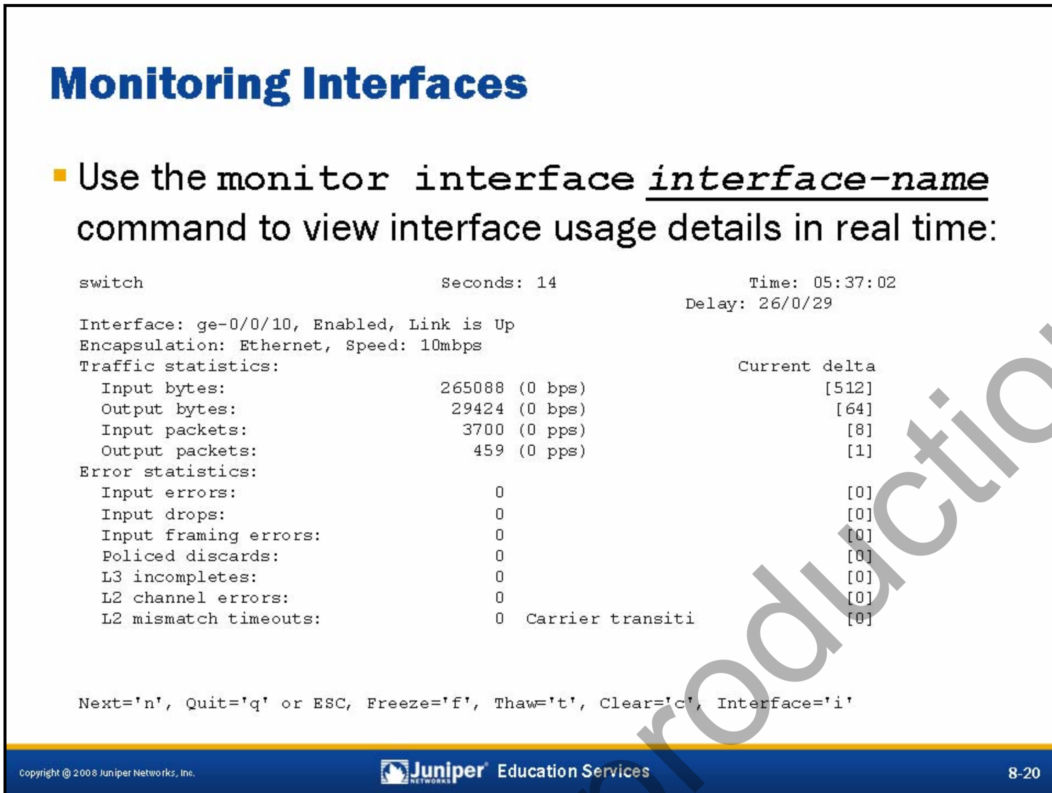
Terse Output Example

The example on the slide illustrates the **`show interfaces terse`** command. In this example the ***`interface-name`*** option is omitted, which causes all installed interfaces and their accompanying details to be displayed. This command is ideal when you simply need to verify state information for physical and logical interfaces. The output from this command displays all installed interfaces in the left column and provides state, protocol family, and addressing details to the right of each listed interface.



Gathering Extensive Interface Information

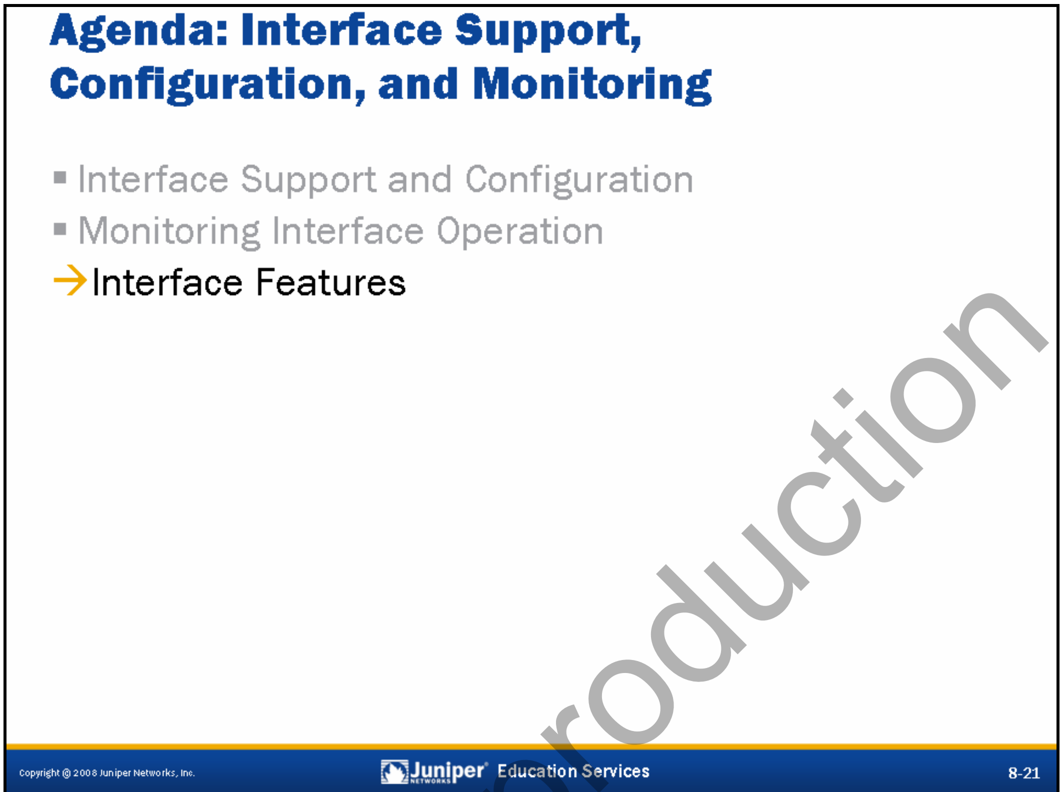
Use the **`show interface extensive`** command to view detailed information for a named interface (or all interfaces when a specific interface is not identified). The example on the slide shows a portion of the generated output when using the **`extensive`** option. This command is ideal when investigating or troubleshooting interfaces because it shows extensive physical and logical interface properties. This is also a great command when determining default settings for interfaces.



Monitoring an Interface

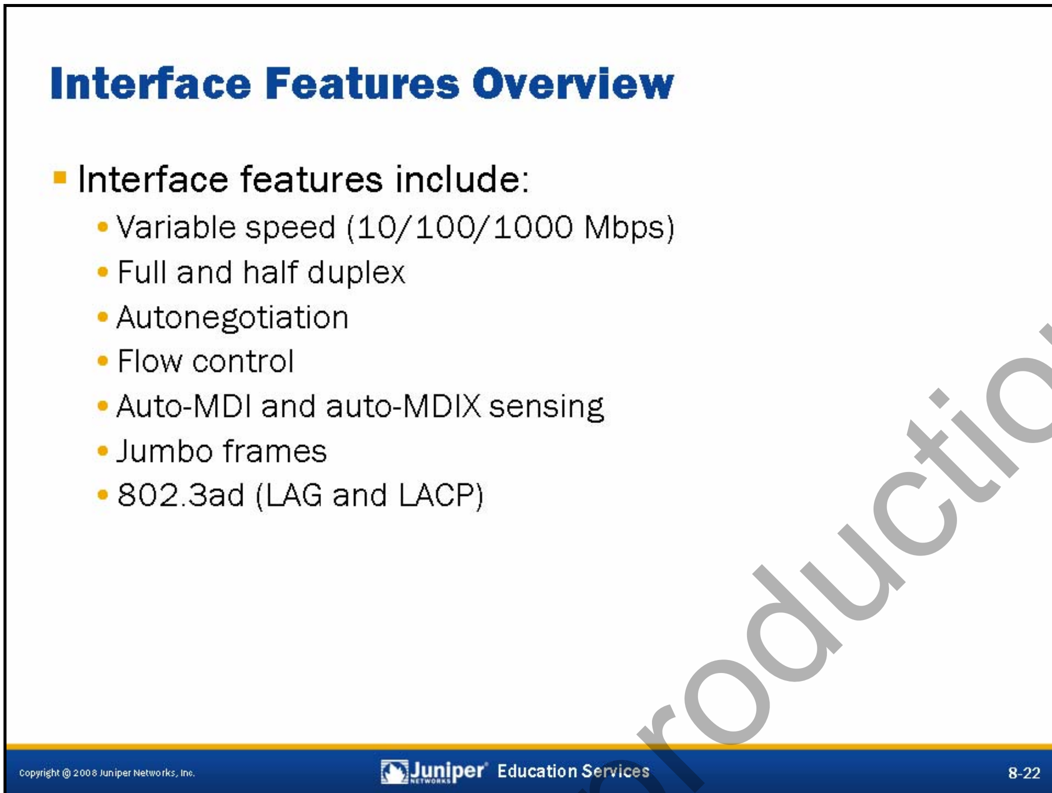
The slide depicts typical output from the **monitor interface** command. Your terminal session must support VT100 emulation for the screen to display correctly. This command provides real-time packet and byte counters as well as displaying error and alarm conditions. To view real-time usage statistics for all interfaces, use the **monitor interface traffic** command. A sample of this command's output is shown:

```
user@switch> monitor interface traffic
switch                             Seconds: 2                    Time: 04:08:36

Interface     Link  Input packets         (pps)     Output packets          (pps)
 ge-0/0/0     Down           3568           (0)                  0            (0)
 ge-0/0/1     Down              0           (0)                  0            (0)
 ge-0/0/2     Down              0           (0)                  0            (0)
 ge-0/0/3     Down              0           (0)                  0            (0)
 ge-0/0/4     Down              0           (0)                  0            (0)
 ge-0/0/5       Up         127389           (0)               3049            (0)
 ...
Bytes=b, Clear=c, Delta=d, Packets=p, Quit=q or ESC, Rate=r, Up=^U, Down=^D
```



Interface Features

The slide highlights the topic we discuss next.



Interface Features

This slide details interface features supported on the EX-series switches.

802.3ad Link Aggregation

- Definition: Method of grouping multiple Ethernet interfaces to form a single link layer interface, also known as a link aggregation group (LAG) or bundle
  - Uses 802.3ad LACP as its discovery protocol
  - Participating interfaces are known as *member links*
  - Commonly used to aggregate trunk links
- Usage and benefits:
  - Increases bandwidth
  - Provides link efficiency
  - Creates physical layer redundancy

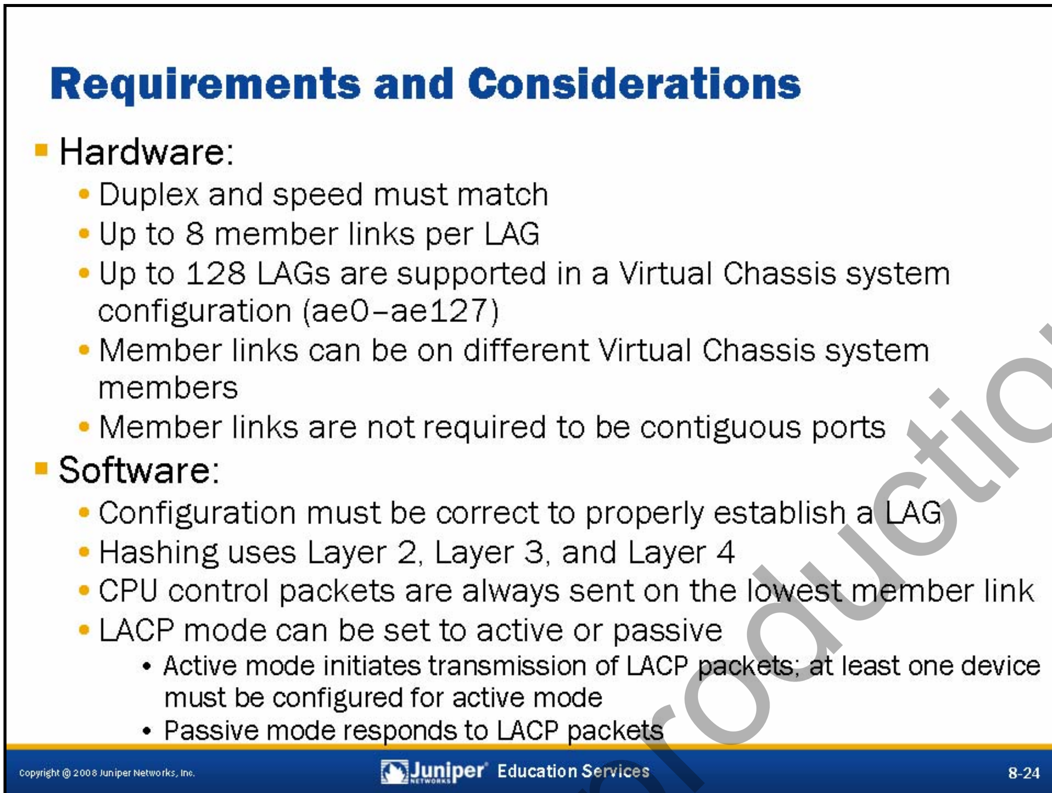
 Juniper Networks Education Services 8-23

Link Aggregation

The Institute of Electrical and Electronics Engineers (IEEE) 802.3ad link aggregation specification enables multiple Ethernet interfaces to be grouped together and form a single link layer interface, also known as a link aggregation group (LAG) or bundle. IEEE 802.3ad uses the Link Aggregation Control Protocol (LACP) as its discovery protocol. All links participating in a LAG are considered members. A typical deployment for LAG is to aggregate trunk links between an access switch and a distribution switch or customer edge (CE) router.

Usage and Benefits

Link aggregation takes place on point-to-point connections between two devices. A LAG balances traffic across the member links within an aggregated Ethernet bundle and effectively increases the uplink bandwidth. Another advantage of link aggregation is increased availability because the LAG is composed of multiple member links. If one member link fails, the LAG continues to carry traffic over the remaining links.



Hardware Requirements and Considerations

A number of hardware requirements and considerations exist when working with link aggregation. The following list highlights these details:

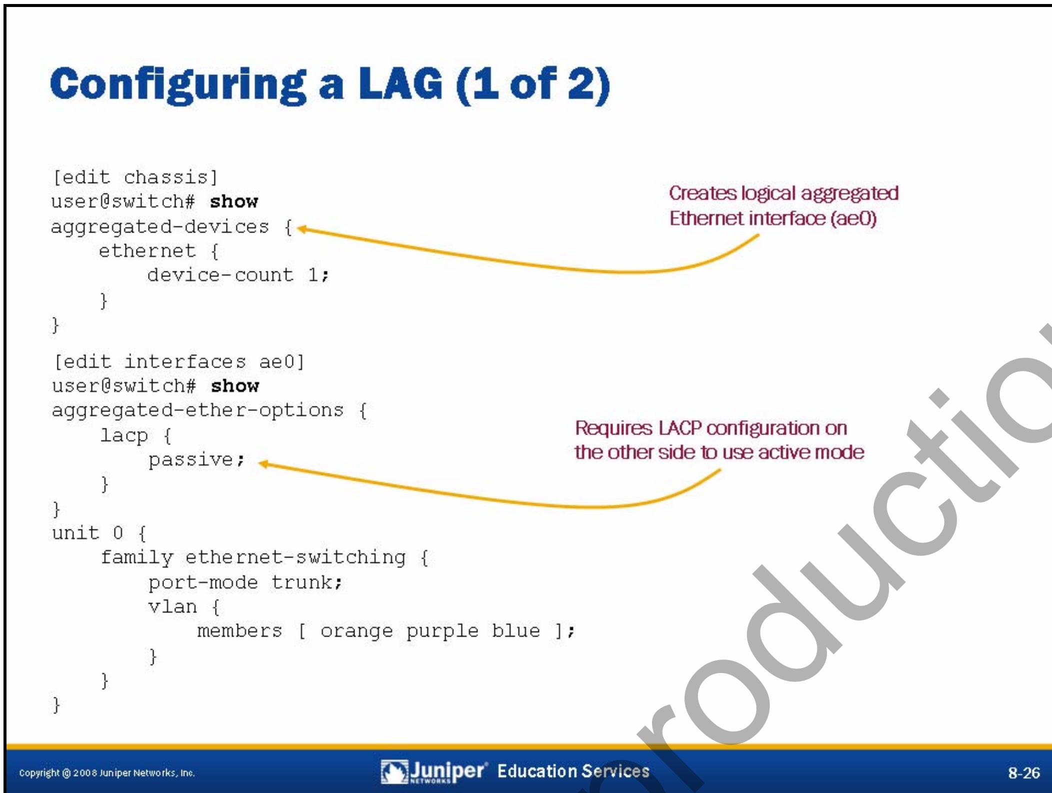
- Duplex and speed settings must match on both participating devices.
- Up to eight member links can belong to a single LAG.
- Up to 64 LAGs are supported on the EX 4200 switch, and up to 32 LAGs are supported on the EX 3200 switch.
- Member links can reside on different members within a Virtual Chassis system.
- Member links within a bundle are not required to be contiguous ports.

Continued on next page.

Software Requirements and Considerations

A number of software requirements and considerations exist when working with link aggregation. The following list highlights these details:

- The load-balancing hash algorithm uses criteria at Layers 2, 3, and 4, dependant upon the layer at which the interface is configured. No configuration is necessary to enable load balancing.
- All control packets that traverse the LAG use the lowest member link.
- If LACP is used for a LAG, at least one side must be configured in active mode. You can configure LACP in active or passive mode. If the device is configured for active mode, it actively transmits protocol data units (PDUs). If the device is configured in passive mode, it responds to the PDUs it receives.



LAG Configuration Example: Part 1

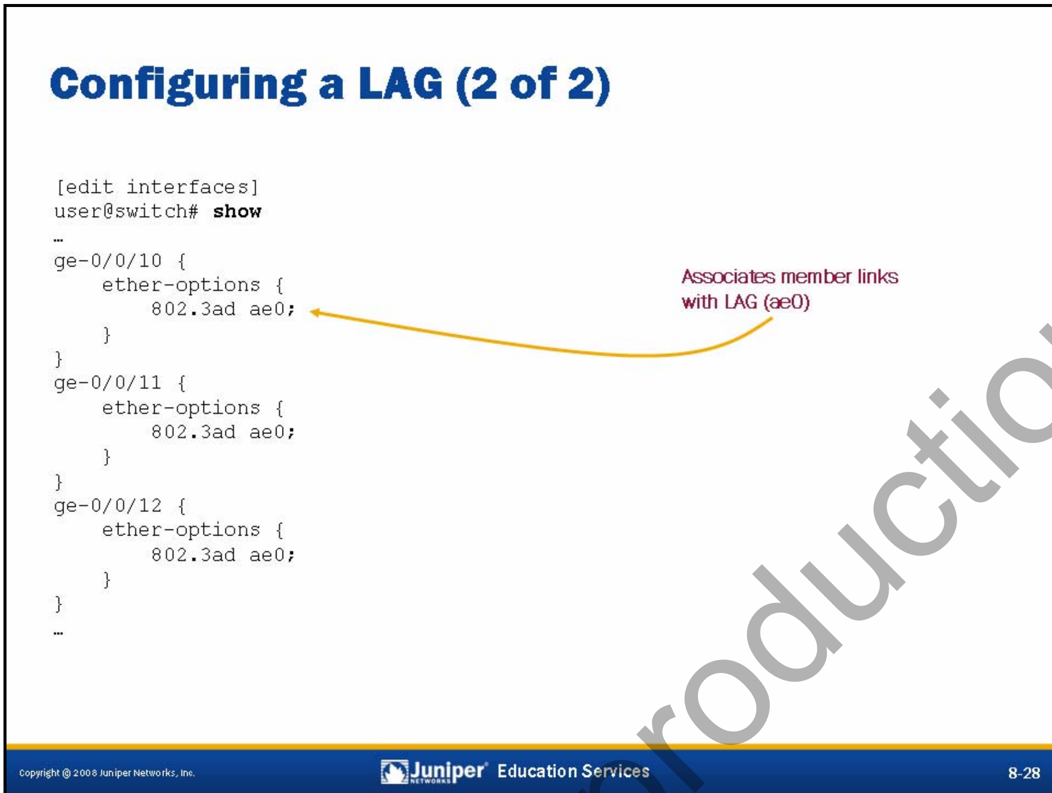
This and the next slide illustrate the steps used to configure link aggregation. The first step creates a logical aggregated Ethernet interface. In this example a single aggregated interface, ae0, is created. By default, no aggregated interfaces exist. To create an aggregated interface, simply add an aggregated device under the `[edit chassis]` hierarchy, as shown in the example on the slide. Once this portion of the configuration is committed, the switch creates the ae0 interface. An example, which illustrates this behavior, is shown here:

```
[edit]
user@switch# run show interfaces terse |match ae

[edit]
user@switch# edit chassis
[edit chassis]
user@switch# set aggregated-devices ethernet device-count 1
[edit chassis]
user@switch# commit
commit complete
[edit chassis]
user@switch# run show interfaces terse |match ae
ae0                         up     down
```

Continued on next page.

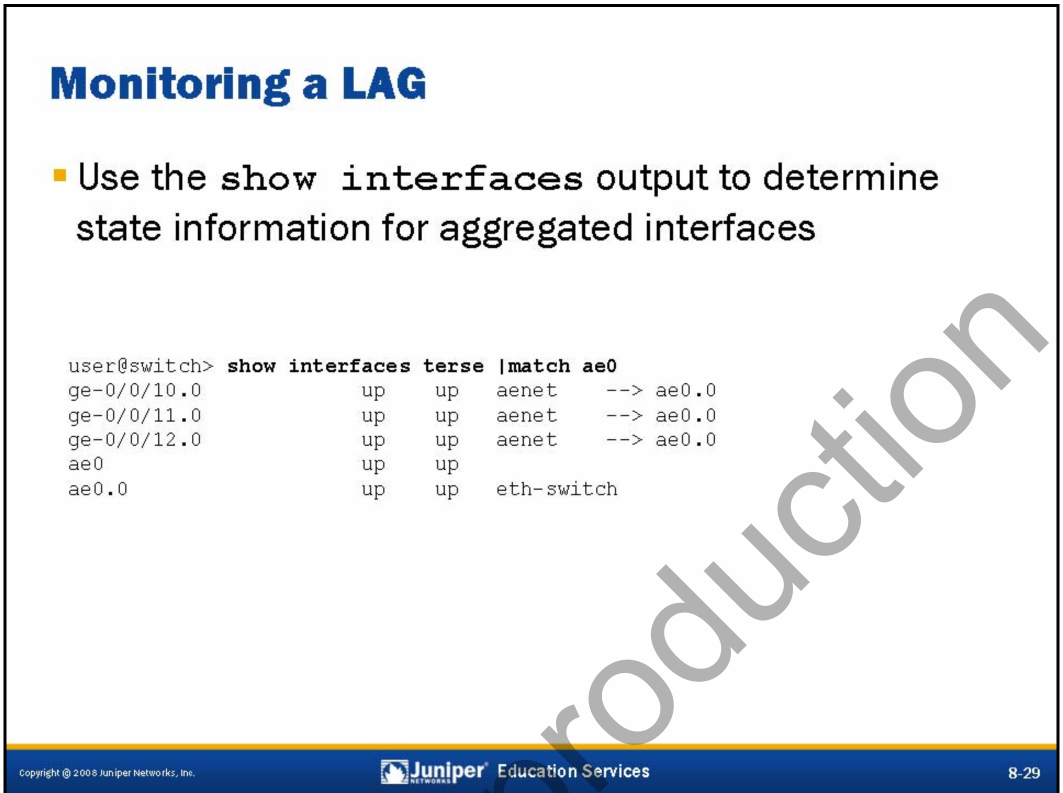
LAG Configuration Example: Part 1 (contd.)

The next step is to define the parameters associated with the ae0 interface. As shown on the slide, the ae0 interface configuration includes at least one logical unit along with the desired logical interface properties. The example on the slide also includes LACP under the `aggregated-ether-options` hierarchy level. As previously indicated, if LACP is used, at least one side must be configured in active mode to successfully establish the connection.



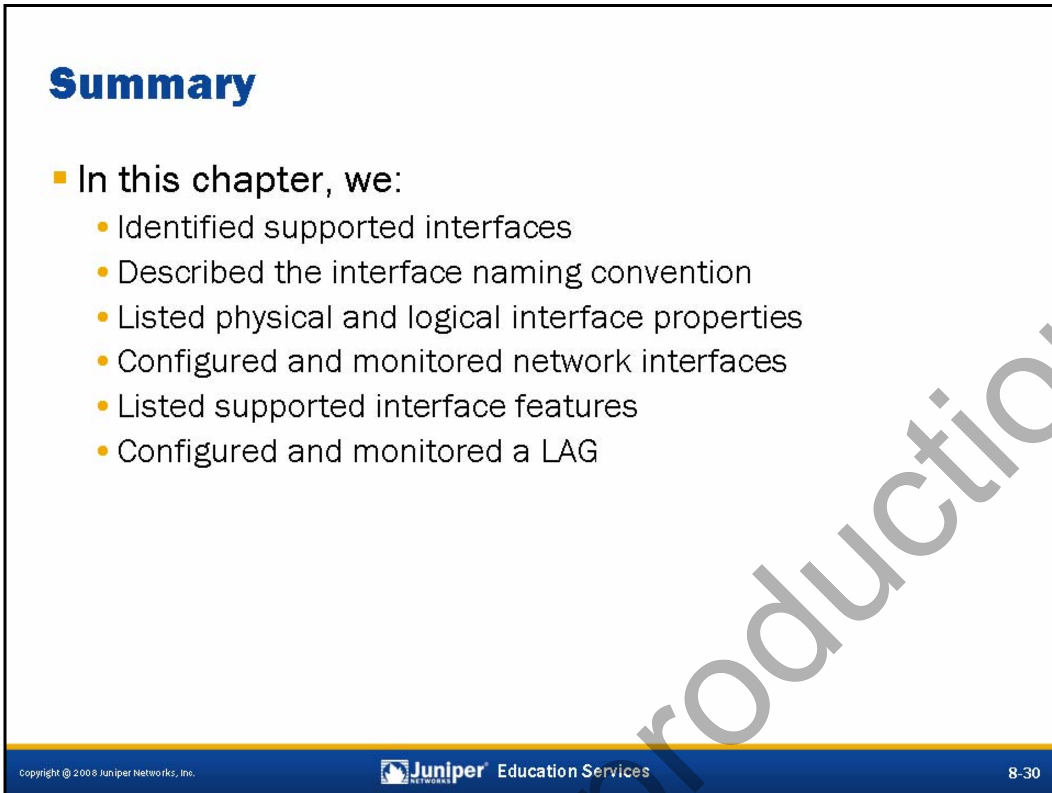
LAG Configuration Example: Part 2

Once the ae0 interface is created and defined within the configuration, the individual member links must be defined and associated with the ae0 bundle. The example on the slide illustrates a typical configuration example, which links member interfaces ge-0/0/10, ge-0/0/11, and ge-0/0/12 with the associated ae0 interface.



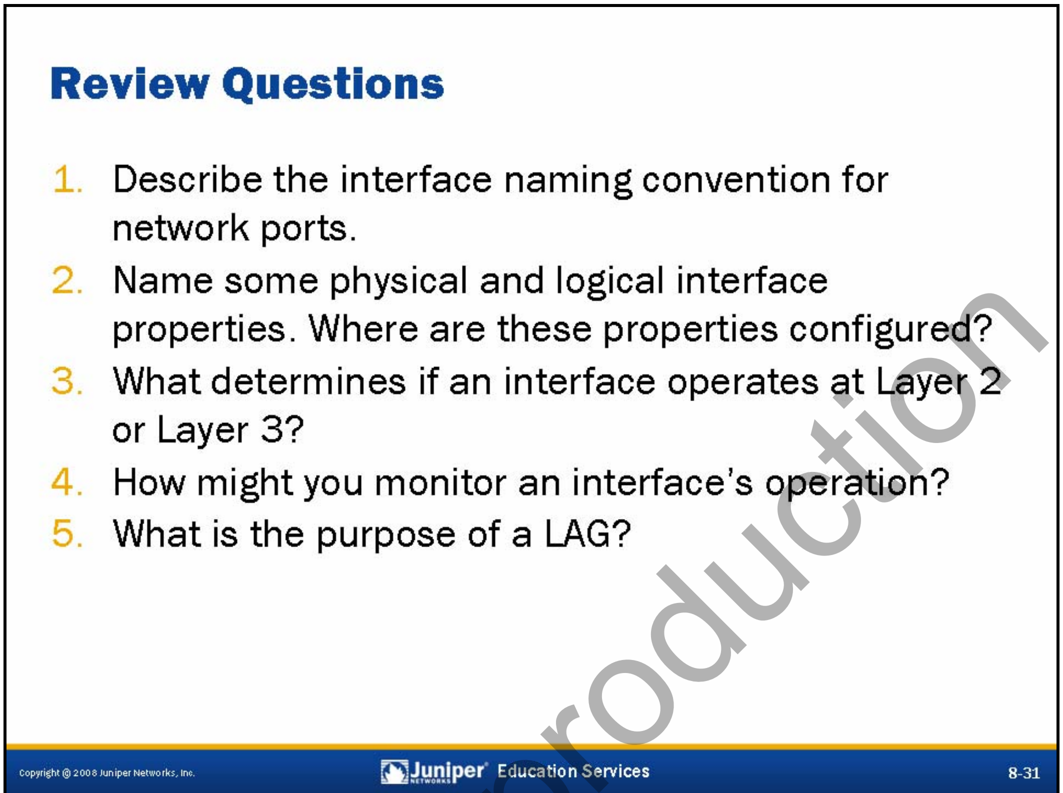
Monitoring the Operation of a LAG

Use the **`show interfaces`** command with the desired option to verify the operational state for the aggregated interface and member links. The example on the slide makes use of the **`terse`** command option along with the pipe (**|**) option to filter the generated output. This example shows that the ae0 interface along with all member links are operationally up.



This Chapter Discussed:

- Supported interface types;
- Interface naming convention;
- Physical and logical interface properties;
- Configuration and monitoring details for network interfaces;
- Interface features; and
- Configuration and monitoring of a LAG.



Review Questions

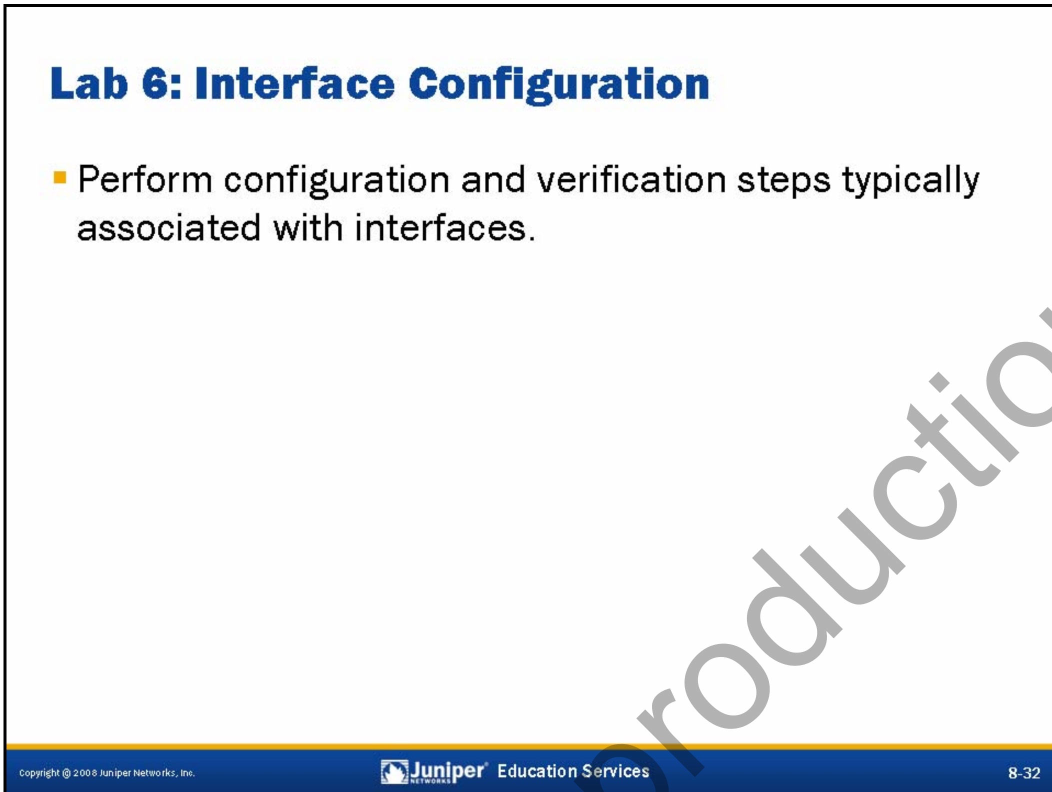
1.

2.

3.

4.

5.



Lab 6: Interface Configuration

The slide provides the objective for this lab.

Operating Juniper Networks Switches in the Enterprise

Chapter 9: Ethernet Switching and Virtual LANs

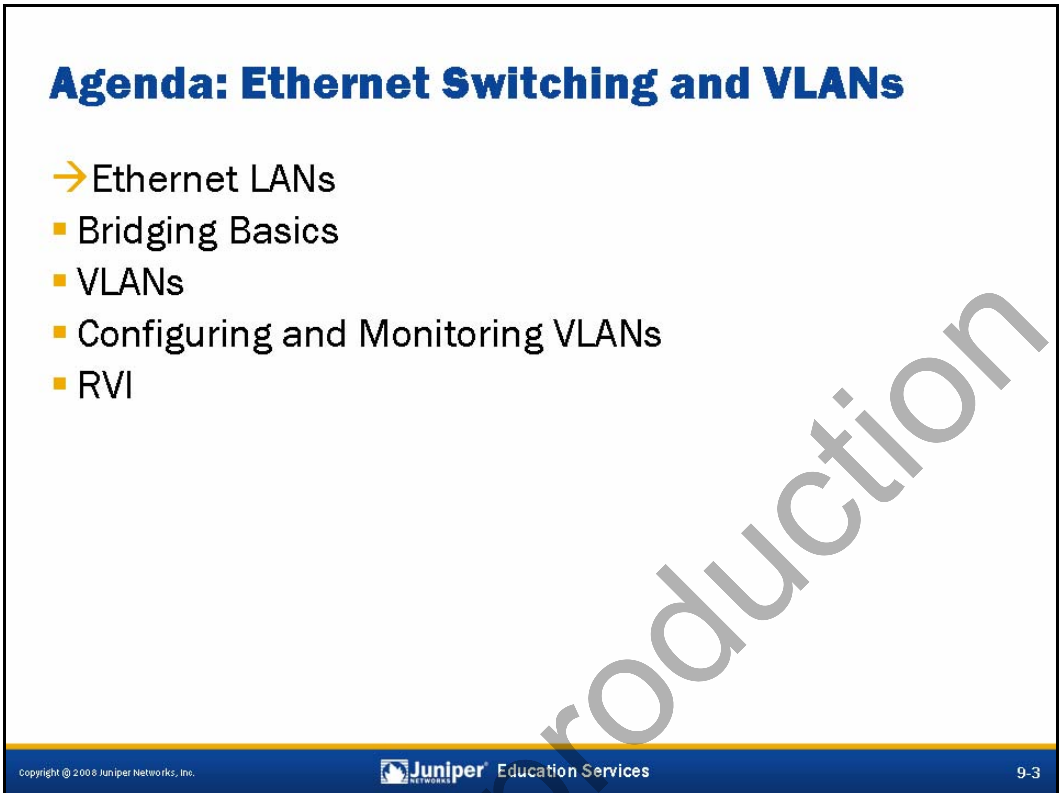
Chapter Objectives

- After successfully completing this chapter, you will be able to:
  - Describe the basic operation of an Ethernet LAN
  - Explain the operation and benefits of bridging
  - Describe the purpose and list the benefits of VLANs
  - Configure and verify proper operation of VLANs
  - Describe the purpose of an RVI
  - Configure and verify proper operation of an RVI

 Juniper NETWORKS Education Services 9-2

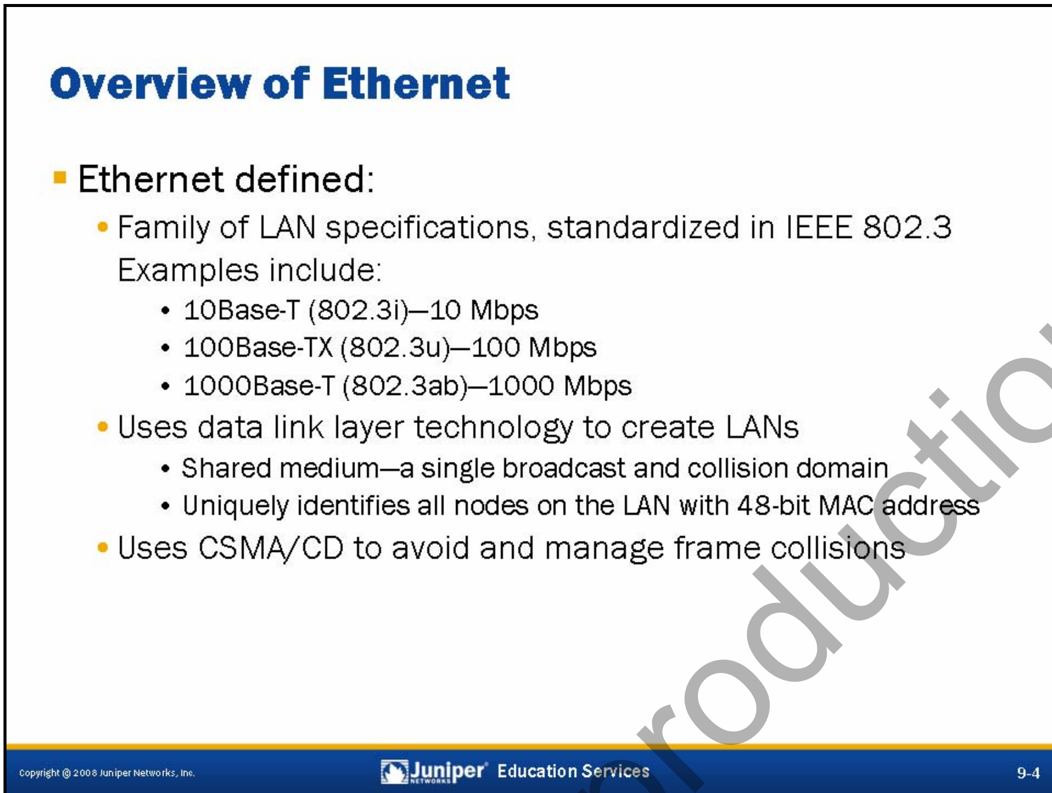
This Chapter Discusses:

- Basic operation of an Ethernet LAN;
- Operation and benefits of bridging;
- The purpose and benefits of virtual LANs (VLANs);
- Configuring and monitoring VLANs;
- The purpose and benefits of a routed VLAN interface (RVI); and
- Configuring and monitoring an RVI.



Ethernet LANs

The slide lists the topics we cover in this chapter. We discuss the highlighted topic first.

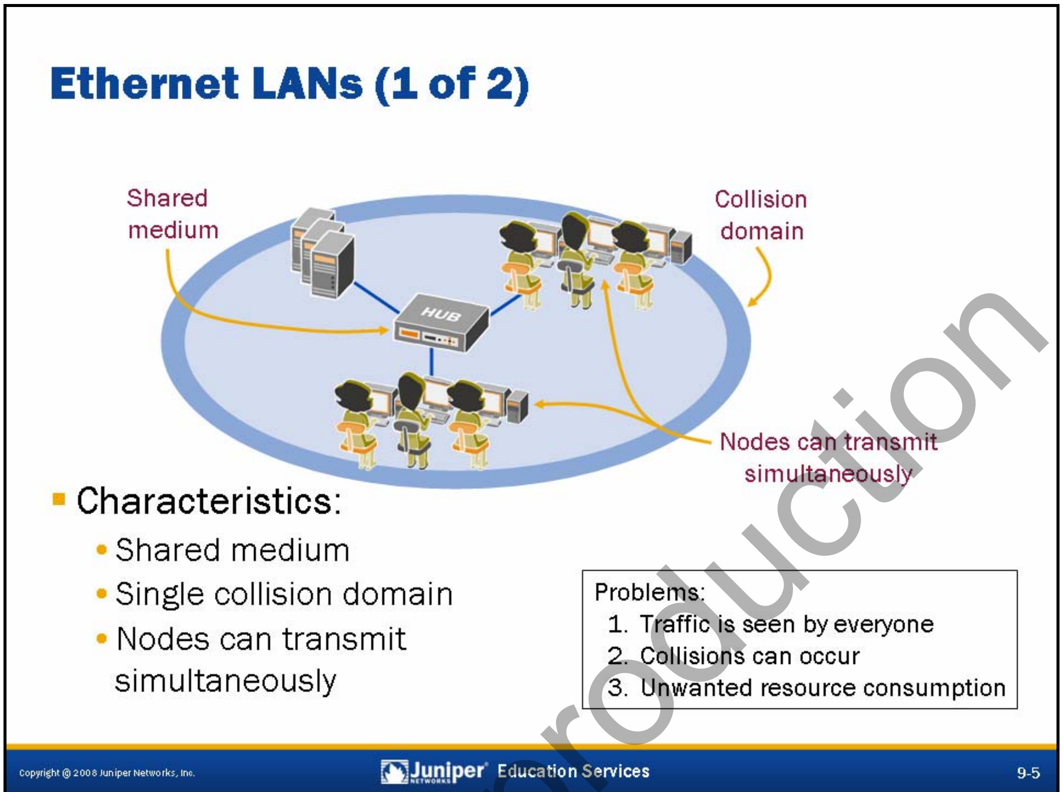


Ethernet Defined

Ethernet is a family of LAN specifications defined in the Institute of Electrical and Electronics Engineers (IEEE) 802.3 standard. The slide provides some common examples, including the 803.2i, 802.3u, and 802.3ab specifications. Each Ethernet implementation uses a unique wiring and signaling standard—typically a copper-based medium or fiber optics—for the physical layer. Although the various implementations of Ethernet can use various wiring and signaling standards, they all use a common addressing format.

Ethernet is a data link layer technology, as defined by Layer 2 of the Open Systems Interconnection (OSI) model of communications. An Ethernet LAN consists of shared medium, which encompasses a single broadcast and collision domain. Network devices, called nodes, on the Ethernet LAN transmit data in bundles that are generally called frames or packets. Each node on a LAN has a unique identifier so that it can be unambiguously located on the network. Ethernet uses the Layer 2 media access control (MAC) address for this purpose. MAC addresses are 48-bit hardware addresses programmed into the Ethernet processor of each node.

Ethernet uses the carrier-sense multiple access with collision detection (CSMA/CD) protocol to avoid and manage frame collisions.

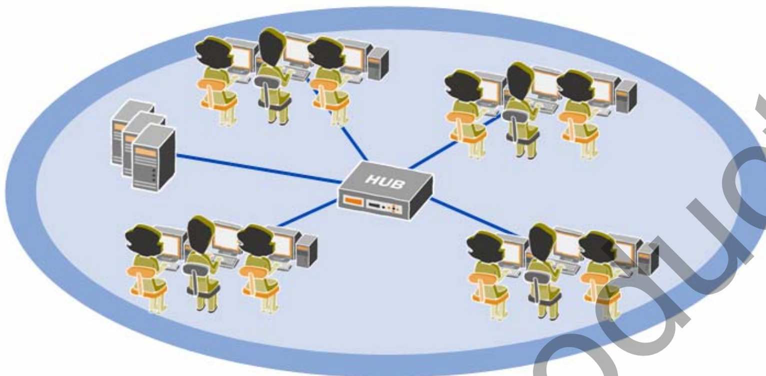


Ethernet LANs: Part 1

Ethernet LANs consist of a shared medium, which defines a single collision domain. As previously mentioned, Ethernet uses the CSMA/CD protocol to help avoid and manage frame collisions. The sample topology on the slide shows a series of nodes connected through a hub using a copper-based physical medium. This type of implementation only allows a single stream of data at a time. All nodes participating in this shared Ethernet LAN listen to verify that the line is idle before transmitting. If the line is idle, the nodes begin transmitting data frames. If multiple nodes listen and detect that the line is idle and then begin transmitting data frames simultaneously, a collision will occur. When collisions occur an error is generated and sent back to the transmitting devices. When a node receives a collision error message, it stops transmitting immediately and waits for a period of time before trying to send the frame again. If the node continues to detect collisions, it progressively increases the time between retransmissions in an attempt to find a time when no other data is being transmitted on the LAN. The node uses a backoff algorithm to calculate the increasing retransmission time intervals. When a node does successfully transmit traffic, that traffic is replicated out all ports on the hub and is seen by all other nodes on the shared Ethernet segment. This traffic-flooding approach, coupled with collisions, consumes network resources.

Ethernet LANs (2 of 2)

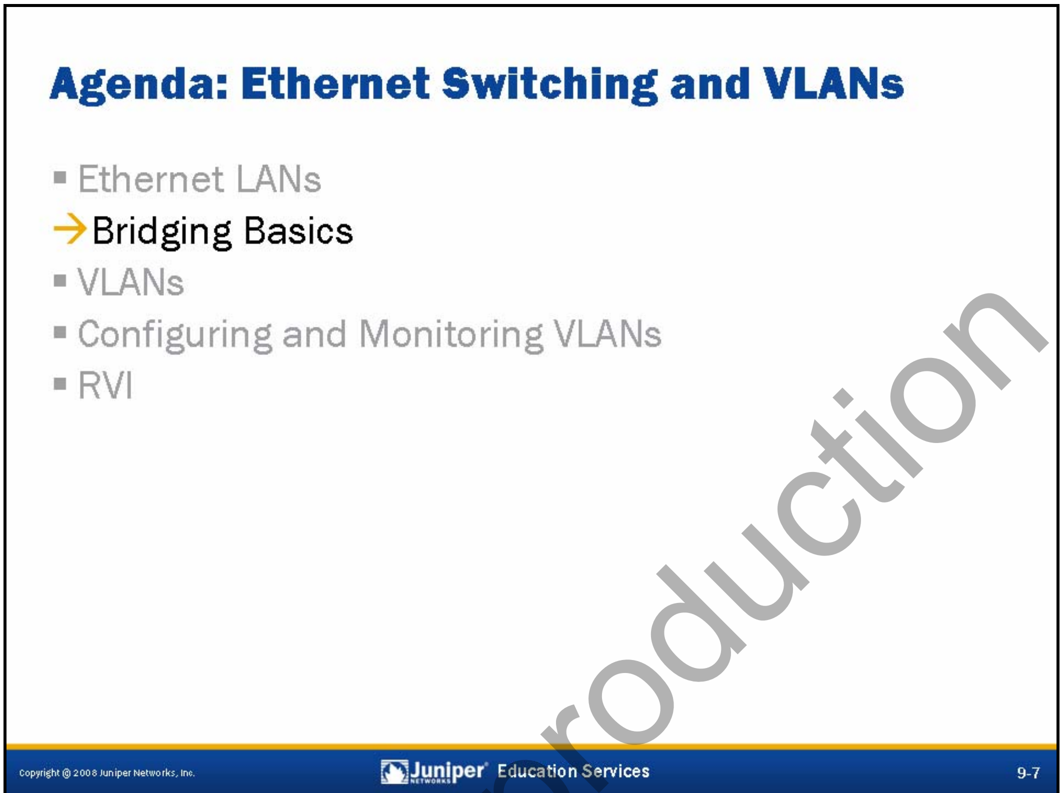
- As the network grows, the likelihood of collisions increases
  - As collisions increase, overall LAN efficiency decreases



Juniper Education Services
9-6

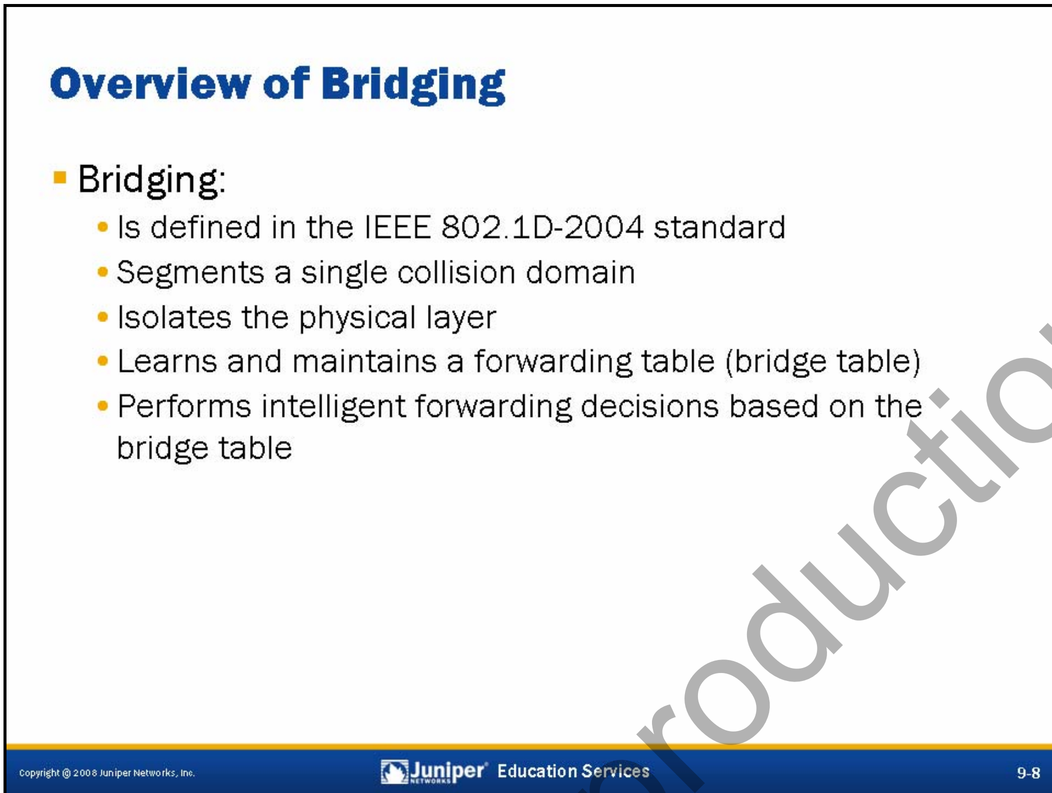
Ethernet LANs: Part 2

Ethernet LANs were originally implemented for small, simple networks. Over time, LANs have become larger and more complex. As an Ethernet LAN grows, the likelihood of collisions on that LAN also grows. As more users are added to a shared Ethernet segment, each participating node receives an increase of traffic from all other participating nodes for which it is not the actual destination. This unwanted consumption of network resources along with an increase of collisions inevitably decreases the overall efficiency on the LAN.



Bridging Basics

The slide highlights the topic we discuss next.

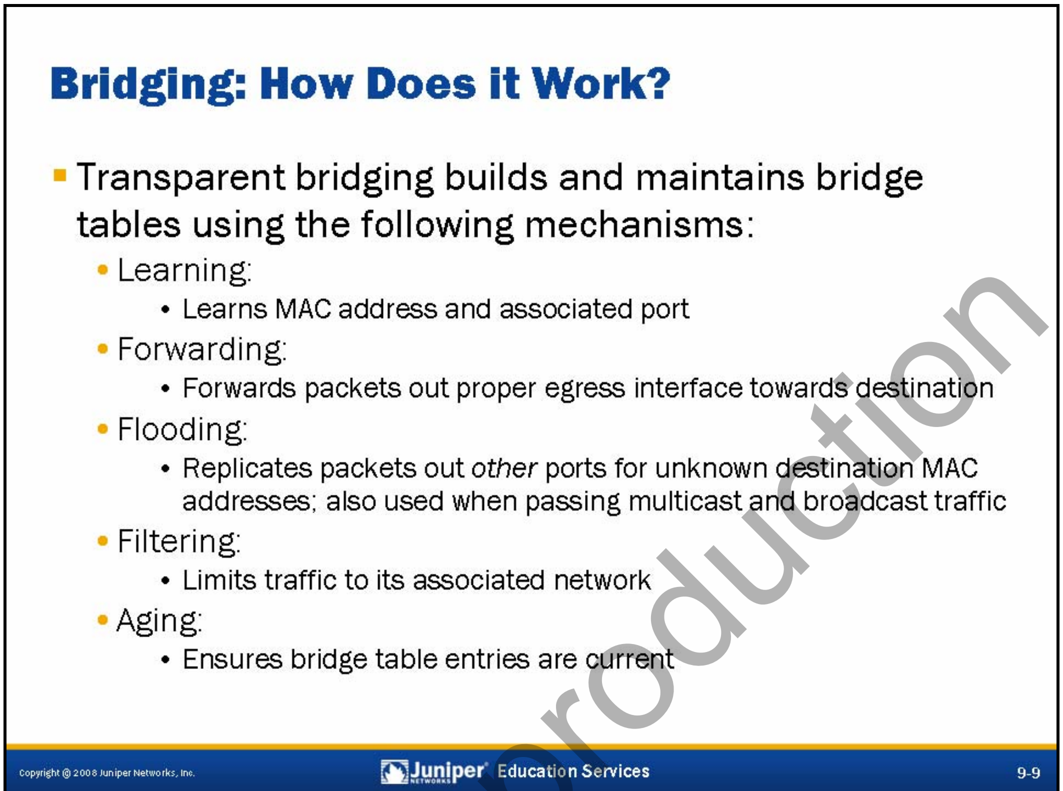


Bridging Defined

Defined in the IEEE 802.1D-2004 standard, bridging addresses some of the inherent problems of large shared Ethernet LANs. Bridging uses microsegmentation to divide a single collision domain into multiple, smaller bridged collision domains. Reducing the size of a collision domain effectively reduces the likelihood that collisions will occur. This approach also enhances performance by allowing multiple streams of data to flow through the switch within a common LAN or broadcast domain.

Bridging allows a mixed collection of interface types and speeds to be logically grouped within the same bridged LAN. The ability to logically group dissimilar interfaces in a bridged LAN environment provides design flexibility not found in a shared Ethernet LAN environment.

Bridging builds and maintains a forwarding table, known as a *bridge table*, for all destinations within the bridged LAN. The bridge table is based on the source MAC address for all devices participating in the bridged LAN. The bridge table is used to aid in intelligent forwarding decisions. This approach reduces unnecessary traffic on the LAN.



Bridging Mechanics

The transparent bridging protocol allows a switch to learn information about all nodes on the LAN. The switch uses this information to create the address-lookup tables called bridge tables that it consults when forwarding traffic to (or toward) a destination on the LAN.

When a switch is first connected to an Ethernet LAN or VLAN, it has no information about other nodes on the network. *Learning* is a process the switch uses to obtain the MAC addresses of all the nodes on the network. It stores these addresses in an address book called a bridge table. To learn MAC addresses, the switch reads all packets that it detects on the LAN or on the local VLAN, looking for MAC addresses of sending nodes. It places these addresses into its bridge table, along with two other pieces of information—the interface (or port) on which the traffic was received and the time when the address was learned.

The *forwarding* mechanism is used by the switch to deliver traffic, passing it from an incoming interface to an outgoing interface that leads to (or toward) the destination. To forward frames, the switch consults the bridge table to see whether the table contains the MAC address corresponding to the frames' destination. If the bridge table contains an entry for the desired destination address, the switch sends the traffic out the interface associated with the MAC address. The switch also consults the bridge table in the same way when transmitting frames that originate on devices connected directly to the switch.

Continued on next page.

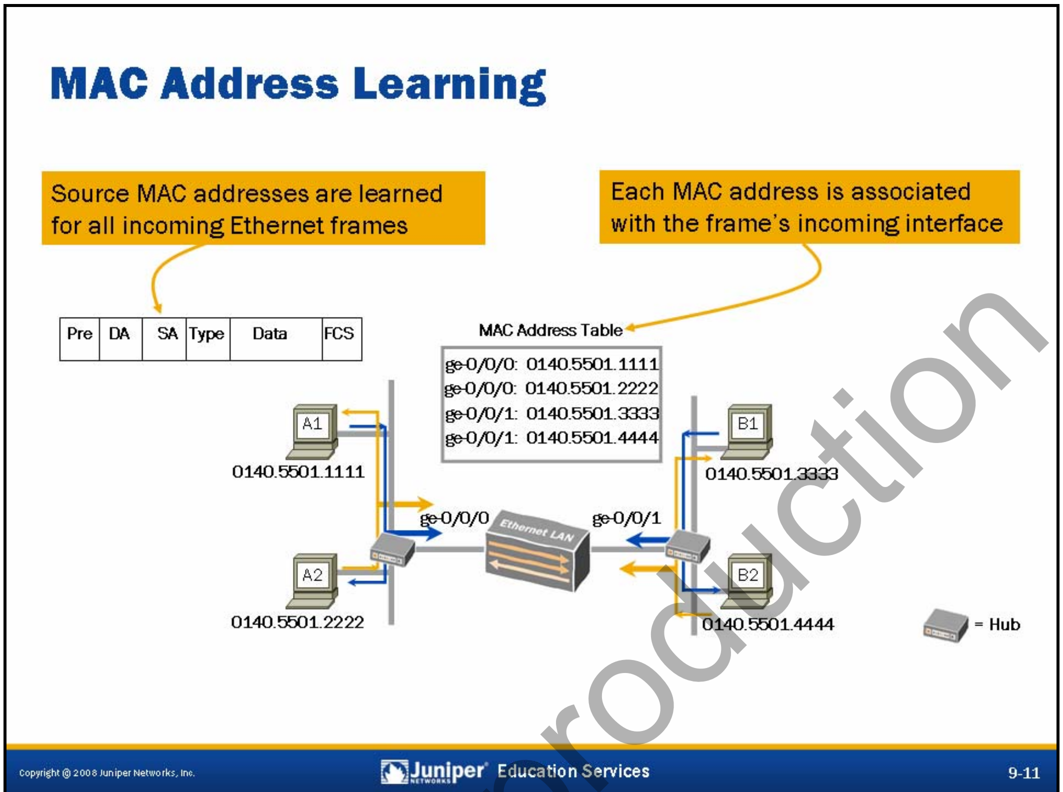
Bridging Mechanics (contd.)

Flooding is a transparent mechanism used to deliver packets to unknown MAC addresses. If the bridging table has no entry for a particular destination MAC address or if the packet received is a broadcast or multicast packet, the switch floods the traffic out all interfaces except the interface on which it was received. (If traffic originates on the switch, the switch floods that traffic out all interfaces.) When an unknown destination responds to traffic that has been flooded through a switch, the switch learns the MAC address of that node and updates its bridge table with the source MAC address and ingress port.

The *filtering* mechanism is used to limit broadcast traffic to its associated network or VLAN. As the number of entries in the bridge table grows, the switch pieces together an increasingly complete picture of the individual networks—the picture clarifies which nodes belong to which network. The switch uses this information to filter traffic. Filtering prevents the switch from forwarding traffic from one network to another network.

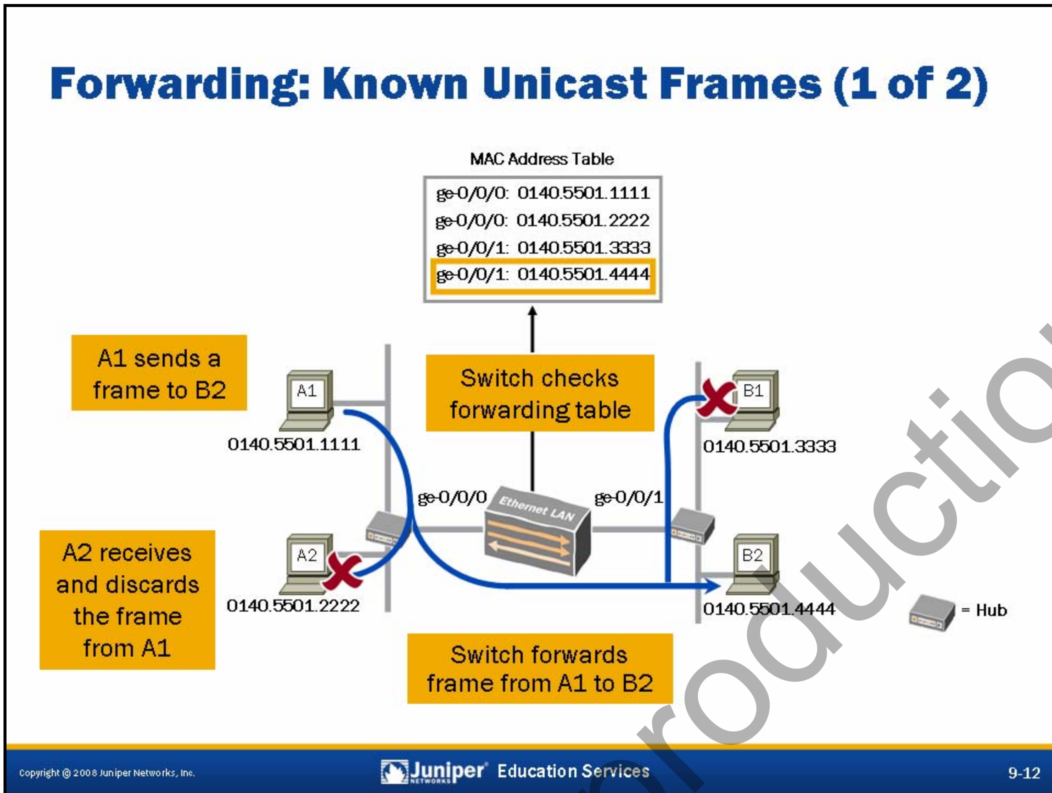
Finally, the switch uses *aging* to ensure that only active MAC address entries are in the bridge table. For each MAC address in the bridge table, the switch records a timestamp of when the information about the network node was learned. Each time the switch detects traffic from a MAC address, it updates the timestamp. A timer on the switch periodically checks the timestamp; if the timestamp is older than a user-configured value, the switch removes the node's MAC address from the bridge table. The default aging timer interval is 300 seconds and is configured on a per-VLAN basis as shown here:

```
[edit]
user@switch# set vlans vlan-name mac-table-aging-time ?
Possible completions:
  <mac-table-aging-time>  MAC aging time (60..1000000 seconds)
```



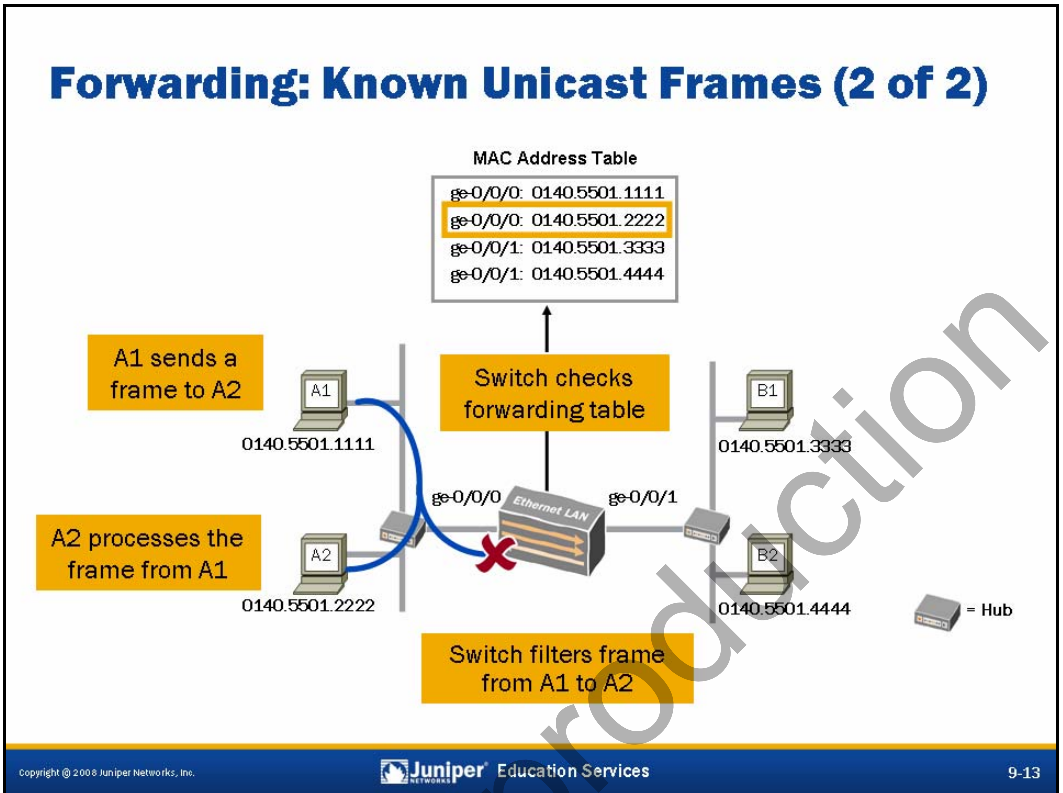
MAC Address Learning

The slide illustrates a basic view of the MAC address learning process. In this example, each switch port is connected to a hub, whereas the individual hubs have multiple nodes connected. As each node sends traffic toward the other nodes on the bridged LAN, the switch reviews that traffic and creates a MAC address table (a bridge table) based on the source address of the sender along with the switch port on which the traffic was received. In this example, we see that the MAC addresses for A1 and A2 are associated with port ge-0/0/0, whereas the MAC addresses for B1 and B2 are associated with port ge-0/0/1.



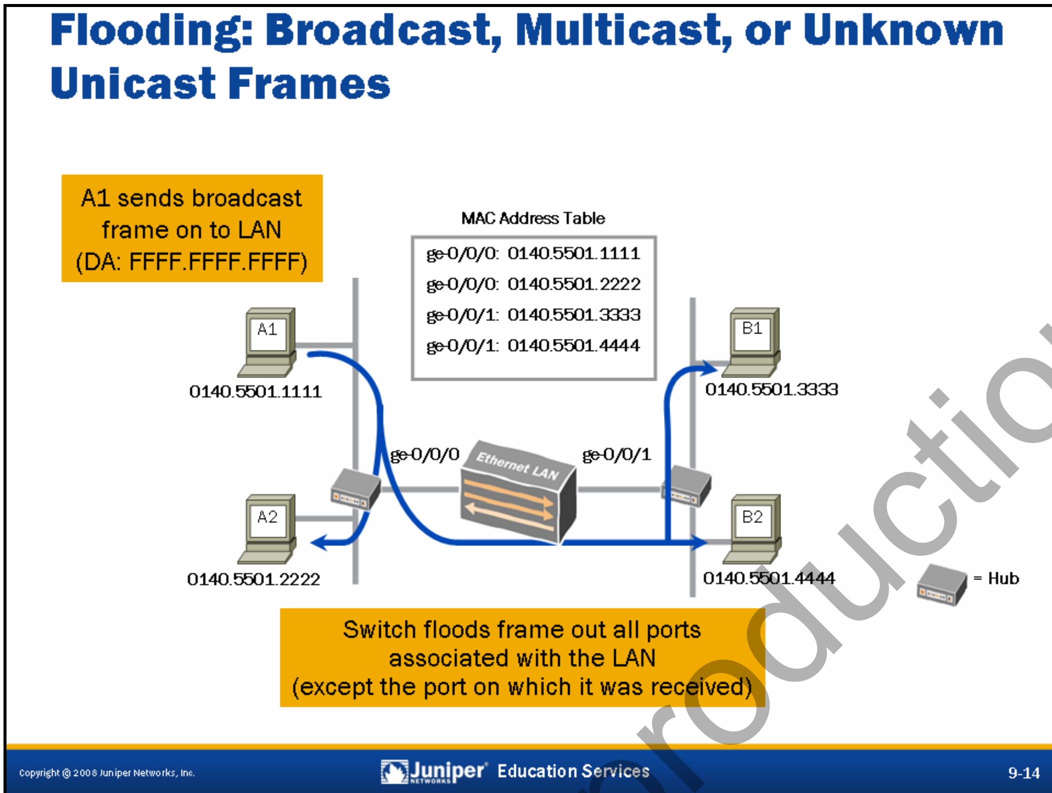
Forwarding: Known Unicast Frames: Part 1

In the example on the slide, A1 sends a frame to B2. The frame is repeated out all ports on the attached hub, which results in frames being sent to both A2 as well as the switch shown in the middle of the illustration. A2 receives the frame and detects that the destination MAC address does not match its own MAC address, at which time A2 discards the frame. The switch receives the frame, checks the MAC address table for a matching entry, and forwards the frame out the associated port based on the lookup results. Ultimately, B2 receives and processes the frame.



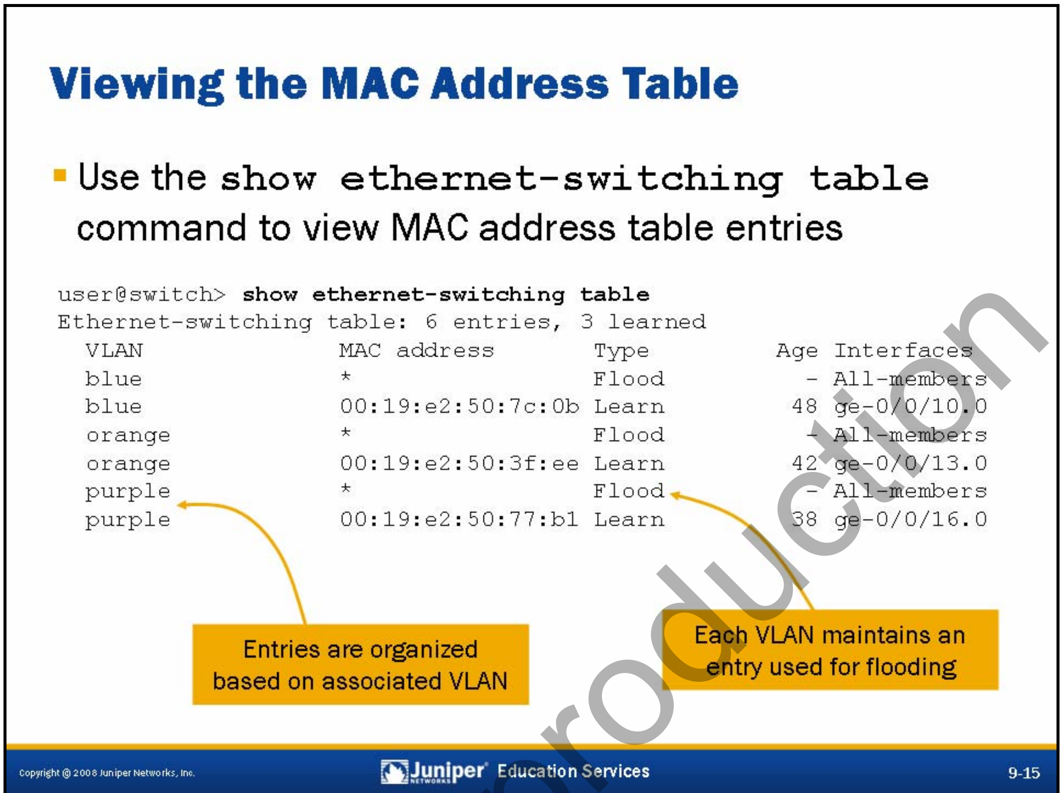
Forwarding: Known Unicast Frames: Part 2

In this example, A1 sends a frame to A2. The frame is received by the attached hub and sent out all ports, which results in duplicate frames sent to A2 as well as the switch. A2 receives the frame and detects that the destination MAC address matches its own MAC address, at which time A2 processes the frame. The switch receives the frame and checks the MAC address table for a matching entry. The entry in the MAC address table shows the egress port, which, in this example, is the same port on which the frame was received. Because the egress port in the MAC address table is the same port where the frame was received, the switch filters the frame.



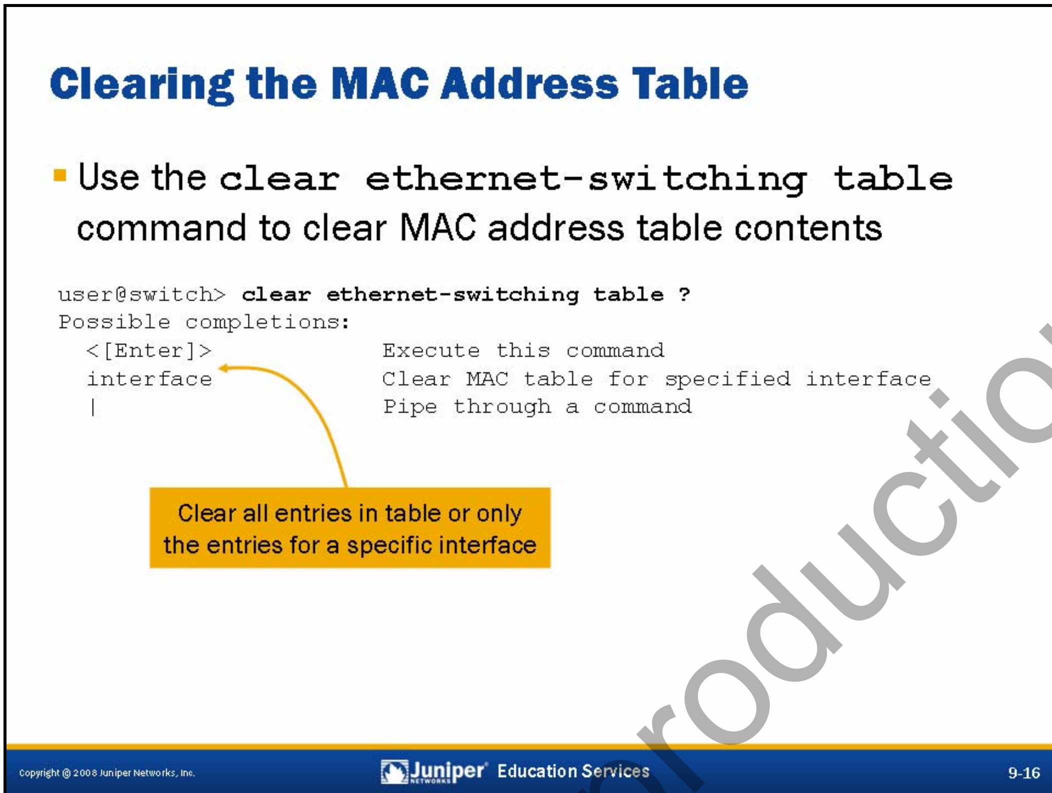
Flooding Frames

Flooding is often used to learn a MAC address not recorded in the bridge table. This mechanism is also used when sending broadcast and, in many cases, multicast frames. The example on the slide shows A1 sending a broadcast frame with a destination MAC address of FFFF.FFFF.FFFF to the LAN. The attached hub sends the frame out all ports. The switch floods the broadcast frame out all ports associated with the LAN, except for the port on which the frame was received. The result shows that the frame is ultimately received by all nodes on the LAN.



Viewing the MAC Address Table

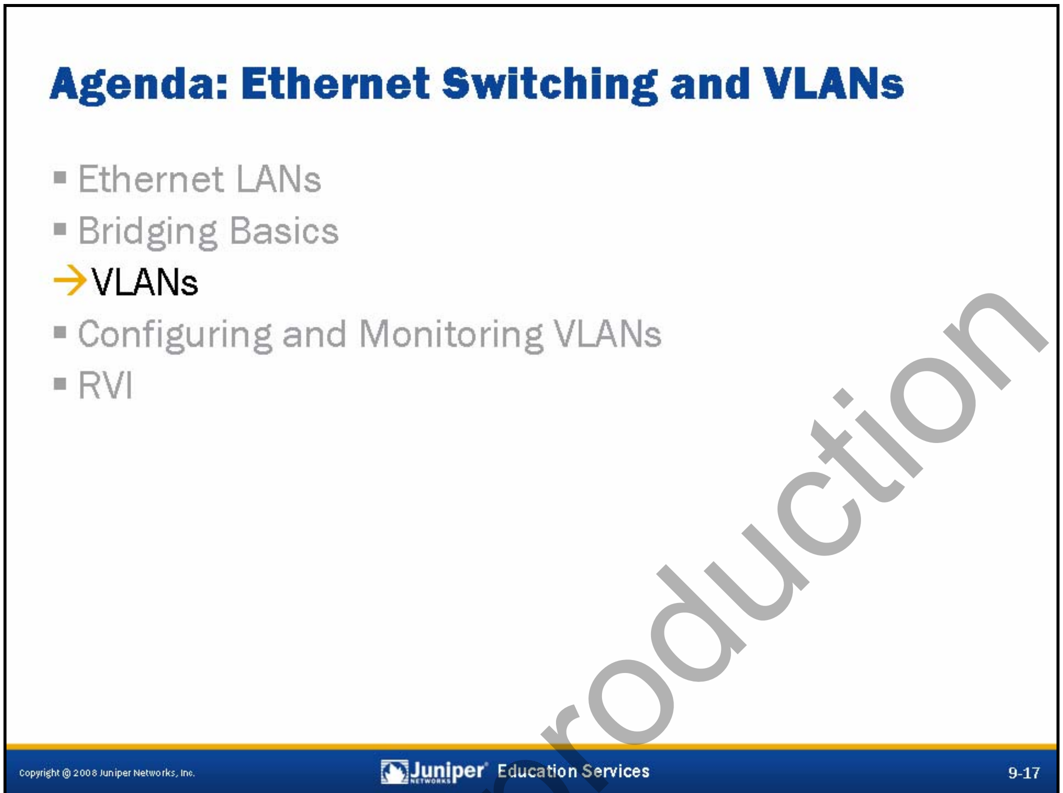
Use the **`show ethernet-switching table`** command to view all entries within the MAC address table. This command generates a list of learned MAC addresses along with the corresponding VLAN, age, and interface. All entries are organized based on their associated VLAN. The sample output on the slide also highlights each VLAN's flood entry, which is associated with all interfaces for the VLAN. This entry is used to flood traffic, destined to an unknown destination, through all interfaces that belong to the same VLAN on which the traffic was received.



Clearing MAC Address Table Entries

Use the **clear ethernet-switching table** command to clear all entries within the MAC address table. Optionally, you can use the **interface** statement to clear only those MAC table entries learned through the specified interface. The following example highlights the use of the **interface** option:

```
user@switch> show ethernet-switching table
Ethernet-switching table: 6 entries, 3 learned
  VLAN              MAC address       Type         Age Interfaces
  blue              *                 Flood          - All-members
  blue              00:19:e2:50:7c:0b Learn         23 ge-0/0/10.0
  orange            *                 Flood          - All-members
  orange            00:19:e2:50:3f:ee Learn          0 ge-0/0/13.0
  purple            *                 Flood          - All-members
  purple            00:19:e2:50:77:b1 Learn         24 ge-0/0/16.0

user@switch> clear ethernet-switching table interface ge-0/0/10.0

user@switch> show ethernet-switching table
Ethernet-switching table: 5 entries, 2 learned
  VLAN              MAC address       Type         Age Interfaces
  blue              *                 Flood          - All-members
  orange            *                 Flood          - All-members
  orange            00:19:e2:50:3f:ee Learn         26 ge-0/0/13.0
  purple            *                 Flood          - All-members
  purple            00:19:e2:50:77:b1 Learn         32 ge-0/0/16.0
```

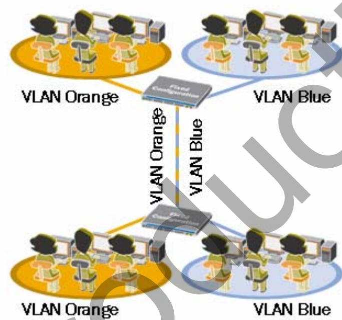


VLANs

The slide highlights the topic we discuss next.

Overview of VLANs

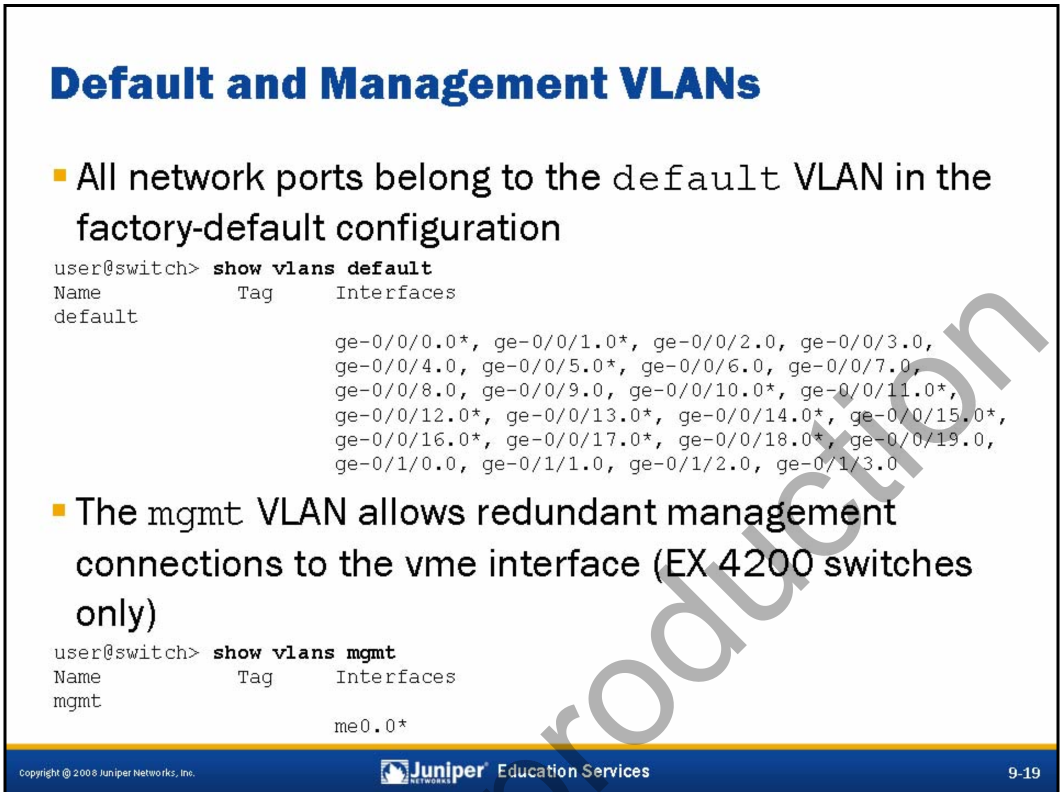
- VLANs:
  - Segment a single broadcast domain into multiple broadcast domains
  - Allow for grouping users based on business needs, regardless of physical location



 Juniper Education Services 9-18

VLANs Defined

A virtual LAN (VLAN) is a collection of network nodes that are logically grouped together to form separate broadcast domains. A VLAN has the same general attributes as a physical LAN, but it allows all nodes for a particular VLAN to be grouped together, regardless of physical location. One advantage of using VLANs is design flexibility. VLANs allow individual users to be grouped based on business needs. Connectivity within a VLAN is established and maintained through software configuration, which makes VLANs such a dynamic and flexible option in today's networking environments.



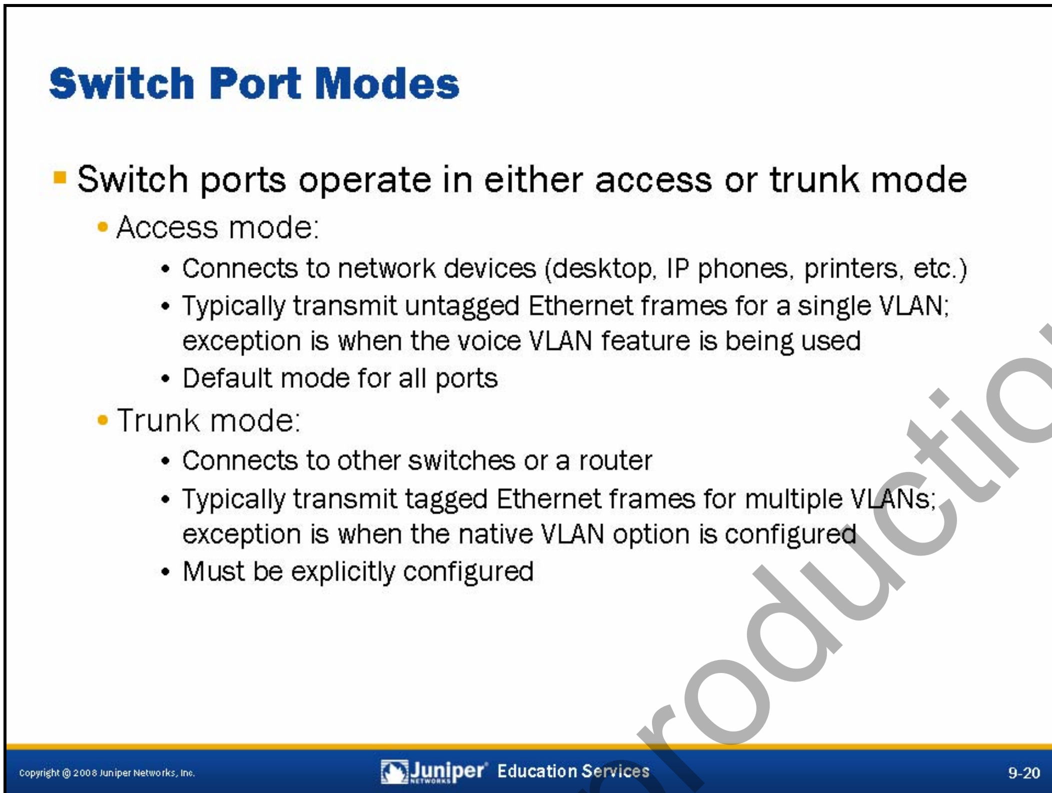
Default VLAN

The factory-default configuration associates all network ports, including installed uplink ports, with the `default` VLAN. The sample capture on the slide illustrates the expected output for the **`show vlans default`** command. In this sample output we can see that the `default` VLAN does not use an 802.1Q tag. An 802.1Q tag can be assigned to the `default` VLAN through user configuration.

Management VLAN

The `mgmt` VLAN is an untagged VLAN and is typically used to facilitate redundant management connections to a Virtual Chassis system's vme interface. We discussed the vme interface in a previous chapter. The `mgmt` VLAN is used only with the EX 4200 switches. If the me0 interface is not configured as a Layer 3 interface, the me0 interface will be associated with this management VLAN, as shown on the slide. If the me0 interface is not configured for Layer 3 operations, it is not displayed in the generated output. A sample capture of this scenario is shown here:

```
user@switch> show vlans mgmt
Name             Tag      Interfaces
mgmt
                          None
```



Switch Port Modes

Switch ports operate in either access mode or trunk mode.

An access port connects to network devices such as desktop computers, IP phones, printers, or file servers. Access ports typically belong to a single VLAN and transmit and receive untagged Ethernet frames. The exception is when the voice VLAN feature is used. If the voice VLAN feature is enabled on an access port, that port might participate in multiple VLANs and can pass both untagged (data) and tagged (voice) traffic. We cover the voice VLAN feature in more detail in a subsequent chapter. All ports default to access mode in the factory-default configuration.

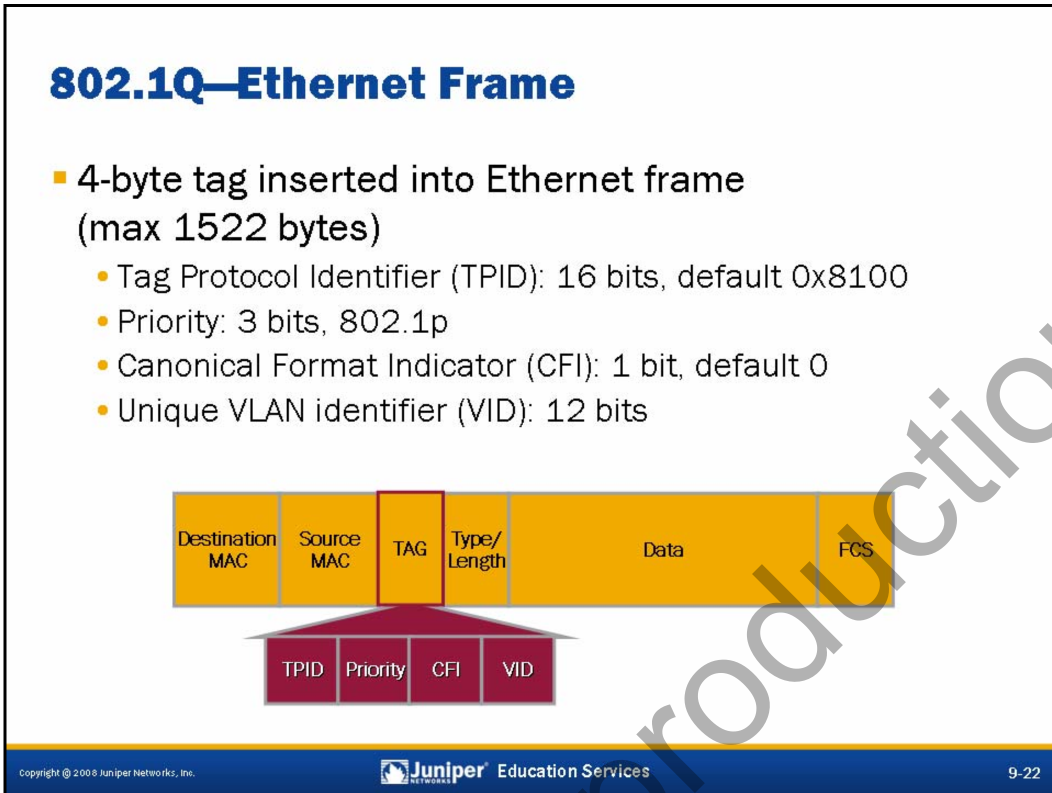
Continued on next page.

Switch Port Modes (contd.)

A trunk port typically connects to another switch or to a customer edge router. Interfaces configured for trunk mode handle traffic for multiple VLANs, multiplexing the traffic for all configured VLANs over the same physical connection, and separating the traffic by tagging it with the appropriate VLAN ID. Trunk ports can also carry untagged traffic when configured with the **`native-vlan-id`** statement. The following sample configuration illustrates the use of the **`native-vlan-id`** configuration option. In this example, the ge-0/0/15 interface is configured as a trunk port and will carry tagged traffic for the `v100` and `v200` VLANs as well as untagged traffic for the `default` VLAN:

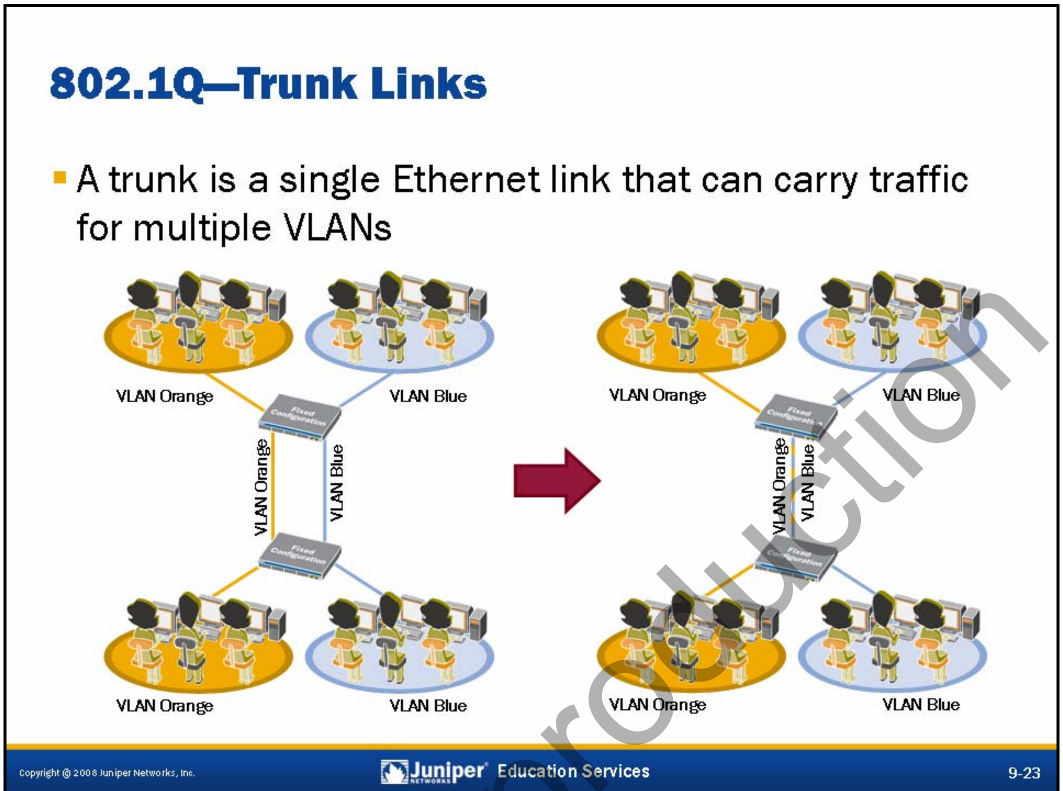
```
[edit interfaces ge-0/0/15]
user@switch# show
unit 0 {
    family ethernet-switching {
        port-mode trunk;
        vlan {
            members [ v100 v200 ];
        }
        native-vlan-id default;
    }
}
```

The remote switch must also be configured to permit untagged traffic for the `default` VLAN.



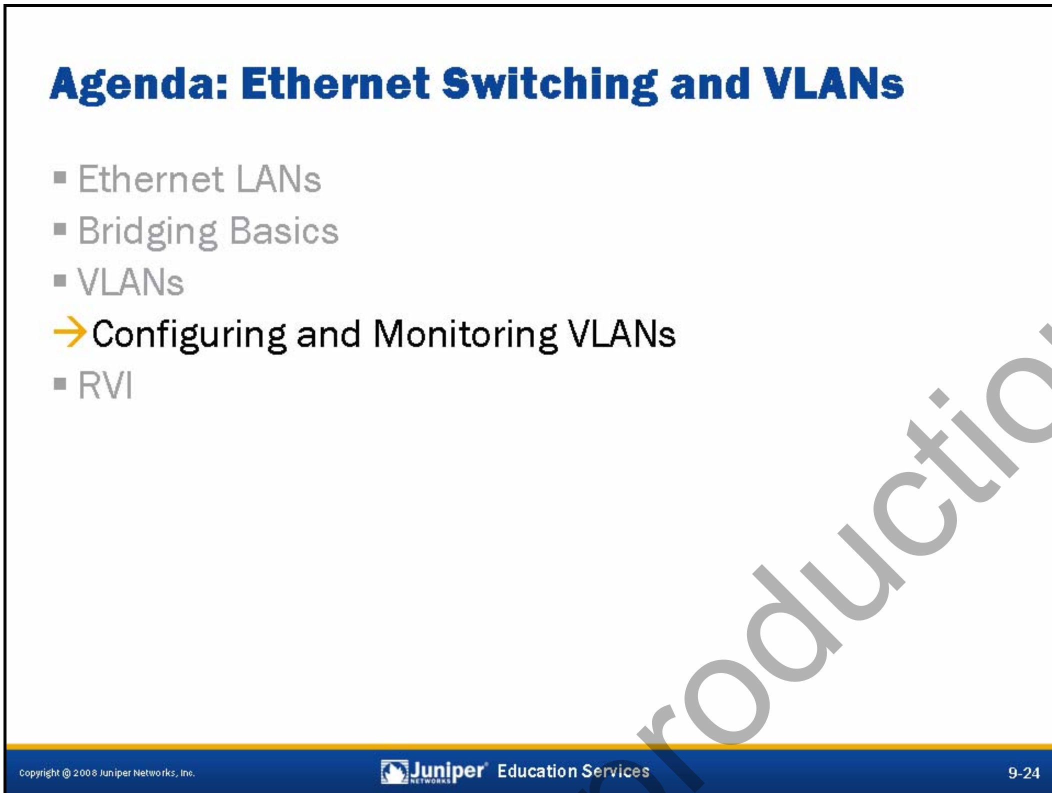
802.1Q—Ethernet Frame

To consistently associate traffic with a particular VLAN, the individual frames must be tagged as they pass throughout a network. The slide illustrates an 802.1Q-tagged Ethernet frame along with the key components of the tag.



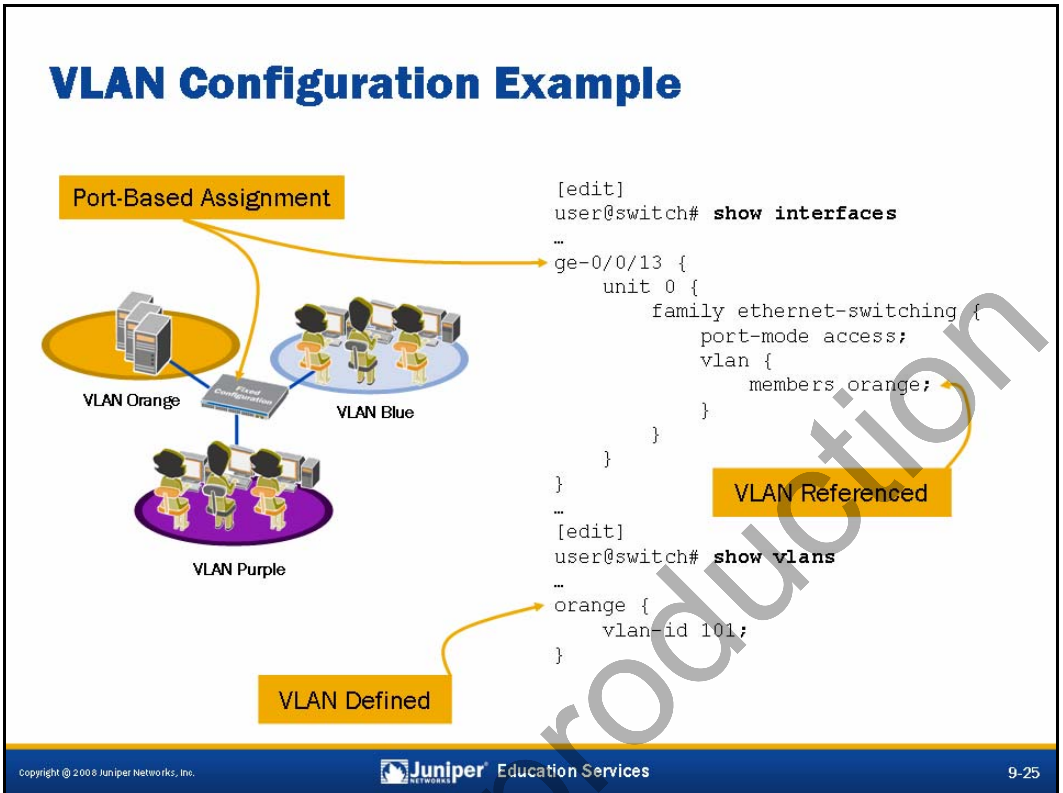
802.1Q Trunk Links

A trunk is a single Ethernet link used to carry traffic for multiple VLANs. A trunk link typically interconnects multiple switches or a switch with a customer edge router. As shown on the slide, interfaces configured as trunk ports handle traffic for multiple VLANs, multiplexing the traffic for all configured VLANs over a single physical connection rather than using separate physical links for each configured VLAN.



Configuring and Monitoring VLANs

The slide highlights the topic we discuss next.



VLAN Configuration Example

The slide provides a basic VLAN configuration example on an access port. The VLAN *orange* is assigned a `vlan-id` of *101* and is defined under the `[edit vlans]` hierarchy. This VLAN is then referenced within the interface's configuration. This example demonstrates port-based assignment.

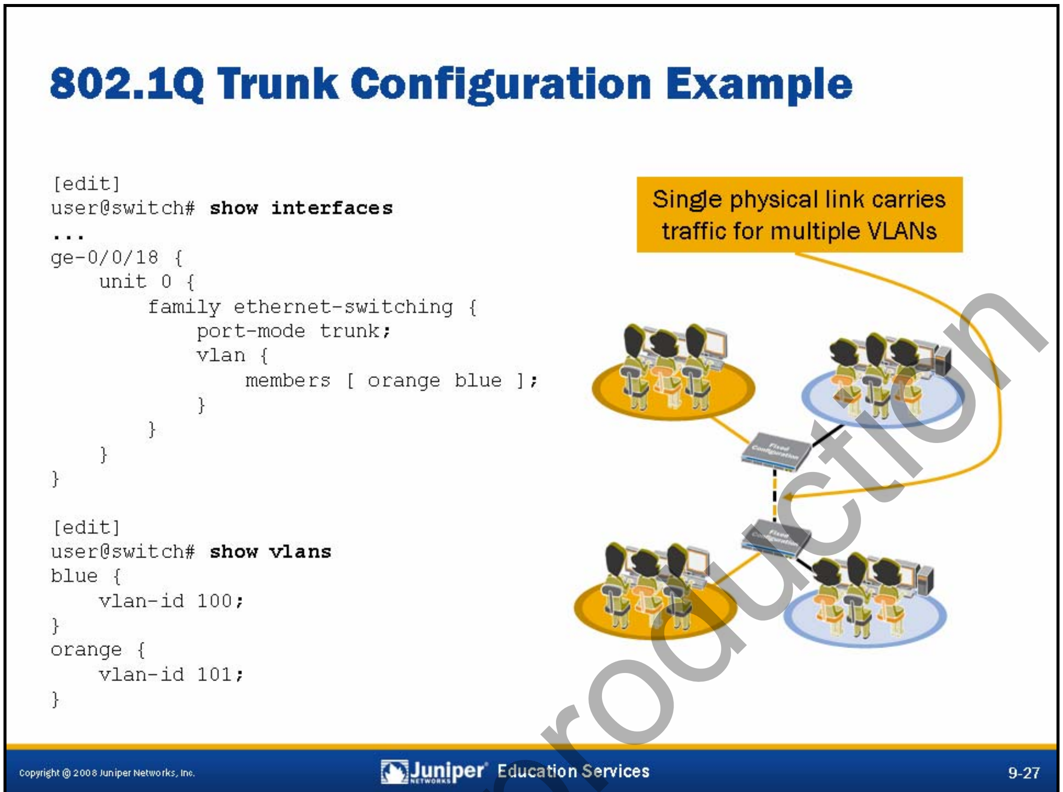
Monitoring VLAN Assignments

```
user@switch> show vlans
Name           Tag     Interfaces
blue           100
                       ge-0/0/10.0*
default
                       ge-0/0/0.0, ge-0/0/5.0*
orange         101
                       ge-0/0/13.0*
purple         102
                       ge-0/0/16.0*
mgmt
                       me0.0*

user@switch> show vlans orange detail
VLAN: orange, 802.1Q Tag: 101, Admin state: Enabled
Number of interfaces: 1 (Active = 1)
  Untagged interfaces: ge-0/0/13.0*
```

 Juniper Education Services 9-26

Monitoring VLAN Assignments

The slide shows some key commands used to monitor VLAN assignments. In this example the ge-0/0/13 belongs to the VLAN named `orange`, which has an 802.1Q tag of 101. Because this interface is configured as an access port it will receive and transmit untagged frames only. If a trunk port were also configured to pass traffic for the `orange` VLAN, it would add and remove the 802.1Q tag value of 101 for all traffic for the `orange` VLAN. We look at a trunk port configuration and monitoring example next.

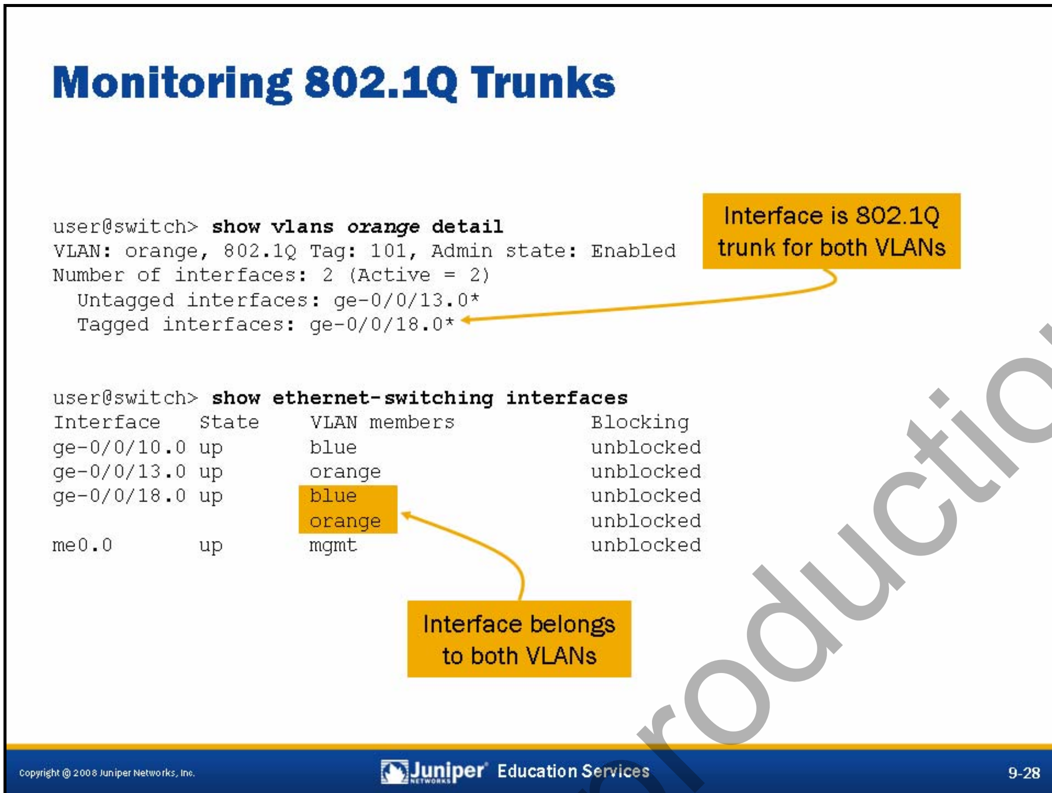


802.1Q Trunk Configuration Example

The slide illustrates an 802.1Q configuration example. In this case, the interface is configured as a trunk port and is associated with the `orange` and `blue` VLANs. The partnering switch would have a similar configuration for the interface functioning as a trunk.

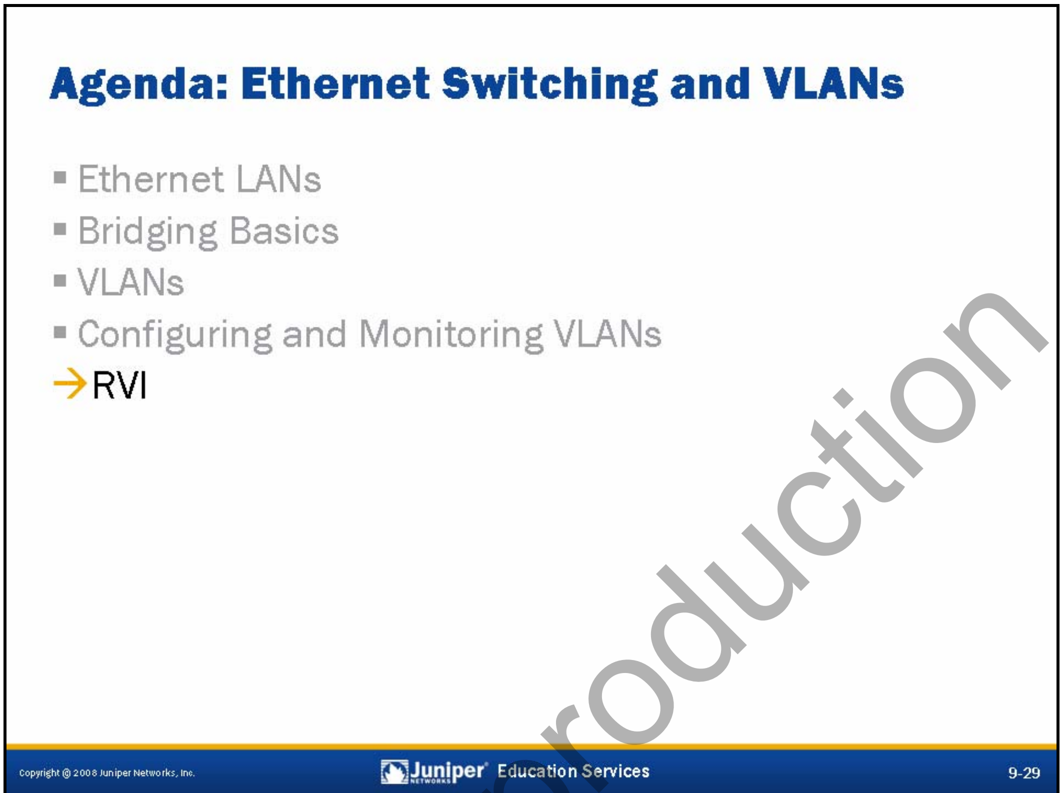
Optionally, you can use the keyword **`all`** to associate all configured VLANs with a given trunk port. The following example accomplishes the same goal as the configuration shown on the slide:

```
[edit interfaces ge-0/0/18]
user@switch# show
unit 0 {
    family ethernet-switching {
        port-mode trunk;
        vlan {
            members all;
        }
    }
}
```



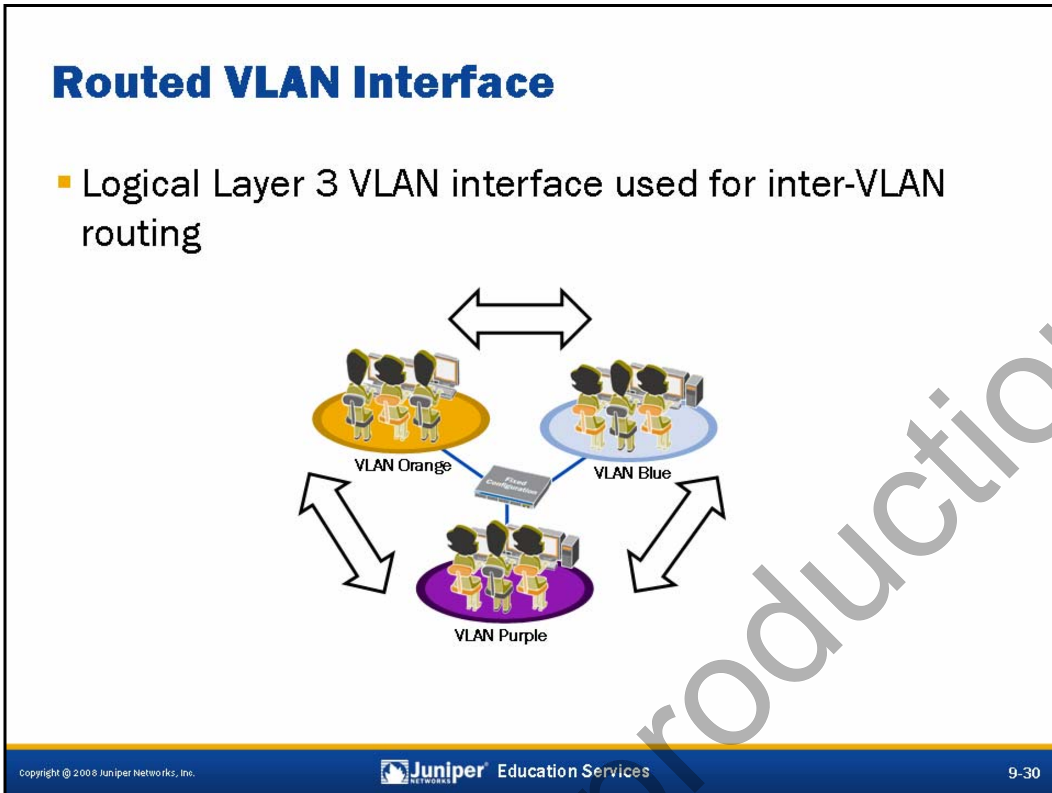
Monitoring 802.1Q Trunks

The slide shows some key commands when monitoring 802.1Q trunks. In this example all traffic that is sent from or received through the ge-0/0/18 interface will have an 802.1Q tag for the *blue* or *orange* VLAN.



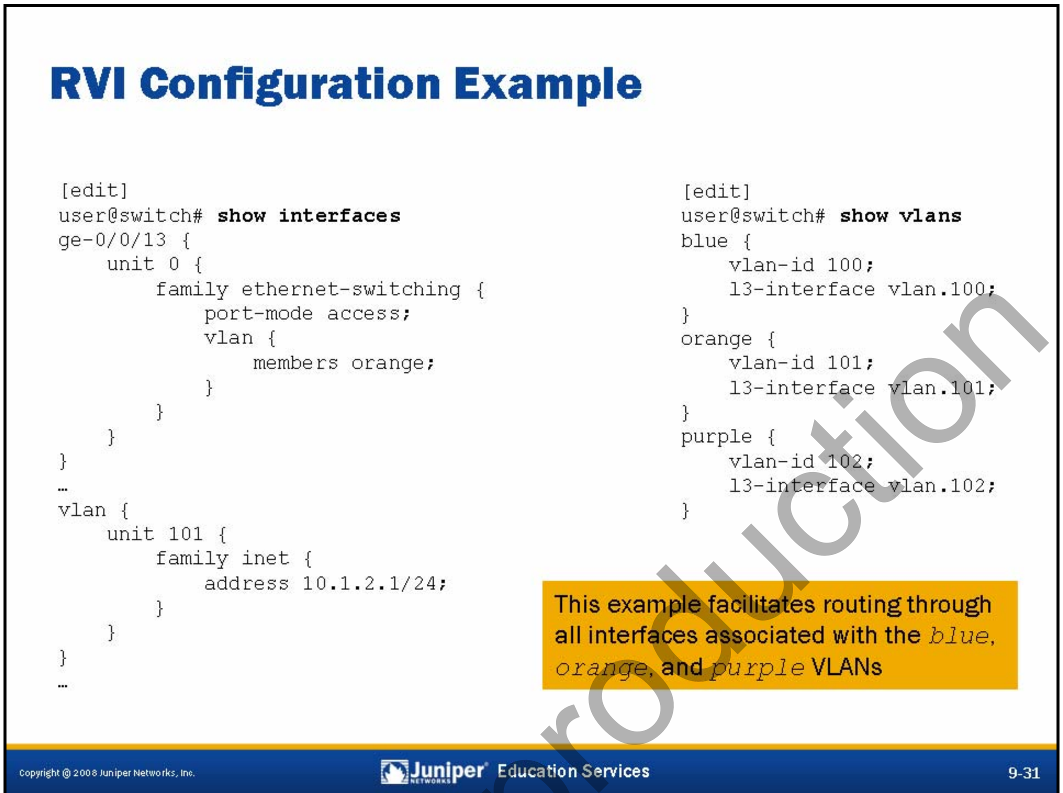
Routed VLAN Interface

The slide highlights the topic we discuss next.



Routed VLAN Interface

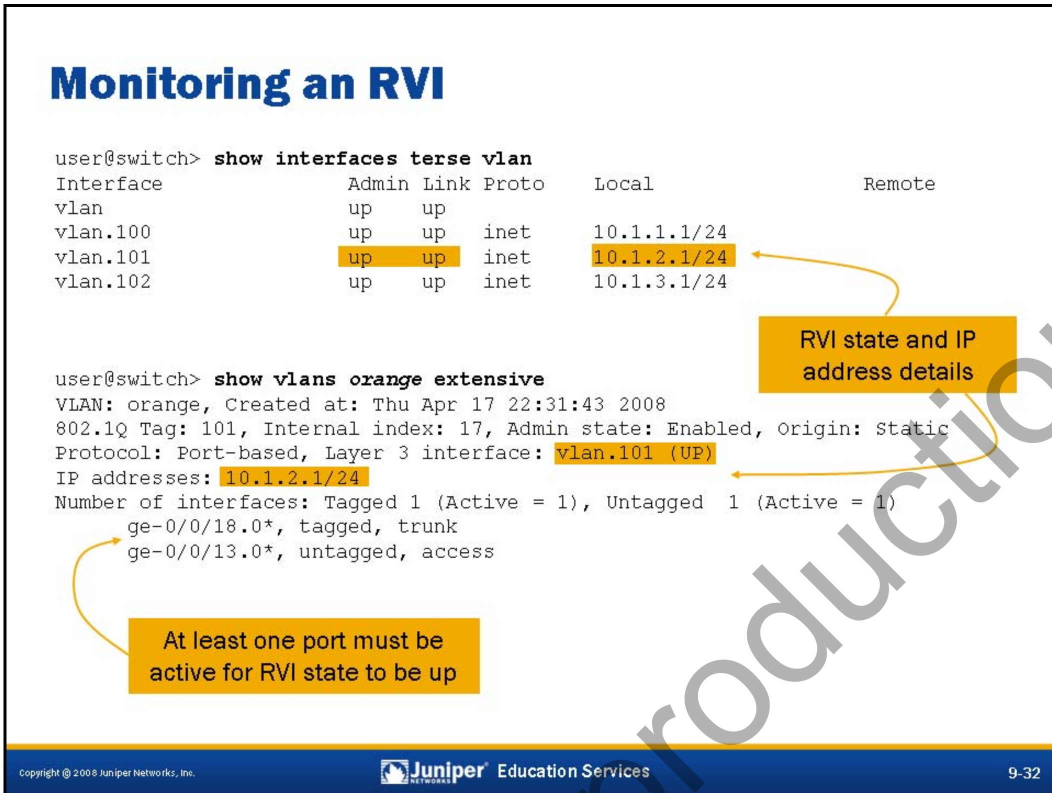
A routed VLAN interface (RVI) is a logical Layer 3 VLAN interface used to route traffic between VLANs. The following slides provide a configuration and monitoring example for an RVI.



RVI Configuration Example

The slide provides a configuration example for an RVI. In this example, the switch will perform a Layer 3 lookup when it receives traffic with a destination MAC address that matches its own MAC address. For the switch to perform this routing operation, the attached devices must have a configured gateway address that matches the IP address associated with the corresponding logical VLAN interface. Because the Layer 3 VLAN interface is a logical interface, it uses the switch's system MAC address, which is shown here:

```
user@switch> show interfaces vlan extensive | match hardware
  Current address: 00:19:e2:50:82:e0, Hardware address: 00:19:e2:50:82:e0

user@switch> show chassis mac-addresses
    FPC 0   MAC address information:
      Public base address      00:19:e2:50:82:e0
      Public count             32
```



Monitoring an RVI

The slide provides some key commands used to monitor an RVI. The slide shows the output from the **show interfaces terse vlan** and **show vlans *vlan_name* extensive** commands. Both of the referenced commands show the state and IP address information for an RVI. As indicated on the slide, at least one active port must be associated with the VLAN for the RVI to be administratively up. The following sample captures show the expected output when no active port is associated with the VLAN:

```
user@switch> show interfaces terse vlan
Interface               Admin Link Proto    Local                 Remote
vlan                    up    up
...
vlan.101                down  up   inet     10.1.2.1/24
...

user@switch> show vlans orange extensive
VLAN: orange, Created at: Thu Apr 17 22:31:43 2008
802.1Q Tag: 101, Internal index: 17, Admin state: Enabled, Origin: Static
Protocol: Port-based, Layer 3 interface: vlan.101 (DOWN)
IP addresses: 10.1.2.1/24
Number of interfaces: Tagged 0 (Active = 0), Untagged  1 (Active = 0)
      ge-0/0/7.0, untagged, access
```

Summary

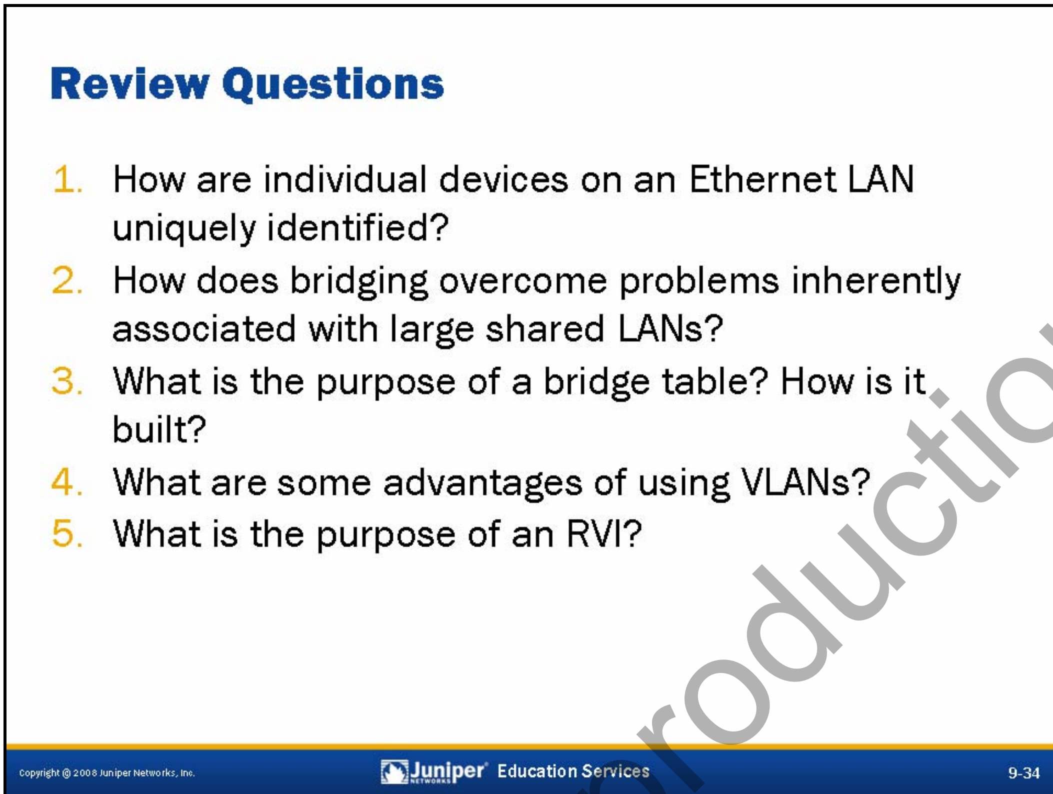
- In this chapter, we:
  - Described the basic operation of an Ethernet LAN
  - Explained the operation and benefits of bridging
  - Described the purpose and listed the benefits of VLANs
  - Configured and verified proper operation of VLANs
  - Described the purpose of an RVI
  - Configured and verified proper operation of an RVI

Juniper Education Services

9-33

This Chapter Discussed:

- Basic operation of an Ethernet LAN;
- Operation and benefits of bridging;
- The purpose and benefits of VLANs;
- Configuring and monitoring VLANs;
- The purpose and benefits of a RVI; and
- Configuring and monitoring an RVI.



Review Questions

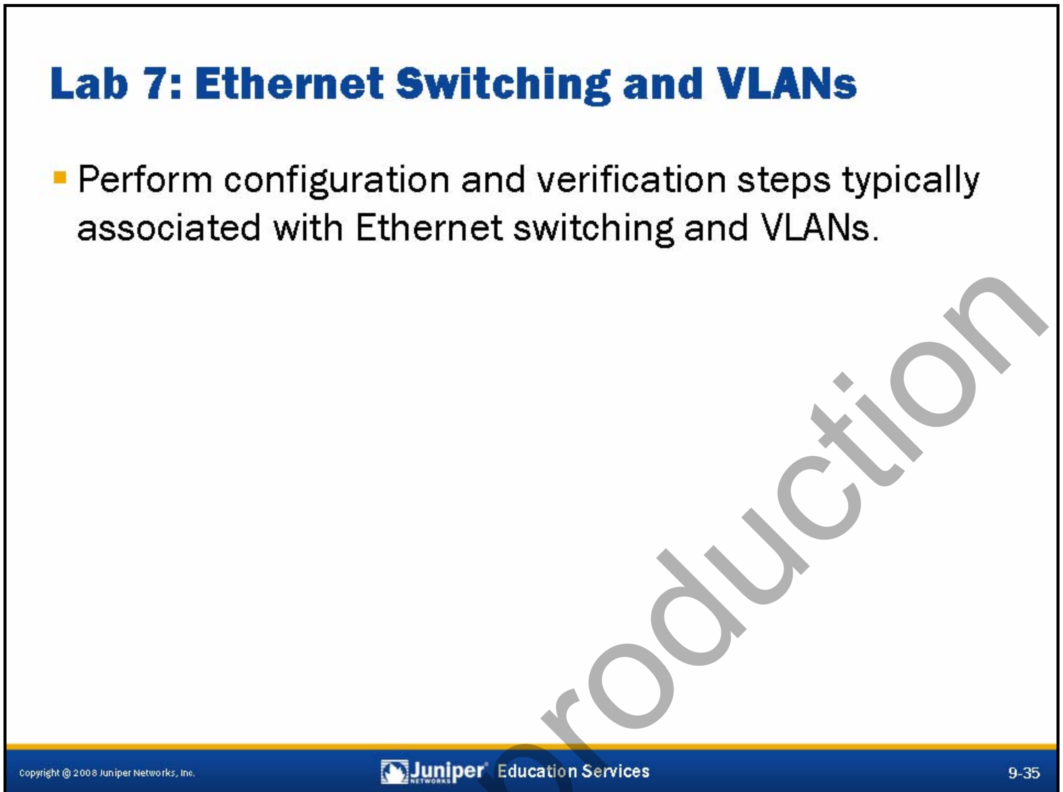
1.

2.

3.

4.

5.

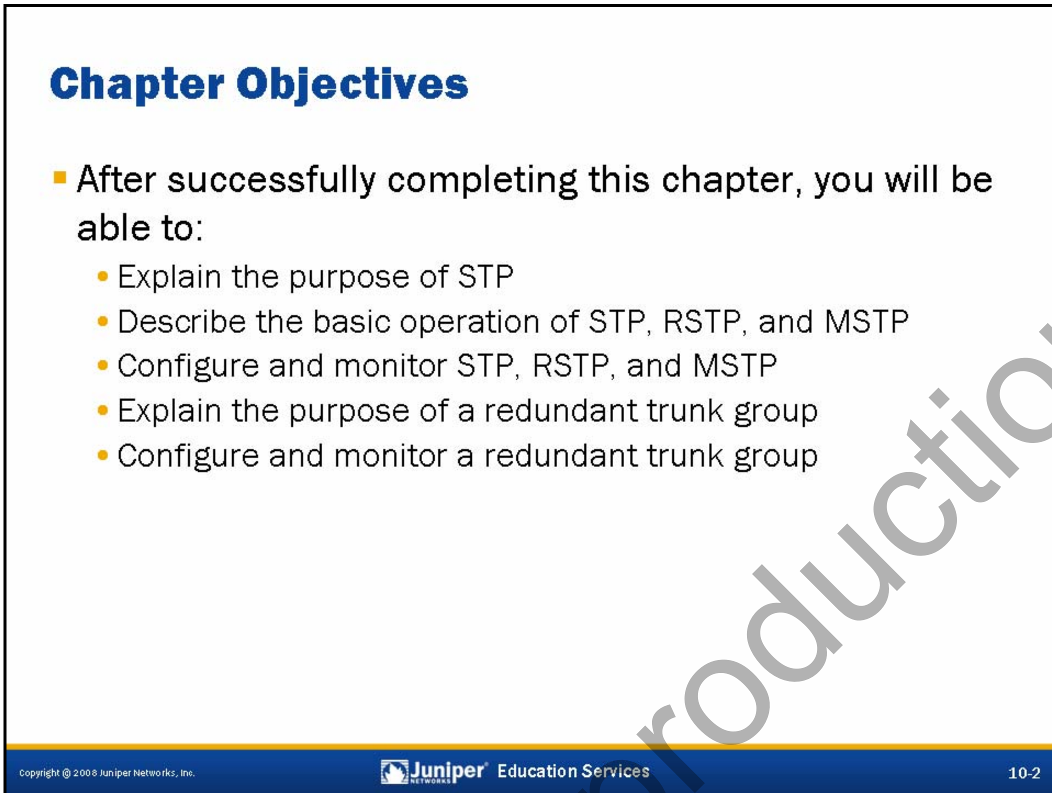


Lab 7: Ethernet Switching and VLANs

The slide provides the objective for this lab.

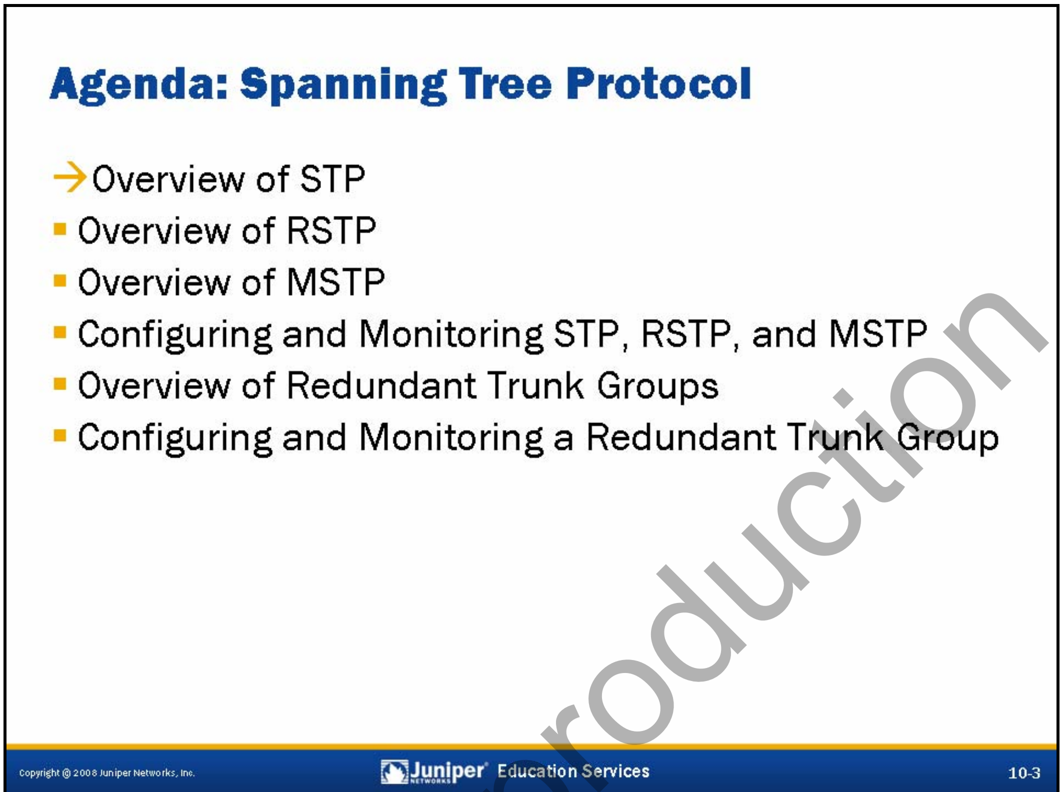
Operating Juniper Networks Switches in the Enterprise

Chapter 10: Spanning Tree Protocol



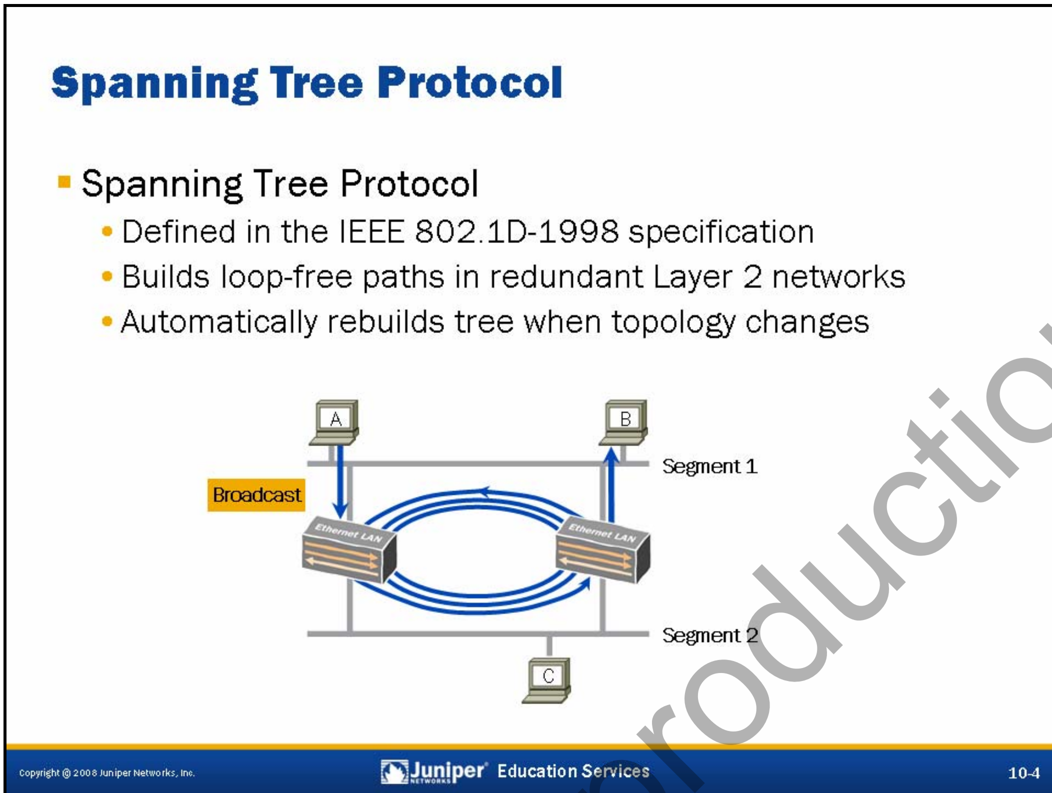
This Chapter Discusses:

- The purpose of the Spanning Tree Protocol (STP);
- The basic operation of STP, the Rapid Spanning Tree Protocol (RSTP), and the Multiple Spanning Tree Protocol (MSTP);
- Configuring and monitoring STP, RSTP, and MSTP;
- The purpose of a redundant trunk group; and
- Configuring and monitoring a redundant trunk group.



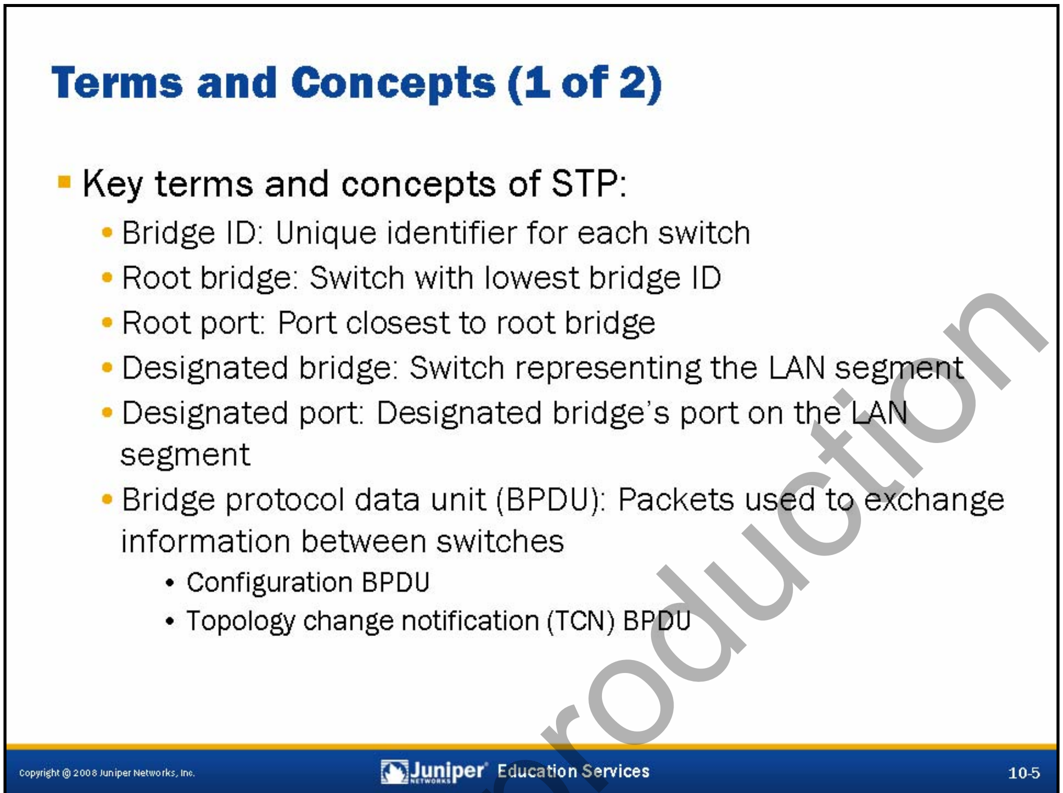
Overview of the Spanning Tree Protocol

The slide lists the topics we cover in this chapter. We discuss the highlighted topic first.



Spanning Tree Protocol

The Spanning Tree Protocol (STP) is defined in the Institute of Electrical and Electronics Engineers (IEEE) 802.1D 1998 specification. STP is a simple Layer 2 protocol that prevents loops and calculates the best path through a switched network that contains redundant paths. STP is required only when redundant paths exist within a Layer 2 network. STP automatically rebuilds the tree when a topology change occurs.



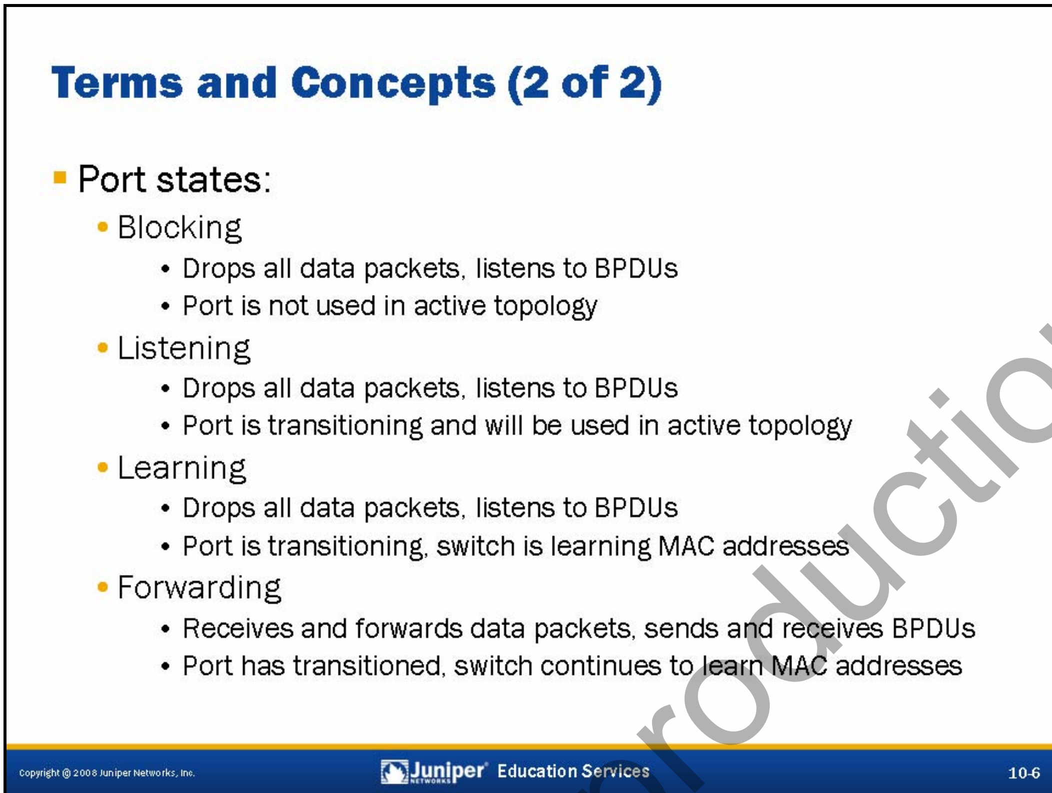
STP Terms and Concepts: Part 1

All switches participating in STP have a unique bridge ID. The bridge ID is a combination of the system MAC address and a configurable priority value. The lowest bridge ID determines the *root bridge*.

Once the root bridge is determined, each nonroot switch determines the least-cost path from itself to the root bridge. The port associated with the least-cost path becomes the *root port* for the switch.

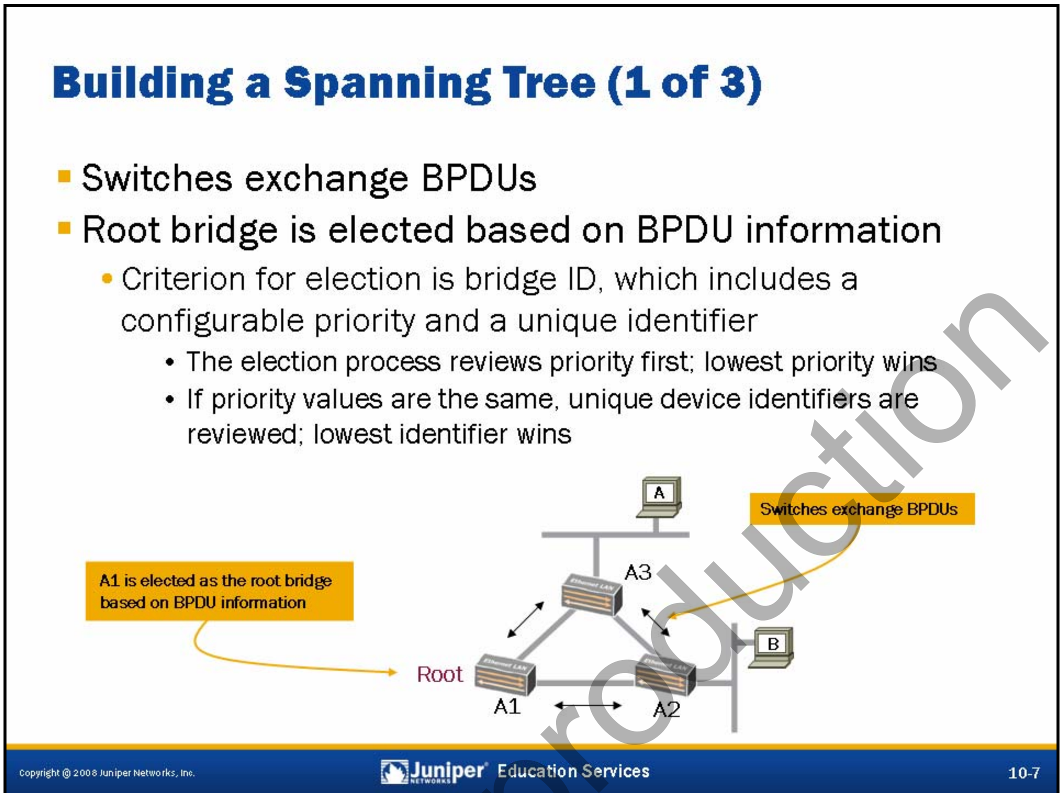
All switches participating on a common network segment must determine which switch offers the least-cost path from the network segment to the root bridge. The switch with the best path becomes the *designated bridge* for the LAN segment, and the port connecting this switch to the network segment becomes the *designated port* for the LAN segment. If equal-cost paths to the root bridge exist from a given LAN segment, the bridge ID is used as a tiebreaker. If the bridge ID is used to help determine the designated bridge, the lowest bridge ID is selected. The designated port transmits bridge protocol data units (BPDUs) on the segment.

STP uses BPDU packets to exchange information between switches. There are two types of BPDUs: configuration BPDUs and topology change notification (TCN) BPDUs. Configuration BPDUs determine the tree topology of a LAN. STP uses the information provided by the BPDUs to elect a root bridge, identify root ports for each switch, identify designated ports for each physical LAN segment, and prune specific redundant links to create a loop-free tree topology. Topology change notification BPDUs are used to report and acknowledge topology changes within a switched network.



STP Terms and Concepts: Part 2

The slide highlights the STP port states along with a brief description of each state. In addition to the states listed on the slide, an interface can also be administratively disabled. A port that is administratively disabled does not participate in the spanning tree but does flood any BPDUs it receives to other ports associated with the same VLAN. Administratively disabled ports continue to perform basic bridging operations and will forward data traffic based on the MAC address table.



Exchange of BPDUs

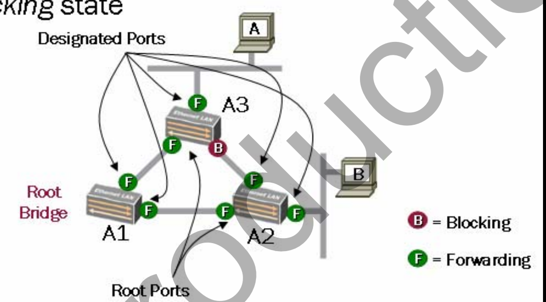
All switches participating in a switched network exchange BPDUs with one another. It is through the exchanged BPDUs that neighboring switches become familiar with one another and learn the information needed to select a root bridge.

Root Bridge Election

STP elects the root bridge device based on the bridge ID, which actually consists of two distinct elements: a configurable priority value and a unique device identifier, which is the system MAC address. The priority values are reviewed first and determine the root bridge. If the priority value of one device is lower than the priority value of all other devices, that device is elected as the root bridge. If the priority values are equal for all devices, STP evaluates the system MAC addresses, and the device with the lowest MAC address is elected as the root bridge.

Building a Spanning Tree (2 of 3)

- Port role is determined by the least-cost path calculation to the root bridge; port state is determined by the port role
  - Ports on root bridge assume designated port role and *forwarding* state
  - Root ports on nonroot switches are placed in *forwarding* state
  - Designated ports on designated bridges are placed in *forwarding* state
  - All other ports are placed in *blocking* state



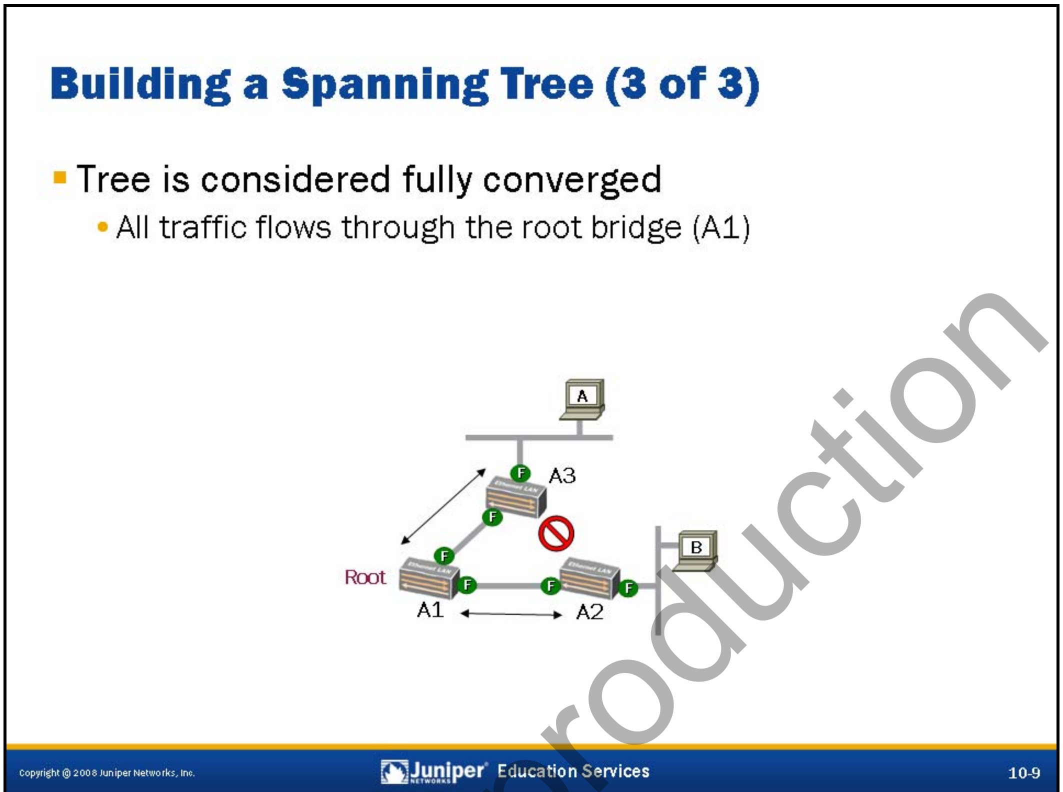
Juniper Education Services
10-8

Port Role and State Determination

Once the root bridge is elected, all nonroot devices perform a least-cost path calculation to the root bridge. The results of these calculations determine the role of the switch ports. The role of the individual switch ports determines the port state.

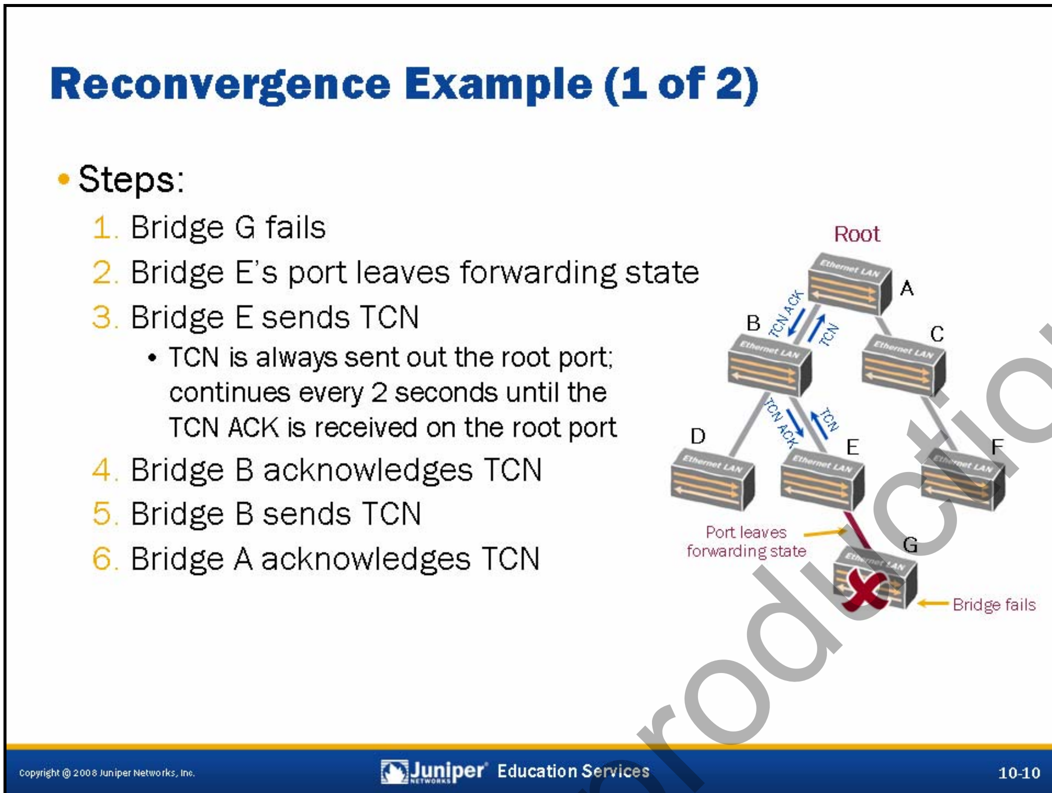
All switch ports belonging to the root bridge assume the designated port role and forwarding state. Each nonroot switch determines a root port, which is the port closest to the root bridge, based on its least-cost path calculation to the root bridge. Each interface has an associated cost that is based on the configured speed. An interface operating at 10 Mbps assumes a cost of 2000000, an interface operating at 100 Mbps assumes a cost of 200000, and an interface operating at 1 Gbps assumes a cost of 20000. If a switch has two equal-cost paths to the root bridge, the switch port with the lowest port number is selected as the root port. The root port for each nonroot switch is placed in the forwarding state.

STP selects a designated bridge on each LAN segment. This selection process is also based on the least-cost path calculation from each switch to the root bridge. Once the designated bridge is selected, its port, which connects to the LAN segment, is chosen as the designated port. If the designated bridge has multiple ports connected to the LAN segment, the lower-numbered port participating on that LAN segment is selected as the designated port. All designated ports assume the forwarding state. All ports not selected as a root port or as a designated port assume the blocking state.



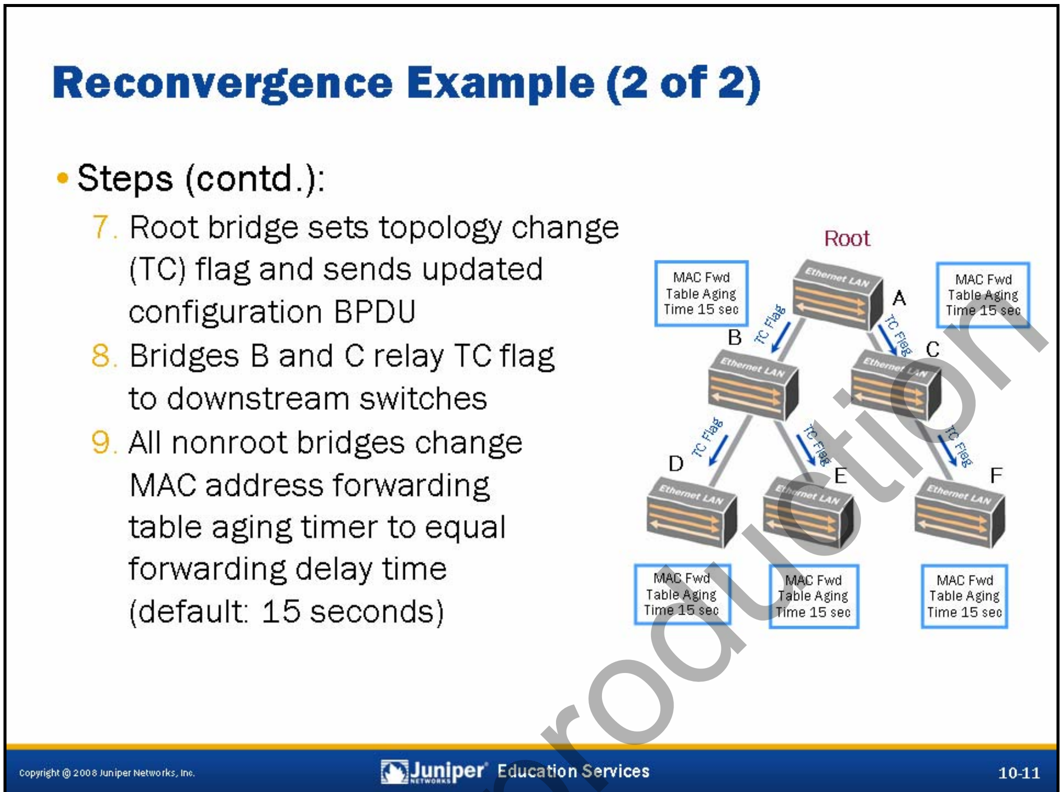
Tree Is Converged

Once the role and state for all switch ports is determined, the tree is considered fully converged. The convergence delay can take up to 50 seconds when the default forwarding delay (15 seconds) and max age timer (20 seconds) values are in effect. The formula used to calculate the convergence delay for STP is 2 x the forwarding delay + the maximum age. In the example shown on the slide, all traffic passing between Host A and Host B transit the root bridge (Switch A1).



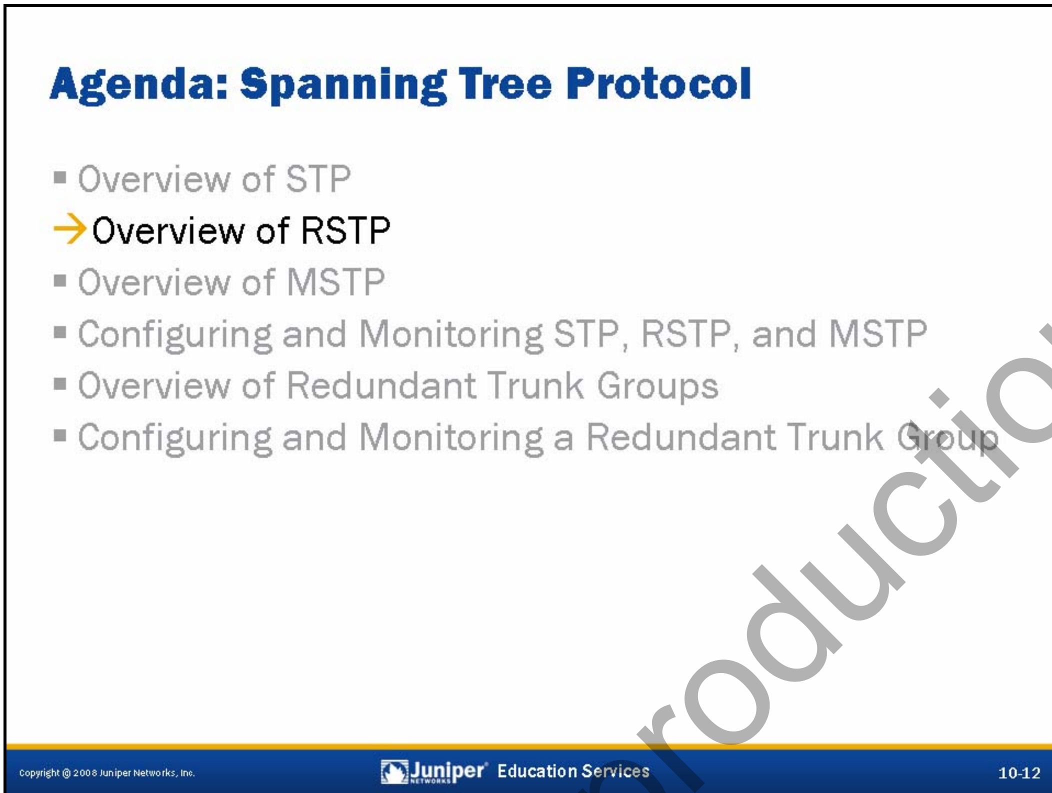
Reconvergence Example: Part 1

This slide shows the first several steps that occur during a failure and reconvergence scenario.



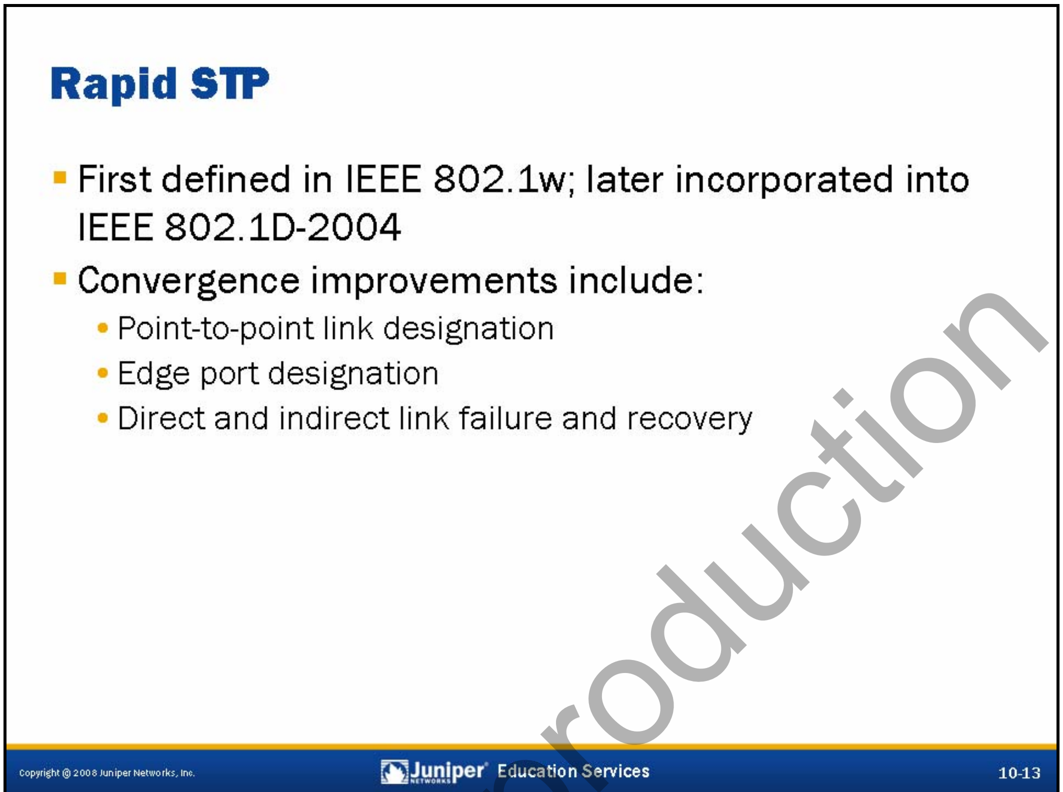
Reconvergence Example: Part 2

This slide shows the remainder of the steps involved in a failure and reconvergence scenario. Once the nonroot bridges change their MAC address forwarding table aging timer to the shortened interval and wait that period of time (15 seconds by default), they then delete all entries from the MAC table that were not refreshed within that time frame. All deleted entries must then be learned once again through the normal learning process.



Overview of the Rapid Spanning Tree Protocol

The slide highlights the topic we discuss next.



RSTP Defined

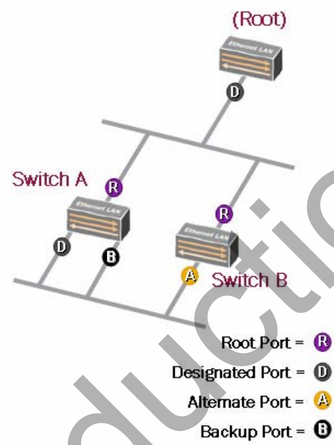
The Rapid Spanning Tree Protocol (RSTP) was originally defined in the IEEE 802.1w draft and was later incorporated into the IEEE 802.1D-2004 specification. RSTP introduces a number of improvements to STP while performing the same basic function.

RSTP Convergence Improvements

RSTP provides better reconvergence time than the original STP. RSTP identifies certain links as point to point. When a point-to-point link fails, the alternate link can transition to the forwarding state without waiting for any protocol timers to expire. RSTP provides fast network convergence when a topology change occurs. RSTP greatly decreases the state transition time compared to STP. To aid in the improved convergence, RSTP uses additional features and functionality, such as edge port definitions and rapid direct and indirect link failure detection and recovery. We examine these features in more detail in subsequent pages of this chapter.

RSTP Port Roles

- RSTP introduces new port roles:
  - Alternate port:
    - Provides alternate path to root bridge (nondesignated switches)
    - Blocks traffic while receiving superior BPDUs from neighboring switch
  - Backup port:
    - Provides redundant path to a segment (designated switches only)
    - Blocks traffic while a more preferred port functions as designated port
- RSTP continues to use root and designated port roles



 Juniper Education Services 10-14

RSTP Introduces New Port Roles

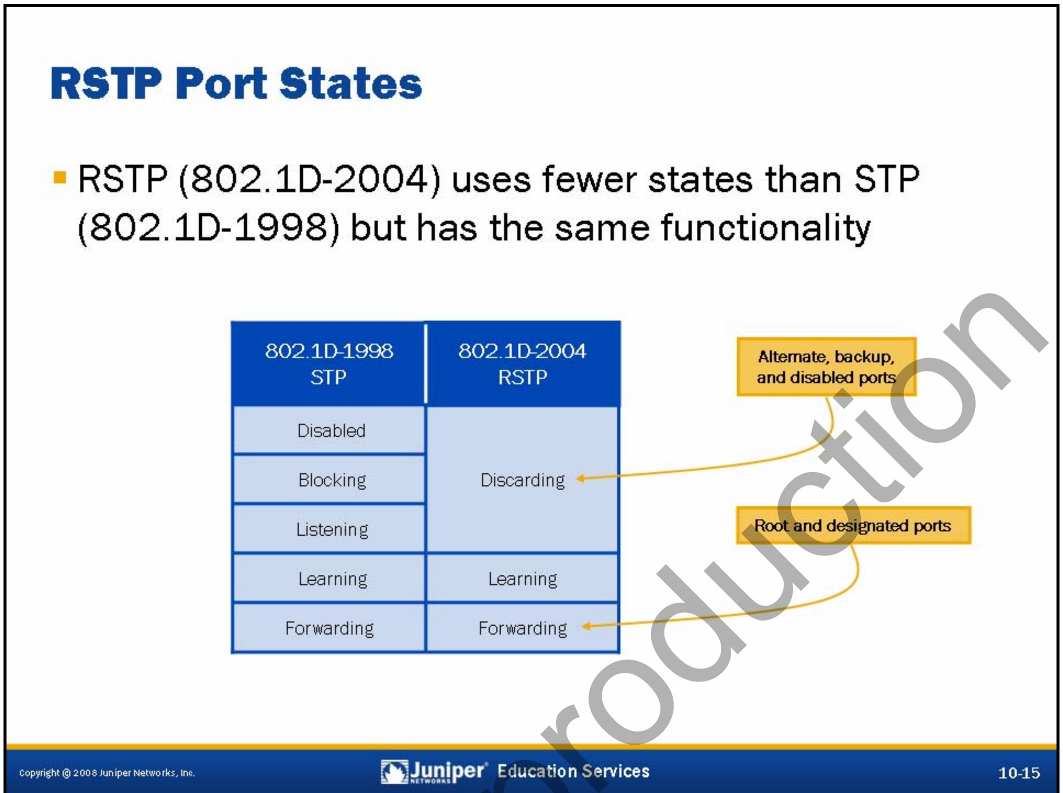
RSTP introduces the alternate and backup port roles. An alternate port is a switch port that has an alternate—generally higher-cost—path to the root bridge. In the event that the root port fails, the alternate port assumes the role of the root port and is placed in the forwarding state. Alternate ports are placed in the discarding state but receive superior BPDUs from neighboring switches. Alternate ports are found on switches participating in a shared LAN segment for which they are not functioning as the designated bridge.

When a designated bridge has multiple ports connected to a shared LAN segment, one of those ports is selected as the designated port. The designated port is typically the port with the lower port number. RSTP considers all other ports on the designated switch that are connected to that shared LAN segment as backup ports. In the event that the designated port is unable to perform its role, one of the backup ports assumes the designated port role upon successful negotiation and is placed in the forwarding state.

Backup ports are placed in the discarding state. While in the discarding state, backup ports receive superior BPDUs from the designated port.

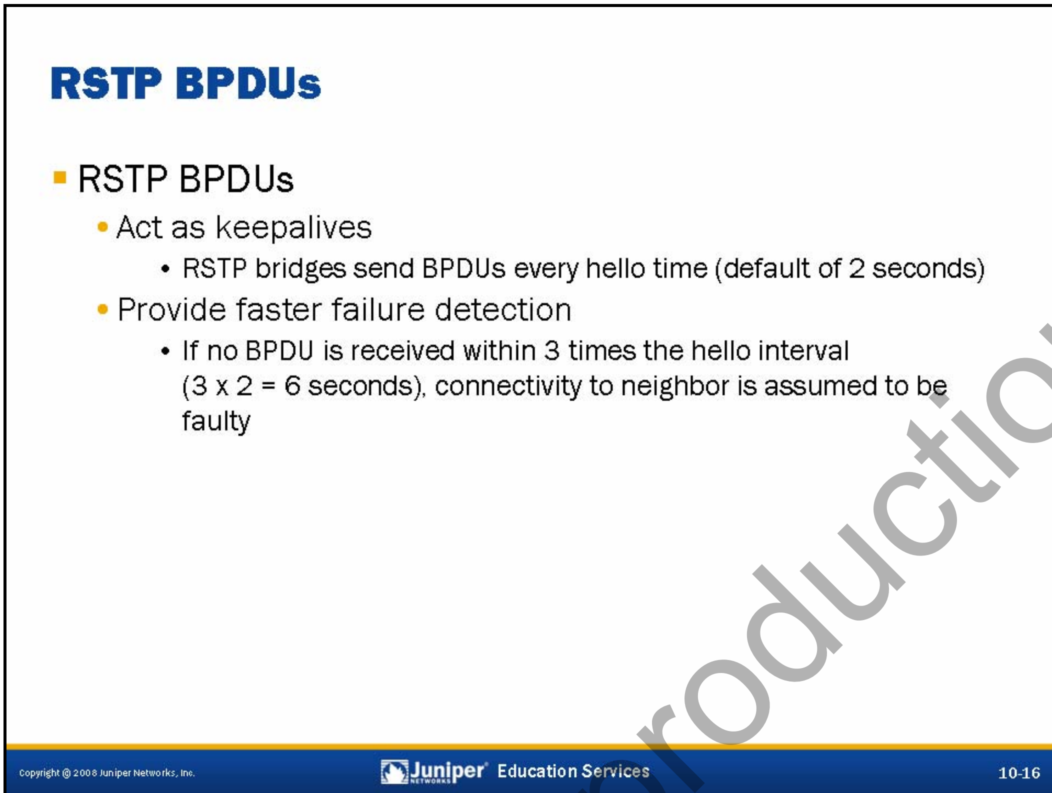
Continued Use of Root and Designated Ports

RSTP continues to use the root and designated port roles. Only ports selected for the root or designated port role participate in the active topology. We described the purpose of the root and designated ports previously in this chapter.



RSTP Port States

RSTP uses fewer port states than STP. The three possible port states found in RSTP are *discarding*, *learning*, and *forwarding*. Any port that is administratively disabled, excluded from the active topology through configuration, or dynamically excluded from forwarding and learning is placed in the discarding state. Ports that are actively learning but not currently forwarding are in the learning state, whereas ports that are both learning and forwarding frames simultaneously are in the forwarding state. As the slide indicates, only those ports selected as root and designated ports use the forwarding state.



RSTP BPDUs

As previously mentioned, STP uses BPDUs to elect a root bridge, identify root ports for each switch, identify designated ports for each physical LAN segment, prune specific redundant links to create a loop-free tree topology, and report and acknowledge topology changes. RSTP configuration BPDUs also function as keepalives. All RSTP bridges send configuration BPDUs every 2 seconds by default. You can alter this value, if needed.

By monitoring neighboring switches through the use of BPDUs, RSTP can detect failures of network components much more quickly than STP. If no BPDU is received within three times the hello interval, connectivity to the neighboring device is assumed to be faulty and the tree is updated. By default, failures are detected within 6 seconds when using RSTP, whereas it might take up to 50 seconds for STP.

On EX-series switches, network ports operating in full-duplex mode are considered point-to-point links. When a failure occurs, a switch port operating as a point-to-point link can become a new root port or designated port and transition to the forwarding state without waiting for the timer to expire. Switch ports operating in half-duplex mode are considered to be shared (or LAN) links and must wait for the timer to expire before transitioning to the forwarding state.

Transitioning to Forwarding State

- Original Spanning Tree Protocol (802.1D-1998)
  - Takes 30 seconds before ports start forwarding traffic after being enabled
    - 2x forwarding delay (listening + learning)
- Rapid Spanning Tree Protocol (802.1D-2004)
  - Uses proposal/agreement handshake on point-to-point links instead of timers
    - Root and edge ports transition to forwarding state immediately
    - Nonedge-designated ports transition to forwarding state once explicit agreement is received

 Juniper Networks Education Services 10-17

STP Forwarding State Transition

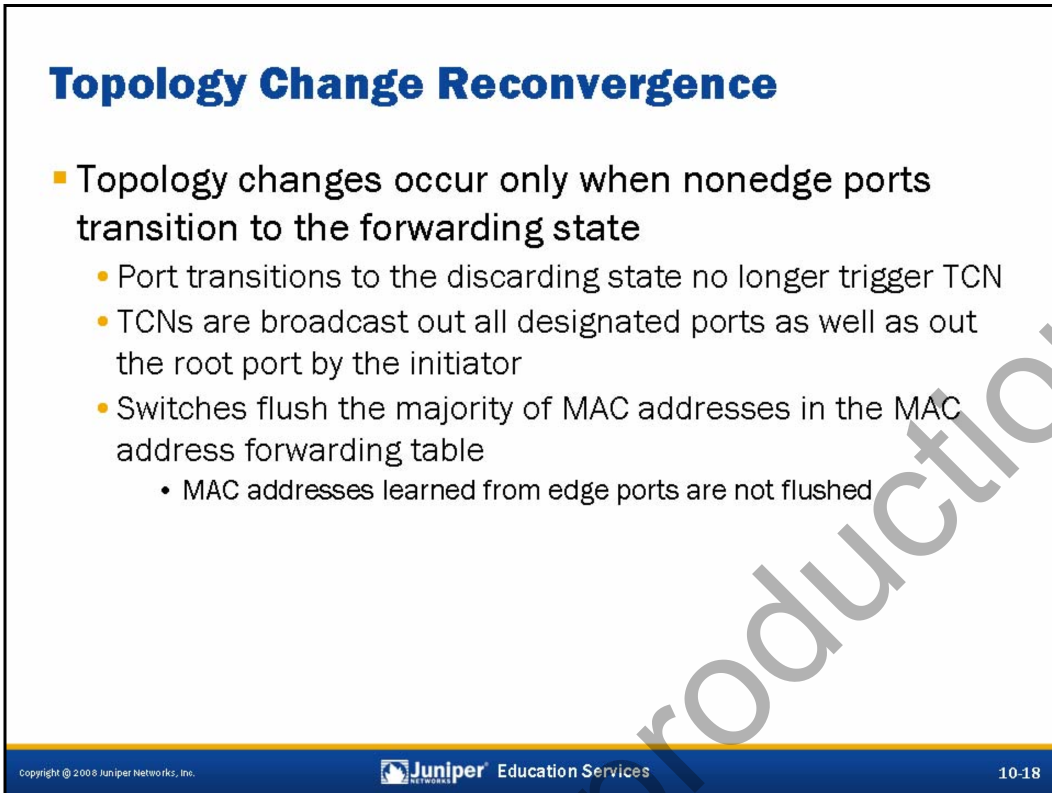
With the original STP, as defined in 802.1D-1998, a port can take more than 30 seconds before it forwards user traffic. As a port is enabled, it must transition through the listening and learning states before graduating to the forwarding state. STP allows two times the forwarding delay (15 seconds by default) for this transition to occur.

RSTP Forwarding State Transition

RSTP offers considerable improvements when transitioning to the forwarding state. RSTP converges faster because it uses a proposal/agreement handshake mechanism on point-to-point links instead of the timer-based process used by STP. On EX-series switches, network ports operating in full-duplex mode are considered point-to-point links, whereas network ports operating in half-duplex mode are considered to be shared (LAN) links.

Root ports and edge ports transition to the forwarding state immediately without exchanging messages with other switches. Edge ports are ports that have direct connections to end stations. Because these connections cannot create loops, they are placed in the forwarding state without any delay. If a switch port does not receive BPDUs from the connecting device, it automatically assumes the role of an edge port. When an EX-series switch receives configuration messages on a switch port that is configured to be an edge port, it immediately changes the port to a normal spanning tree port (nonedge port).

Nonedge-designated ports transition to the forwarding state only after an explicit agreement is received from the attached switch.

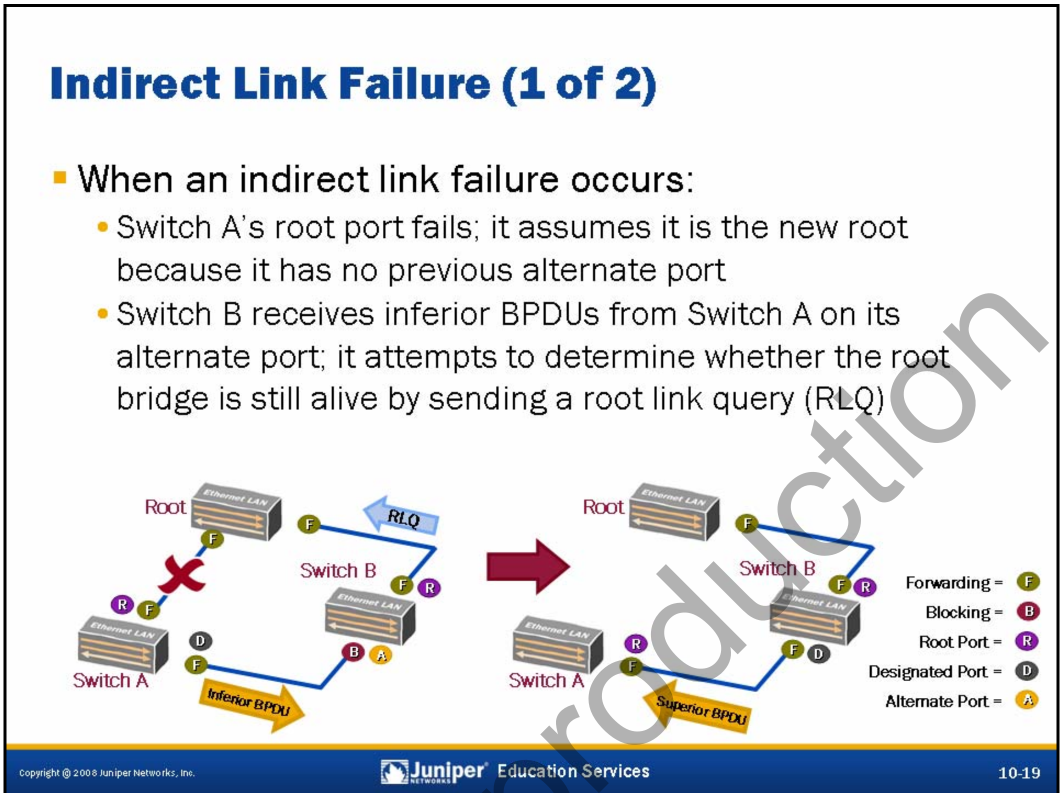


Topology Changes

When using STP, state transitions on any participating switch port cause a topology change to occur. RSTP reduces the number of topology changes and improves overall stability within the network by generating topology change notifications (TCNs) only when nonedge ports transition to the forwarding state. Nonedge ports are typically defined as ports that interconnect switches. Edge ports are typically defined as ports that connect a switch to end stations.

RSTP also provides improved network stability because it does not generate a TCN when a port transitions to the discarding state. With RSTP, TCNs are not generated when a port is administratively disabled, excluded from the active topology through configuration, or dynamically excluded from forwarding and learning.

When a TCN is required and generated, it is broadcast out all designated ports as well as out the root port by the initiating device. Unlike traditional STP, neighboring switches that are not in the path of the initiator to the root bridge do not need to wait for this information from the root bridge. As the changes are propagated throughout the network, the switches flush the majority of the MAC addresses located in their MAC address forwarding tables. The individual switches do not, however, flush MAC addresses learned from their locally configured edge ports.

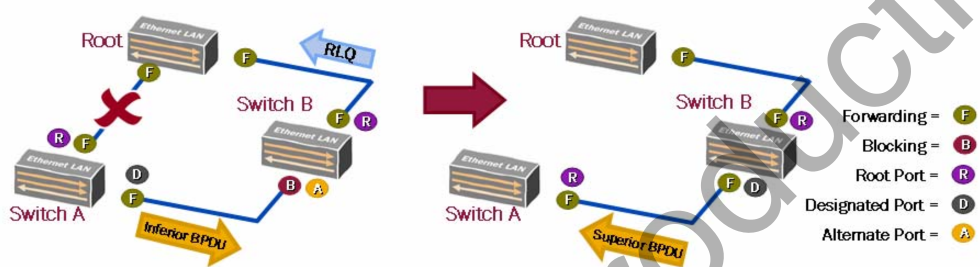


Indirect Link Failure: Part 1

RSTP performs rapid recovery for link failures. This slide, along with the subsequent slide, illustrate a typical scenario for an indirect link failure.

Indirect Link Failure (2 of 2)

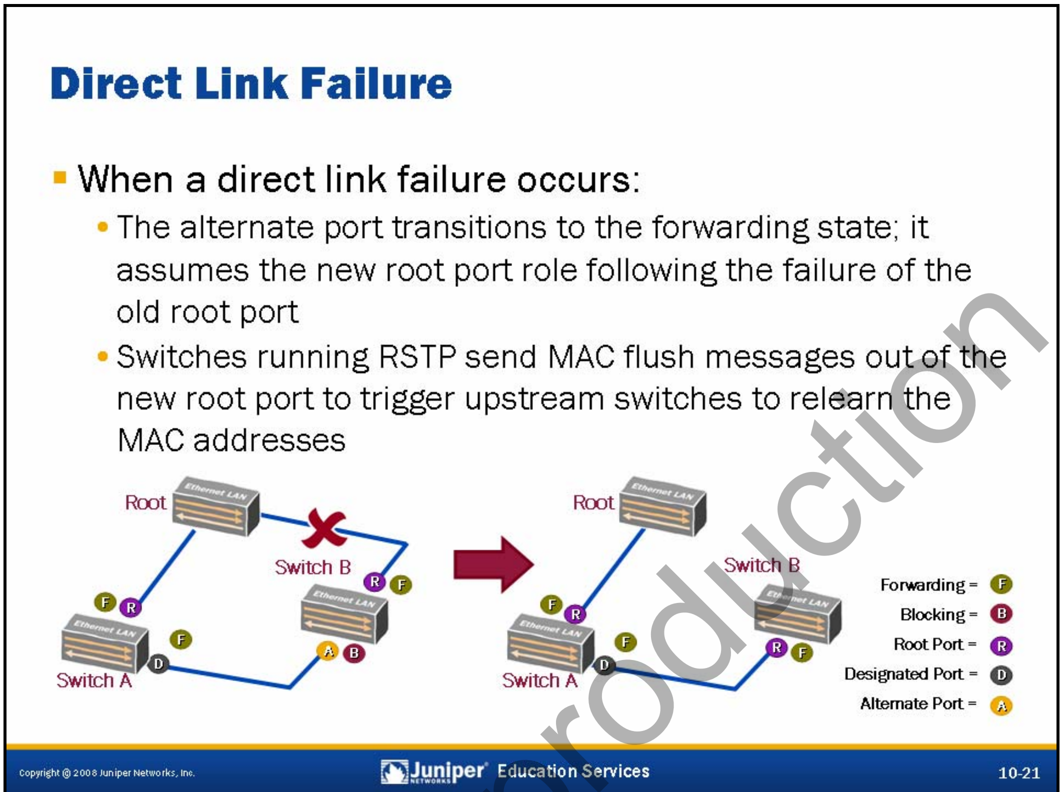
- Remaining steps:
  - After confirming that the root bridge is still alive, Switch B moves the alternate port to the designated port role and begins sending superior BPDUs downstream to Switch A
  - Switch A receives superior BPDU, knows it is not the root, and places the port connecting to Switch B as the root port



 Juniper Education Services 10-20

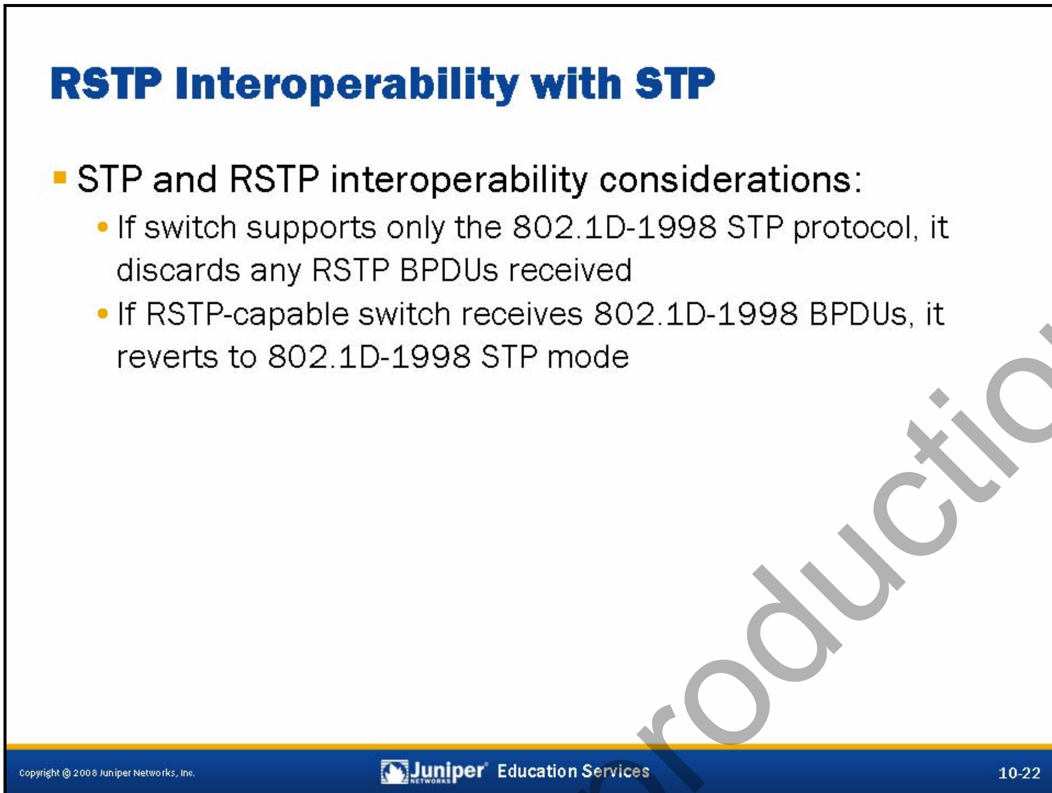
Indirect Link Failure: Part 2

This slide highlights the remainder of the steps involved during an indirect link failure.



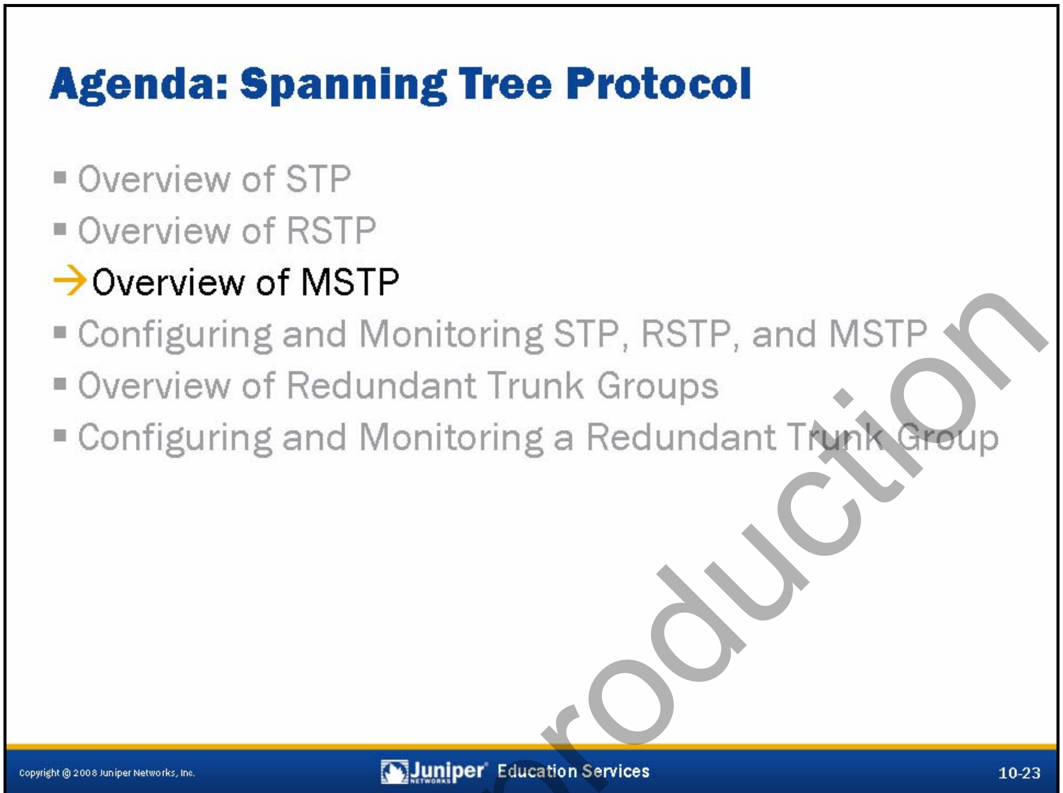
Direct Link Failure

The slide illustrates a typical scenario in which a direct link failure occurs.



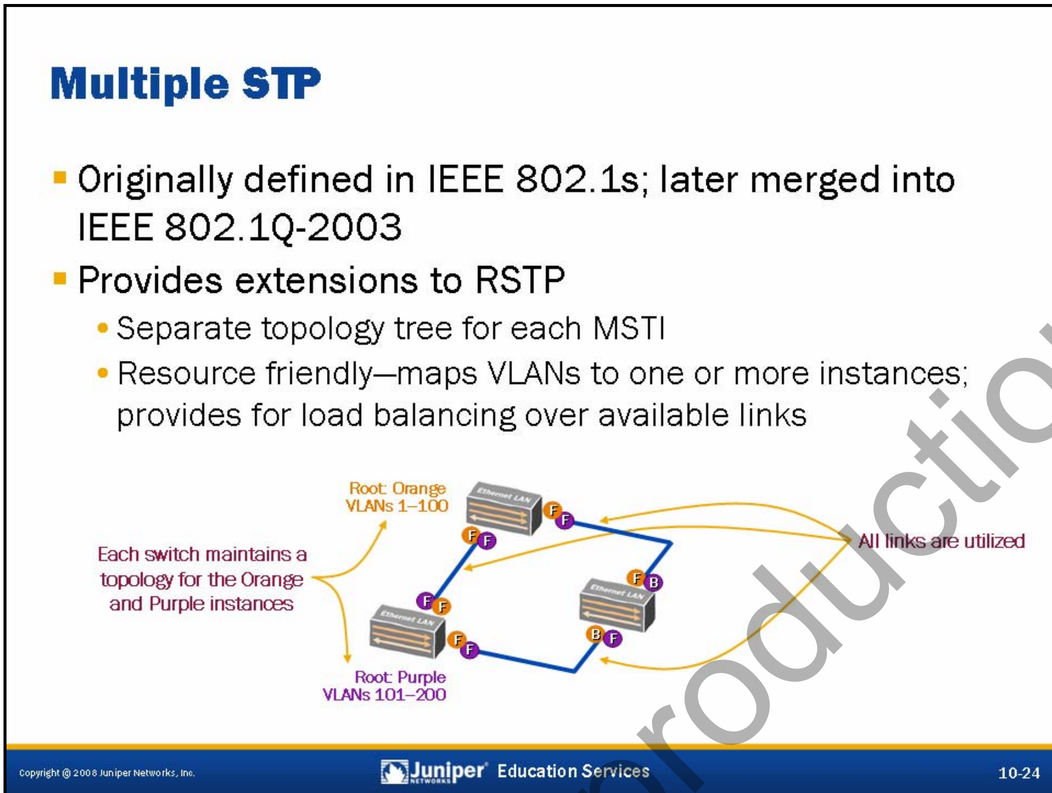
Interoperability Considerations

Switches configured for STP and RSTP will interoperate with one another. There are, however, a few basic considerations to keep in mind. If a switch supports only STP and interconnects with a switch running RSTP, that switch will discard the RSTP BPDUs. The RSTP-capable switch, upon receiving STP BPDUs, reverts to STP mode, thus allowing interoperability between the two devices.



Overview of the Multiple Spanning Tree Protocol

The slide highlights the topic we discuss next.



MSTP Defined

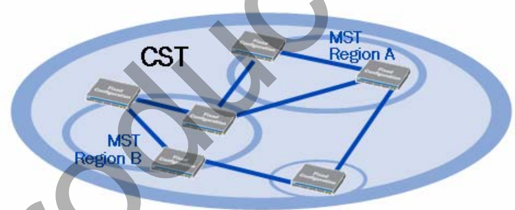
The Multiple Spanning Tree Protocol (MSTP) was originally defined in the IEEE 802.1s draft and later incorporated into the IEEE 802.1Q-2003 specification.

MSTP Enhancements over RSTP

Although RSTP provides faster convergence than STP, it still does not make good use of all available paths within a redundant Layer 2 network. With RSTP, all traffic from all VLANs follows the same path as determined by the spanning tree; therefore, redundant paths are not utilized. MSTP overcomes this limitation through the use of multiple spanning tree instances (MSTI). Each MSTI creates a separate topology tree and can be administratively mapped to one or more VLANs. Allowing users to administratively map VLANs to MSTIs facilitates better load sharing across redundant links within a Layer 2 switching environment.

Multiple Spanning Tree Region

- An MST region is a group of switches with the same region name, revision level, and VLAN-to-instance mapping
  - Max of 64 MSTIs per region
  - One regional root bridge per instance
- Backward compatible with STP and RSTP through common spanning tree (CST)



 Juniper Education Services 10-25

MST Region

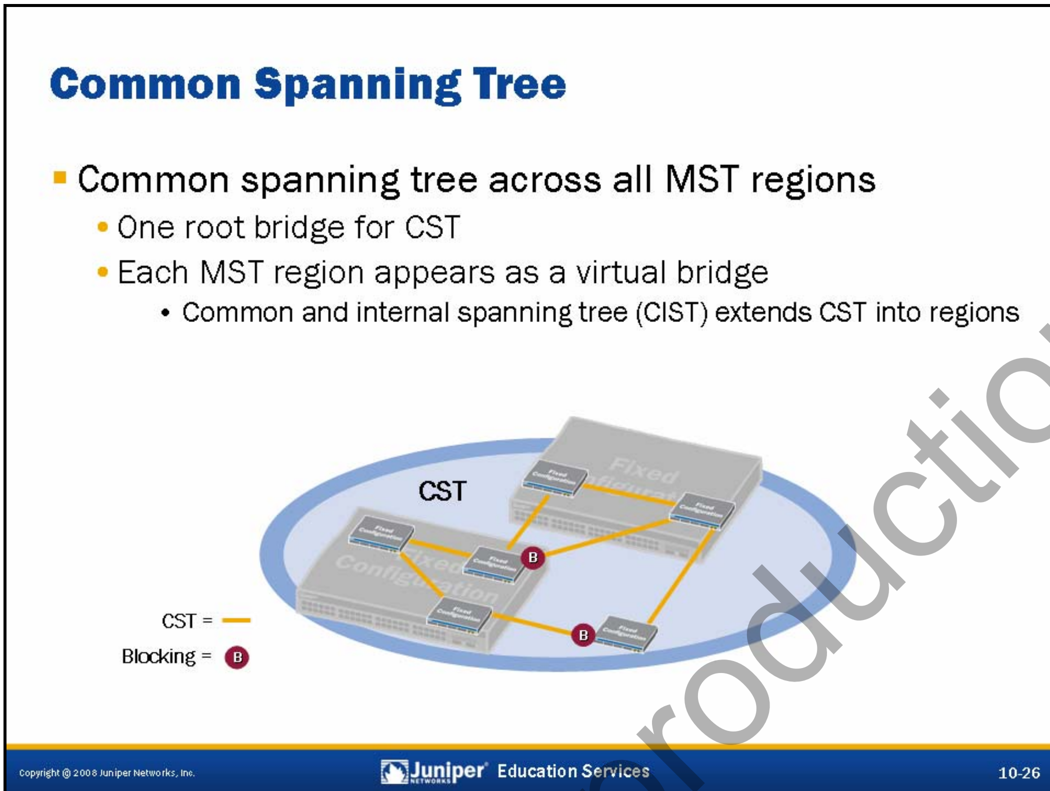
MSTP allows switches to be logically grouped into manageable clusters, known as MST regions. An MST region is a group of switches that share the same region name, revision level, and VLAN-to-instance mapping parameters.

Each MST region supports up to 64 MSTIs. MSTP greatly reduces the number of BPDUs on a LAN by including the spanning tree information for all MSTIs in a single BPDU. MSTP encodes region information after the standard RSTP BPDU along with individual MSTI messages. The MSTI configuration messages convey spanning tree information for each instance.

MSTP elects a regional root bridge for each MSTI. The regional root bridge is elected based on the configured bridge priority and calculates the spanning tree within its designated instance.

MSTP Compatibility with STP and RSTP

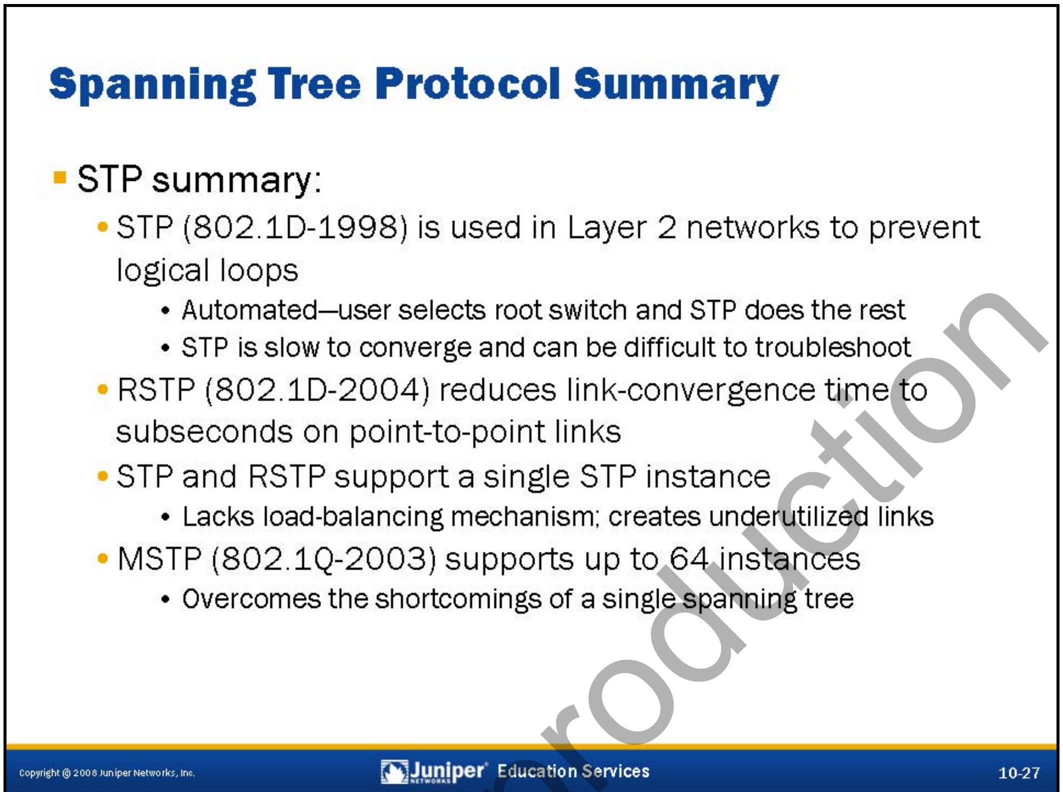
Because MSTP encodes region information after the standard RSTP BPDU, a switch running RSTP interprets MSTP BPDUs as RSTP BPDUs. This behavior facilitates full compatibility between devices running MSTP and devices running STP or RSTP. All RSTP switches outside of an MST region view the MST region as a single RSTP switch. The common spanning tree (CST), which interconnects all MST regions as well as STP devices not bound to a particular region, facilitates end-to-end paths within an MSTP environment.



Common Spanning Tree

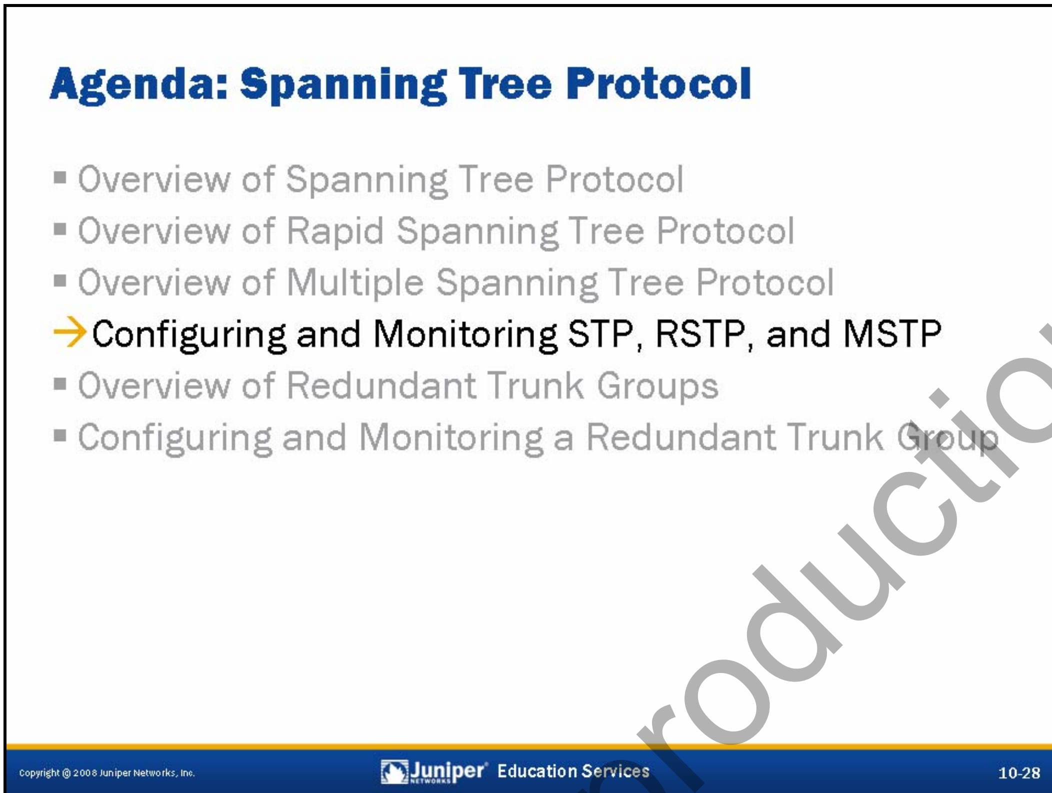
All MSTP environments contain a CST, which is used to interconnect individual MST regions and independent STP devices. A single root bridge is elected and tasked with path calculation for the CST. As illustrated on the slide, each MST region is treated as a virtual bridge within the environment, regardless of the actual number of devices participating in the MST region.

The common and internal spanning tree (CIST) is a single topology that connects all switches (STP, RSTP, and MSTP devices) through an active topology. The CIST includes a single spanning tree as calculated by STP and RSTP together with the logical continuation of connectivity through MST regions. The CIST is calculated by MSTP and ensures connectivity between LANs and devices within a bridged network.



Spanning Tree Protocol Summary

This slide provides a quick overview along with the highlights of STP, RSTP, and MSTP.



Configuring and Monitoring STP, RSTP, and MSTP

The slide highlights the topic we discuss next.

Configuring STP

```
[edit protocols stp]
user@switch# set ?
Possible completions:
+ apply-groups          Groups from which to inherit configuration data
+ apply-groups-except   Don't inherit configuration data from these groups
  bridge-priority       Priority of the bridge (in increments of 4k - 0,4k,8k,..60k)
  disable               Disable STP
  forward-delay         Time spent in listening or learning state (4..30 seconds)
  hello-time            Time interval between configuration BPDUs (1..10 seconds)
> interface
  max-age               Maximum age of received protocol bpdu (6..40 seconds)
> traceoptions          Tracing options for debugging protocol operation
```

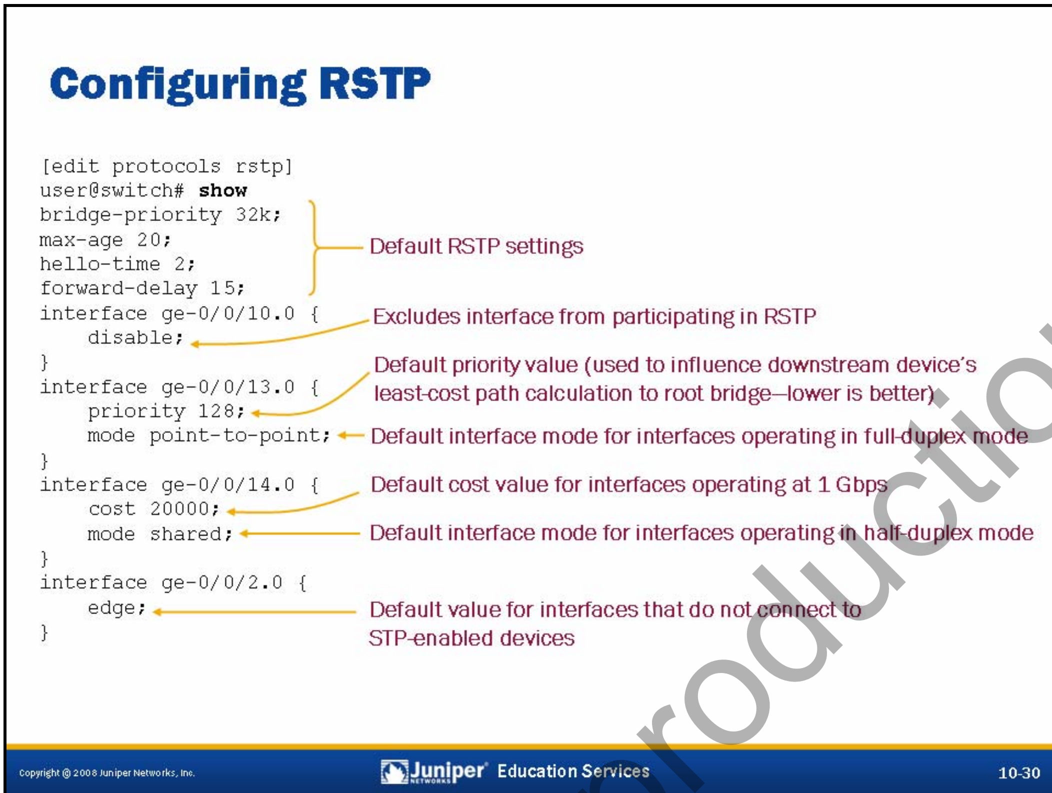
```
[edit protocols stp]
user@switch# show
bridge-priority 32k;
max-age 20;
hello-time 2;
forward-delay 15;
```

Configuration example illustrates default STP settings

 Juniper Education Services 10-29

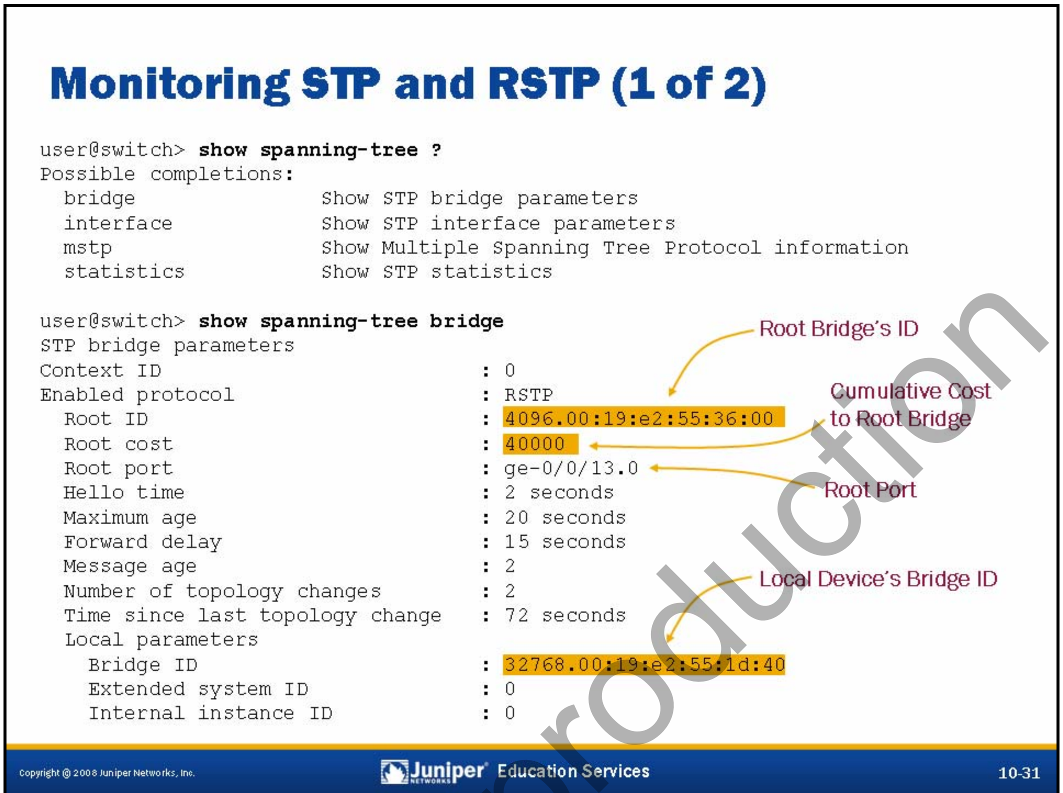
Configuring STP

This slide shows some STP configuration options along with a basic STP configuration. EX-series switches use a version of STP based on IEEE 802.1D-2004, with a forced protocol version of 0, running RSTP in STP mode. Because of this implementation, you can define RSTP configuration options, such as `hello-time`, under the `[edit protocols stp]` configuration hierarchy.



Configuring RSTP

The sample RSTP configuration provided on the slide shows the typical configuration structure along with various settings called out.



Monitoring Spanning Tree Operation: Part 1

This slide and the next illustrate some common operational-mode commands used to monitor the operation of STP and RSTP.

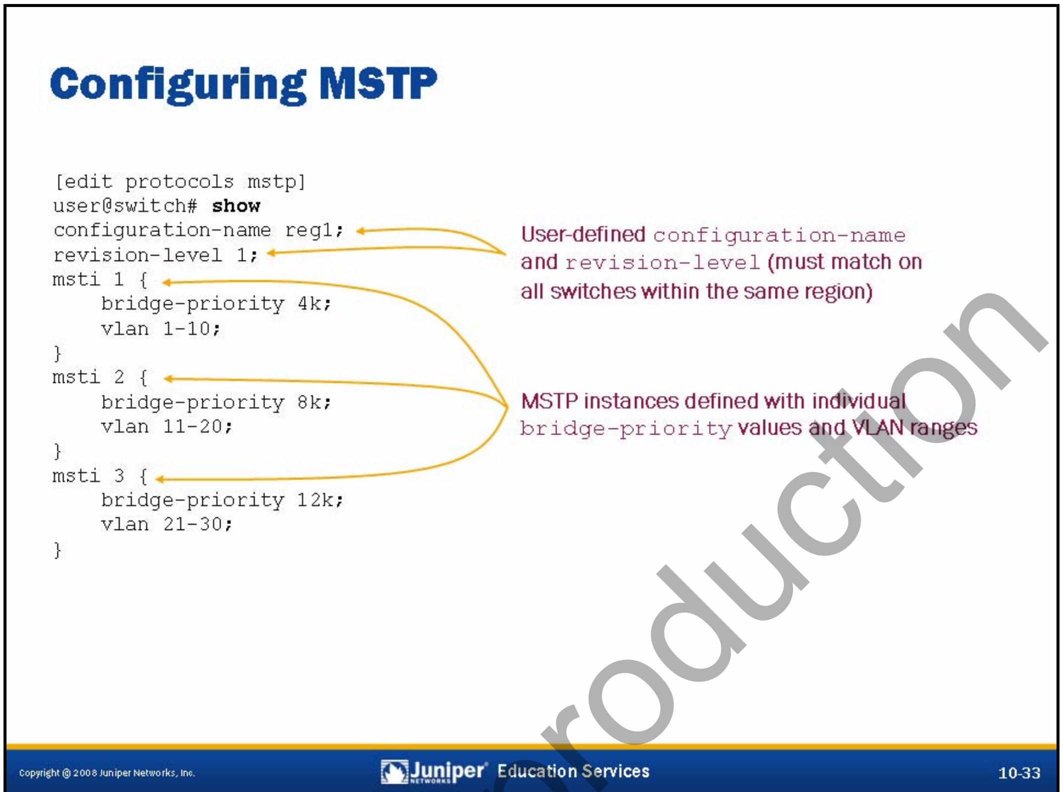
Monitoring STP and RSTP (2 of 2)

```
user@switch> show spanning-tree interface

Spanning tree interface parameters for instance 0

Interface     Port ID    Designated      Designated          Port     State  Role
                           port ID        bridge ID           Cost
ge-0/0/10.0    128:523      128:523  32768.0019e2507c00  20000     BLK    ALT
ge-0/0/11.0    128:524      128:524  32768.0019e2507c00  20000     BLK    ALT
ge-0/0/12.0    128:525      128:525  32768.0019e2507c00  20000     BLK    ALT
ge-0/0/13.0    128:526      128:526  32768.0019e2503fe0  20000     FWD    ROOT
ge-0/0/14.0    128:527      128:527  32768.0019e2503fe0  20000     BLK    ALT
ge-0/0/15.0    128:528      128:528  32768.0019e2503fe0  20000     BLK    ALT

user@switch> show spanning-tree statistics interface

Interface    BPDUs sent   BPDUs received      Next BPDU
                                            transmission
ge-0/0/10.0          7              5                0
ge-0/0/11.0          7              5                0
ge-0/0/12.0          7              5                0
ge-0/0/13.0          7              4                0
ge-0/0/14.0          7              5                0
ge-0/0/15.0          7              5                0
```

 Juniper Education Services 10-32

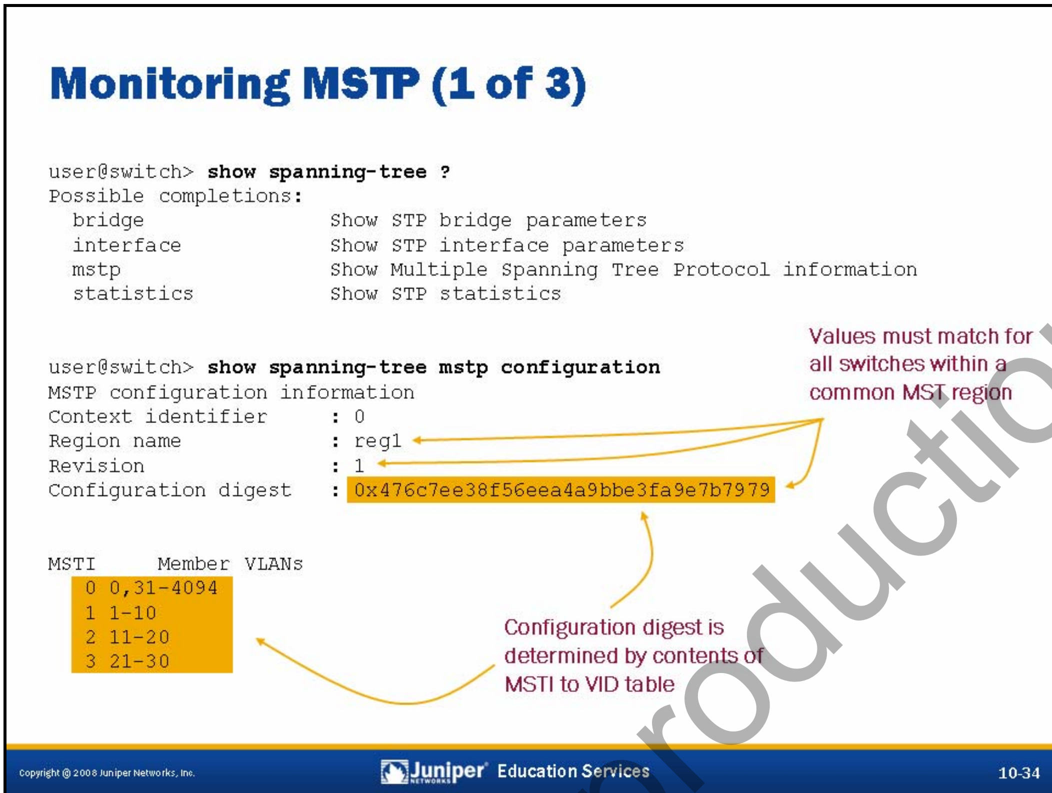
Monitoring Spanning Tree Operation: Part 2

This slide shows typical output for the **show spanning-tree interface** and **show spanning-tree statistics interface** commands.



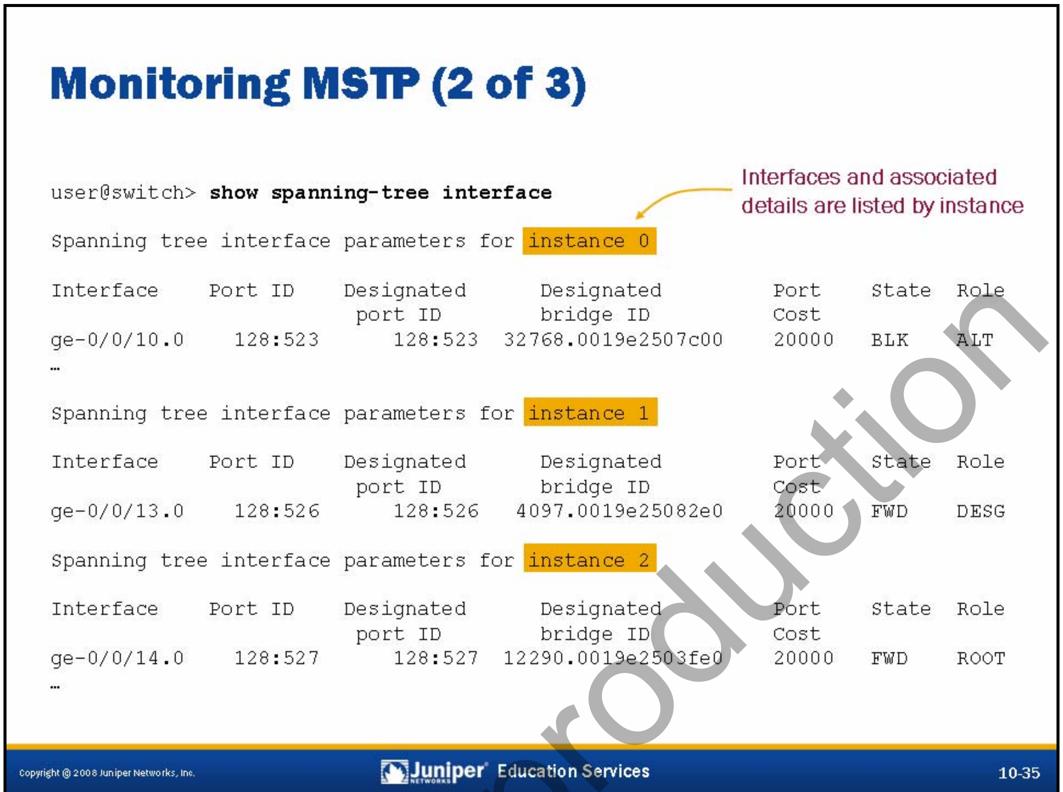
Configuring MSTP

The sample MSTP configuration provided on the slide shows the typical configuration structure along with various settings called out.



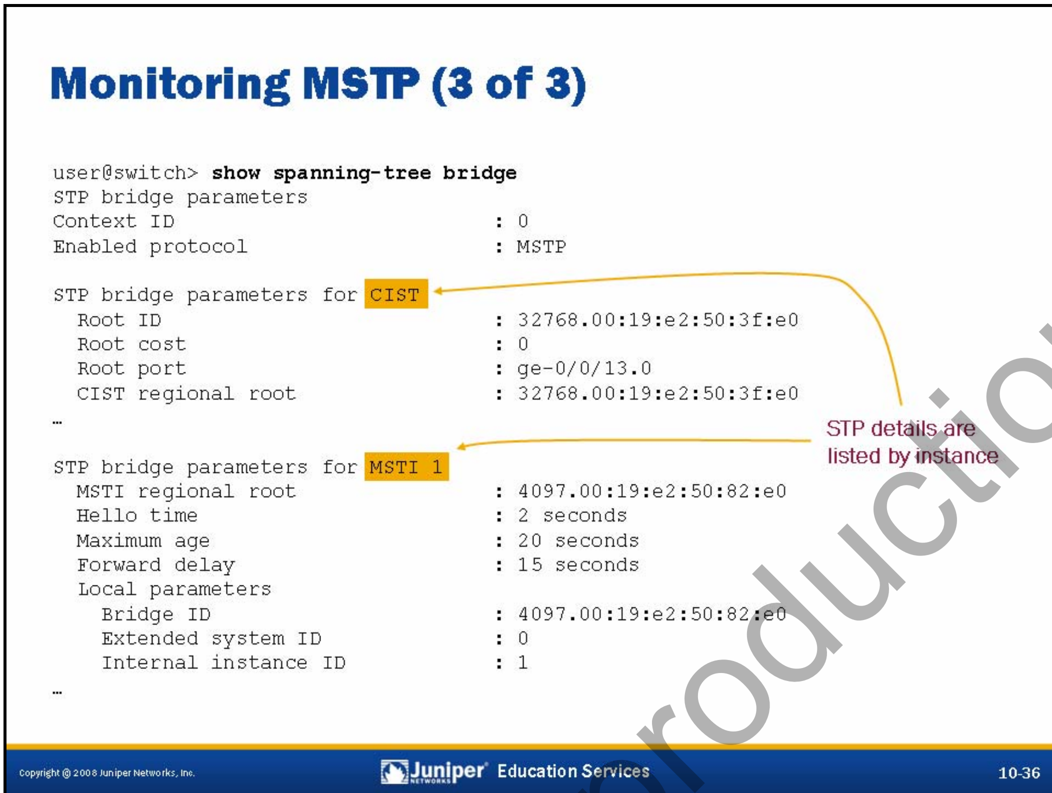
Monitoring MSTP Operation: Part 1

This slide and the next two slides illustrate some common operational-mode commands used to monitor MSTP. This slide highlights the **`show spanning-tree mstp configuration`** command, which you can use to verify MSTP configuration parameters including region, revision, and assigned MSTI parameters.



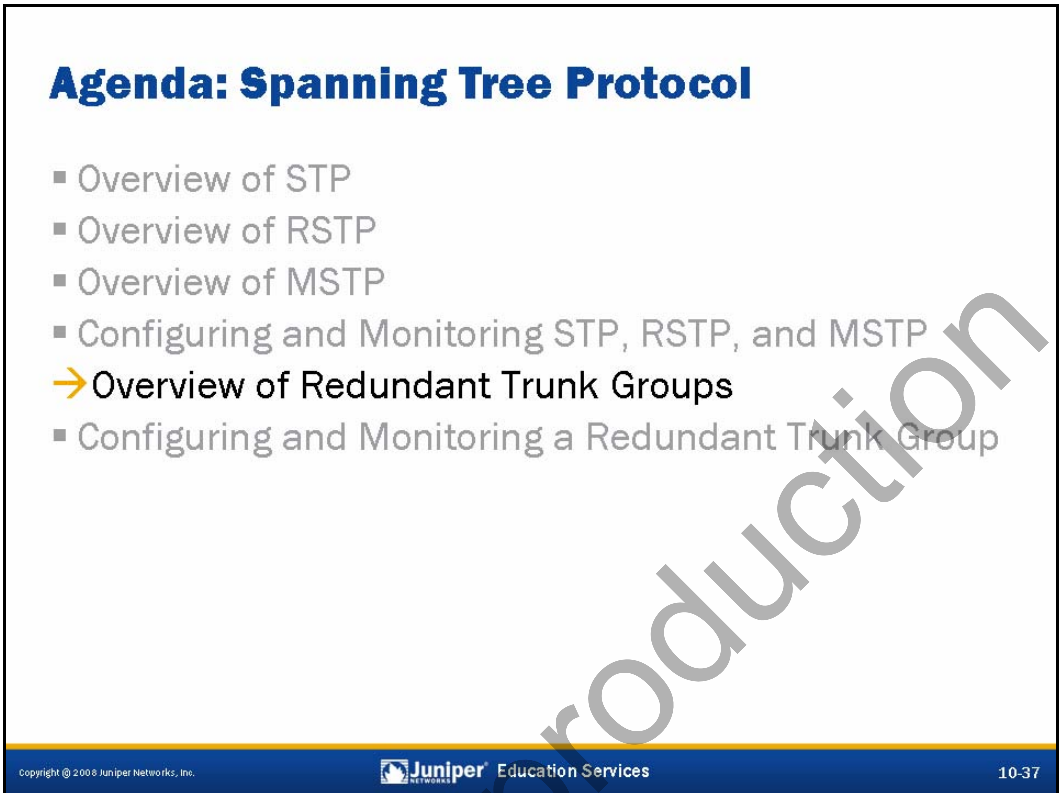
Monitoring MSTP Operation: Part 2

This slide highlights the use of the **show spanning-tree interface** command, which you use to verify the MSTP interface status and role assignment along with various other details.



Monitoring MSTP Operation: Part 3

This slide highlights the **`show spanning-tree bridge`** command, which you use to display STP bridge parameters for the CIST and individual MSTIs.



Overview of a Redundant Trunk Group

The slide highlights the topic we discuss next.

Redundant Trunk Group

- Redundant trunk group:
  - Provides quick and simple failover mechanism for redundant Layer 2 links without requiring STP
  - Primary application is in enterprise environments where each access switch is dual homed to two distribution switches
  - Has one active link to forward traffic, while the other link acts as a backup and does not forward traffic; when active link fails, backup link becomes active and forwards traffic

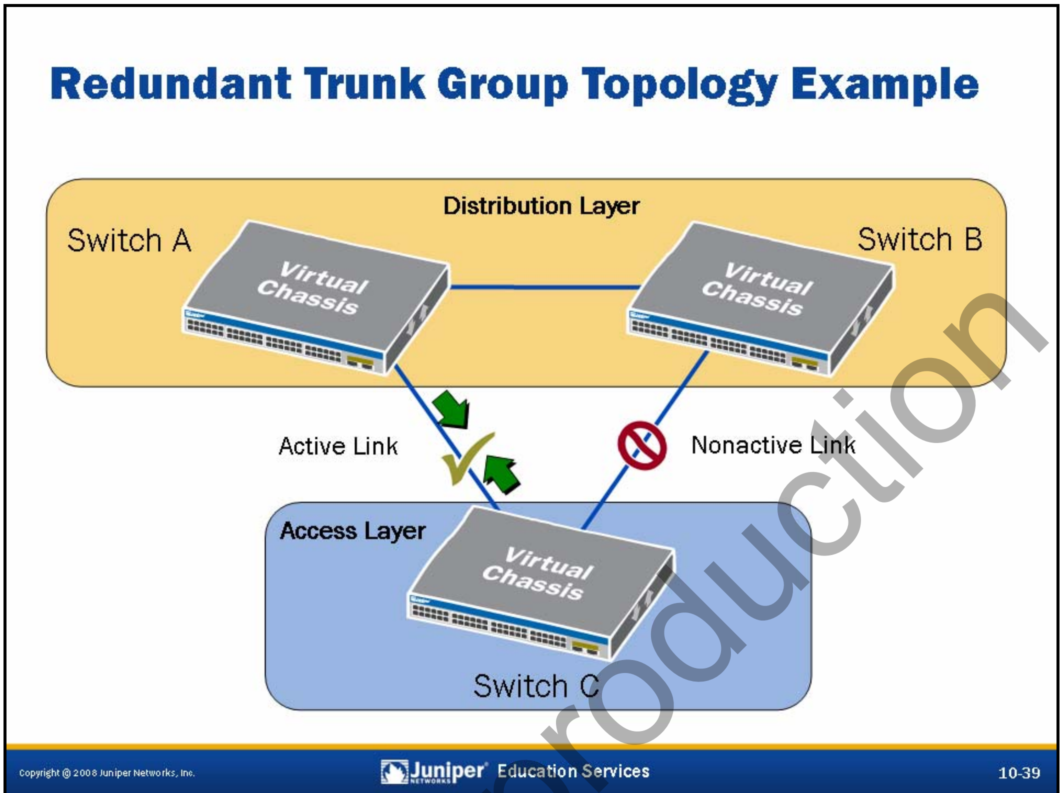
Juniper Education Services
10-38

Redundant Trunk Group

A redundant trunk group facilitates a basic failover mechanism between redundant Layer 2 links without requiring STP. If the primary or active trunk should become unavailable, a redundant trunk group ensures that traffic is routed to a secondary trunk port, while keeping network convergence time to a minimum.

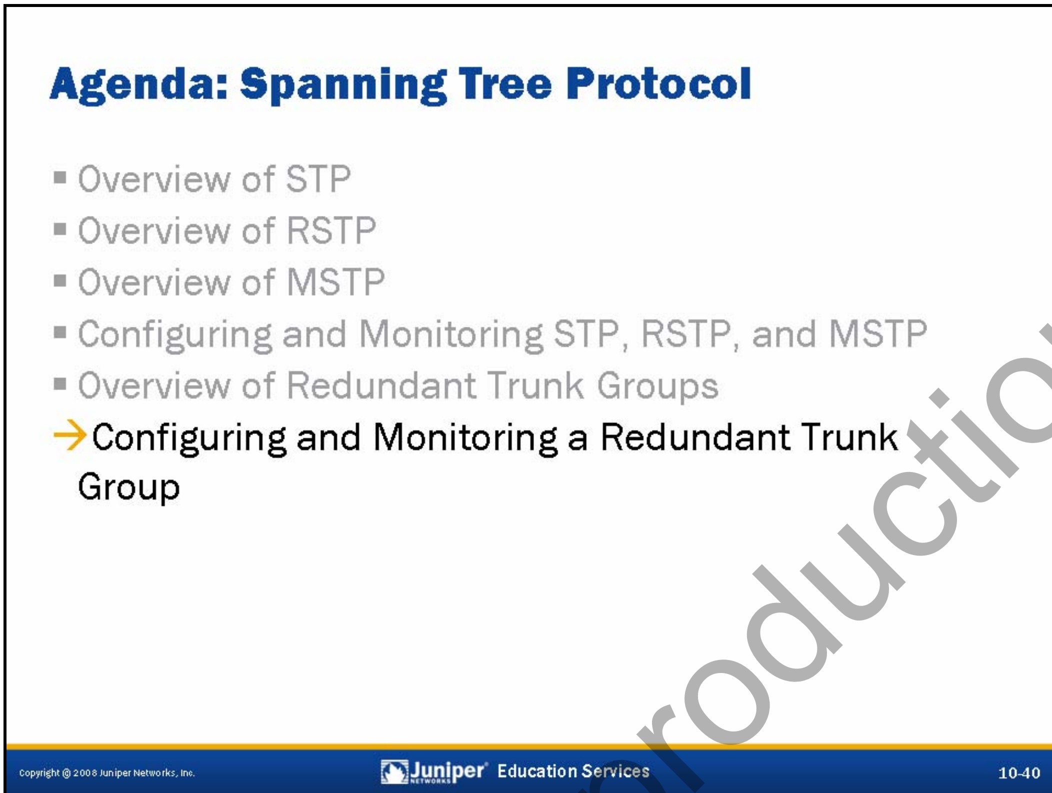
Redundant trunk groups are typically configured in a redundant enterprise network environment that consists of access and distribution layers. If an access switch has dual-homed trunk links to the distribution layer, you can configure the redundant trunk link as part of a redundant trunk group to provide a simple recovery solution, should the active trunk go down.

You configure the redundant trunk group on the access switch, and it contains two links: a primary (or active) link, and a secondary link. If the active link fails, the secondary link automatically assumes the active role and starts forwarding data traffic. JUNOS software forwards data traffic only on the active link. The software drops data traffic on the secondary link. JUNOS software tracks these dropped packets. You can view these packets by using the **`show interfaces`** ***`interface-name`*** command. While data traffic is blocked on the secondary link, Layer 2 control traffic is still permitted. For example, you can run a Link Layer Discovery Protocol (LLDP) session between two EX-series switches on the secondary link.



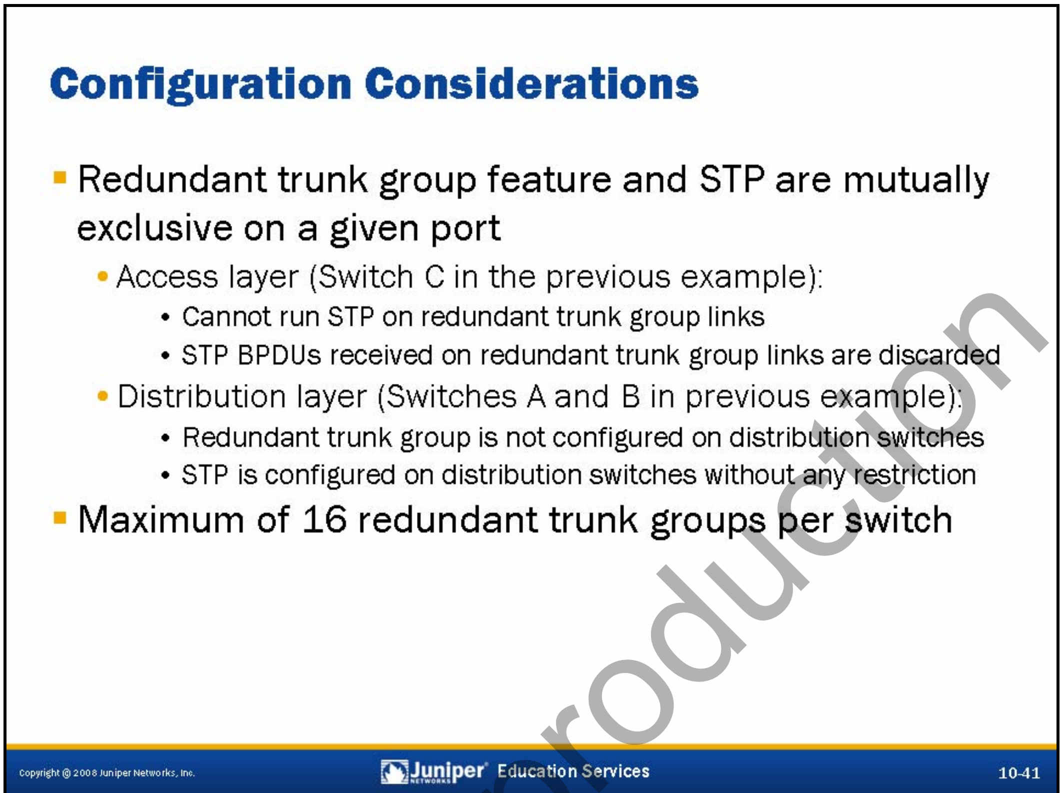
Redundant Trunk Group Topology Example

The slide illustrates a typical topology example in which the redundant trunk group feature might be used. In this example, Switch C is functioning as an access layer switch and has two, multihomed trunks connecting to Switch A and Switch B, which are operating as distribution layer switches. (Access switches are typically used to connect users to the network, whereas distribution switches often interconnect multiple access switches.) Switch C has a redundant trunk group configured and has the trunk connecting to Switch A set as active, whereas the trunk connecting to Switch B is set as nonactive (or secondary). In this scenario, Switch A and Switch B do not require any special configuration, nor do they have any restrictions. Switch C, however, does have a special configuration in the form of a redundant trunk group definition. Because the redundant trunk group is configured, Switch C cannot run STP or RSTP on the network ports that are participating in the redundant trunk group. All other ports can, if needed, participate in STP or RSTP.



Configuring and Monitoring Redundant Trunk Groups

The slide highlights the topic we discuss next.



Redundant Trunk Group and STP

When implementing the redundant trunk group feature, keep in mind that some special configuration considerations exist. For example, the redundant trunk group feature and STP are mutually exclusive on a given port.

Based on the "Redundant Trunk Group Topology Example" on page 39, the redundant trunk group feature is configured only on the access layer switch, Switch C. The ports participating in the redundant trunk group cannot participate in STP. If STP BPDUs are received on the network ports configured within the redundant trunk group, those BPDUs are discarded.

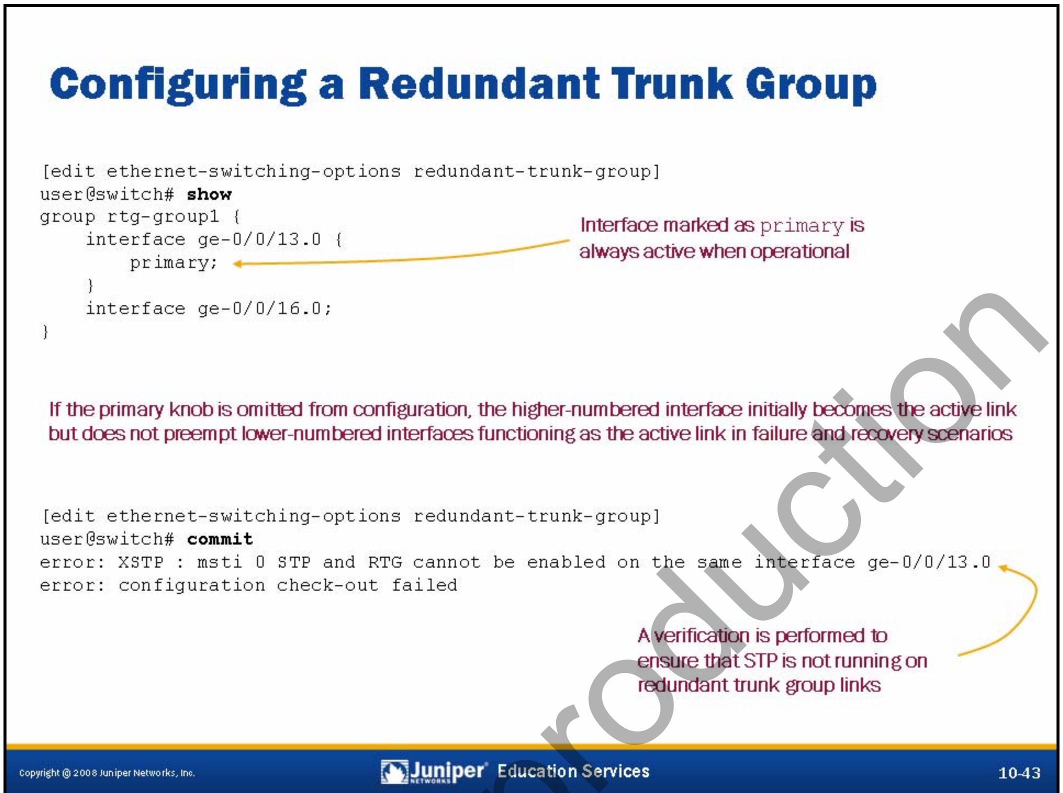
The distribution layer switches, Switch A and Switch B in the previous example, do not require any special configuration, nor do they have any restrictions pertaining to STP.

Continued on next page.

Maximum Number of Groups

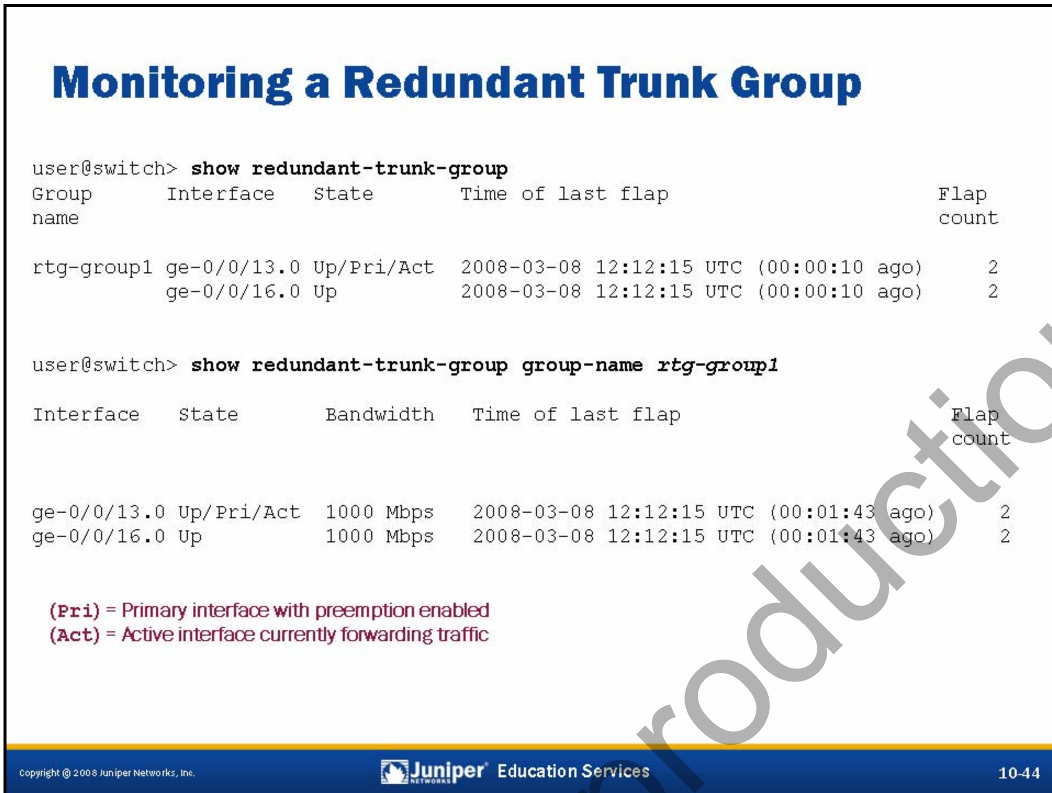
Each EX-series device can have up to sixteen redundant trunk groups configured. This consideration is highlighted in the following configuration snippet:

```
[edit ethernet-switching-options]
root# show
redundant-trunk-group {
    group rtg1 {
        interface ge-0/0/0.0;
        interface ge-0/0/1.0;
    }
...
    group rtg17 {
        interface ge-1/0/11.0;
        interface ge-1/0/12.0;
    }
}

[edit ethernet-switching-options]
root# commit
error: ESWD rtg : Only 16 max groups are allowed
error: configuration check-out failed
```



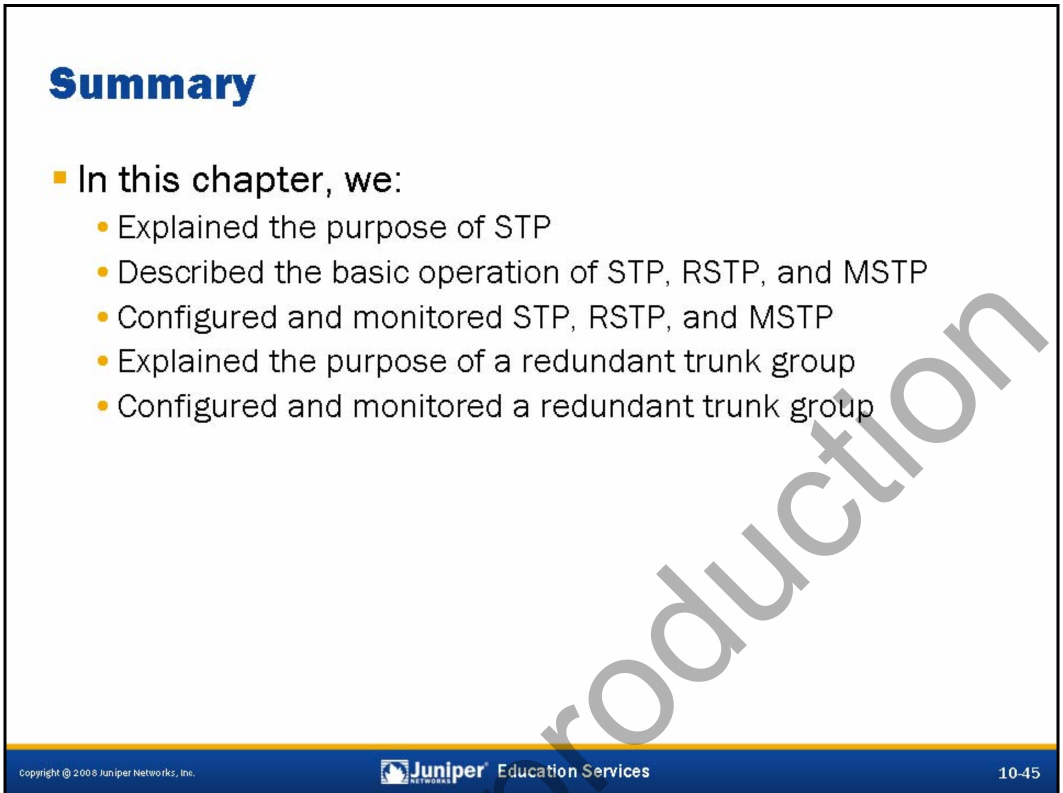
Configuring a Redundant Trunk Group

The sample redundant trunk group configuration provided on the slide shows the typical configuration structure. This example also highlights the **`primary`** configuration knob, which marks the associated interface as the active interface whenever it is operational. As noted on the slide, if the **`primary`** knob is omitted from the configuration, the higher-numbered interface initially becomes the active link but does not preempt any lower-numbered interfaces functioning as the active link in failure and recovery scenarios. The slide also illustrates the resulting error message when a commit is issued and both STP and the redundant trunk group feature are configured on the same network port simultaneously.



Monitoring a Redundant Trunk Group

This slide illustrates some common operational-mode commands used to monitor a redundant trunk group.



This Chapter Discussed:

- The purpose of the STP;
- The basic operation of STP, RSTP, and MSTP;
- Configuring and monitoring STP, RSTP, and MSTP;
- The purpose of a redundant trunk group; and
- Configuring and monitoring a redundant trunk group.

Review Questions

1. What is the purpose of the Spanning Tree Protocol?
2. Describe the operation of the STP port states.
3. Describe how a spanning tree is built.
4. How are STP, RSTP, and MSTP different?
5. What is a redundant trunk group? Describe the environment in which a redundant trunk group is typically found.

 Juniper Education Services 10-46

Review Questions:

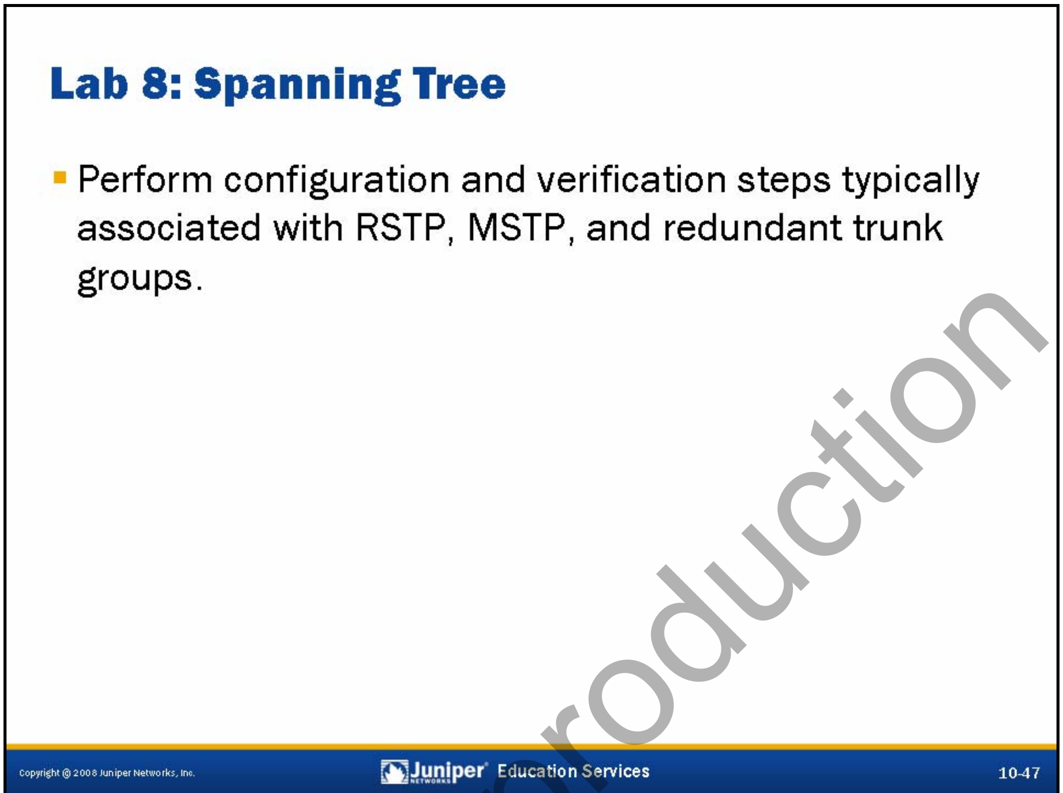
1.

2.

3.

4.

5.

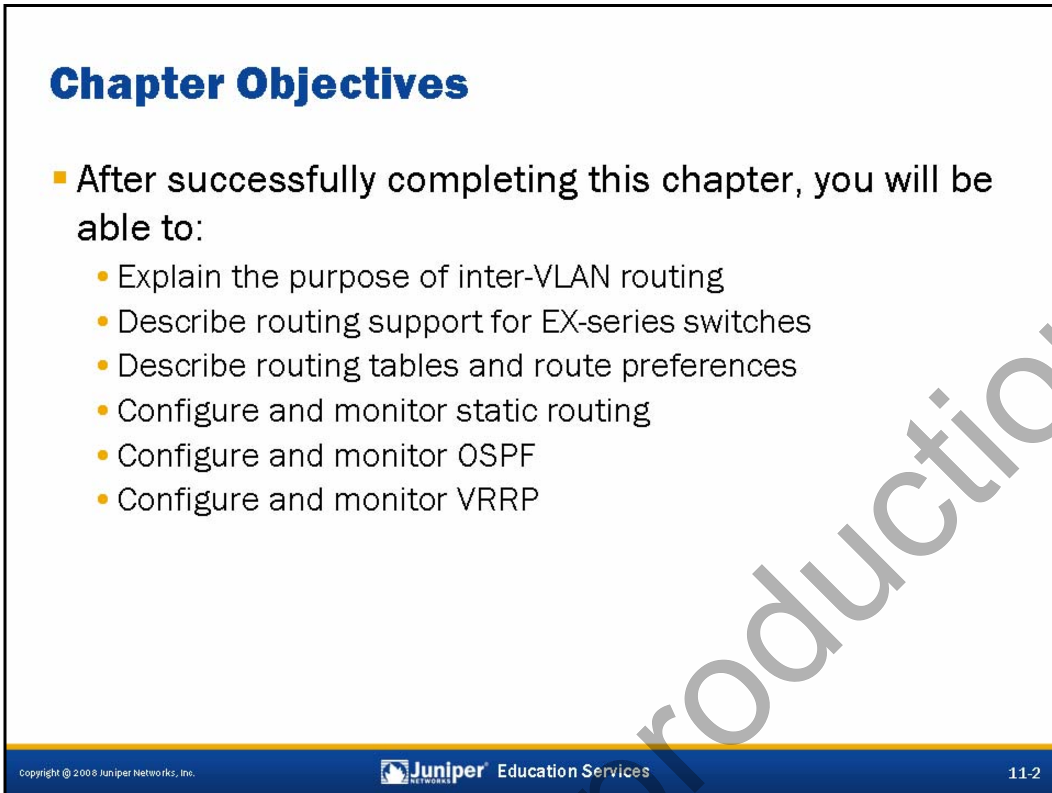


Lab 8: Spanning Tree

The slide provides the objective for this lab.

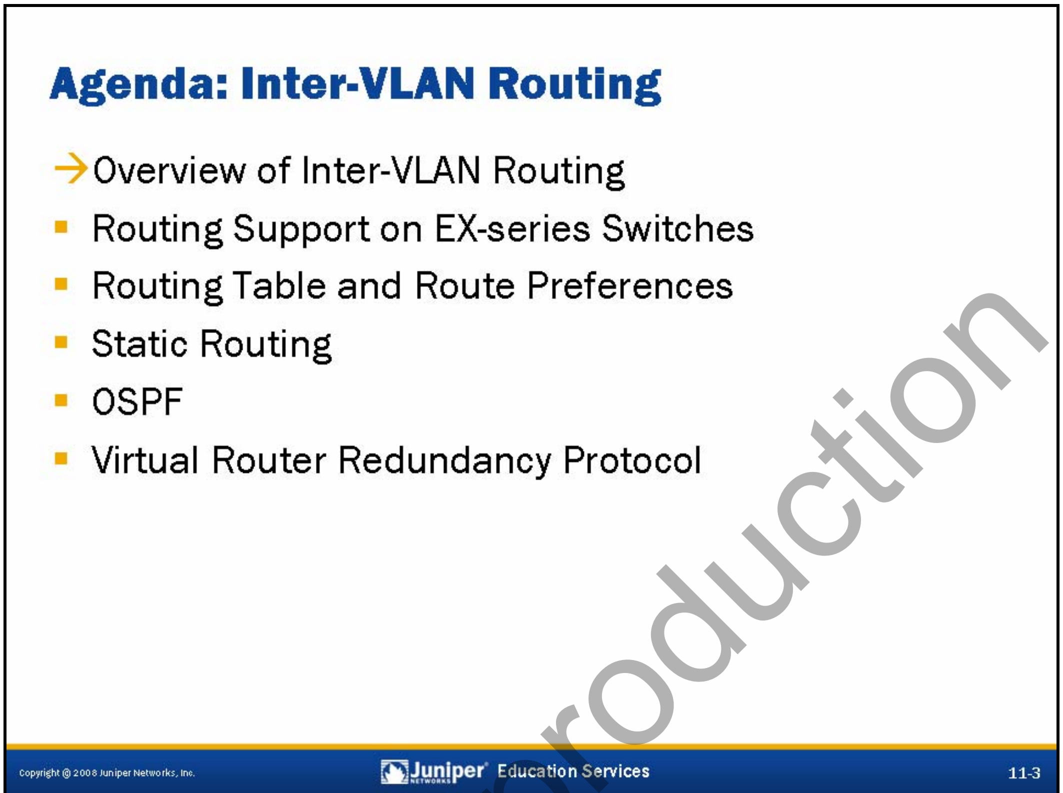
Operating Juniper Networks Switches in the Enterprise

Chapter 11: Inter-VLAN Routing



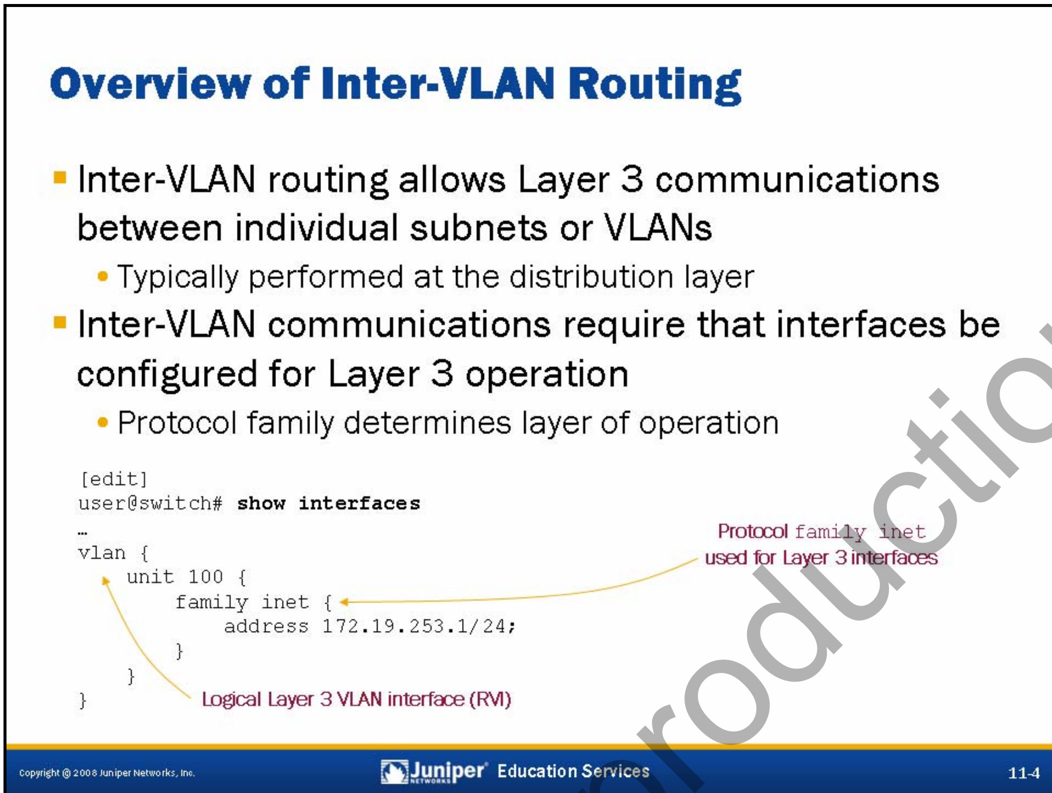
This Chapter Discusses:

- The purpose of inter-VLAN routing;
- Routing support for EX-series switches;
- Route tables and route preference;
- Configuring and monitoring static routing;
- Configuring and monitoring OSPF; and
- Configuring and monitoring the Virtual Router Redundancy Protocol (VRRP).



Overview of Inter-VLAN Routing

The slide lists the topics we cover in this chapter. We discuss the highlighted topic first.



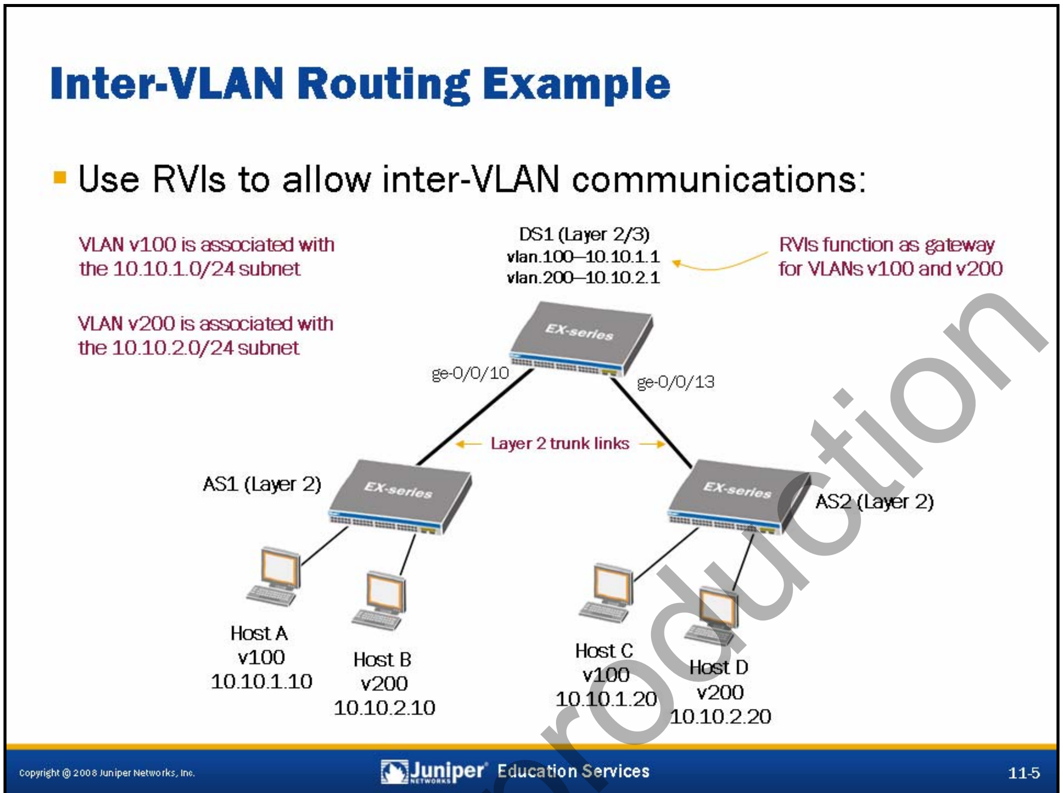
Allows Layer 3 Communications

Inter-VLAN routing facilitates communication between different IP subnets or virtual LANs (VLANs). All EX-series switches are considered multilayer switches because they can perform both switching and routing operations. EX-series switches support a number of routing protocols and forwarding mechanisms, which we discuss on subsequent pages in this chapter.

Inter-VLAN routing operations are typically performed in the distribution or aggregation layer within a hierarchical network.

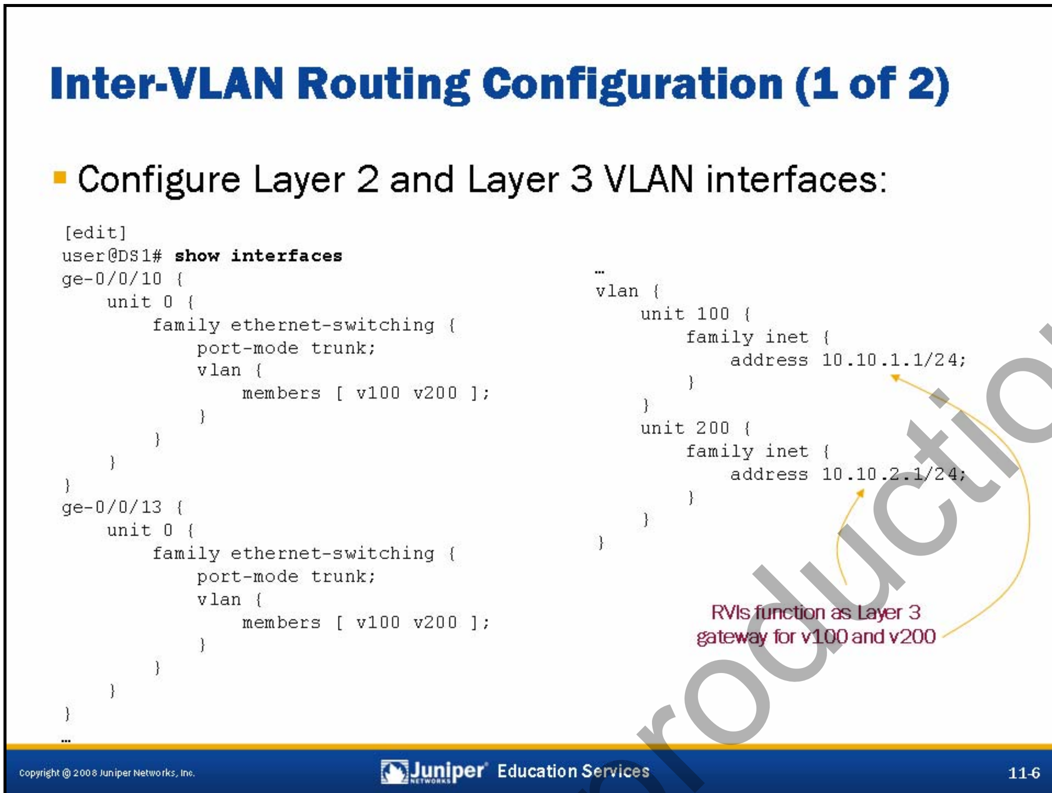
Requires Layer 3 Interface

Because routing between subnets is performed at the network layer, inter-VLAN communications require a Layer 3 interface on the device performing this operation. The example shown on the slide illustrates a logical Layer 3 VLAN interface (RVI) and highlights the `family inet` designation, which is required for an interface to operate in Layer 3 mode.



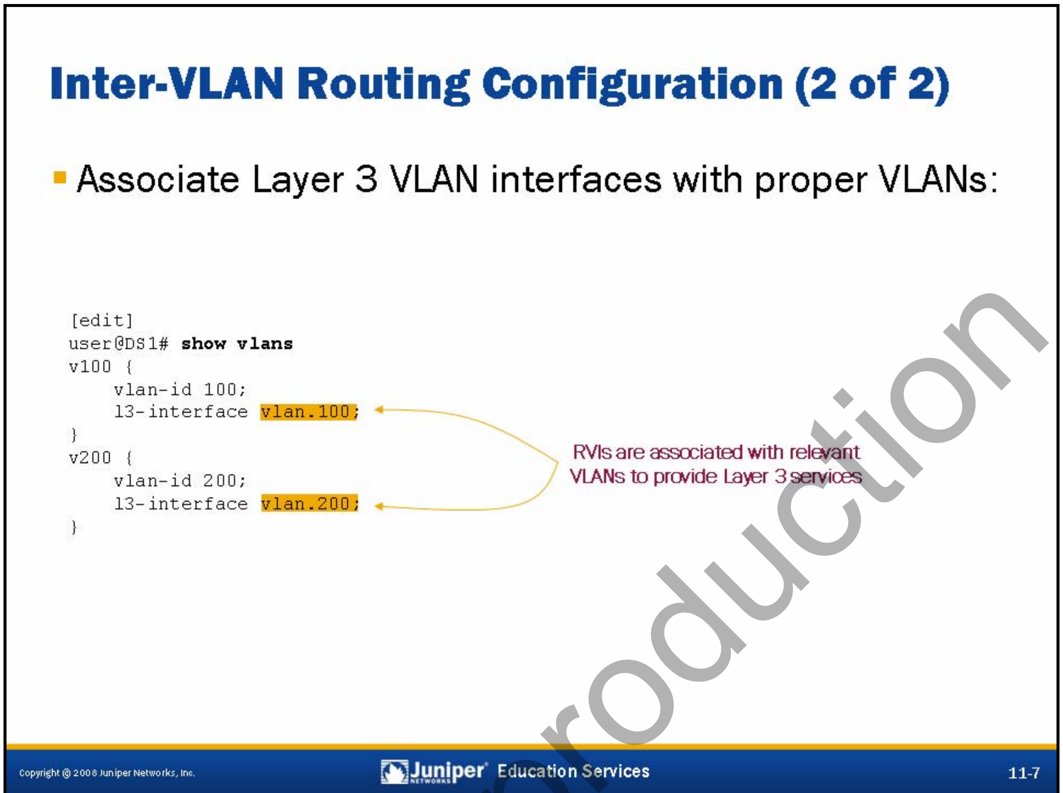
Inter-VLAN Routing Example

In the example on the slide, there are two distinct subnets: 10.10.1.0/24 and 10.10.2.0/24. These two subnets are associated with the v100 and v200 VLANs, respectively. To allow inter-VLAN communications, a Layer 3 VLAN interface (RVI) must be defined and associated with each VLAN or subnet on the DS1 distribution switch. The IP address associated with each Layer 3 VLAN interface functions as the default gateway for the associated subnet. All inter-VLAN communications are sent to the gateway, which is the DS1 device in this example, then relayed to the appropriate subnet and destination host.



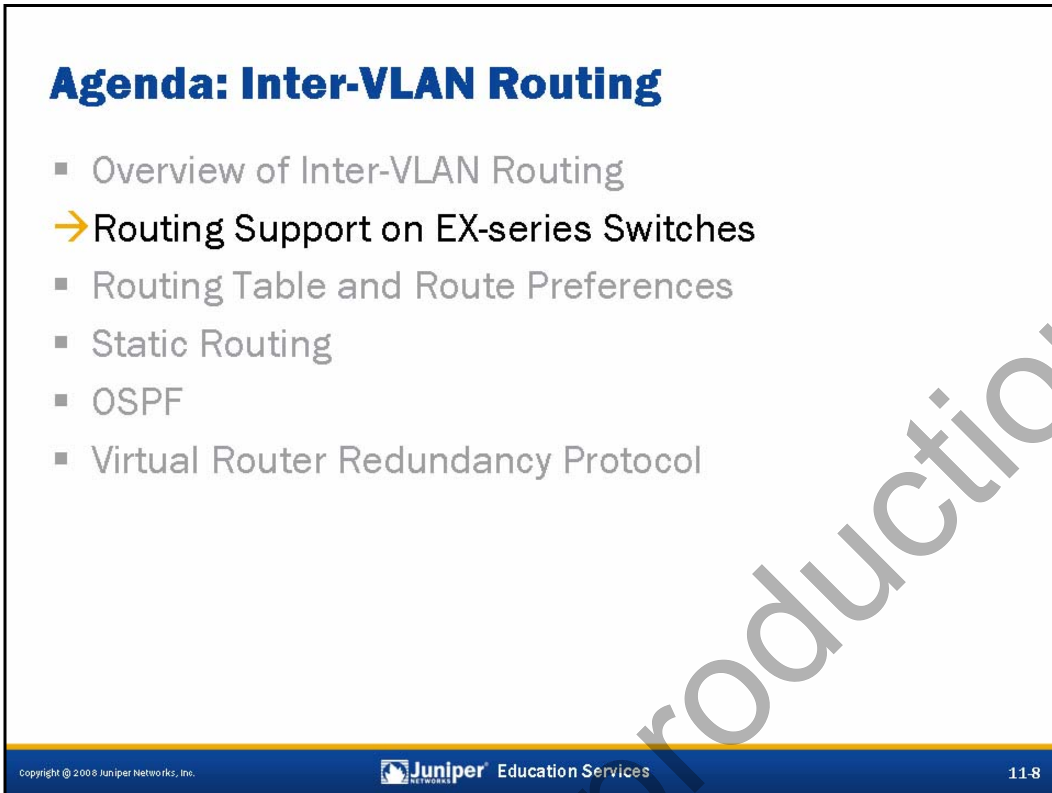
Configuring the Interfaces

The slide shows the interface configuration required on the DS1 switch. The vlan.100 and vlan.200 routed VLAN interfaces function as gateways for VLAN v100 and VLAN v200 respectively. Although not shown in this example, the access switches (AS1 and AS2) must also have their access and trunk ports properly configured to permit communications.



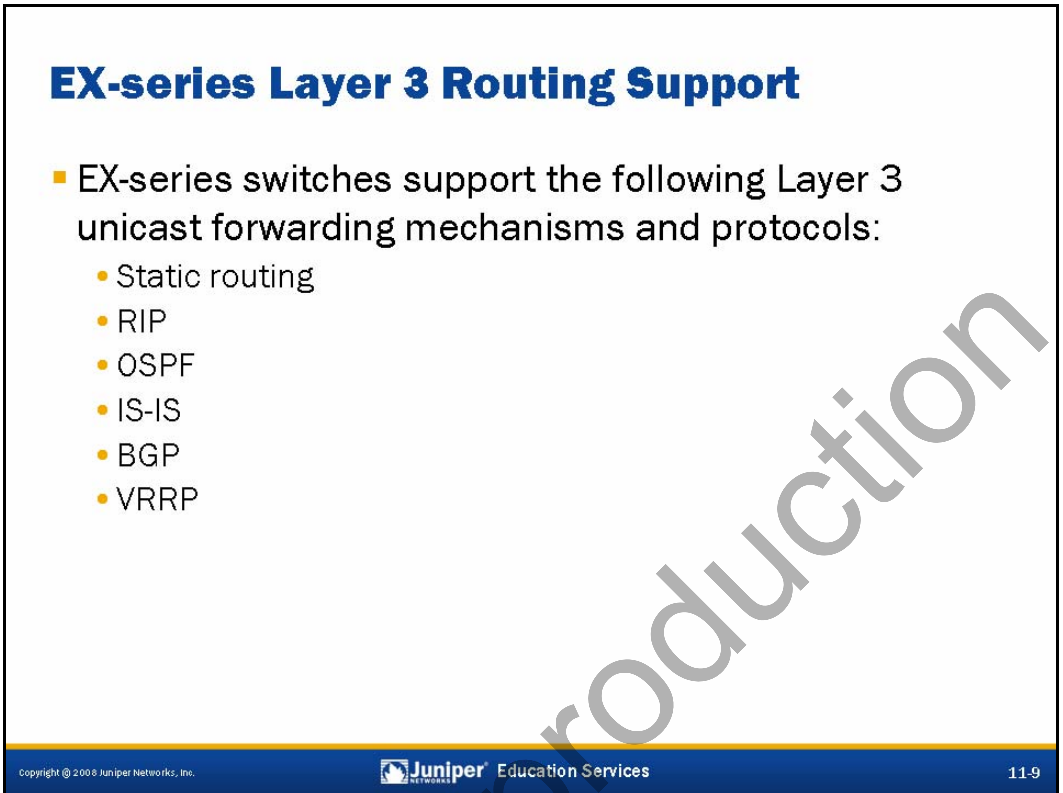
Associating RVIs with VLANs

This slide shows the association of the vlan.100 and vlan.200 RVIs with their respective VLANs. This association allows the referenced RVIs to provide Layer 3 services to hosts participating on the v100 and v200 VLANs.



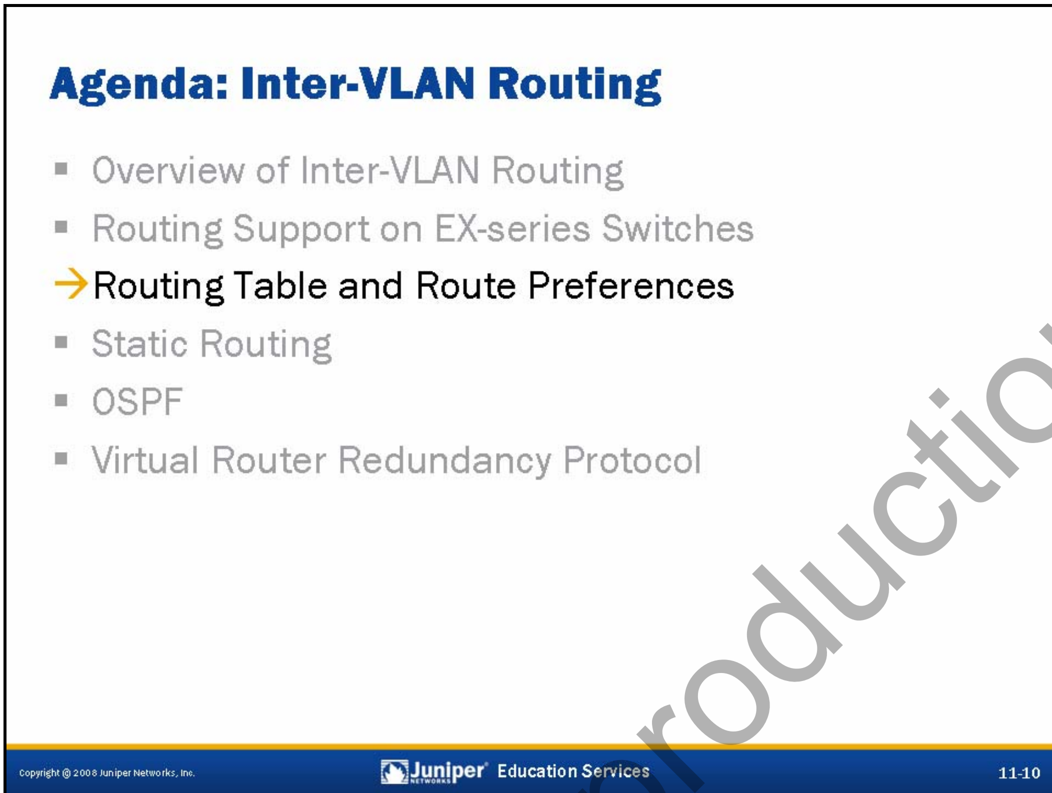
Routing Support on EX-series Switches

The slide highlights the topic we discuss next.



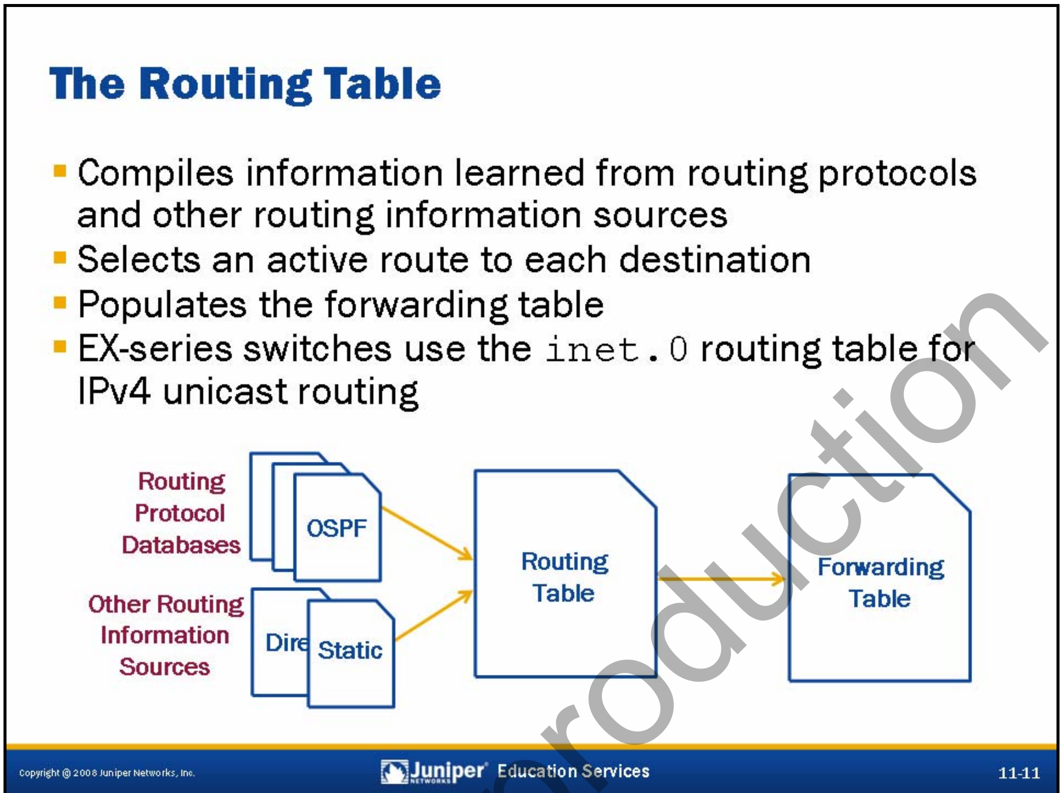
Layer 3 Routing Support

This slide introduces the various Layer 3 unicast forwarding mechanisms and routing protocols. We cover static routing, OSPF, and the Virtual Router Redundancy Protocol (VRRP) in subsequent sections in this chapter.



Routing Table and Route Preference

The slide highlights the topic we discuss next.



Routing Information Sources

The JUNOS software routing table consolidates prefixes from multiple routing information sources including various routing protocols, static routes, and directly connected routes.

Active Route Selection

When the switch receives multiple routes for a given prefix, it selects a single route as the active route. With additional configuration, JUNOS software supports multiple, equal-cost routes.

Forwarding Table

The active route for each destination is used to populate the router's forwarding table. The forwarding table determines the outgoing interface and Layer 2 rewrite information for each packet the router forwards.

Multiple Routing Tables

Juniper Networks switches can accommodate multiple routing tables. The primary routing table, `inet.0`, is used to store IPv4 unicast routes. This course concentrates solely on the `inet.0` routing table.

Route Preference

- Ranks routes received from different sources
- Primary criterion for selecting the active route
- Ranges from 0 to 4,294,967,295, with lower value preferred

Route Preference Values

Routing Information Source	Default Preference
Direct	0
Local	0
Static	5
OSPF internal	10
RIP	100
OSPF AS external	150
BGP (both EBGP and IBGP)	170

 Juniper Education Services 11-12

Preferred Routing Information Sources

JUNOS software uses route preference to differentiate routes received from different routing protocols or routing information sources. Route preference is equivalent to administrative distance on other vendors' equipment.

Primary Tiebreaker

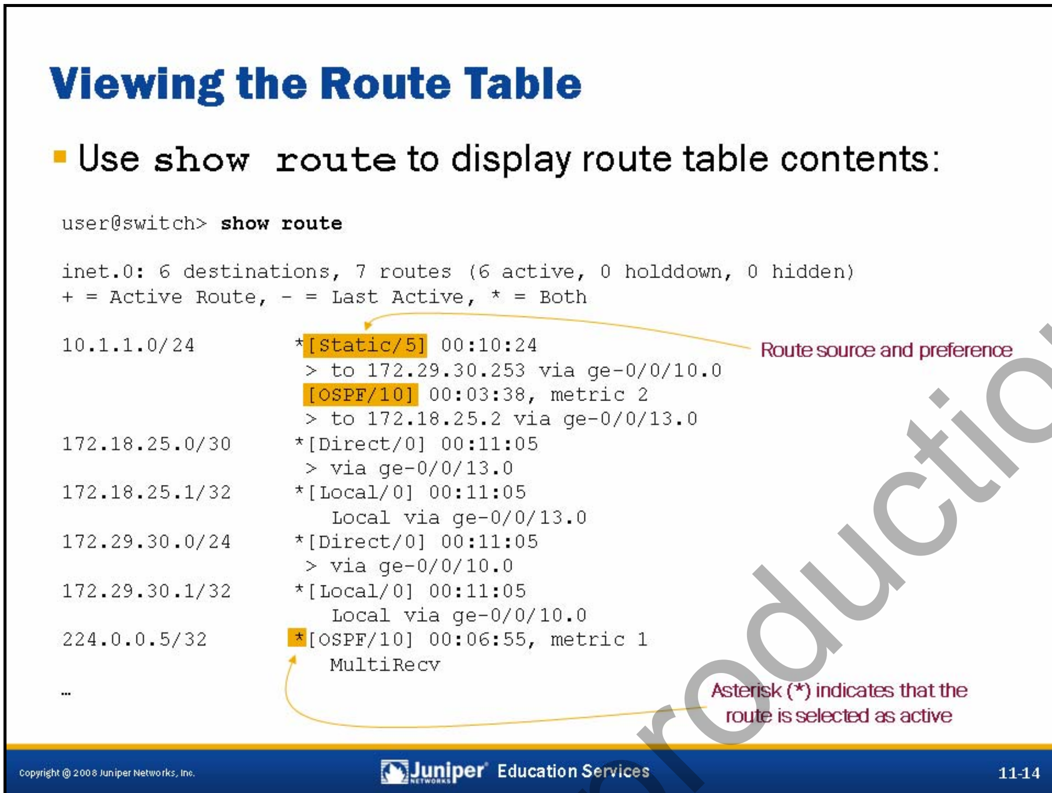
JUNOS software uses route preference as the primary criterion for selecting the active route. Preference values cause routes from certain information sources to be ranked more preferably than the same route received from another information source. The table at the bottom of the slide shows the default preference values for a selected set of routing information sources.

Continued on next page.

Lower Is Better

Routing preference values can range from 0 to 4,294,967,295. Lower preference values are preferred over higher preference values. The following command output demonstrates that a static route with a preference of 5 is preferred over an OSPF internal route with a preference of 10:

```
user@switch> show route 192.168.36.1 exact

inet.0: 5 destinations, 6 routes (5 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

192.168.36.1/32    *[Static/5] 00:00:31
                    > to 10.1.1.2 via ge-0/0/10.0
                    [OSPF/10] 00:02:21, metric 1
                    > to 10.1.1.2 via ge-0/0/10.0
```

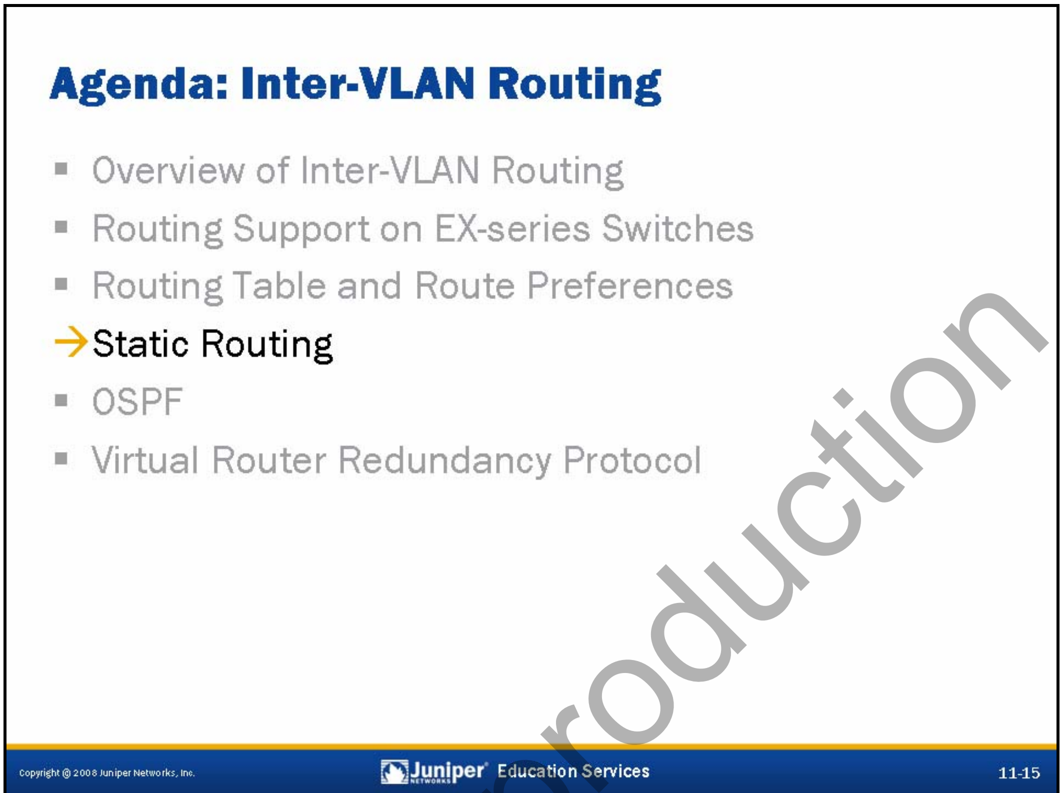


Viewing the Route Table

This slide shows the use of the **show route** command, which displays all route entries in the routing table. As identified on the slide, all active routes are marked with an asterisk (*) next to the selected entry. Each route entry displays the source from which it was learned along with the route preference for that source.

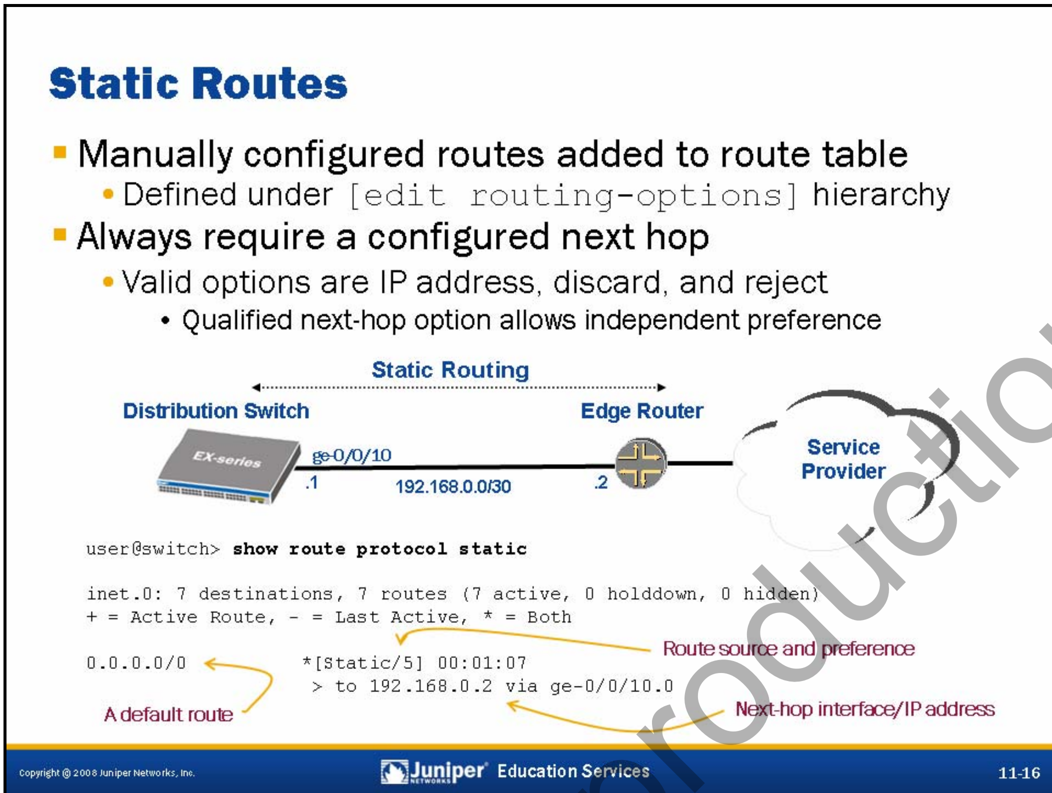
You can filter the generated output by destination prefix, protocol type, and other distinguishing attributes. The following sample capture illustrates the use of the protocol filtering option:

```
user@switch> show route protocol ospf

inet.0: 6 destinations, 7 routes (6 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.1.1.0/24         [OSPF/10] 04:57:41, metric 2
                     > to 172.18.25.2 via ge-0/0/13.0
224.0.0.5/32       *[OSPF/10] 05:00:58, metric 1
                       MultiRecv
```



Static Routing

The slide highlights the topic we discuss next.



Static Routes

Static routes are used in a networking environment for multiple purposes, including a default route for the autonomous system (AS) and as routes to customer networks. Unlike dynamic routing protocols, the routing information provided by static routes is manually configured on each router or multilayer switch in the network. All configuration for static routes occurs at the `[edit routing-options]` level of the hierarchy.

Next Hop Required

Static routes must have a valid next-hop defined. Often that next-hop value is the IP address of the neighboring router headed toward the ultimate destination. Another possibility is that the next-hop value is the *bit bucket*. This phrase is analogous to dropping the packet off the network. Within JUNOS software, the way to represent the dropping of packets is with the keywords **`reject`** or **`discard`**. Both options drop the packet from the network. The difference between them is in the action the EX-series switch takes after the drop action. If you specify **`reject`** as the next-hop value, the switch sends an ICMP message (that is, the network unreachable message) back to the source of the IP packet. If you specify **`discard`** as the next-hop value, the switch does not send back an ICMP message; the switch drops the packet silently.

Continued on next page.

Next Hop Required (contd.)

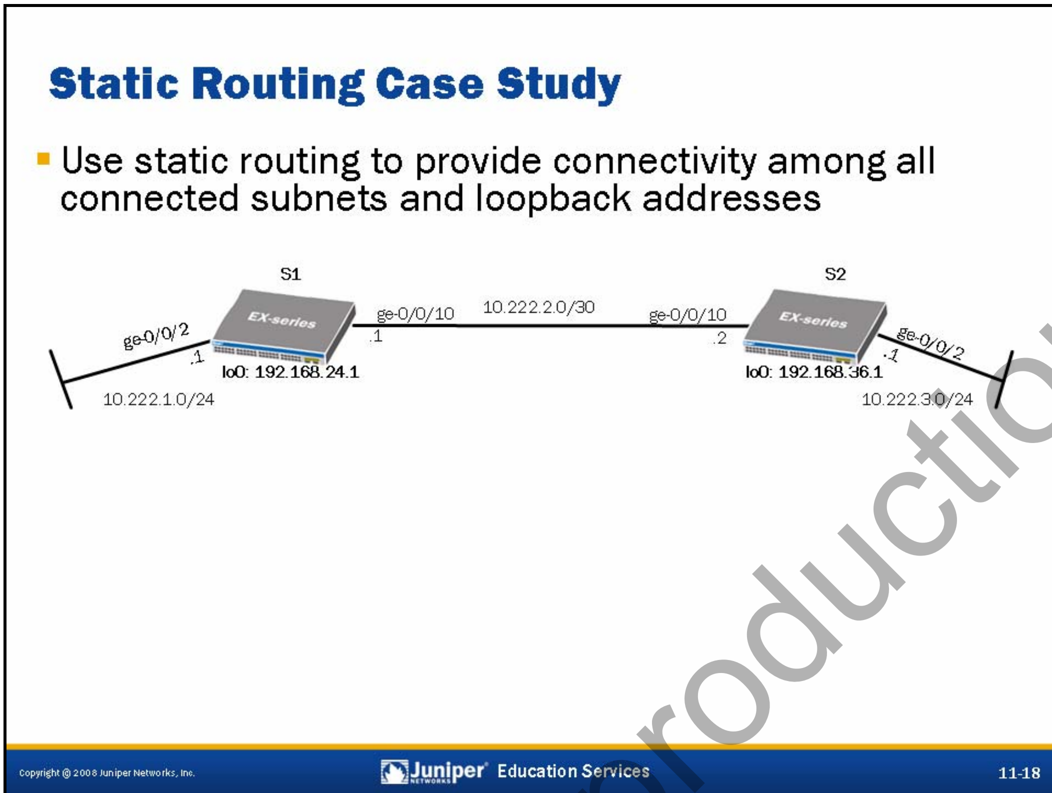
The **qualified-next-hop** option allows independent preferences for static routes to the same destination. An example using the **qualified-next-hop** option is shown here:

```
[edit]
user@switch# show routing-options
static {
    route 0.0.0.0/0 {
        next-hop 10.1.1.1;
        qualified-next-hop 10.2.2.1 {
            preference 6;
        }
    }
}
```

In the sample configuration shown, the next hop 10.1.1.1 assumes the default static route preference of 5, whereas the qualified next hop 10.2.2.1 uses the defined route preference of 6. All traffic using this static route will use the 10.1.1.1 next hop unless it becomes unavailable. If the 10.1.1.1 next hop becomes unavailable, the 10.2.2.1 next hop will be used. Some vendors refer to this implementation as a *floating static route*.

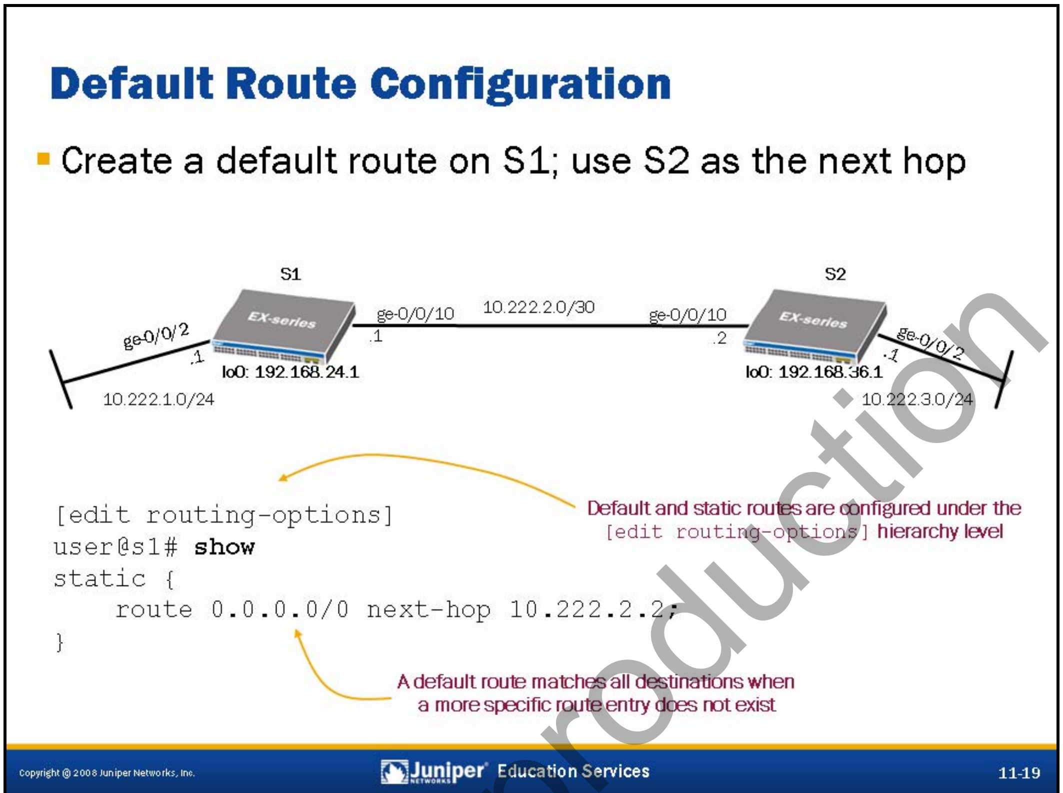
By default, the next-hop IP address of static routes configured in the JUNOS software must be reachable using a direct route. Unlike other vendors, recursive lookups of next hops are not performed by default.

Static routes remain in the routing table until you remove them or until they become inactive. One possible way for a static route to become inactive is when the IP address used as the next hop becomes unreachable.



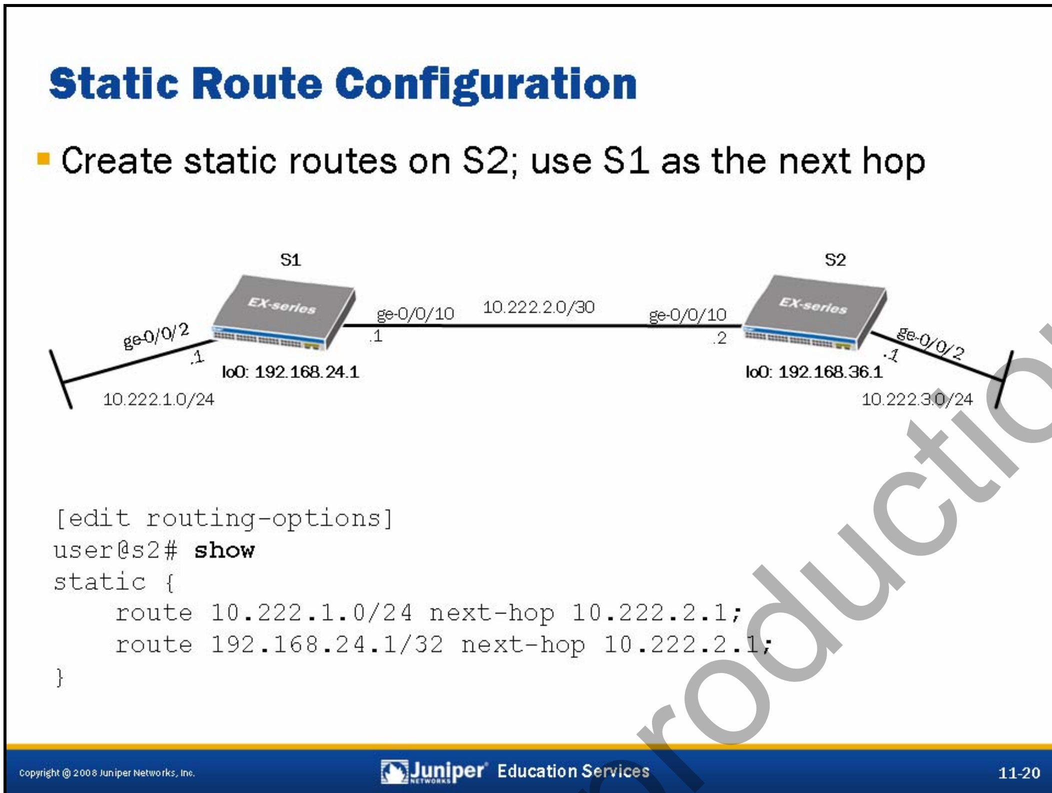
Static Routing Example

We use the topology shown on the slide to demonstrate static routing. We will configure a default static route to allow S1 to reach S2's 10.222.3.0/24 network and 192.168.36.1/32 loopback address. Likewise, we will configure S2 with static routes that allow it to reach S1's 10.222.1.0/24 network and 192.168.24.1/32 loopback address through S1.



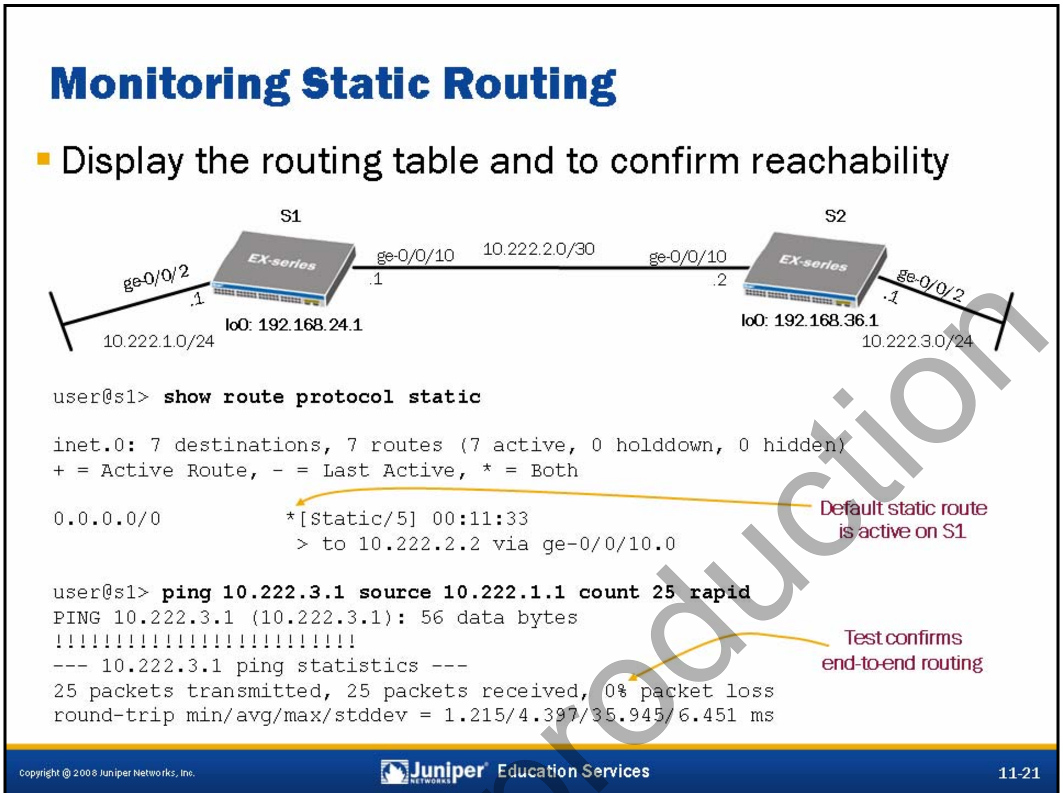
Default Route Configuration

This slide highlights the default route configuration as defined on S1.



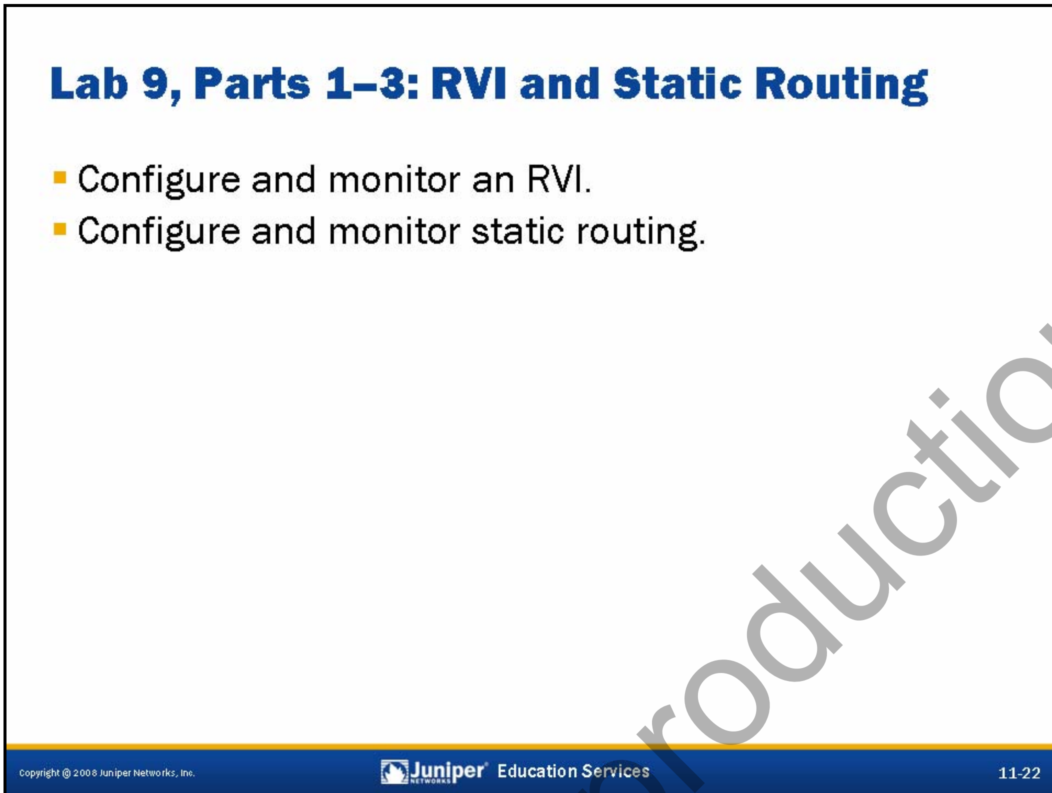
Static Route Configuration

This slide highlights the static route configuration as defined on S2.



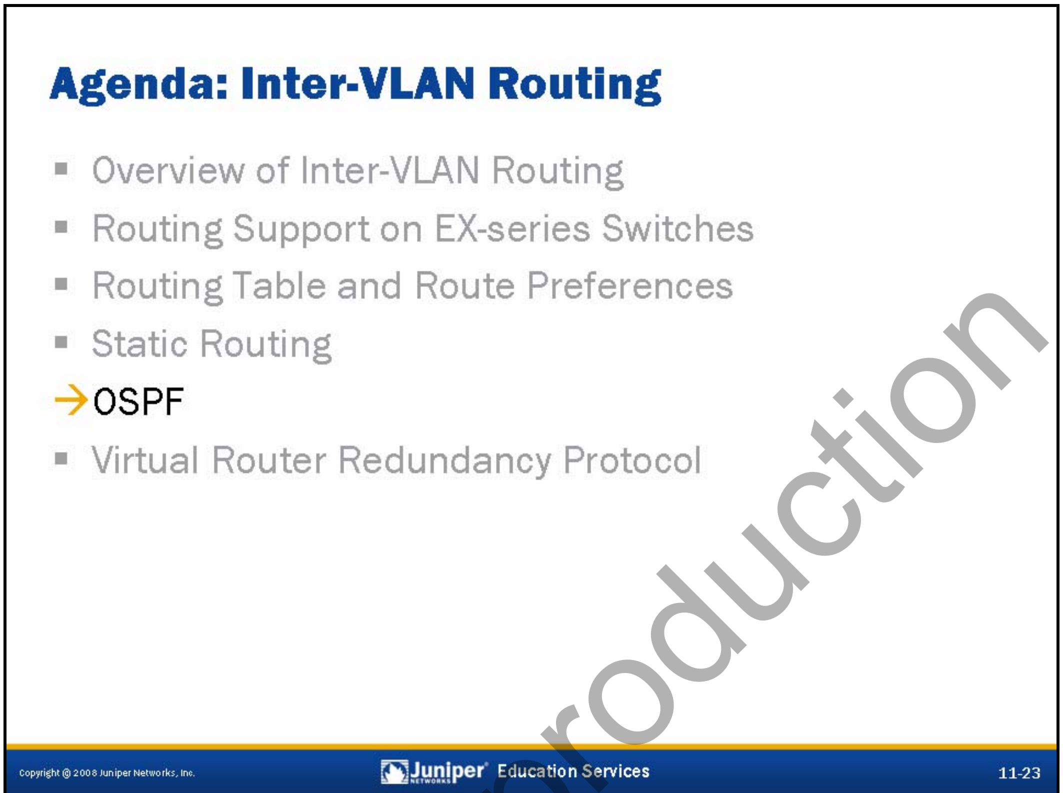
Monitoring Static Routing

This slide shows the basic verification steps when determining proper operation of static routing.



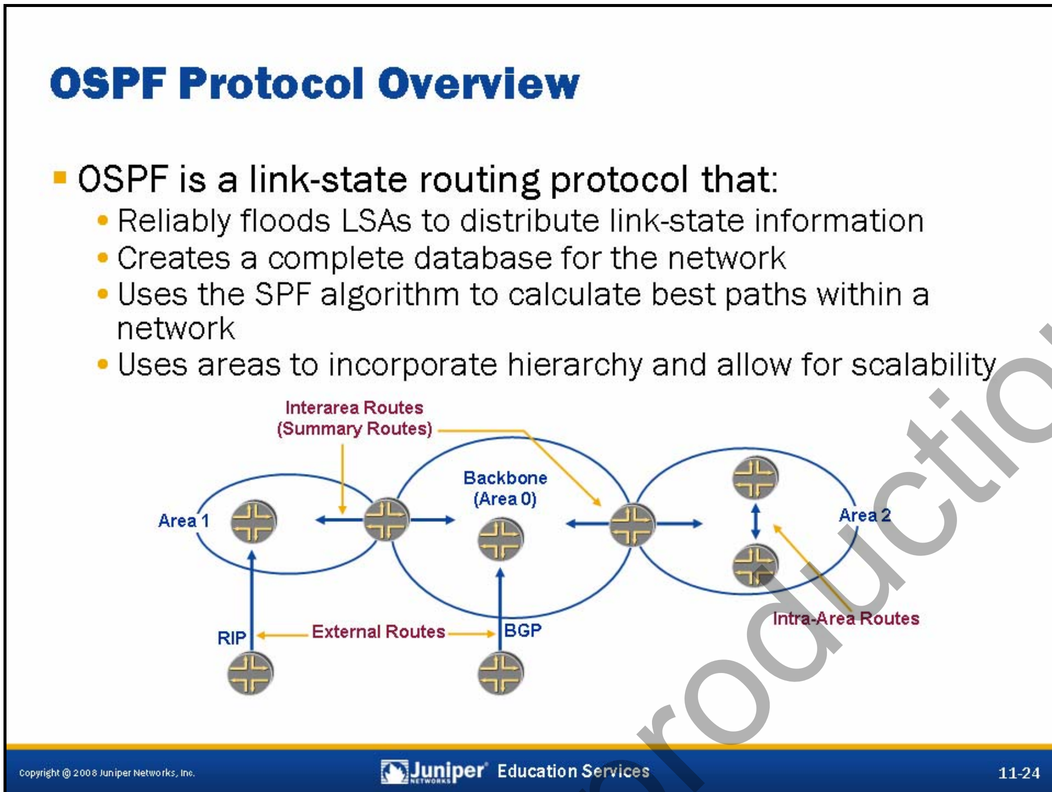
Lab 9, Parts 1–3: RVI and Static Routing

The slide provides the objective for this lab.



OSPF

The slide highlights the topic we discuss next.



OSPF Protocol

EX-series switches support OSPF version 2, which is defined in RFC 2328. OSPF is a link-state routing protocol designed for use within an AS. It is considered an interior gateway protocol (IGP). Link-state protocols allow for faster reconvergence, support larger internetworks, and are less susceptible to bad routing information than distance-vector protocols.

Routers running OSPF send out information about their network links and the state of those links to other routers in the AS. This information is transmitted reliably to all other routers in the AS by means of link-state advertisements (LSAs). The other routers receive this information, and each router stores it locally. This total set of information now contains all possible links in the network.

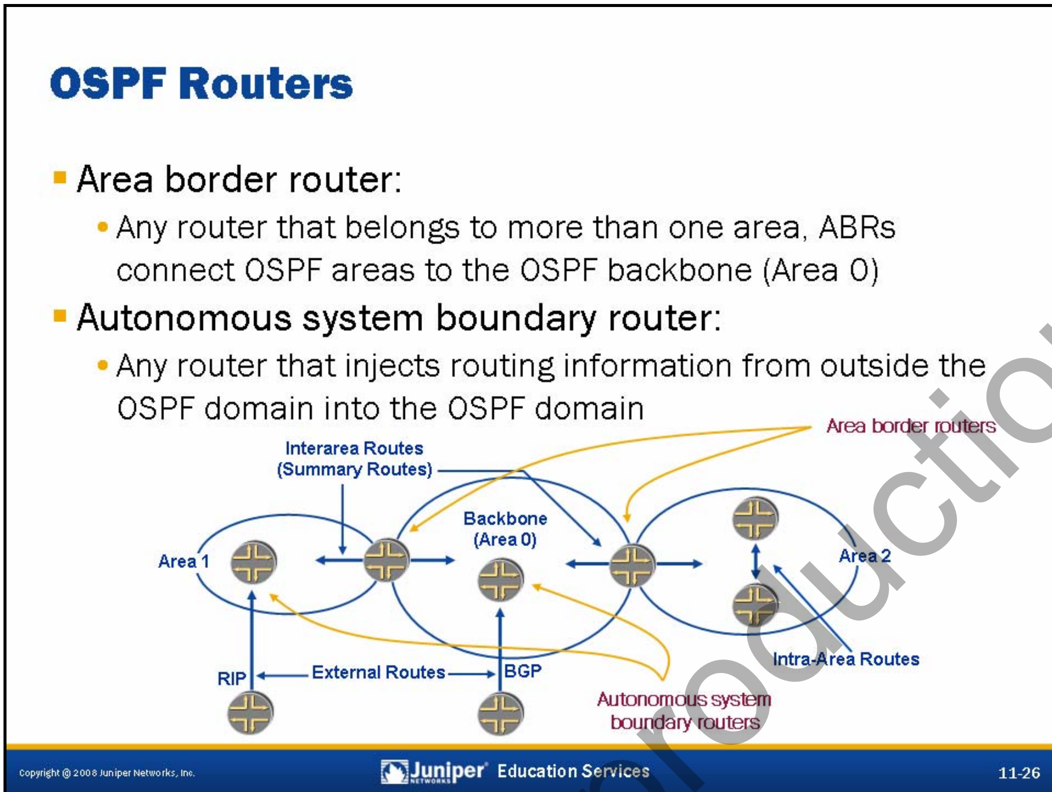
In addition to flooding LSAs and discovering neighbors, a third major task of the link-state routing protocol is establishing the link-state database. The link-state (or topological) database stores the LSAs as a series of records. The important information for the shortest-path determination process is the advertising router's ID, its attached networks and neighboring routers, and the cost associated with those networks or neighbors.

Continued on next page.

OSPF Protocol (contd.)

OSPF uses the shortest-path-first (SPF) algorithm (also called the Dyjkstra algorithm) to calculate all at once the shortest paths to all destinations. It does this calculation by calculating a tree of shortest paths incrementally and picking the best candidate from that tree.

OSPF uses areas to allow for a hierarchical organization and facilitate scalability. An OSPF area is a logical group of routers, or EX-series switches, whose routing information might be summarized and passed to the rest of the network. We discuss OSPF areas in more detail on a subsequent slide.

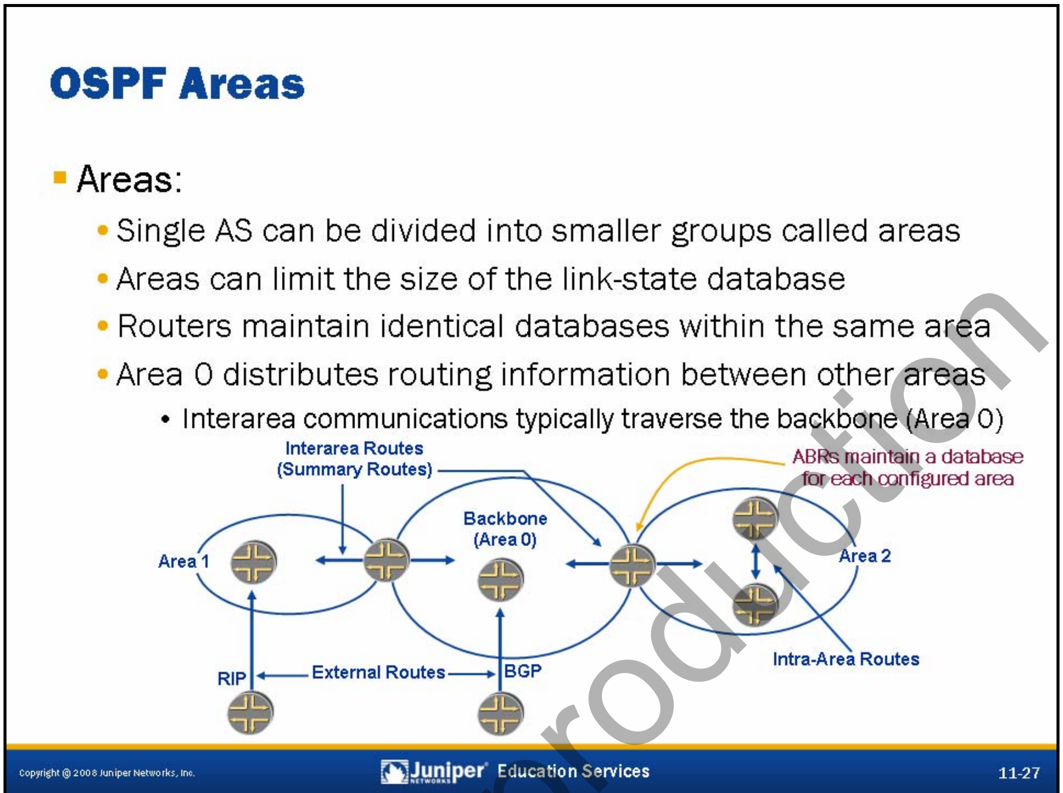


Area Border Routers

An OSPF router with links in two areas is called an *area border router* (ABR). The ABR is responsible for connecting OSPF areas to the backbone. It transmits network information between the backbone and the other areas.

Autonomous System Boundary Routers

An OSPF router that injects routing information from outside the OSPF routing domain is known as an *autonomous system boundary router* (ASBR). Typically, an ASBR is located in the backbone, but the OSPF specification allows an ASBR in other areas as well.



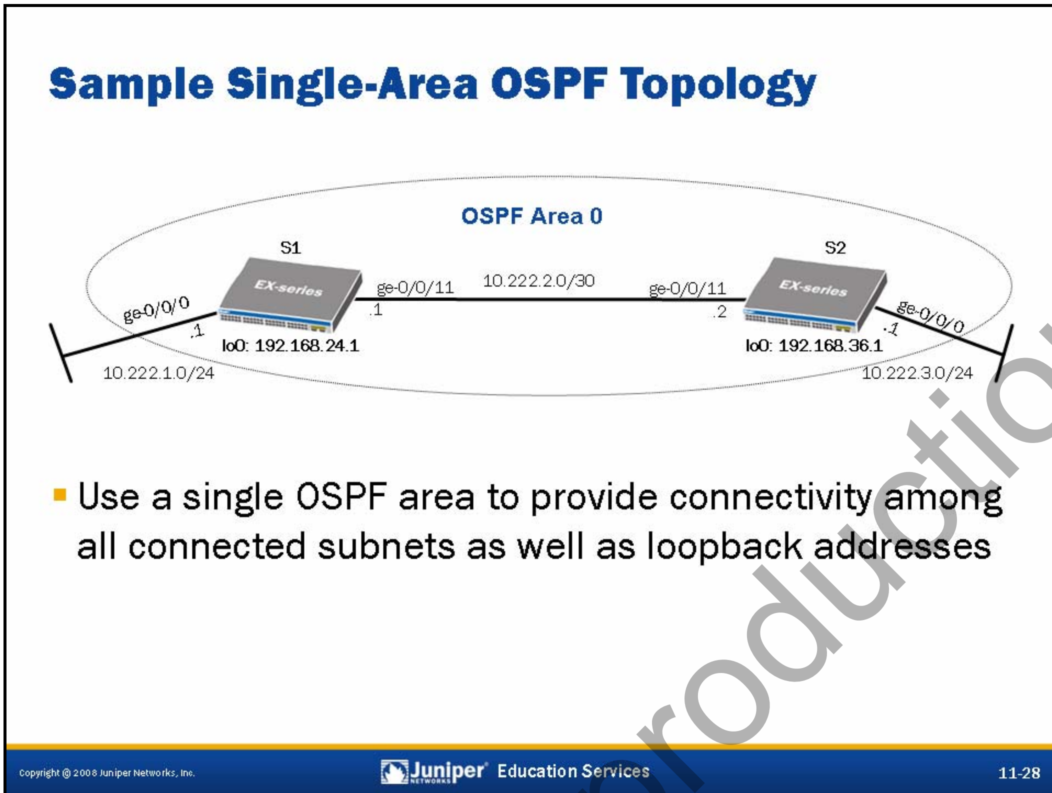
OSPF Areas

Using areas achieves the OSPF hierarchy. As mentioned on the slide, areas can reduce the size of the link-state database on an individual router. Each OSPF router maintains a separate link-state database for each area to which it is connected. The link-state database for a given area will be identical for all participating routers within that area.

To ensure correct routing knowledge and connectivity, OSPF maintains a special area called the *backbone* area. The backbone area is designated as Area 0. All other OSPF areas must connect themselves to the backbone for connectivity. All data traffic between OSPF areas must transit the backbone.

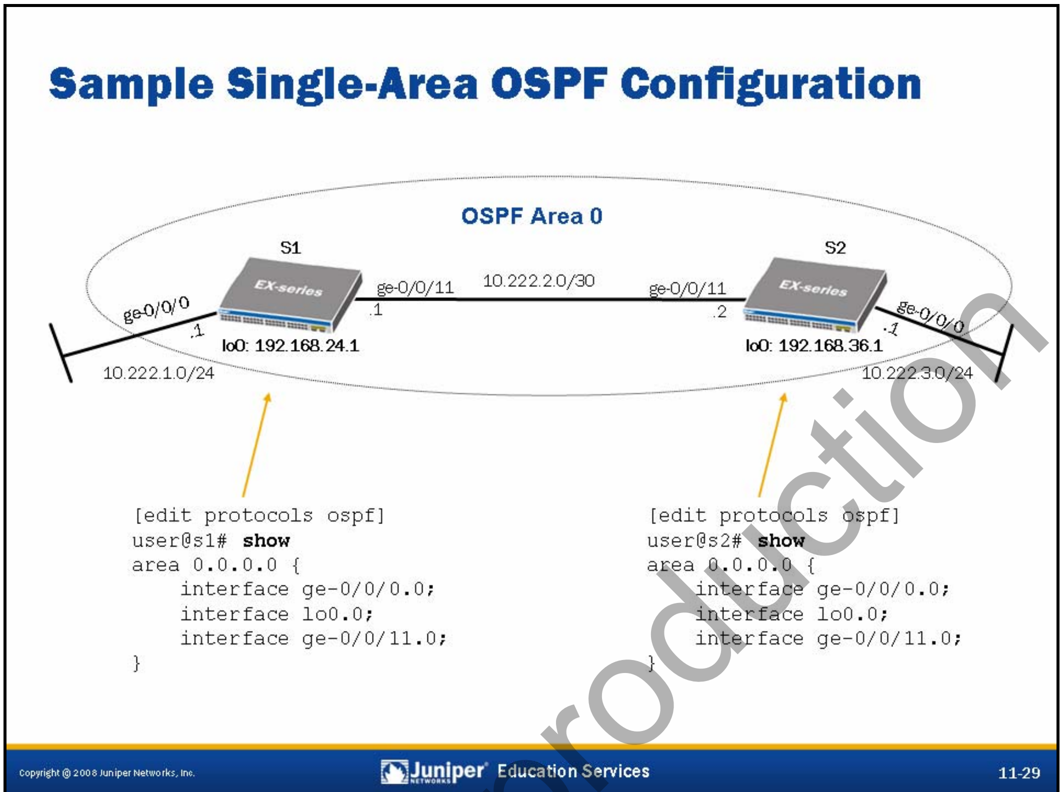
This slide highlights the relationships between OSPF areas and illustrates how route information is exchanged within and between areas. OSPF classifies different types of routing information as follows:

- *Intra-area* (or *internal*) routes: Routes that are generated from within an area, where the destination belongs to the area.
- *Interarea* (or *summary)* routes: Routes that originate from other areas.
- *External routes*: Routes that originate from other routing protocols, or different OSPF processes, and that are injected into OSPF through redistribution.



Single-Area OSPF Topology Example

We use the topology shown on the slide to demonstrate single-area OSPF routing. We enable OSPF on the LAN and loopback interfaces of both S1 and S2. This setup provides connectivity between all networks shown in the topology.



Single-Area OSPF Configuration Example

This slide illustrates the required configuration for S1 and S2 to accommodate the previously mentioned routing objective.

Monitoring OSPF (1 of 3)

- Use the **show ospf neighbor** command to display adjacencies
  - Use the **detail** or **extensive** keyword for added information

```
user@s1> show ospf neighbor
Address          Interface              State     ID               Pri  Dead
10.222.2.2       ge-0/0/11.0            Full      192.168.36.1     128    36
```

- Use the **clear ospf neighbor** command to clear adjacencies
  - Specify individual neighbors or clear all neighbor adjacencies

```
user@s1> clear ospf neighbor ?
Possible completions:
  <[Enter]>            Execute this command
  <neighbor>           Name of neighbor
  |                    Pipe through a command
user@s1> clear ospf neighbor
```

 Juniper Education Services 11-30

Displaying Adjacency Information

The **show ospf neighbor** command displays OSPF adjacency status. The output includes the following fields:

- `Address`: Displays the address of the neighbor.
- `Intf`: Displays the interface through which the neighbor is reachable.
- `State`: Displays the state of the neighbor, which can be `Down`, `Attempt`, `Init`, `2-Way`, `Exstart`, `Exchange`, `Loading`, and `Full`.
- `ID`: Displays the router ID (RID) of the neighbor.
- `Pri`: Displays the priority of the neighbor to become the designated router (DR).
- `Dead`: Displays the number of seconds until the neighbor relationship times out.
- `area` (detailed and extensive output only): Displays the area in which the neighbor is located.
- `opt` (detailed and extensive output only): Displays the option bits from the neighbor.
- `DR` (detailed and extensive output only): Displays the address of the DR.
- `BDR` (detailed and extensive output only): Displays the address of the backup designated router (BDR).

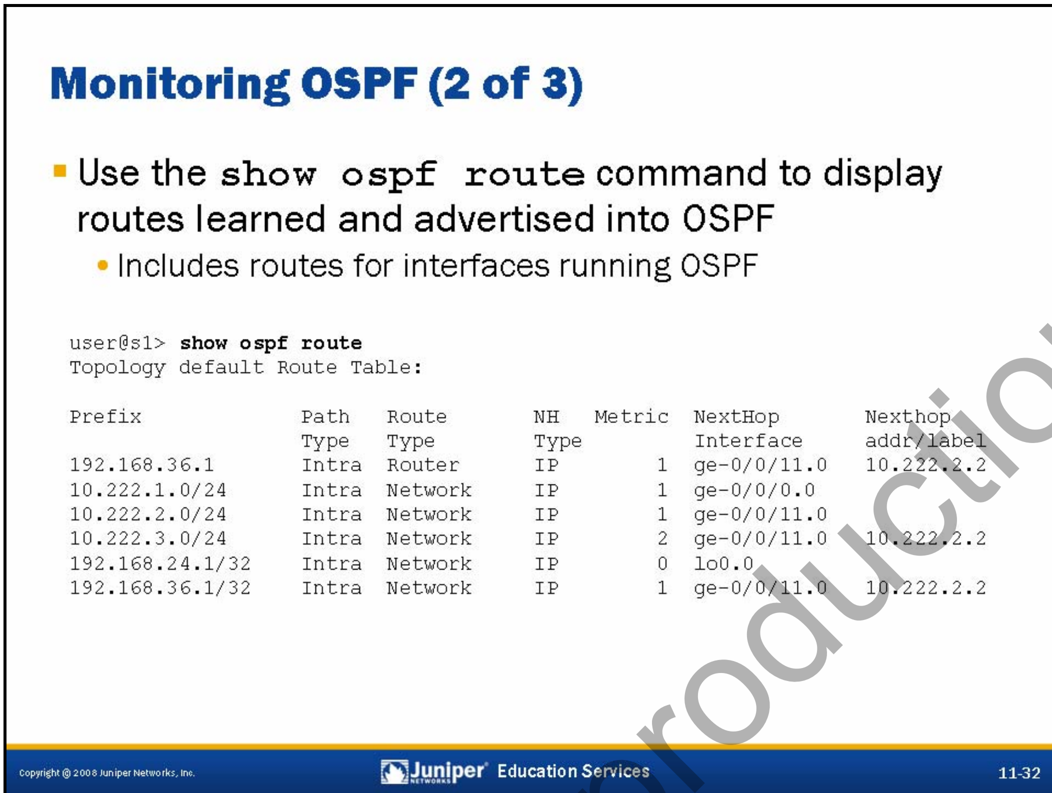
Continued on next page.

Displaying Adjacency Information (contd.)

- `Up` (detailed and extensive output only): Displays the length of time since the neighbor came up.
- `adjacent` (detailed and extensive output only): Displays the length of time since the adjacency with the neighbor was established.

Clearing Adjacencies

Use the **`clear ospf neighbor`** command to clear an OSPF adjacency. Note that in most cases the adjacency will be reformed immediately.



Displaying OSPF Route Information

The **`show ospf route`** command displays those routes in the unicast routing table, `inet.0`, that were installed by OSPF. The use of additional keywords, such as **`abr`**, allows you to display only OSPF routes learned by specific LSA types. The output of the **`show ospf route`** command includes the following fields:

- `Prefix`: Displays the destination of the route.
- `Path Type` and `Route Type`: Display how the route was learned:
  - `ABR`: Route to ABR;
  - `ASBR`: Route to ASBR;
  - `Ext`: External router;
  - `Inter`: Interarea route;
  - `Intra`: Intra-area route; or
  - `Network`: Network route.
- `Metric`: Displays the route's metric value.
- `NextHop Interface`: Displays the interface through which the route's next hop is reachable.

Continued on next page.

Displaying OSPF Route Information (contd.)

- `Nexthop addr/label`: Displays the address of the next hop.
- `area` (detailed output only): Displays the area ID of the route.
- `options` (detailed output only): Displays the option bits from the LSA.
- `origin` (detailed output only): Displays the router from which the route was learned.

Monitoring OSPF (3 of 3)

- Use the `show ospf database` command to display link-state database entries

```
user@s1> show ospf database

    OSPF link state database, Area 0.0.0.0
 Type       ID               Adv Rtr           Seq      Age  Opt  Cksum  Len
Router   *192.168.24.1     192.168.24.1     0x8000000e  1270  0x22 0xedcc   60
Router    192.168.36.1     192.168.36.1     0x8000000d  1271  0x22 0xd0c3   60
Network   10.222.2.2       192.168.36.1     0x8000000a  1271  0x22 0xb0f3   32
```

Self-originated LSAs marked with *

- Use the `clear ospf database` command to clear the link-state database

```
user@s1> clear ospf database
```

 Juniper Education Services 11-34

Displaying OSPF Database Entries

The **`show ospf database`** command displays entries in the protocol's link-state database. The display is organized by LSA types. The **`show ospf database`** command includes the following optional keywords:

- **`brief`**: Displays a brief listing of all entries in the OSPF link-state database. This setting is the default.
- **`detail`**: Displays detailed information about the entries in the OSPF link-state database.
- **`extensive`**: Displays extremely detailed information about the entries in the OSPF link-state database.

Continued on next page.

Displaying OSPF Database Entries (contd.)

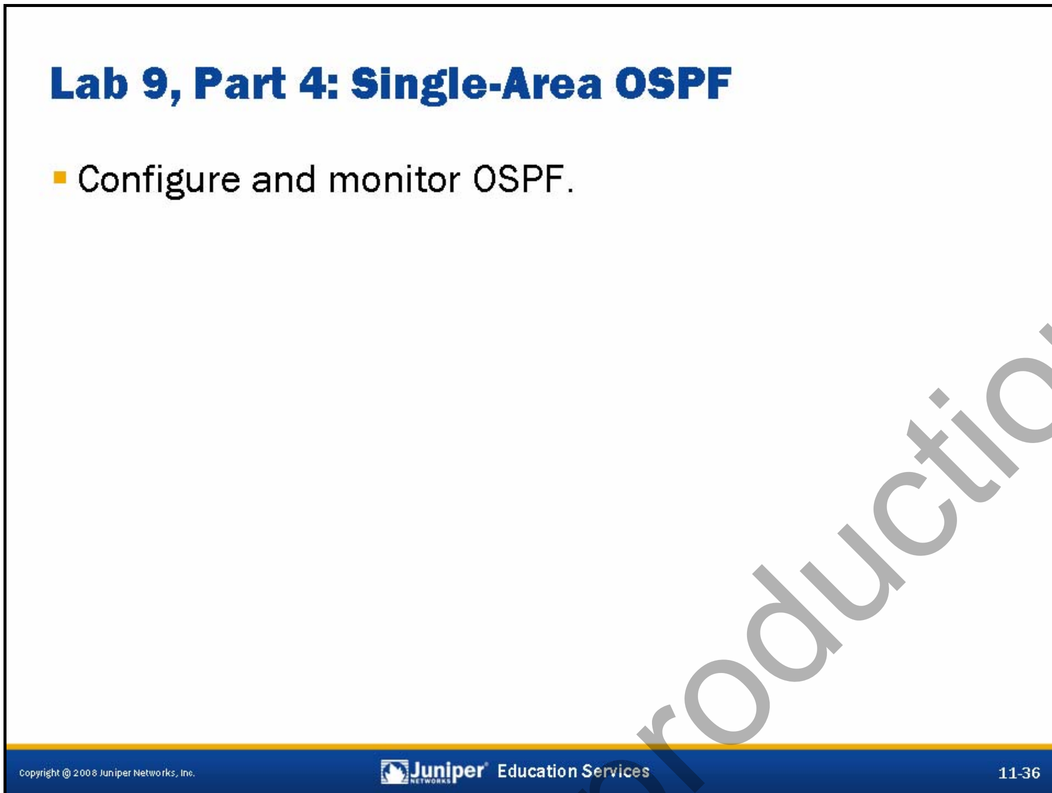
- **LSA filters**: Displays one or more of the following LSA filters. If you specify more than one filter, only LSAs that match all the filters are displayed. For example, the **show ospf database detail router lsa-id 10.0.0.1** command displays all router LSAs in all areas that have an LSA identifier of 10.0.0.1. The filters are the following:
  - `advertising-router address`: Displays the LSAs advertised by a particular router.
  - `area` *area-id*: Displays the LSAs in a particular area.
  - `lsa-id` *lsa-id* (optional): Displays the LSA with the specified LSA identifier.
  - `lsa-type`: Displays specific types of LSAs. You can specify `asbrsummary`, `extern`, `netsummary`, `network`, `nssa`, or `router`.
  - `summary` (optional): Displays summary information about the OSPF link-state database.

Clearing Database Entries

The **clear ospf database** command clears entries from the link-state database. After the command is entered, the router begins the database synchronization process with its neighboring routers such that, in most cases, the database returns to its prior state.

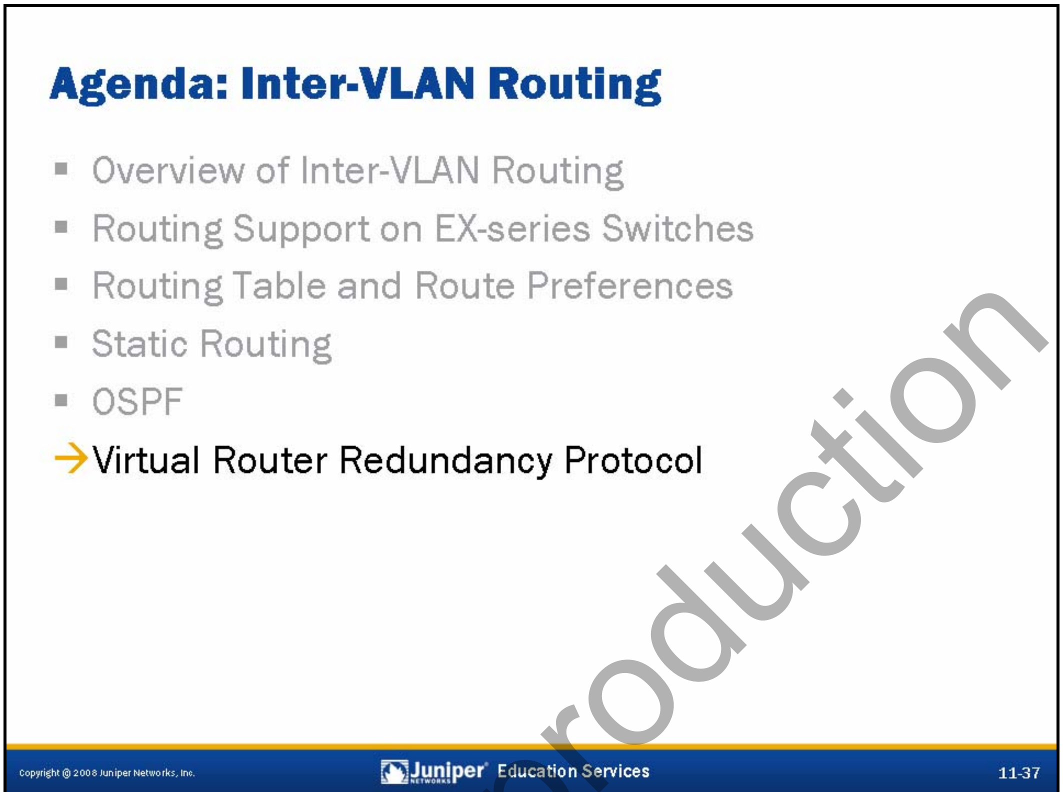
The **clear ospf database** command supports an optional **purge** switch. By including the **purge** switch, you force the local router to set *all* LSAs in its database to the maximum age. These LSAs are then reflooded according to the OSPF specification, which states that a router must regenerate any LSA that it has set to maximum age, regardless of whether the LSA was generated by the local router. All routers receive the newly flooded maximum age LSAs; the router that originated a given LSA is forced to refresh that LSA when it receives a copy of that LSA with an indication that it has reached the maximum age.

Albeit somewhat disruptive, this procedure tends to eliminate stale or bogus database entries without having to wait for the normal aging-out process, which can take as long as 3600 seconds (one hour). Note that other vendors' OSPF implementations might not be prepared for a simultaneous reflooding of every LSA in the network or for another router to increase the age of LSAs that their routers originated. Therefore, you should not use this feature in a production network without prior interoperability testing.



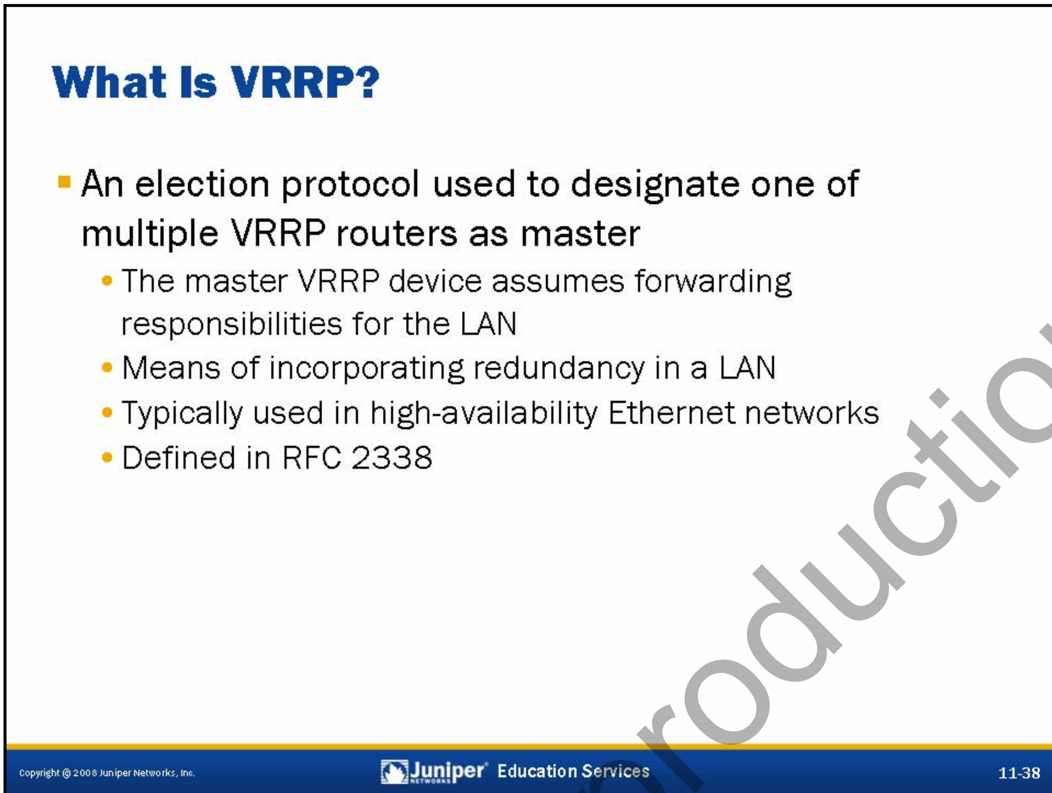
Lab 9, Part 4: Single-Area OSPF

The slide provides the objective for this lab.



Virtual Router Redundancy Protocol

The slide highlights the topic we discuss next.



VRRP Defined

The Virtual Router Redundancy Protocol (VRRP) is a standards-based election protocol created to facilitate redundancy in a LAN environment and eliminate any single point of failure typically found in LAN environments that use static default routing as a means of routing beyond the local subnet. VRRP identifies one router to function as the master VRRP router; thus, that router represents the gateway for that particular subnet and performs the required forwarding for the end hosts on that subnet. All other routers that could potentially assume the role of the master VRRP router for the subnet are known as backup VRRP routers. VRRP is very similar in functionality to Cisco Systems' Hot Standby Router Protocol (HSRP). VRRP is most commonly found in Ethernet environments but can also be used in LAN environments that use Token Ring or Fiber Distributed Data Interface (FDDI). VRRP is an industry standard and is defined in RFC 2338.

VRRP Terminology

- *Virtual router*—Virtual entity that functions as the default router on a LAN; consists of a VRID and an IP address used as a gateway address known as the VIP address
- *VRRP router*—Any router participating in VRRP, including the master and all backup routers
- *Master router*—VRRP router performing packet forwarding and responding to ARP requests
- *Backup router*—VRRP router available to assume the role of the master router upon failure

 Juniper Education Services 11-39

Virtual Router

The virtual router is a logical entity that functions as the default router on a LAN segment or network. The virtual router consists of a virtual router identifier (VRID) and a virtual IP (VIP) address. The VRID uniquely identifies one virtual router from another. The VIP address is managed by the virtual router and is attached to the VRRP router functioning as the master for that network.

VRRP Router

A VRRP router is any router participating in VRRP, including the master and backup routers. A VRRP router might belong to more than one virtual router group.

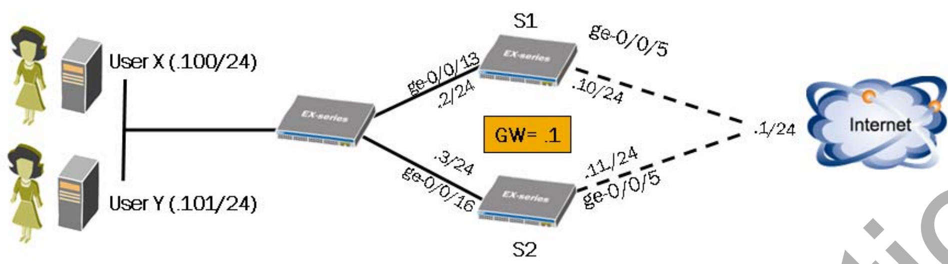
Master Router

The master router is a VRRP router that owns the responsibility of forwarding packets on a given LAN segment. The master router also performs the Address Resolution Protocol (ARP) functions for the virtual router that it represents. The master router selection is deterministic and is typically based on a user-defined priority.

Backup Router

The backup router is a VRRP router that is available to perform all the responsibilities associated with the master router in the event the master router fails.

VRRP Case Study: Sample Topology

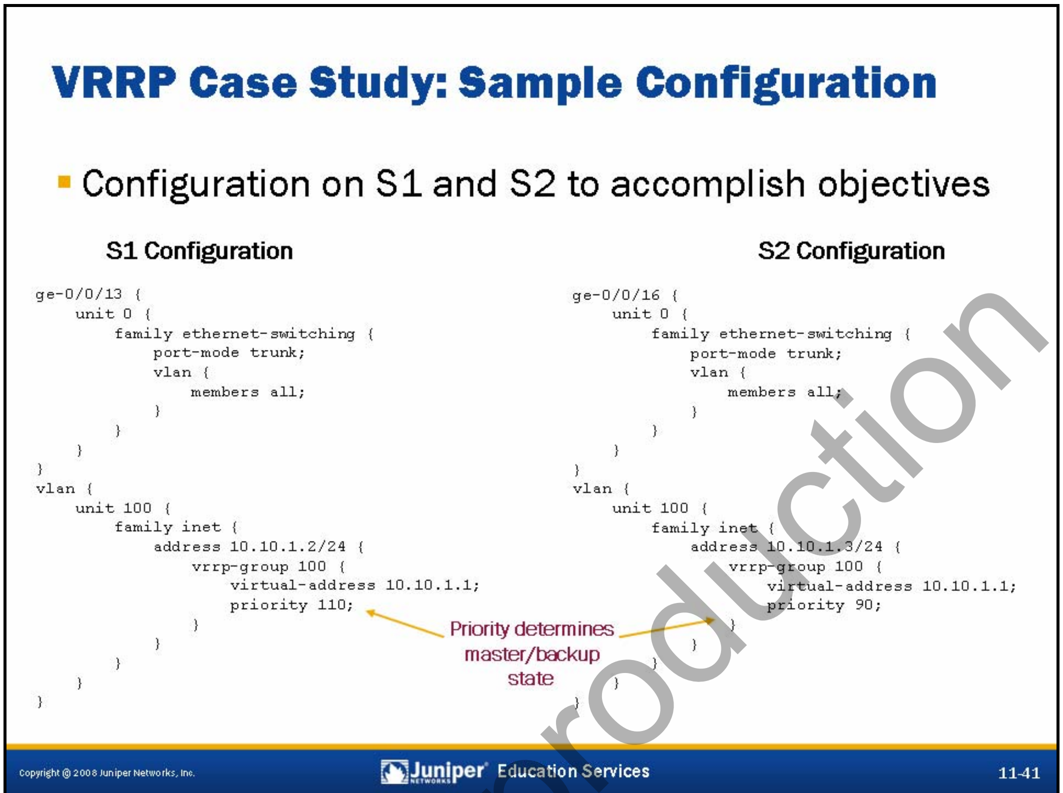


- Goals:
  - Provide a single gateway address on both S1 and S2 for the 10.10.1.0/24 subnet; this common gateway address should be bound to a Layer 3 VLAN interface for both S1 and S2
  - Use VRRP to provide redundancy during failure scenarios; S1 should function as the master during normal operations

 Juniper Education Services 11-40

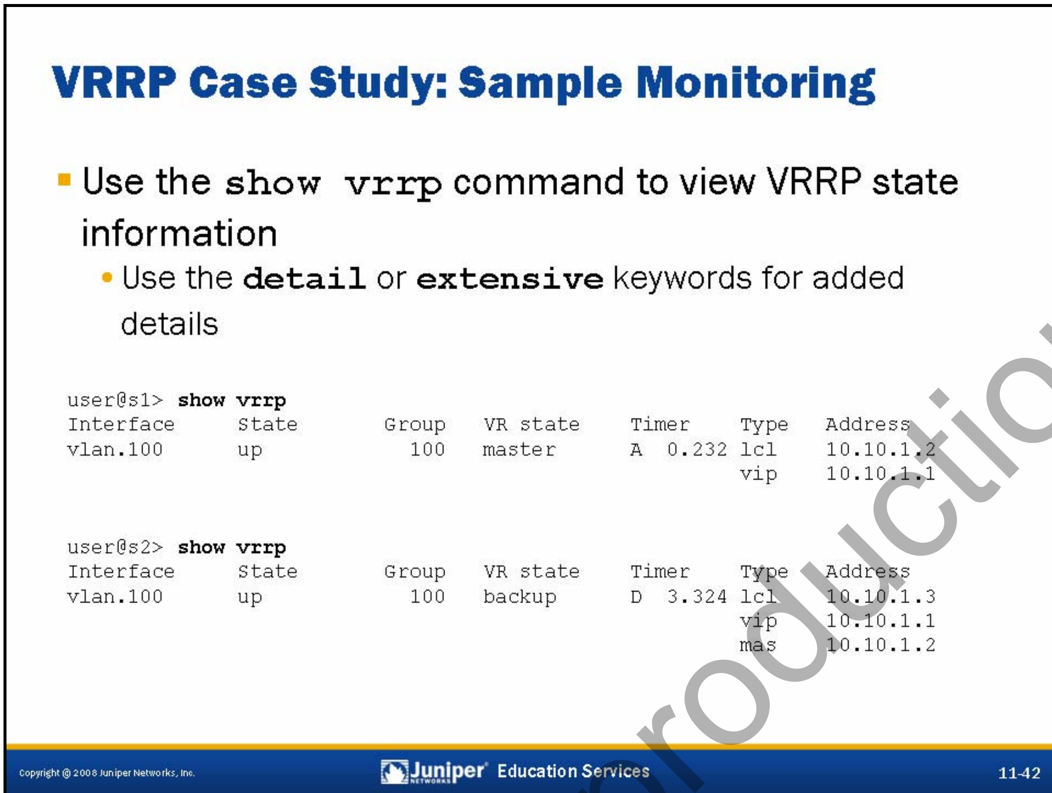
Sample VRRP Topology

This slide illustrates a sample topology and defines objectives for our VRRP case study.



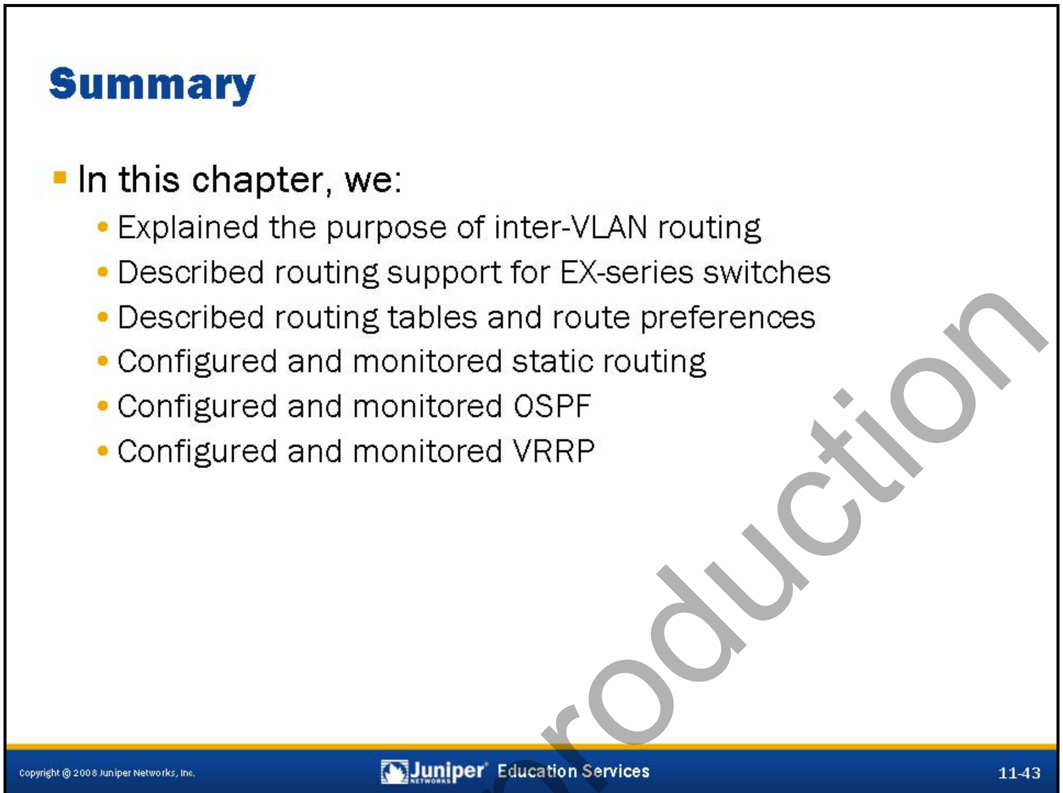
Sample VRRP Configuration

This slide provides the recommended configuration for both S1 and S2 to accomplish the previously stated objectives. Based on the proposed configuration, S1 functions as the master router, whereas S2 assumes the backup router role for VRRP group 100. In the event of a failure of S1, S2 assumes VRRP mastership and begins performing the forwarding functions associated with a master VRRP router functioning as a default gateway for a network. Note that the VRRP configuration is tied to the logical Layer 3 VLAN interface named vlan.100 on both switches, which was also one of the stated objectives.



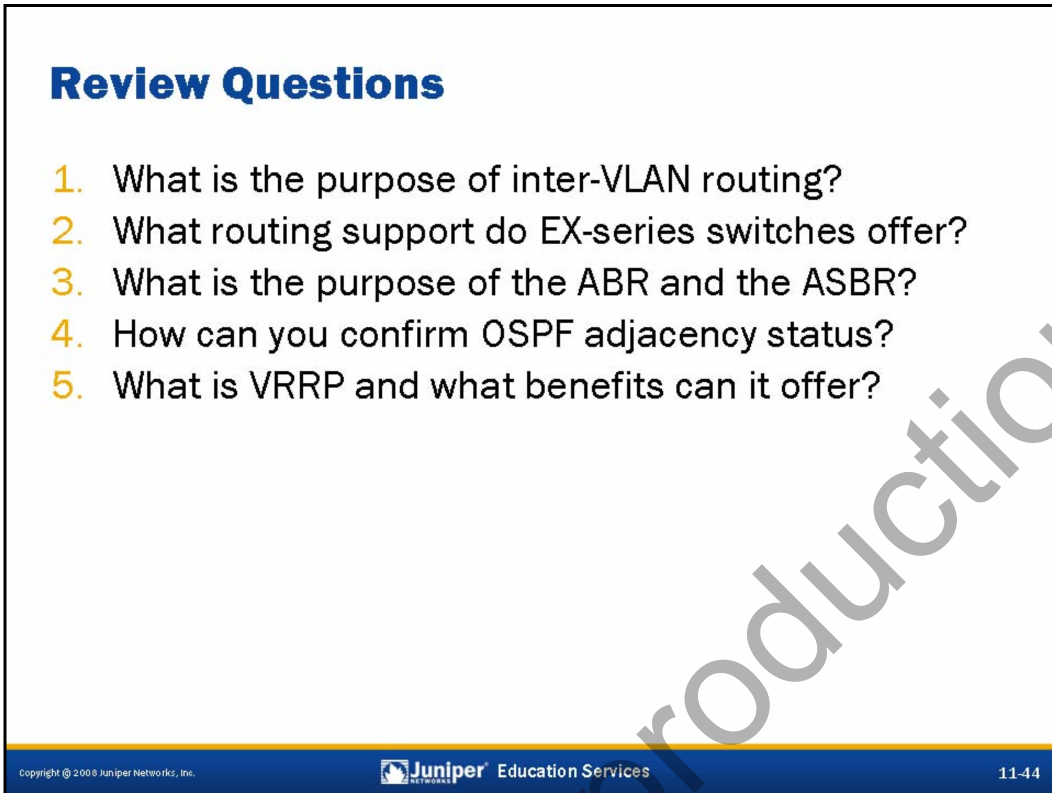
Monitoring VRRP Operation

This slide highlights the **show vrrp** command. This command is useful in identifying the VRRP state on a given router. Use the **detail** or **extensive** options to increase the amount of VRRP-related details.



This Chapter Discussed:

- The purpose of inter-VLAN routing;
- Routing support for EX-series switches;
- Route tables and route preference;
- Configuring and monitoring static routing;
- Configuring and monitoring OSPF; and
- Configuring and monitoring VRRP.



Review Questions:

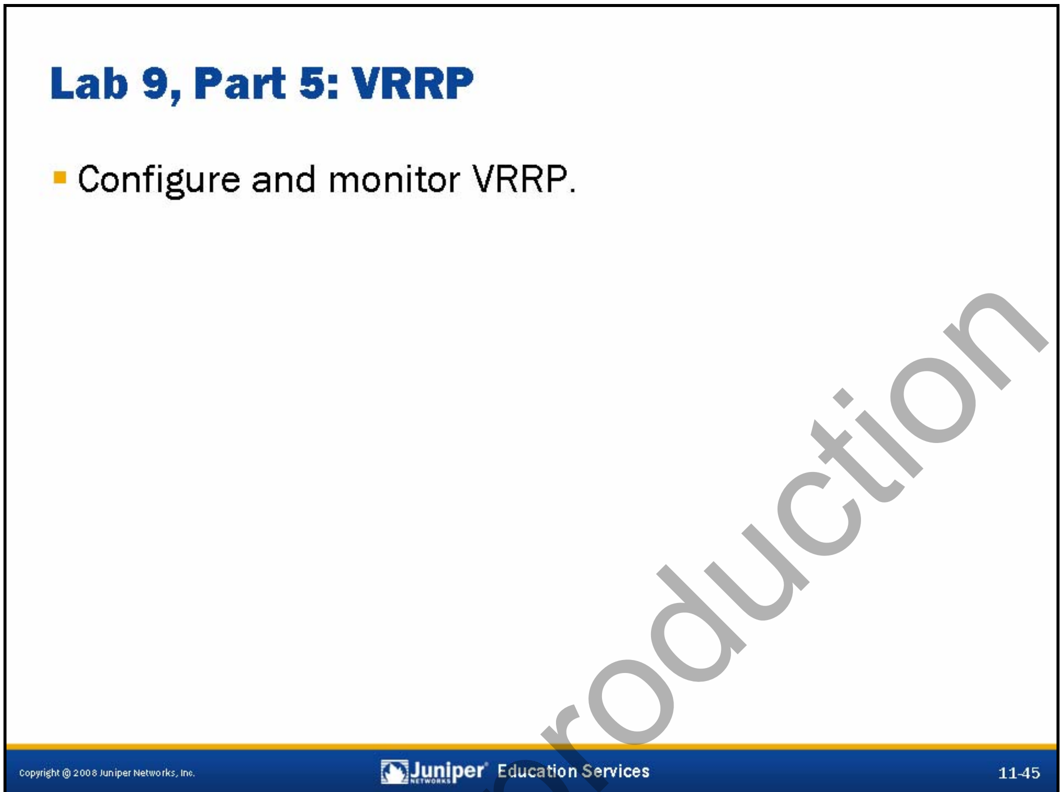
1.

2.

3.

4.

5.



Lab 9, Part 5: VRRP

The slide provides the objective for this lab.

Operating Juniper Networks Switches in the Enterprise

Chapter 12: Routing Policy and Firewall Filters

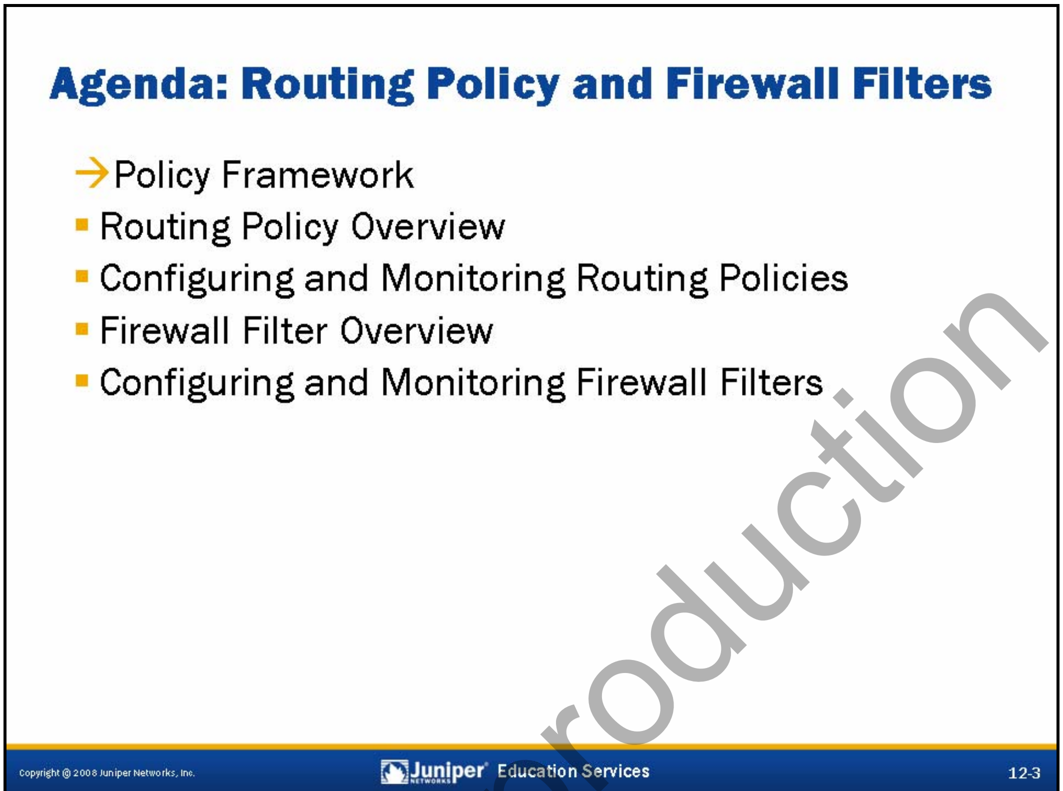
Chapter Objectives

- After successfully completing this chapter, you will be able to:
  - Describe the policy framework on EX-series switches
  - Explain how policies and firewall filters are evaluated
  - Identify common situations in which routing policy is used
  - Write and apply a routing policy
  - Describe the types of firewall filters supported on EX-series switches
  - Write and apply a firewall filter

Juniper Education Services
12-2

This Chapter Discusses:

- The policy framework on EX-series switches;
- Policy and firewall filter evaluation;
- Typical usage scenarios for routing policy;
- Configuring and applying a routing policy;
- The types of firewall filters supported on EX-series switches; and
- Configuring and applying firewall filters.



Policy Framework

The slide lists the topics we cover in this chapter. We discuss the highlighted topic first.

Policy Language Overview

- Policy language hierarchy:
  - A term is composed of a *from* statement and a *then* statement
  - A policy is composed of zero or more terms

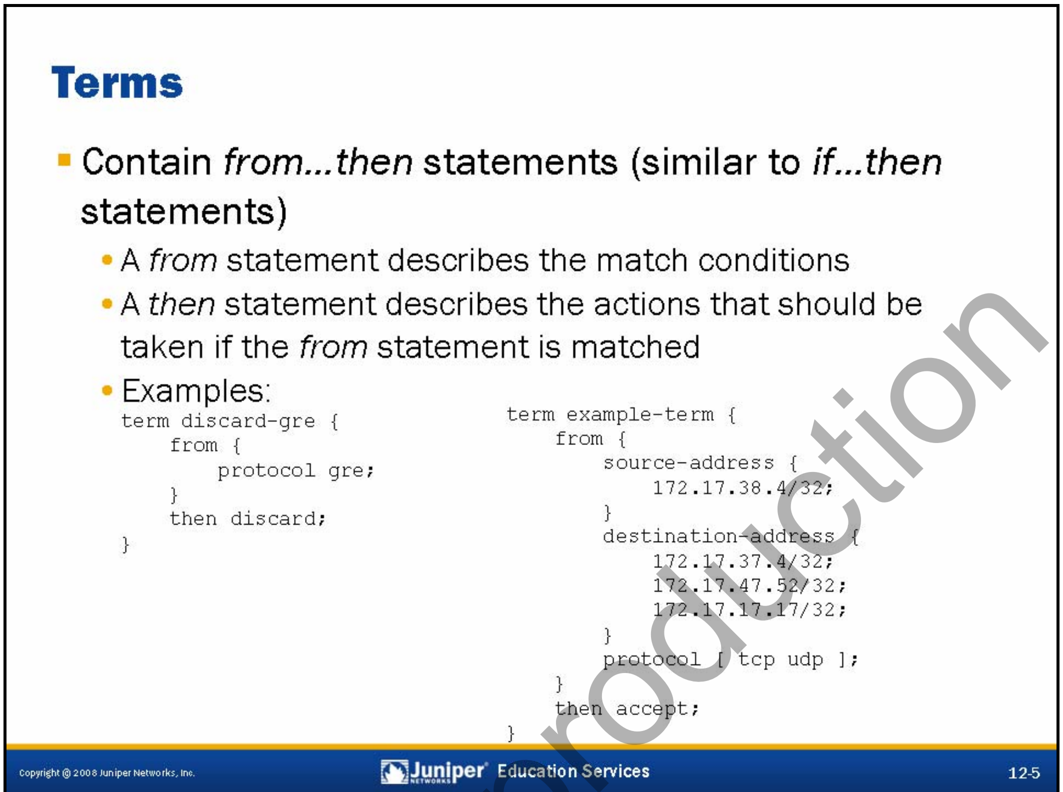
Juniper Education Services

12-4

Policy Language Hierarchy

Although routing policy and firewall filters serve different purposes and have different match and action conditions, they are all built using a common structure. Learning that common structure enables you to better understand all JUNOS software policies.

The fundamental building block of all policy evaluation is the *term*. A term contains one or more match conditions and one or more actions. If all the match conditions are true, JUNOS software takes the action. You use a policy to group together multiple terms and establish the order in which the switch evaluates the terms.



The Term

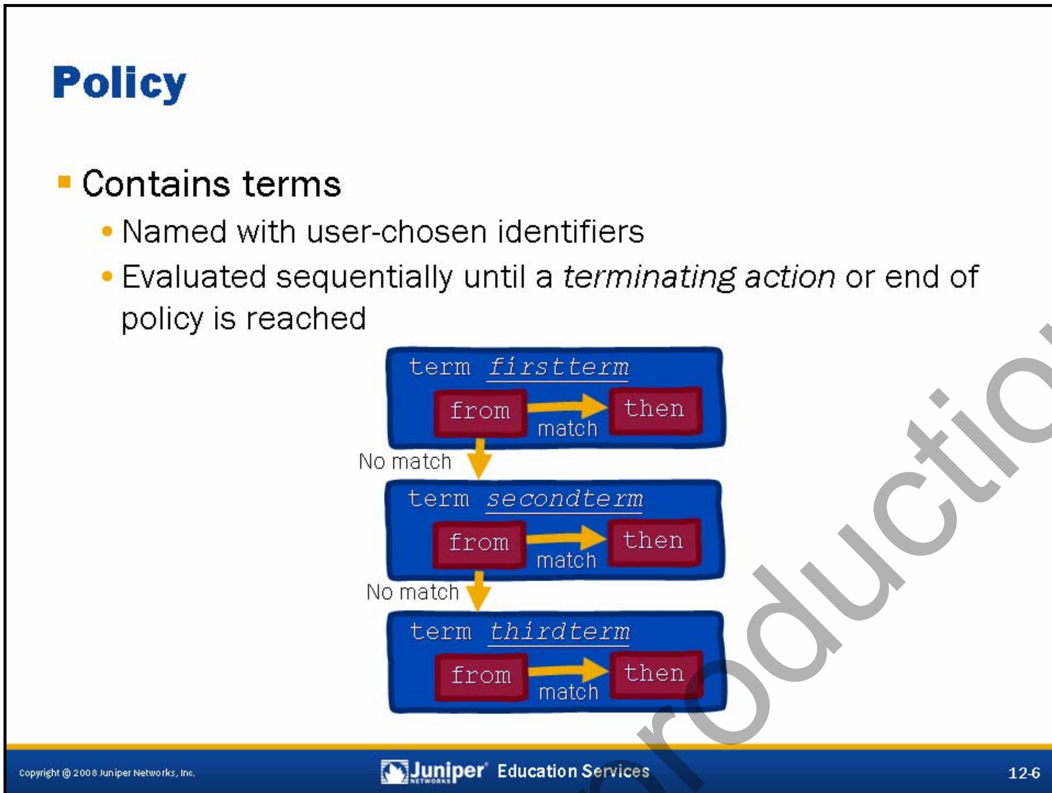
Terms are the basic building blocks of all JUNOS software policy. They are essentially *if...then* statements. If all the match conditions specified in the *from* statement are true (or if no *from* statement is specified), then all the actions in the *then* statement are executed.

You give terms a name. The name has no effect on the evaluation of the term; rather, it gives you a way to provide a meaningful identifier that you can use when referring to the term.

When evaluating the *from* statement, the switch performs the evaluation as a logical OR between arguments to a single match criterion and a logical AND between different match criteria. Put differently, for the *from* statement to be considered true, the item being evaluated must match at least one of the arguments to each match criterion given.

For example, consider the *`example-term`* from a firewall filter provided on the slide. To match this term, a packet must match the `source-address`, `destination-address`, and `protocol` conditions. To match the `source-address` condition, the packet's source address must be 172.17.38.4. However, to match the `destination-address` condition, the packet's destination address can be any of the listed addresses. Likewise, to match the `protocol` condition, the packet's protocol can be either of the listed protocols.

If all the conditions in the *from* statement are true, the switch performs all the actions in the *then* statement. In this example, a single action is given: to `accept` the packet. Thus, in our example, the term accepts a TCP packet from 172.17.38.4 to 172.17.37.4.



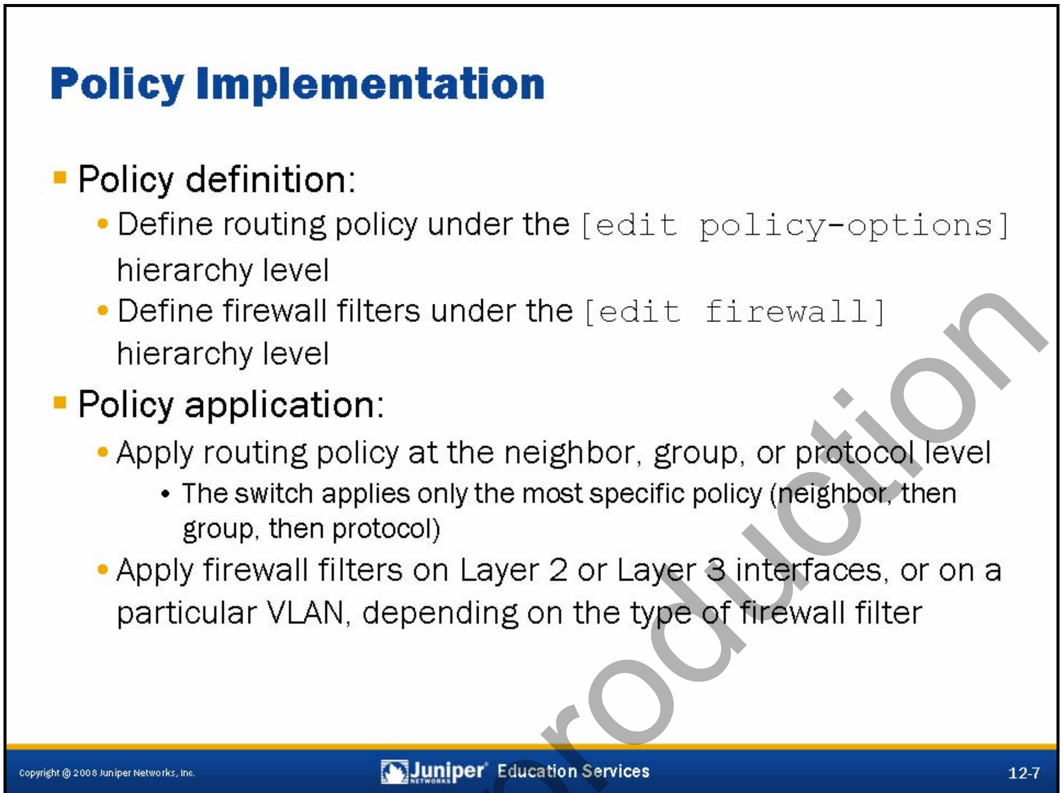
The Policy

Policies contain ordered groups of terms. You give policies a name, which you use to identify them when they are referenced elsewhere in the configuration.

When the switch evaluates a policy, the switch evaluates each term sequentially, in the order in which it appears in the policy. You can use the command-line interface (CLI) **`insert`** command in configuration mode to modify the order in which terms appear in the policy. For example, at the `[edit firewall family inet filter` *`filtername`*`]` level, typing **`insert term`** ***`secondterm`*** **`before term`** ***`firstterm`*** places the term called *`secondterm`* immediately before the term called *`firstterm`*. You can use the **`insert`** command for all ordered data elements in the configuration, including terms within policies.

When the switch evaluates a policy, it evaluates terms sequentially. If an item matches all the conditions in the *from* statement of a term, JUNOS software executes all the actions specified in the *then* statement of that term. Provided that one of those actions is a *terminating action*, the evaluation of the policy stops. Some actions (such as the **`next term`** action) are not terminating actions, and the evaluation of the policy continues. In that case, later terms might overwrite attribute changes made by earlier terms because the switch applies the actions for each matching term as the term is evaluated.

The actions that control the acceptance and rejection of items (**`accept`**, **`reject`**, and **`discard`**) are terminating actions. Using these terminating actions results in a *first-match* policy evaluation because the switch immediately executes the action and causes no further evaluation of the policy.



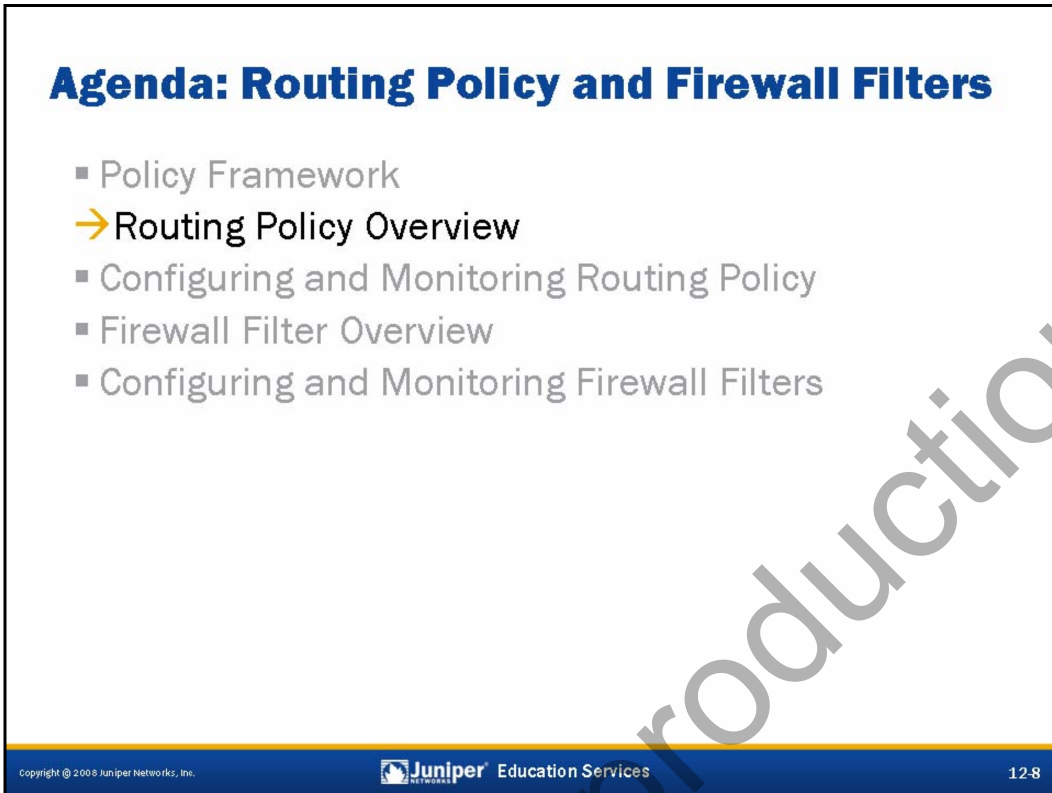
Policy Definition

Implementing policy, whether it be routing policy or firewall filters, requires two distinct steps. The first step is defining the policy. In JUNOS software, you define routing policy is defined under the `[edit policy-options]` hierarchy level and firewall filters under the `[edit firewall]` hierarchy level. We view some examples of the hierarchy structure for both routing policy and firewall filters in subsequent pages.

Policy Application

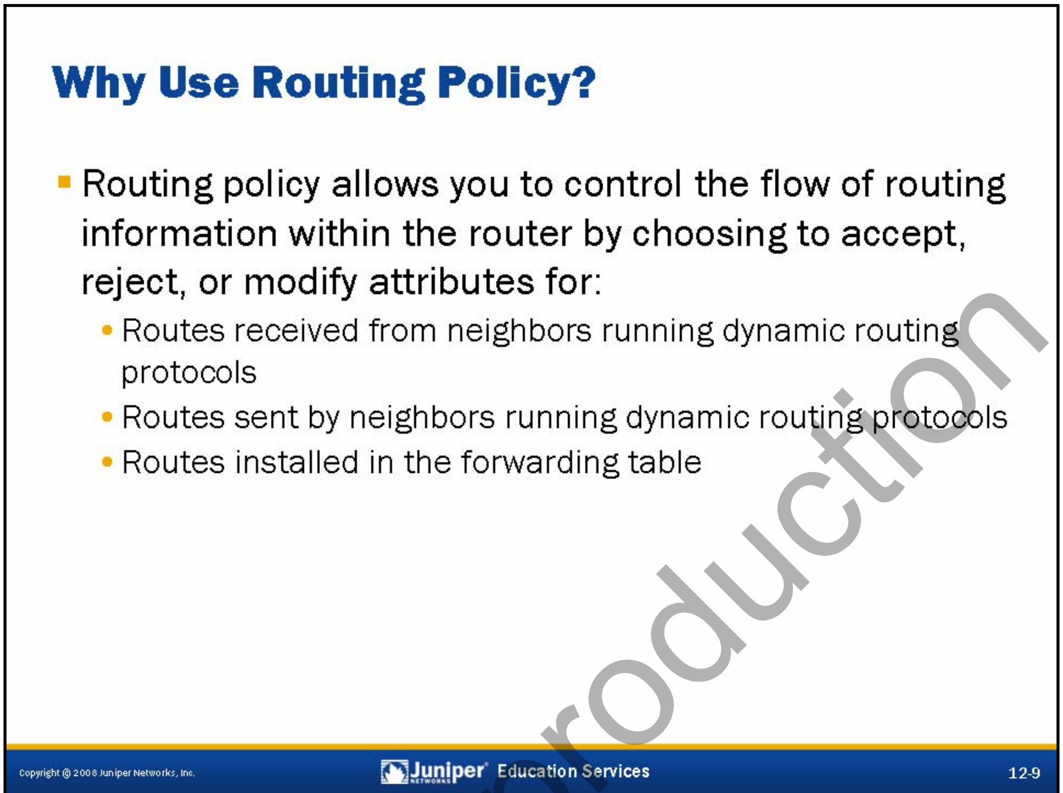
The second step involved with policy implementation is applying the policy. You can apply the routing policy at different levels depending on the situation and protocol to which the policy applies. With BGP, for example, you can apply a policy at the protocol, group, or neighbor level. When you apply different policies at different levels, the switch uses the most specific application. In other words, a neighbor import policy is more specific than the associated group's import policy, and a group policy is more specific than a global policy applied at the `[edit protocols bgp]` hierarchy level.

You apply firewall filters on Layer 2 interfaces, Layer 3 interfaces, or on a particular VLAN. Layer 2 interface and VLAN firewall filter applications require a firewall filter definition at the `[edit firewall family ethernet-switching]` hierarchy level, whereas Layer 3 interface firewall filter applications require a firewall filter definition at the `[edit firewall family inet]` hierarchy level.



Routing Policy Overview

The slide highlights the topic we discuss next.

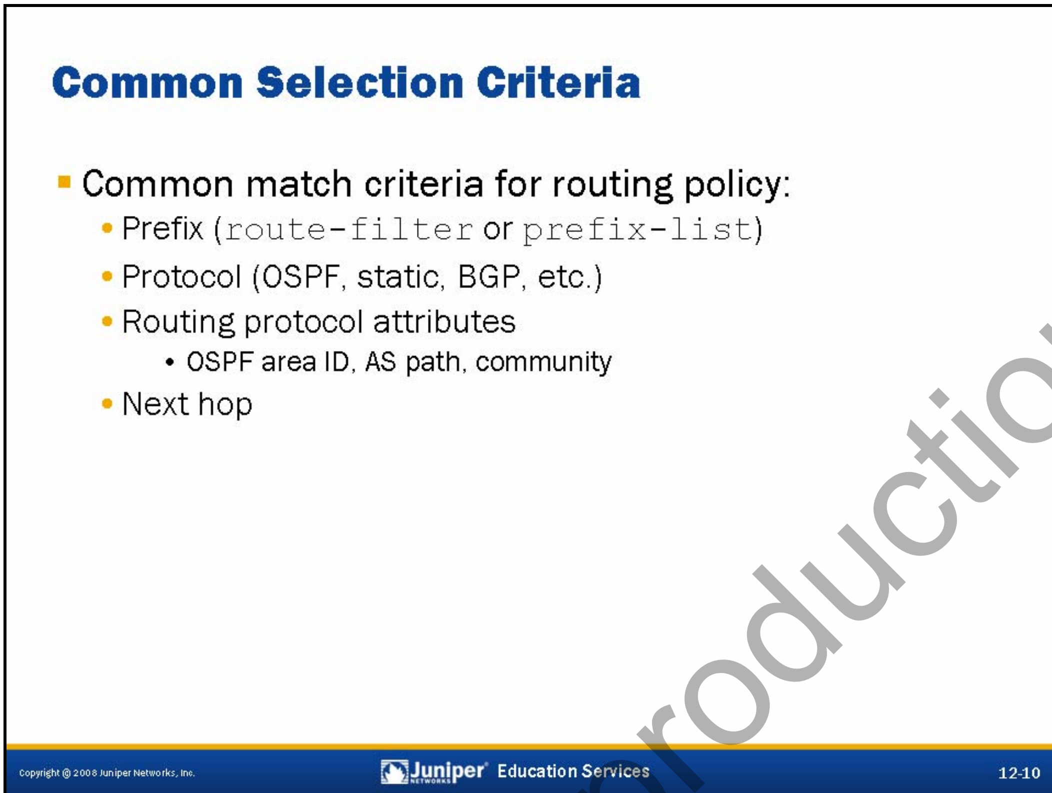


Routing Policy Uses

Routing policy allows you to control the flow of routing information within the switch. You can apply routing policy as information enters the routing table and as information leaves the routing table.

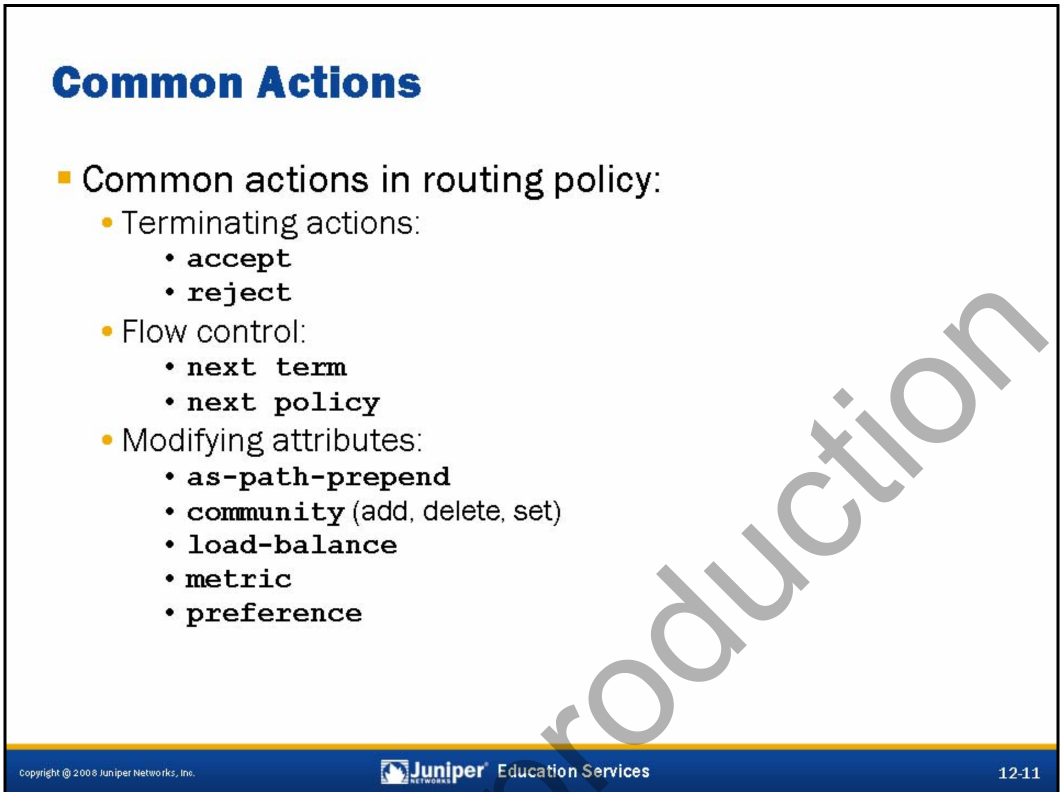
You can use routing policy to choose which routes you accept or reject from neighbors running dynamic routing protocols. You can also use routing policy to choose which routes you send to neighbors running dynamic routing protocols. Routing policy also allows you to modify attributes on routes as they enter or leave the routing table.

You can also use routing policy to control the flow of routing information into the forwarding table. This use allows you to control which routes are installed in the forwarding table and to control some of the attributes associated with those routes.



Common Selection Criteria

This slide shows some of the criteria you can use to select routes with *from* statements. You can select routes based on their prefix, protocol, some routing protocol attributes, or next hop. You can view the full list in the CLI interactive help and in the *JUNOS Policy Framework Configuration Guide*.

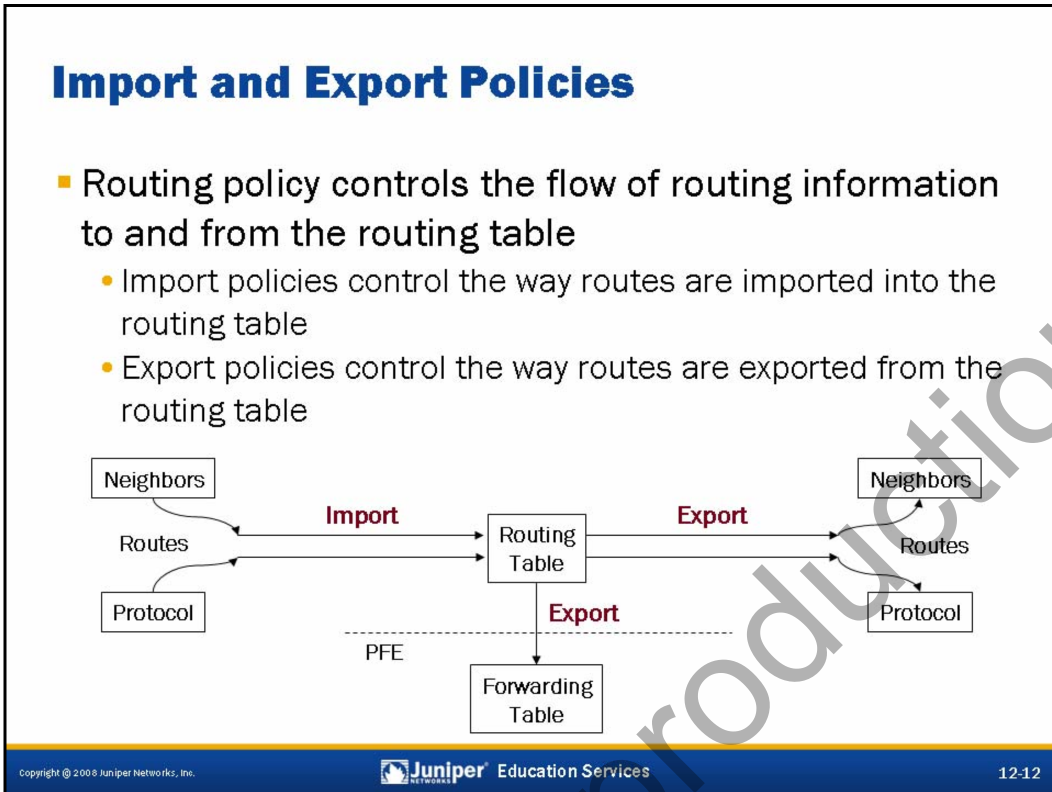


Common Actions

Some common routing policy actions include the terminating actions of **accept** and **reject**. These are called *terminating actions* because they cause the evaluation of the policy (and policy chain) to stop and the route to be accepted or rejected. The nonterminating equivalents of **default-action accept** and **default-action reject** do not cause policy evaluation to stop, but they do overrule the default policy's accept or reject determination.

Other common routing policy actions affect the flow of policy evaluation. The **next term** and **next policy** actions cause the switch to evaluate the next term or next policy, respectively.

Other common actions are to prepend the AS path in BGP announcements, modify BGP communities, instruct the forwarding table to load balance routes, modify an IGP metric or BGP MED (using the `metric` statement), and modify route preference.



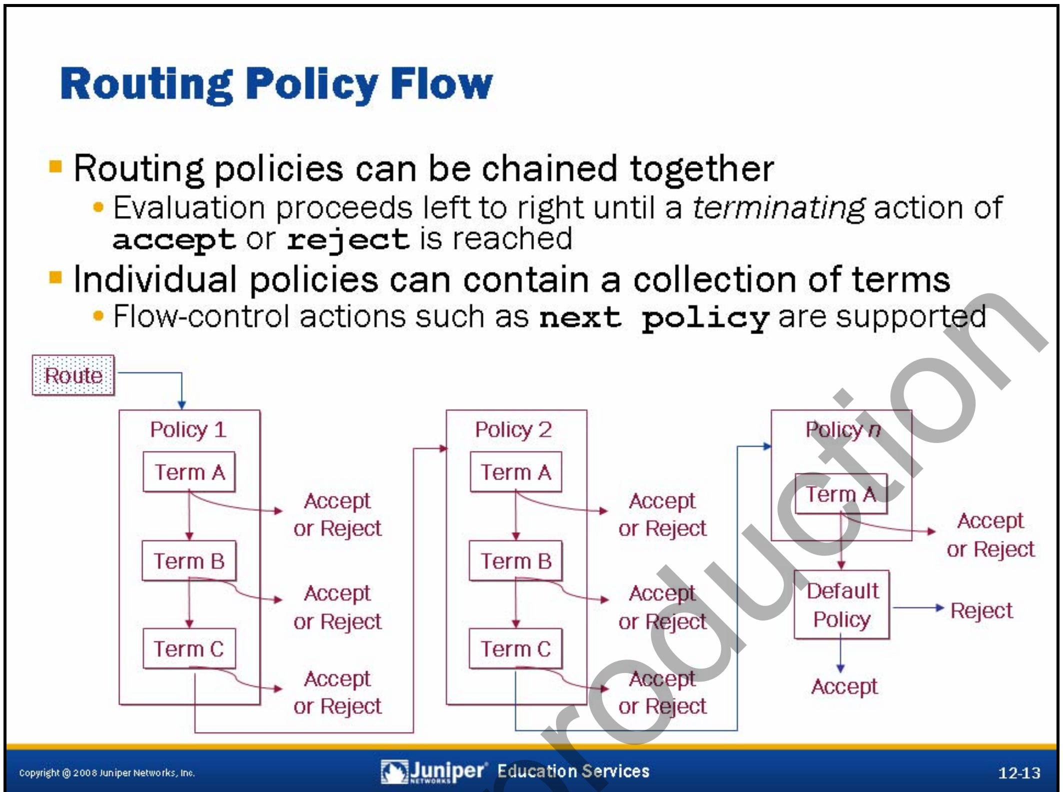
Import and Export Policies

Policies that control the way the switch imports routes into the routing table are called *import policies*. The switch applies these policies before it places the routes in the routing table. Thus, an import policy can change the routes that are available in the routing table and can affect the local route selection process.

Policies that control the way the switch exports routes from the routing table are called *export policies*. The switch applies these policies as it exports routes from the routing table to dynamic routing protocols or to the forwarding table. Only active routes are available for export from the routing table. Thus, while an export policy can choose which active routes to export and can modify attributes of those routes, it cannot cause inactive routes to be exported.

For example, suppose you have an OSPF route (preference 10) and a BGP route (preference 170) for the same prefix. An export policy can determine whether or not to send the active OSPF route and can modify attributes of the route as it is sent, but it cannot cause the inactive BGP route to be sent.

JUNOS software applies export policies as routes are exported from the routing table, so attribute changes do not affect the local routing table; rather, the software applies them to the route as it is exported.



Policy Chaining

You can cascade policies to form a chain of policy processing. You can create this chain of policies to solve a complex set of route manipulation tasks in a modular manner.

JUNOS software evaluates policies from left to right based on the order in which they are applied to a routing protocol. JUNOS software checks each policy's match criteria and performs the associated action when a match occurs. If the first policy does not match or if the match is associated with a nonterminating action, JUNOS software evaluates the route against the next policy in the chain. This pattern repeats itself for all policies in the chain. JUNOS software ultimately applies the default policy for a given protocol when no terminating actions occur while evaluating the user-defined policy chain. We define the default routing policies on the following pages.

Policy processing stops once a route meets a terminating action, unless you are grouping policies with Boolean operators. Grouping policies for logical operations, such as AND or OR, is a subject that is beyond the scope of this class.

Continued on next page.

Individual Policies

Individual policies can comprise multiple entries called terms. Terms are individual match/action pairs that you can name numerically or symbolically.

JUNOS software lists terms sequentially from top to bottom and evaluates them in that manner. Each term is checked for its match criteria. When a match occurs, the software performs the associated action. If no match exists in the first term, the software checks the second term. If no match exists in the second term, JUNOS software checks the third term. This pattern repeats itself for all terms. If no match exists in the last term, JUNOS software checks the next applied policy.

When a match is found within a term, JUNOS software takes the corresponding action. When that action is taken, the processing of the terms and the applied policies stops.

Default Routing Policy

- Each protocol has a default import policy and a default export policy
  - Applied if no policy is specified
  - Applied at the end of any specified policy or policy chain
    - Example:

```
export [ policyA policyB ];
```

can be thought of as

```
export [ policyA policyB default-policy ];
```

And:

```
export only-policy;
```

can be thought of as

```
export [ only-policy default-policy ];
```


Juniper Education Services
12-15

The Default Policy

Every protocol has a default import policy and a default export policy. The switch applies these policies if no import or export policy is configured. In addition, if an export or import policy or policy chain is configured, the switch applies the default policy at the end of the configured policy as if it were the last policy in a policy chain.

We examine each protocol's default policy on the following pages.

Default Policies

Protocol	Import Policy	Export Policy
RIP	Accept all RIP routes from explicitly configured neighbors and import into `inet.0`	Reject everything
OSPF	Accept all OSPF routes and import into `inet.0`	Reject everything (protocol floods by default)
BGP	Accept all BGP routes and import into `inet.0`	Accept all active BGP routes

 Juniper Education Services 12-16

Default Policies

The chart on the slide summarizes the default import and export policies for several common routing protocols.

The default policy for RIP is to import all routes learned from explicitly configured neighbors. The switch ignores all routes learned from neighbors that are not explicitly configured. By default, no routes are exported to RIP neighbors, including routes received via RIP. Thus, to advertise *any* routes to RIP neighbors, you must configure an export policy. For RIP, you can configure import policies at the protocol level and neighbor level, whereas you can configure export policies only at the group level.

The default OSPF import policy is to import all OSPF routes. As a link-state protocol, OSPF maintains a consistent link-state database throughout each OSPF area by flooding link-state advertisements (LSAs). You cannot apply policy to affect the maintenance of the local link-state database or the flooding of LSAs. Additionally, you cannot apply policy that prevents internal (including interarea) routes from being installed in the routing table. (A link-state protocol assumes that all switches or routers have the same routing information for internal routes, which causes all routes to make consistent forwarding decisions. If you could block internal routes from entering the routing table, you could create routing loops or cause certain prefixes to become unreachable.) However, you *can* apply policy that blocks external routes.

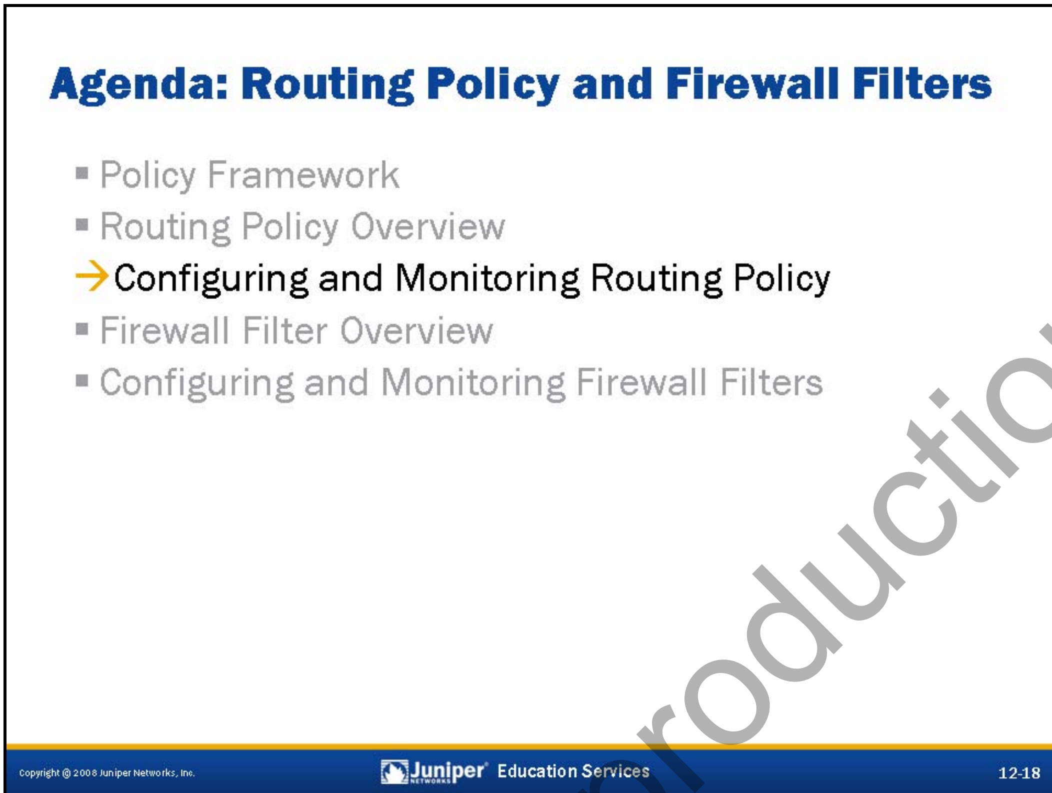
Continued on next page.

Default Policies (contd.)

The default OSPF export policy (which rejects everything) does not cause the switch to stop flooding LSAs through the area. Rather, the switch always floods LSAs throughout the OSPF area, and that behavior cannot be controlled by routing policy. The default export policy simply blocks additional routes from other sources from being advertised to OSPF neighbors. If you want to advertise other routes via OSPF, you must configure an explicit export policy.

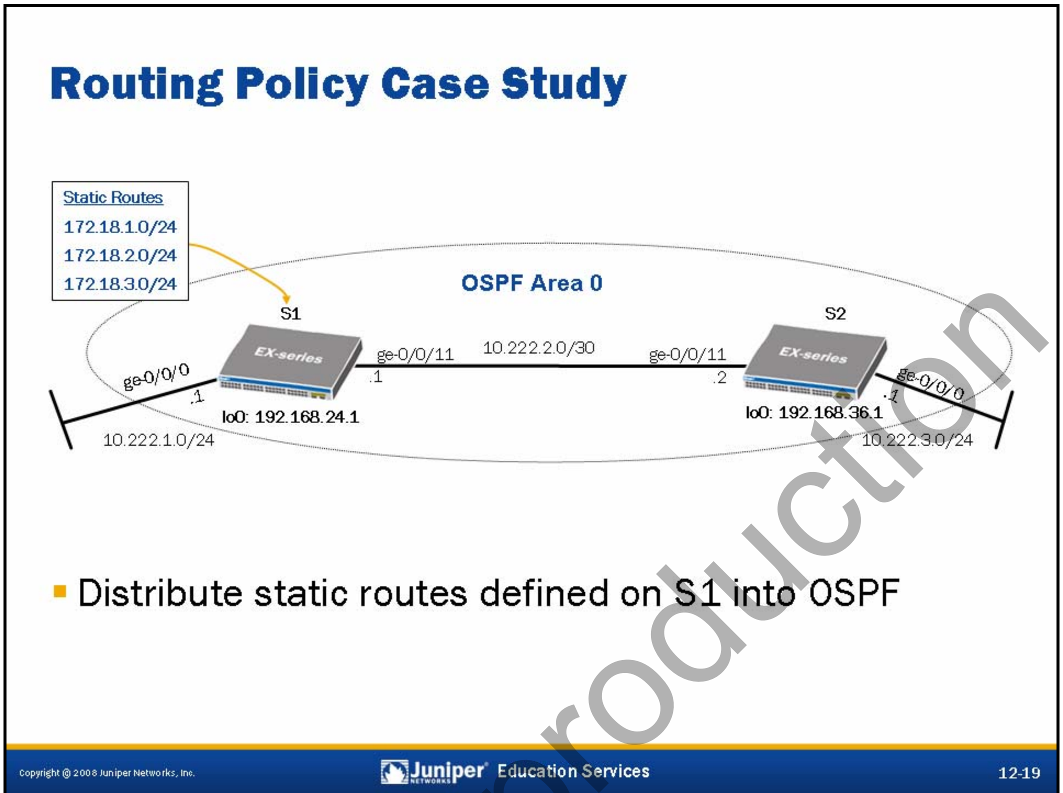
Because link-state protocols rely on all participating devices having consistent link-state databases, you can configure import and export policies only at the protocol level.

BGP’s default import policy is to accept all routes from BGP neighbors and install them in the routing table, and to export all active BGP routes to all BGP neighbors.



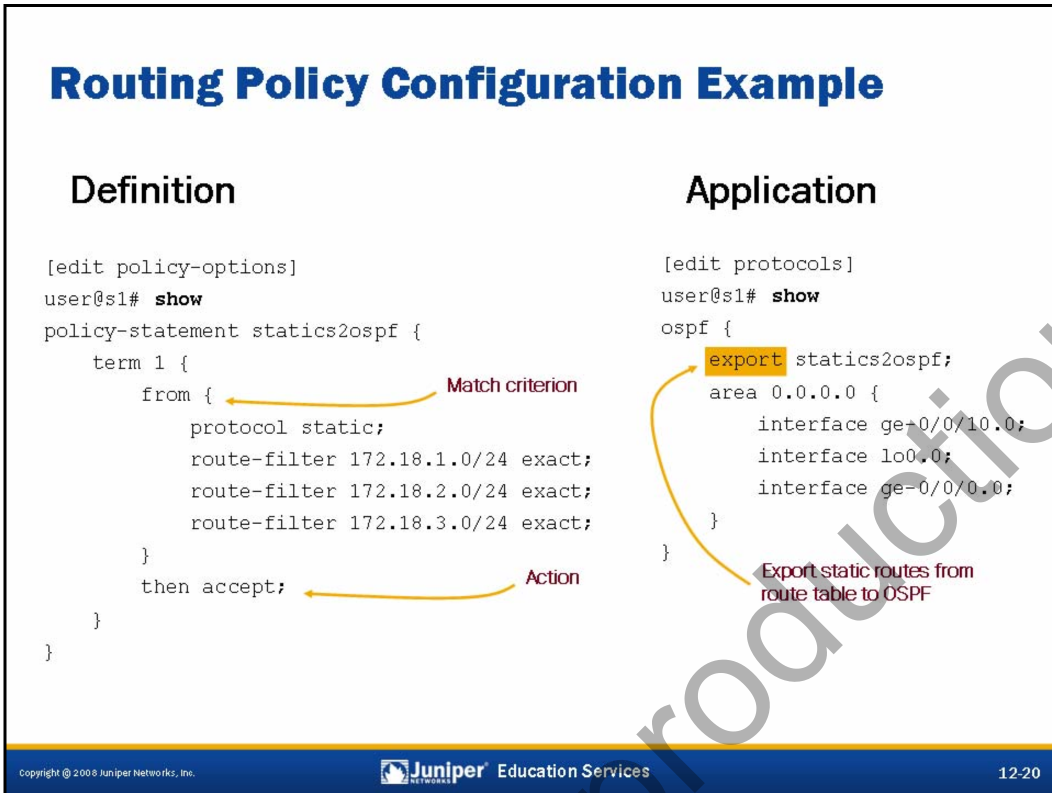
Configuring and Monitoring Routing Policy

The slide highlights the topic we discuss next.



Routing Policy Case Study

This slide introduces a routing policy case study that requires a distribution policy to export into OSPF a series of static routes defined on the S1 device into OSPF.



Routing Policy Configuration Example

This slide shows the required configuration to accomplish our previously stated objective.

Routing Policy Monitoring Example

```
user@s1> show ospf database external
     OSPF AS SCOPE link state database
 Type       ID               Adv Rtr           Seq      Age  Opt  Cksum  Len
Extern  *172.18.1.0      192.168.24.1     0x80000003   276  0x22 0x99d4  36
Extern  *172.18.2.0      192.168.24.1     0x80000002  1176  0x22 0x90dd  36
Extern  *172.18.3.0      192.168.24.1     0x80000002   876  0x22 0x85e7  36

user@s2> show route 172/8
inet.0: 11 destinations, 11 routes (11 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

172.18.1.0/24      *[OSPF/150] 00:37:11, metric 0, tag 0
                    > to 10.222.2.1 via ge-0/0/10.0
172.18.2.0/24      *[OSPF/150] 00:37:11, metric 0, tag 0
                    > to 10.222.2.1 via ge-0/0/10.0
172.18.3.0/24      *[OSPF/150] 00:37:11, metric 0, tag 0
                    > to 10.222.2.1 via ge-0/0/10.0
```

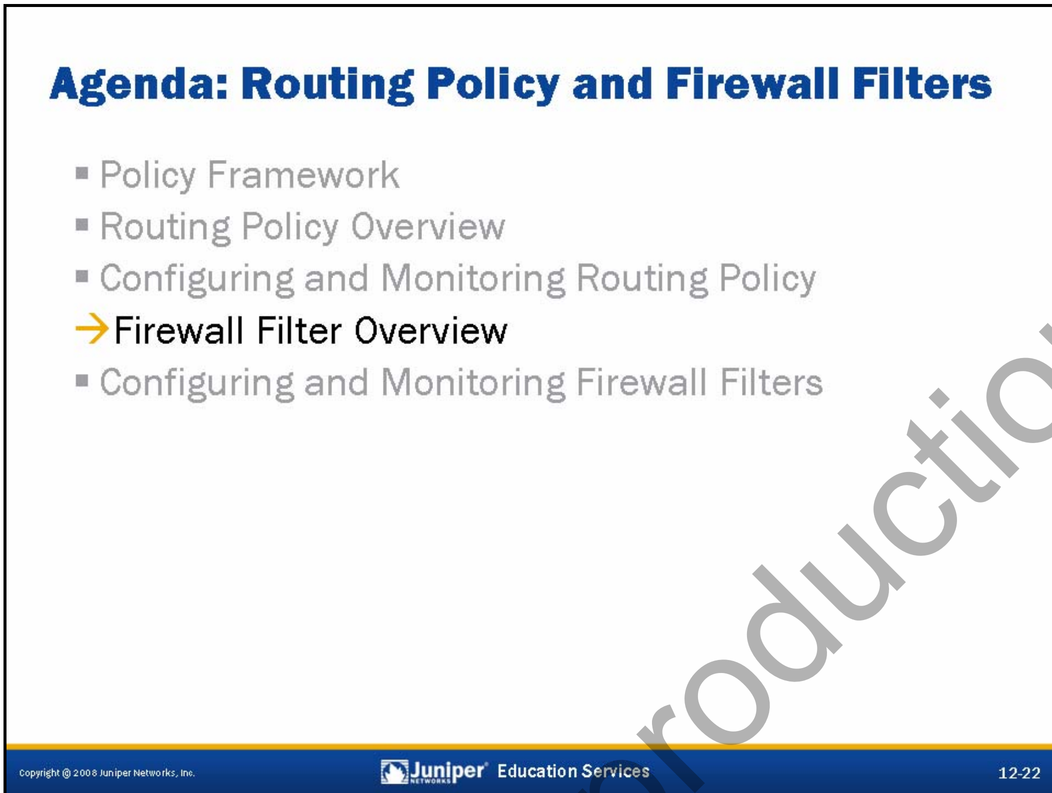
S2 receives the external LSAs flooded by S1 and adds the appropriate OSPF routes

Juniper Education Services

12-21

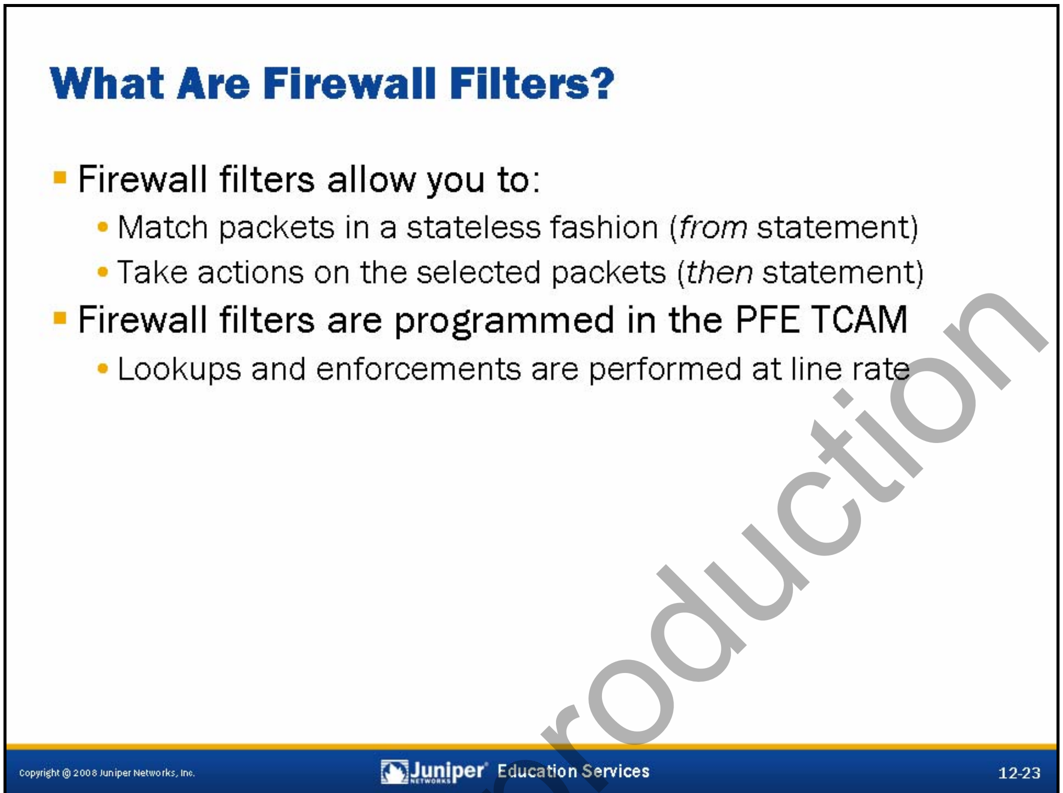
Routing Policy Monitoring Example

Based on the policy definition and application, the static routes, as defined on the S1 device, are now being flooded into OSPF as external LSAs. We also perform route learning verification on the neighboring switch, S2, and see that the routes are in fact being added to S2's route table. Based on these verification steps, it appears that our objective has been met.



Firewall Filter Overview

The slide highlights the topic we discuss next.

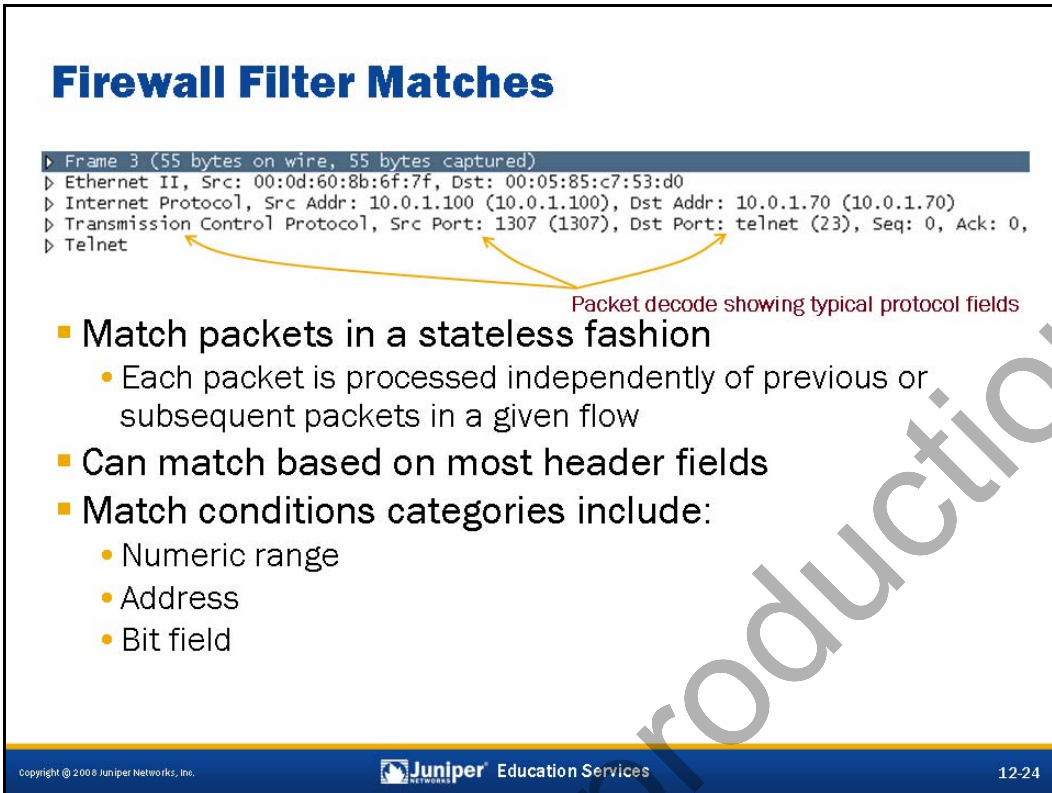


Firewall Filters

Firewall filters follow the policy framework described previously in this chapter. You use the *from* statement to list criteria used to select packets and the *then* statement to specify the actions that the switch takes on matching packets. Stateless firewall filters work on each individual packet in isolation from all others. Thus, unlike a stateful firewall, which tracks connections and allows you to specify an action to be taken on all packets within a flow, a stateless firewall filter has no concept of connections. The stateless nature of these filters can impact the way you write your firewall filters.

Hardware Processing

Unlike some other vendors, EX-series switches always perform firewall filter checks in hardware. Firewall filters are programmed in the Packet Forwarding Engine (PFE) ternary content addressable memory (TCAM). Because firewall filters are implemented in hardware rather than a software process, the result is a very efficient match and enforcement rate when performing packet filtering operations.



Stateless Filtering

The switch processes each packet through the firewall filters independently of all other packets. This processing affects the way you craft firewall filters and also has implications on the information that is available to a switch when it processes packets through these filters.

Because the switch does not keep state information on connections, you must explicitly allow traffic in both directions for each connection that you want to permit. By contrast, stateful firewall filters require you to permit only the initial connection and then permit bidirectional communications for this connection.

The stateless nature of these filters also affects the information available to the switch when processing these filters. For example, if you want to allow all *established* TCP sessions through a switch, you can have the firewall filter permit all TCP packets that have the acknowledgement (ACK) flag set in the TCP header. However, looking for this match condition provides no guarantee that the session was properly established. This packet might instead have been maliciously crafted to have the ACK flag set for an unestablished TCP session.

Continued on next page.

Match on Header Fields

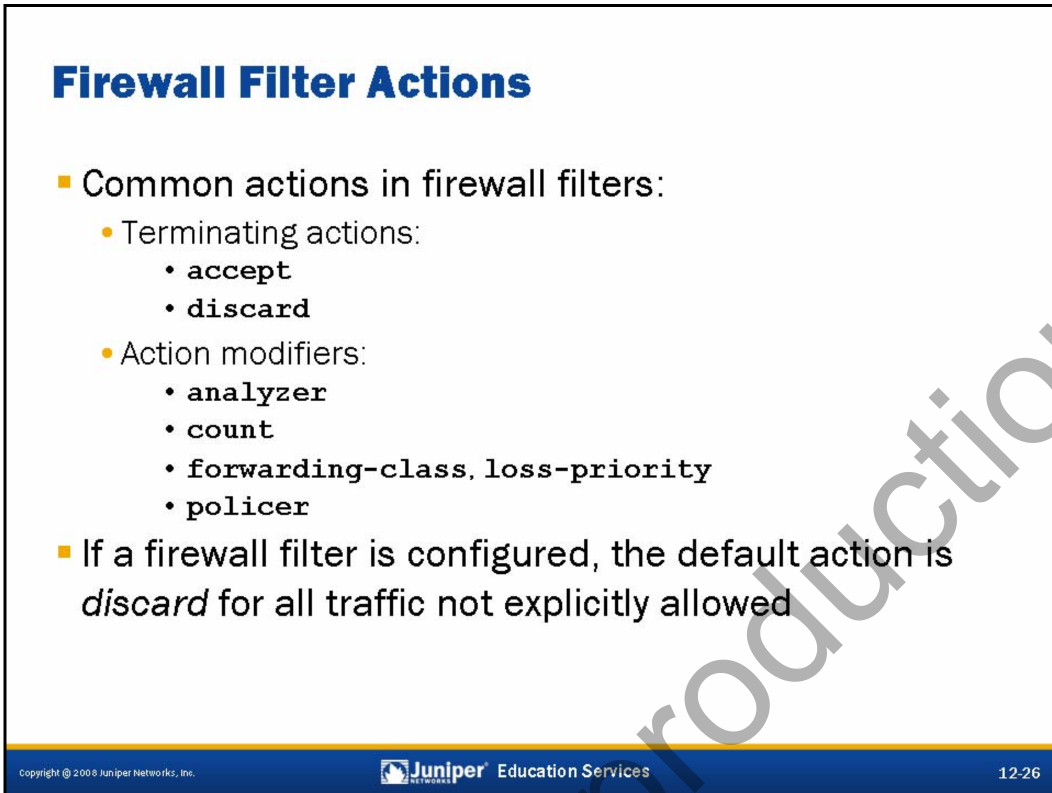
You specify the criteria to be used to match packets in *from* clauses within firewall filter terms. You can use many header fields as match criteria. However, you must remember that all header fields might not be available to you because of the way firewall filters are processed.

When you specify a header field, the JUNOS software looks for a match at the location in the header where that field should exist. However, it does not check to ensure that the header field makes sense in the given context. For example, if you specify that the software should look for the ACK flag in the TCP header, the software looks for that bit to be set at the appropriate location, but it does not check that the packet was actually a TCP packet. Therefore, you must account for how the software looks for a match when writing your filters. In this case, you should have the switch check both that the packet was a TCP packet and that the TCP ACK flag was set.

The stateless nature of firewall filters can affect the information available in the processing of fragmented packets. Processing fragments is trickier with stateless firewall filters than with a stateful firewall filter. The first fragment should have all the Layer 4 headers; however, subsequent fragments will not. Additionally, attempting to check Layer 4 headers in fragments produces unpredictable results. As was explained previously, the JUNOS software still attempts to evaluate the Layer 4 headers; however, the second and subsequent fragments do not contain these headers, so the matches are unpredictable.

Categories of Match Conditions

Match conditions generally fall into three categories: numeric range, address, and bit field match conditions. You can generally use the same evaluation options for each condition within the category. There are also several *synonyms*, which are match conditions that are equivalent to one or more of these match conditions.



Common Actions

You specify actions in the *then* clause of a term. You can specify terminating actions or action modifiers.

Terminating actions cause the policy evaluation to stop. The `accept` action causes the switch to accept the packet and continue the input or output processing of the packet. The `discard` action causes the switch to silently discard the packet, without sending an Internet Control Message Protocol (ICMP) message to the source address.

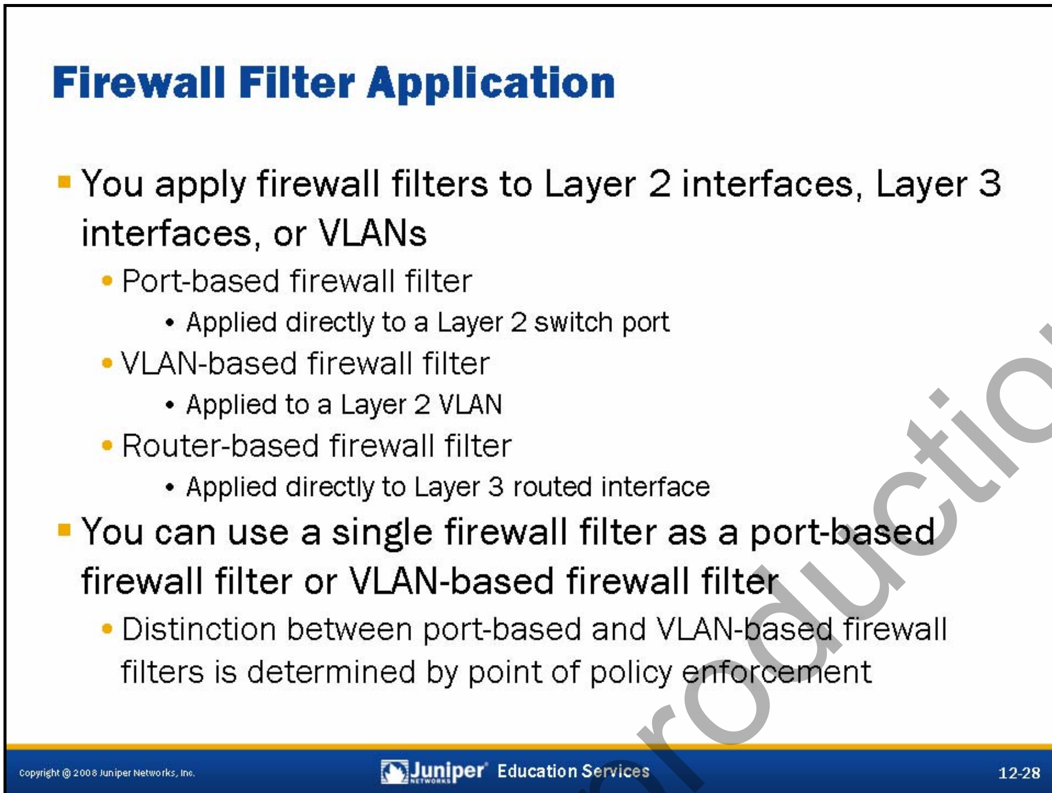
You can specify one or more action modifiers with any terminating action. If you specify an action modifier but do not specify a terminating action, the switch imposes an action of `accept`. You can use the `count` action modifier to count the number of packets processed through the filter that match the specified criteria defined in the *from* statement. The `forwarding-class` and `loss-priority` action modifiers are used to specify class-of-service (CoS) information. The `policer` action modifier allows you to invoke a traffic policer. The `analyzer` action modifier specifies that the switch should mirror the packets for additional analysis.

Continued on next page.

Default Action

The default action when a firewall filter is configured is *discard*. Therefore, if a packet fails to match any term within a firewall filter or chain of firewall filters, the switch silently discards it.

Unlike routing policy, the default action is different when a firewall filter is configured than when no firewall filter is configured. If no firewall filter is configured, the default action is *accept*.

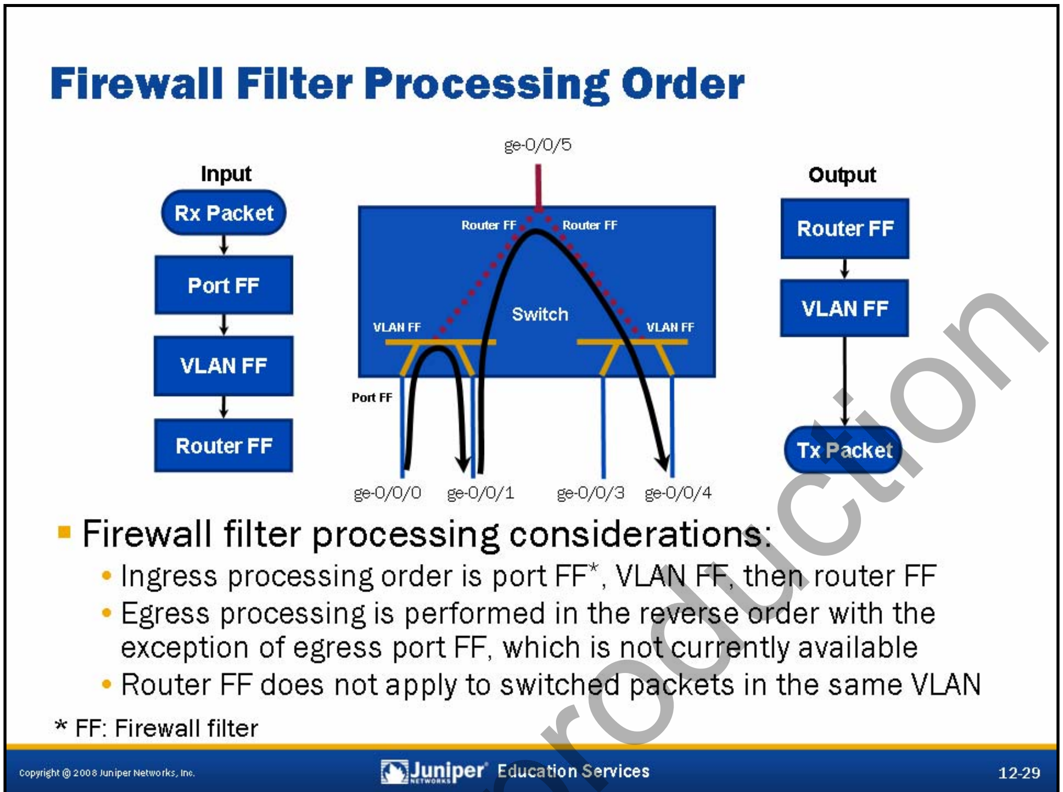


ACL Application

Firewall filters are applied on Layer 2 interfaces, Layer 3 interfaces, or on a particular VLAN. Layer 2 interface and VLAN firewall filter applications require a firewall filter definition at the `[edit firewall family ethernet-switching]` hierarchy level, whereas Layer 3 interface firewall filter applications require a firewall filter definition at the `[edit firewall family inet]` hierarchy level. The name given for a firewall filter is determined by the protocol family and the point of application. For example, firewall filters applied on a Layer 2 interface are often referred to as port-based access control lists (PACLs), firewall filters applied to a Layer 2 VLAN are commonly known as VLAN-based ACLs (VACLs), and firewall filters applied to a Layer 3 routed interface are called router-based ACLs (RACLs).

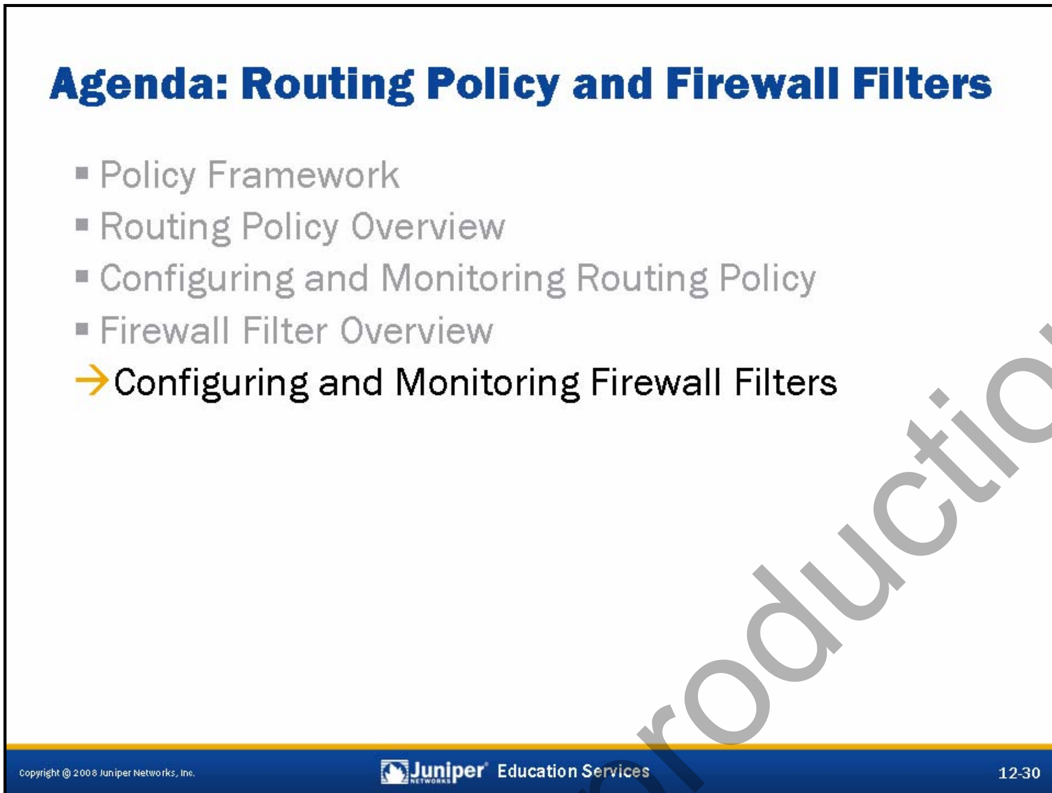
Common ACL, Multiple Enforcement Points

The switch can use a firewall filter defined at the `[edit firewall family ethernet-switching]` hierarchy level as a port-based firewall filter or as a VLAN-based firewall filter. The main distinction between port-based firewall filters and VLAN-based firewall filters is the point of enforcement or application. A port-based firewall filter is applied on a given Layer 2 switch port, whereas a VLAN-based firewall filter is applied under the `[edit vlans]` hierarchy for the designated VLAN.



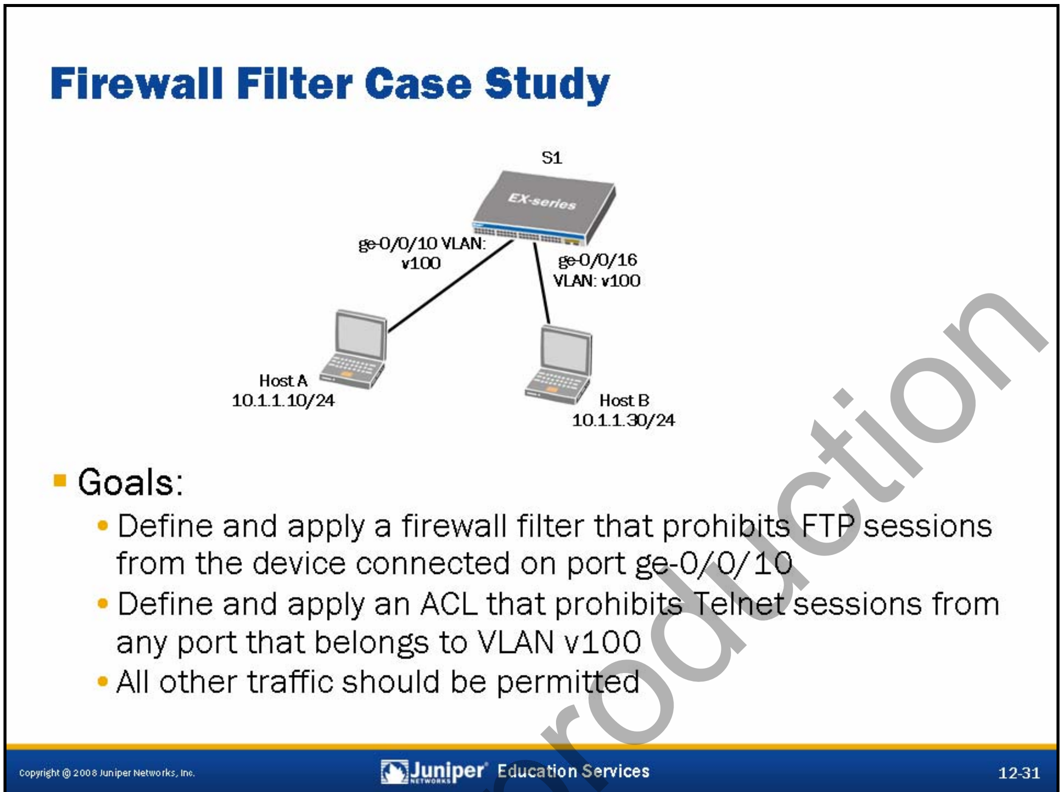
Firewall Filter Processing

This slide illustrates the firewall filter processing order within EX 3200 and EX 4200 switches. Along with the firewall processing order, the slide also includes some basic firewall filter processing considerations.



Configuring and Monitoring Firewall Filters

The slide highlights the topic we discuss next.



Firewall Filter Case Study

This slide introduces the firewall filter case study and provides some basic objectives.

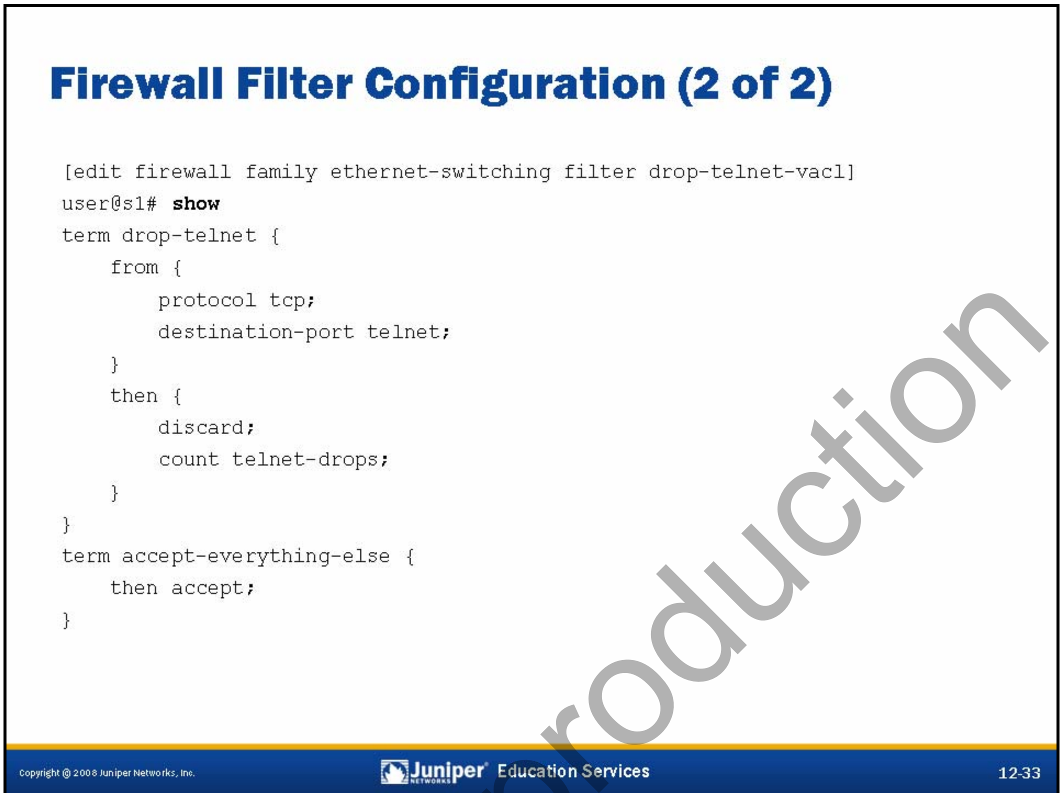
Firewall Filter Configuration (1 of 2)

```
[edit firewall family ethernet-switching filter drop-ftp-pacl]
user@s1# show
term drop-ftp {
    from {
        protocol tcp;
        destination-port ftp;
    }
    then {
        discard;
        count ftp-drops;
    }
}
term accept-everything-else {
    then accept;
}
```


Juniper Education Services
12-32

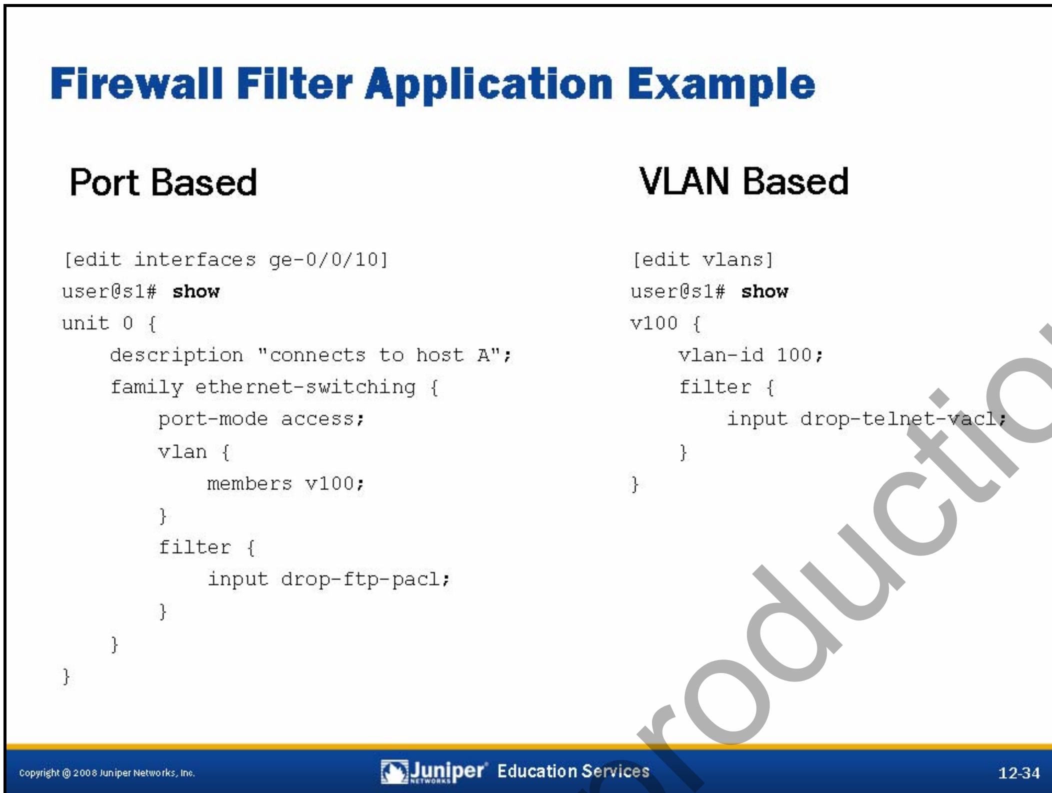
Firewall Filter Configuration: Part 1

This slide illustrates the sample port-based ACL used to meet one of the stated objectives.



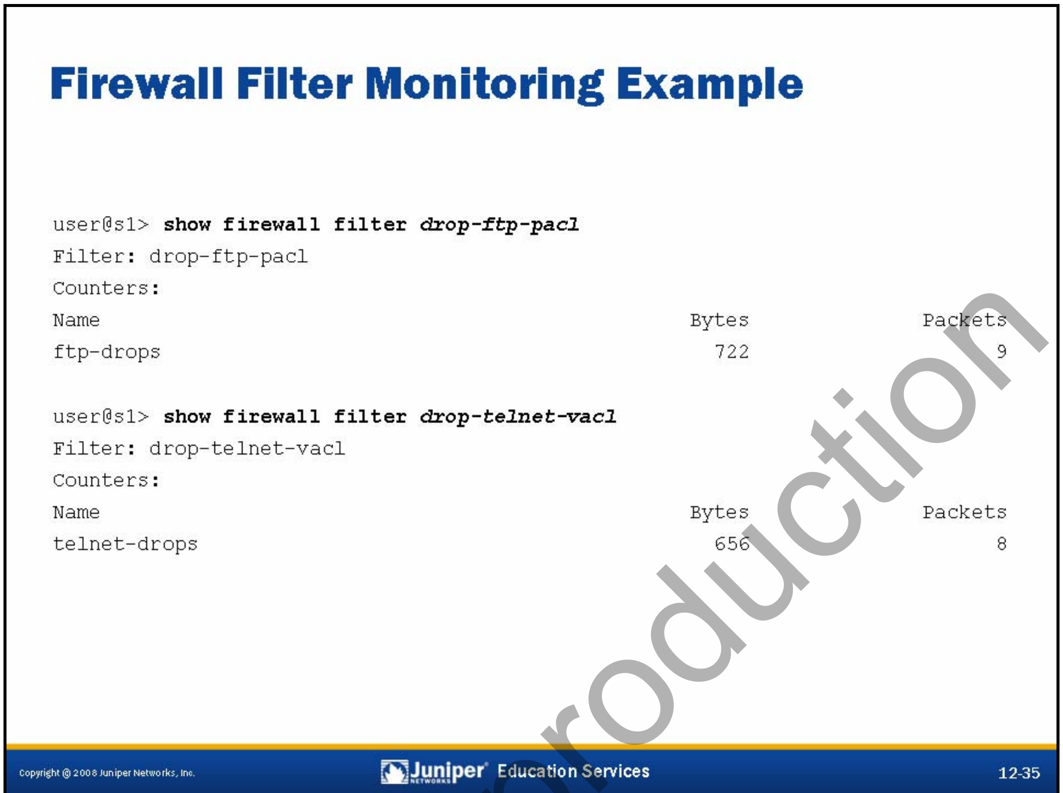
Firewall Filter Configuration: Part 2

This slide illustrates the sample VLAN-based ACL used to meet the remainder of the stated objectives.



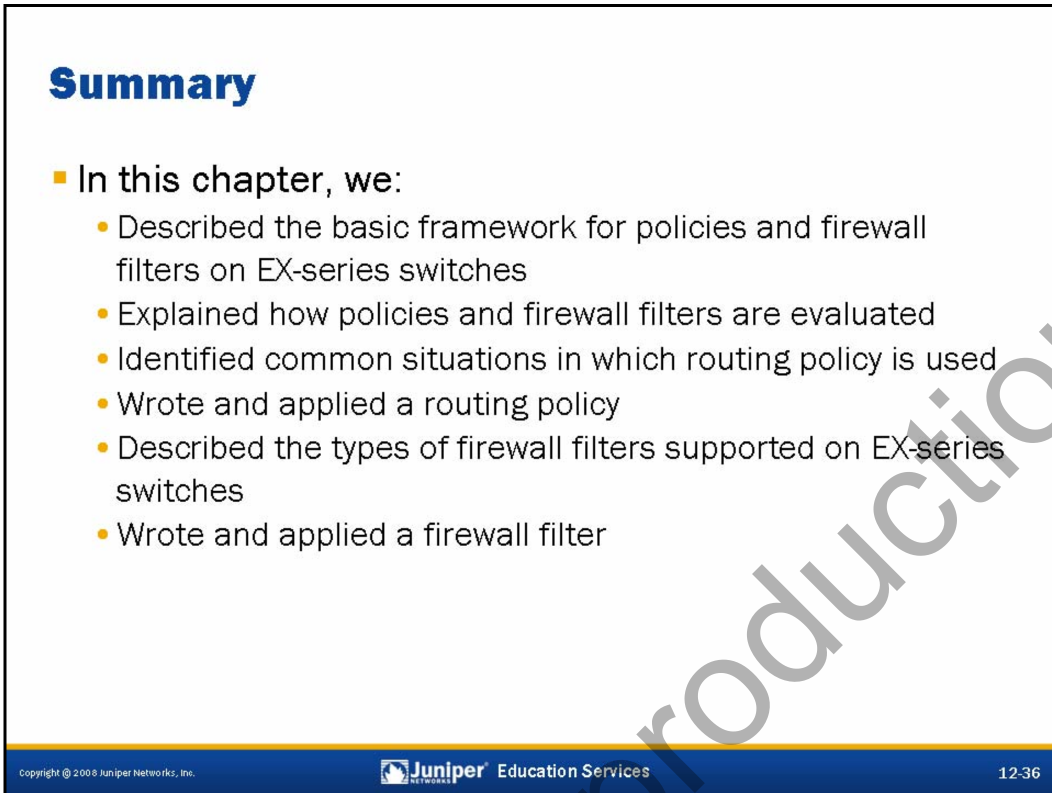
Firewall Filter Application Examples

This slide displays the port-based and VLAN-based ACL applications.



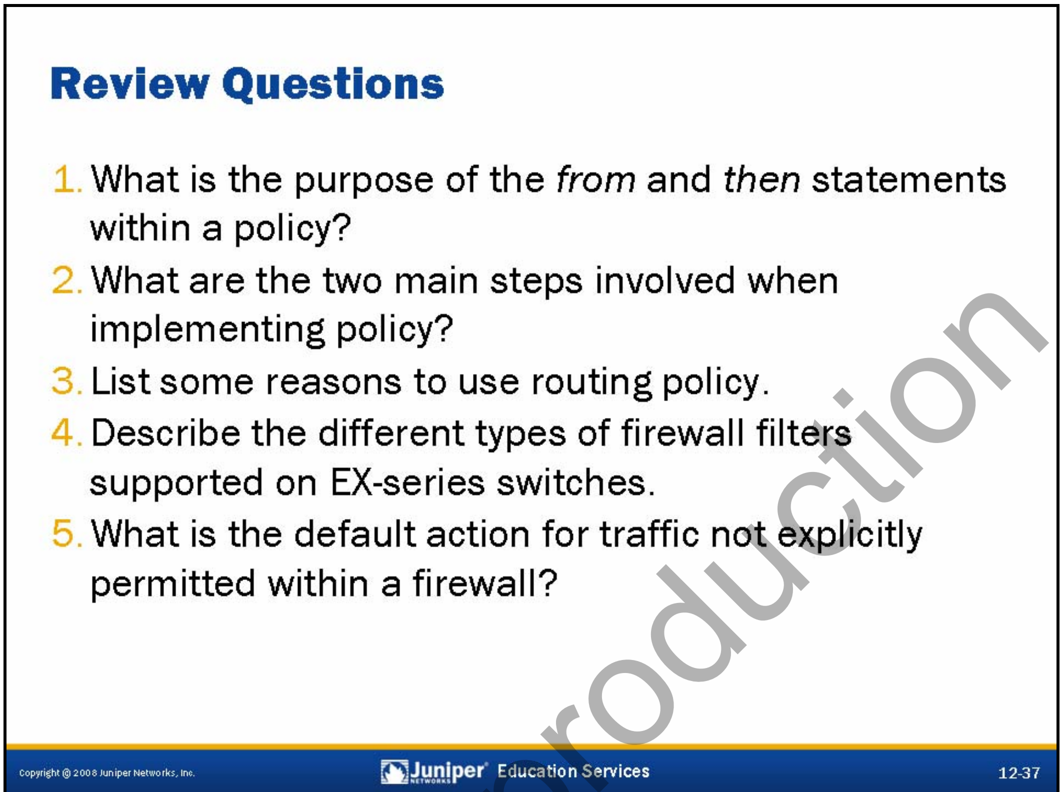
Firewall Filter Monitoring Example

This slide shows the use of the **`show firewall filter`** ***`filter-name`*** command. The highlighted commands display a count for ACL violations because we included the **`count`** modifying action along with the **`discard`** terminating action within the associated ACLs.



This Chapter Discussed:

- The policy framework on EX-series switches;
- Policy and firewall filter evaluation;
- Typical usage scenarios for routing policy;
- Configuration and application of routing policy;
- The types of firewall filters supported on EX-series switches; and
- Configuration and application of firewall filters.



Review Questions:

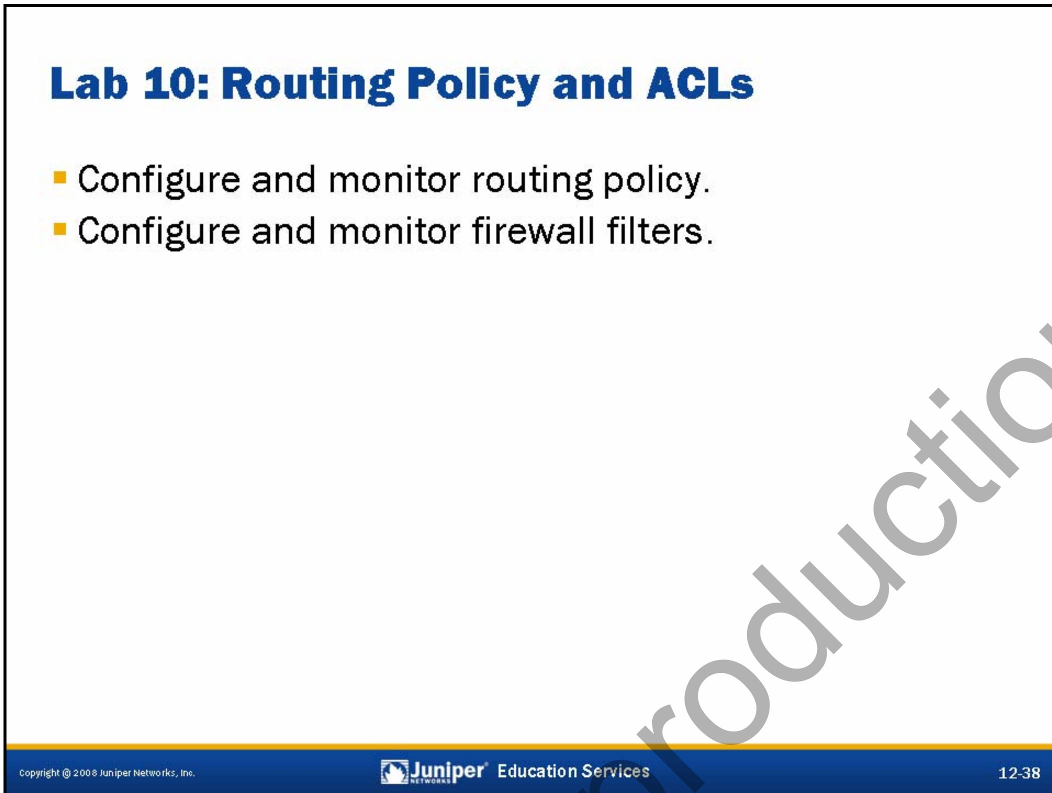
1.

2.

3.

4.

5.

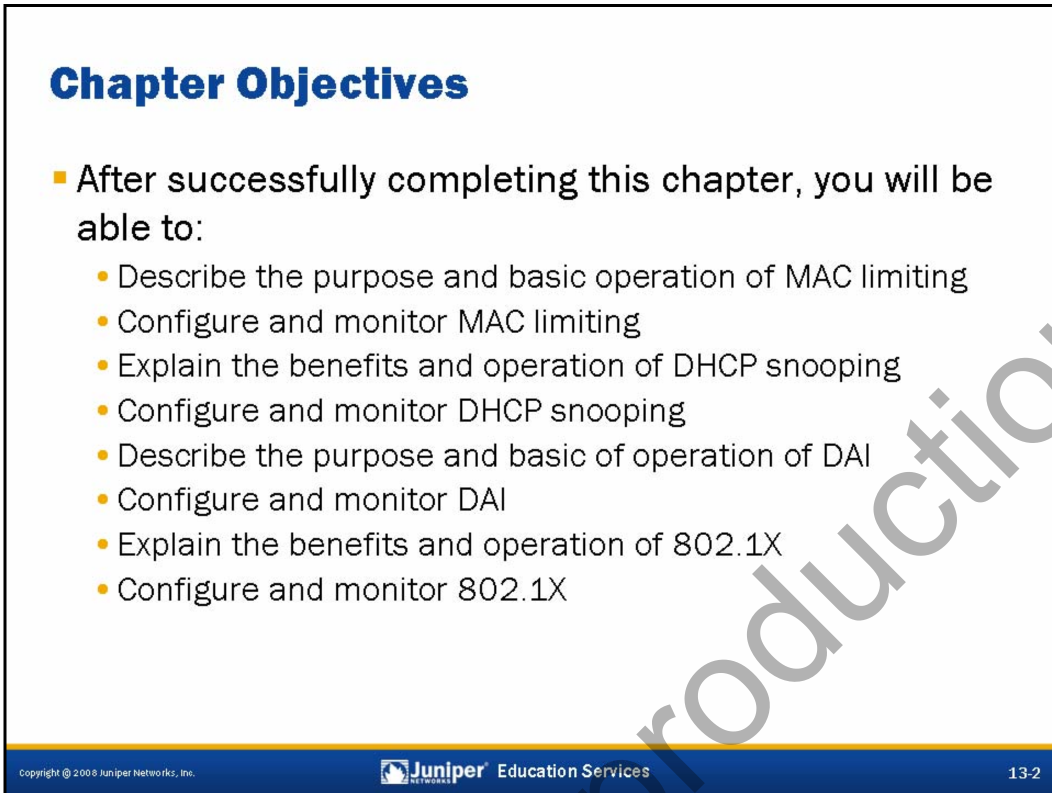


Lab 10: Routing Policy and ACLs

The slide provides the objectives for this lab.

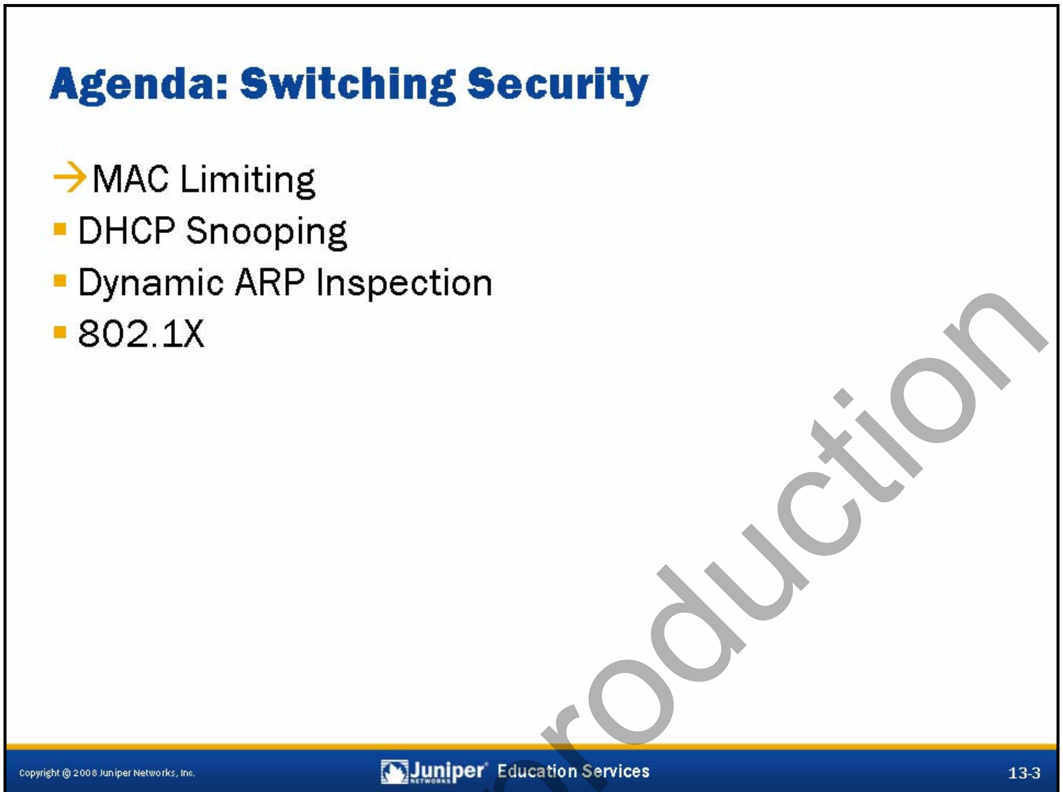
Operating Juniper Networks Switches in the Enterprise

Chapter 13: Switching Security



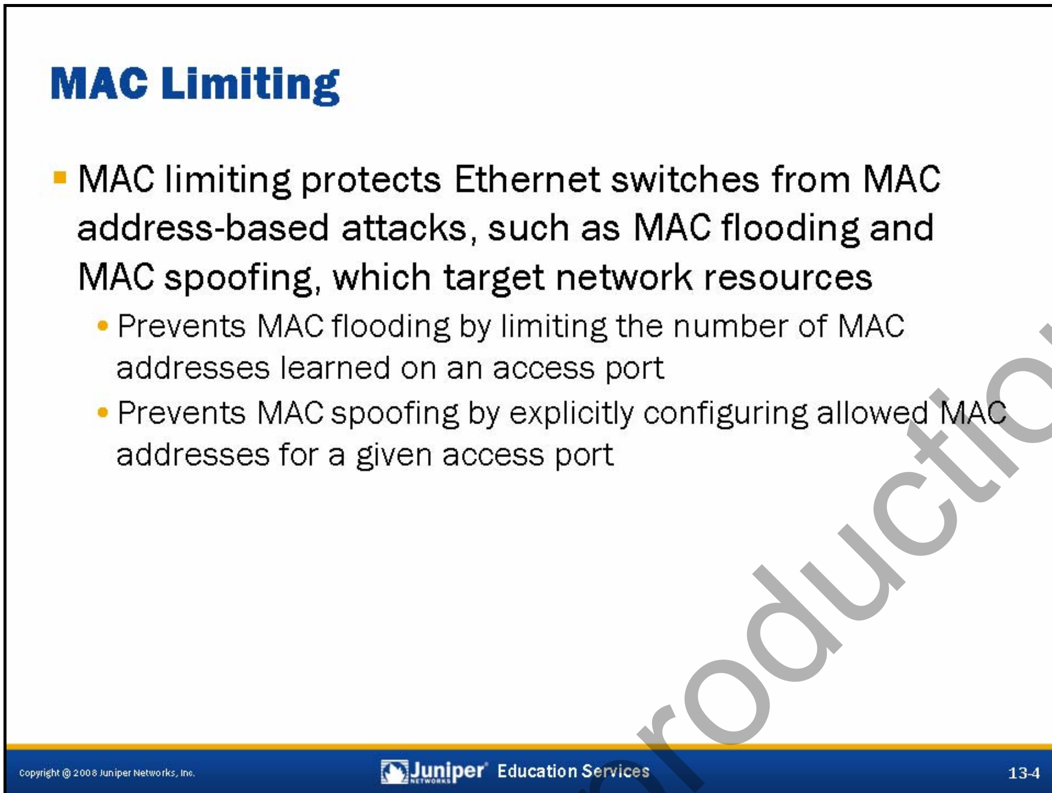
This Chapter Discusses:

- The purpose and basic operation of media access control (MAC) limiting;
- Configuring and operational monitoring of MAC limiting;
- The benefits and operation of Dynamic Host Configuration Protocol (DHCP) snooping;
- Configuring and monitoring DHCP snooping;
- The purpose and basic operation of Dynamic ARP Inspection (DAI);
- Configuring and monitoring DAI;
- The benefits and operation of 802.1X; and
- Configuring and monitoring 802.1X.



MAC Limiting

The slide lists the topics we cover in this chapter. We discuss the highlighted topic first.

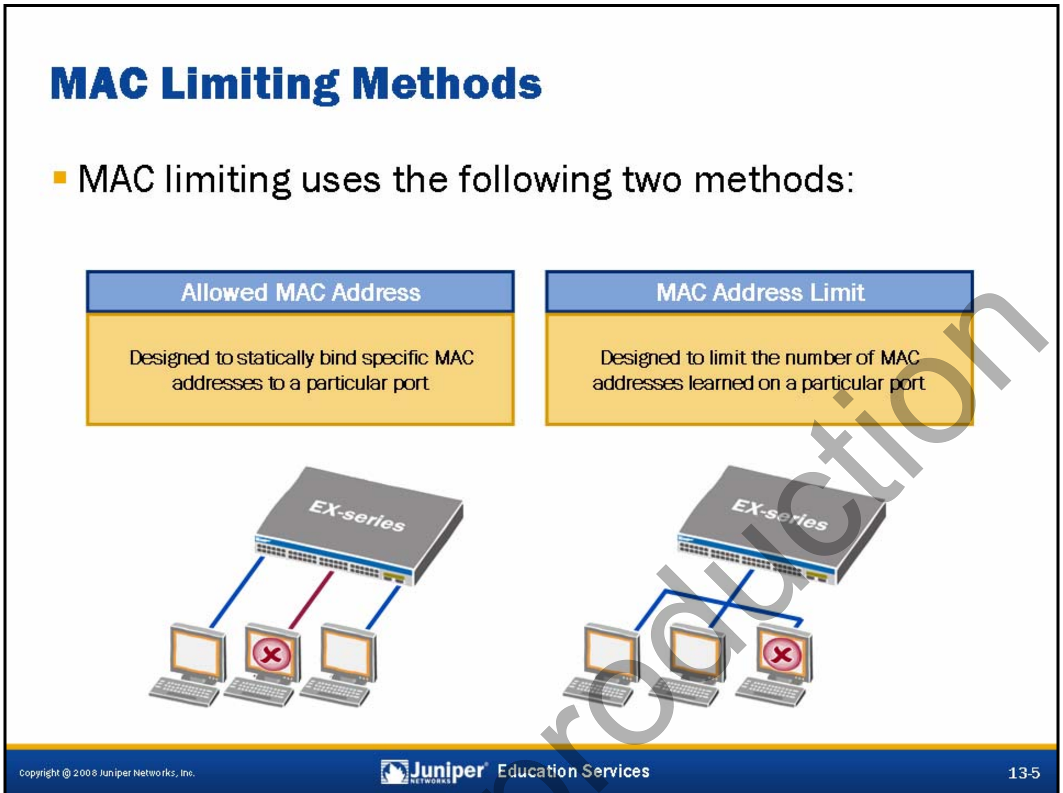


MAC Limiting

MAC limiting protects Ethernet switches, as well as other network resources, against attacks that use MAC addresses. Some examples of attacks that use MAC addresses to disrupt network operations include MAC flooding and MAC spoofing. Both MAC flooding and MAC spoofing can be quite harmful because they facilitate a denial-of-service (DoS) attack, which renders users, systems, or entire networks useless. MAC limiting can be implemented using two different methods.

The first method allows you to specify the maximum number of MAC addresses that can be learned on a single Layer 2 access port. Once the switch reaches the MAC limit, all traffic sourced from new MAC addresses is subject to being dropped based on the configured action.

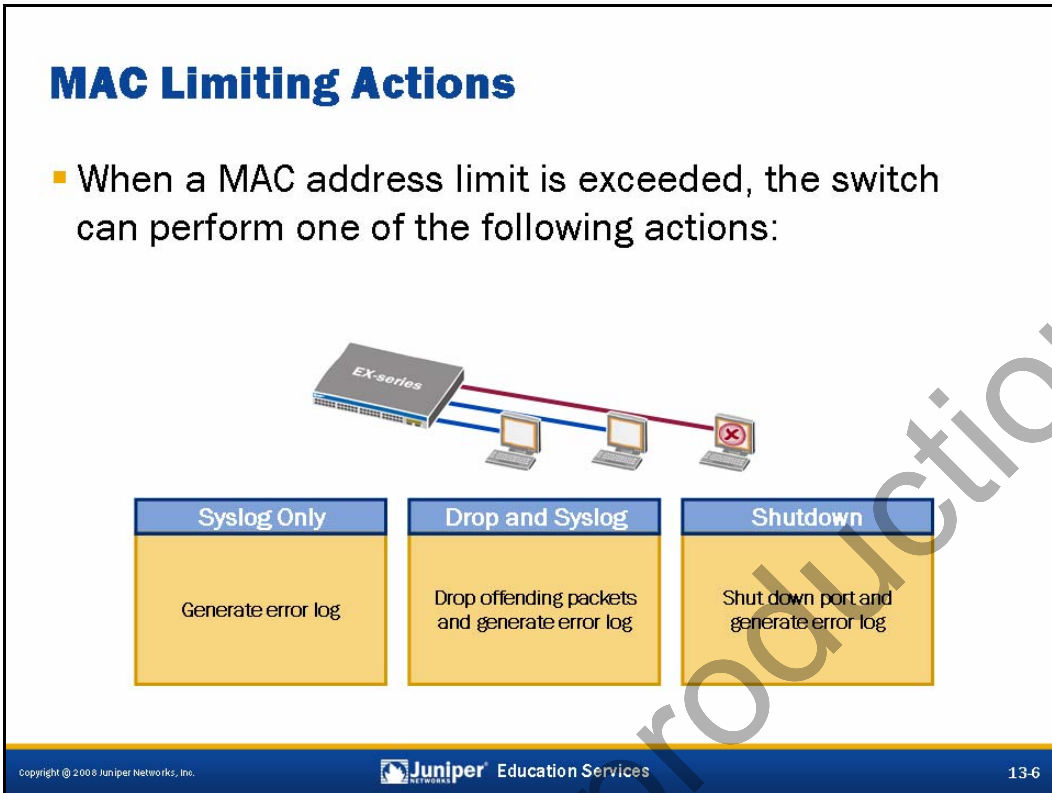
The second method allows you to define allowed MAC addresses for a specific access port. Any MAC address that is not listed will not be learned or permitted network access.



MAC Limiting Methods

The slide illustrates the two MAC limiting methods. In the first method, shown on the left side of the slide, the switch either permits or denies each individual host network access through the attached network port, based on its MAC address. This requires knowledge of the node's MAC address and is not ideal in environments where end-users move from switch port to switch port.

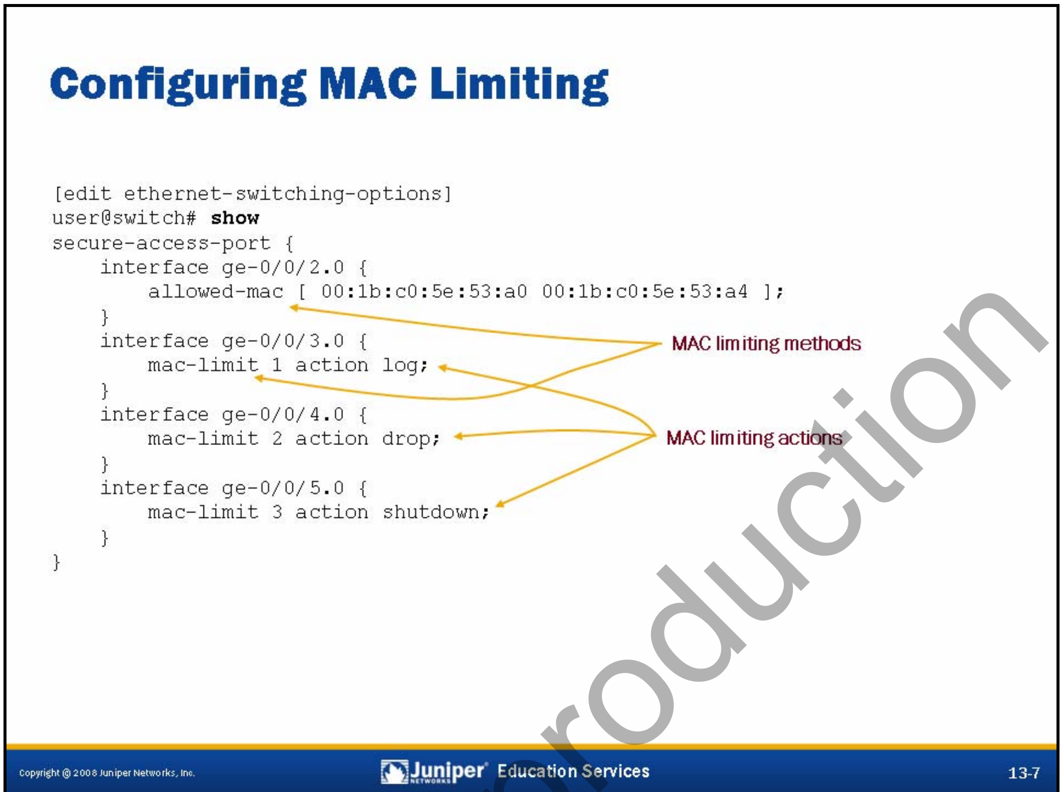
In the second method, shown on the right side of the slide, multiple hosts are attempting to access the network through a single switch port. In this example, the port is configured with a MAC limit of two; thus, the switch permits access to only the first two devices, whereas it denies network access through this port for any subsequent devices beyond the specified limit of two. The MAC limit is user defined and varies depending on the needs within each environment. In environments that use IP telephony, the limit specified should be two when an IP phone and a user's PC are attached to the same switch port. In data-only environments, you can typically specify a limit of one to account for the user's PC connection.



MAC Limiting Actions

When a MAC limiting violation occurs, the switch performs one of the following actions:

- `shutdown`: Blocks data traffic and generates a system log entry.
- `drop`: Drops the packet and generates a system log entry.
- `log`: Does not drop the packet but generates a system log entry.
- `none`: Does nothing.

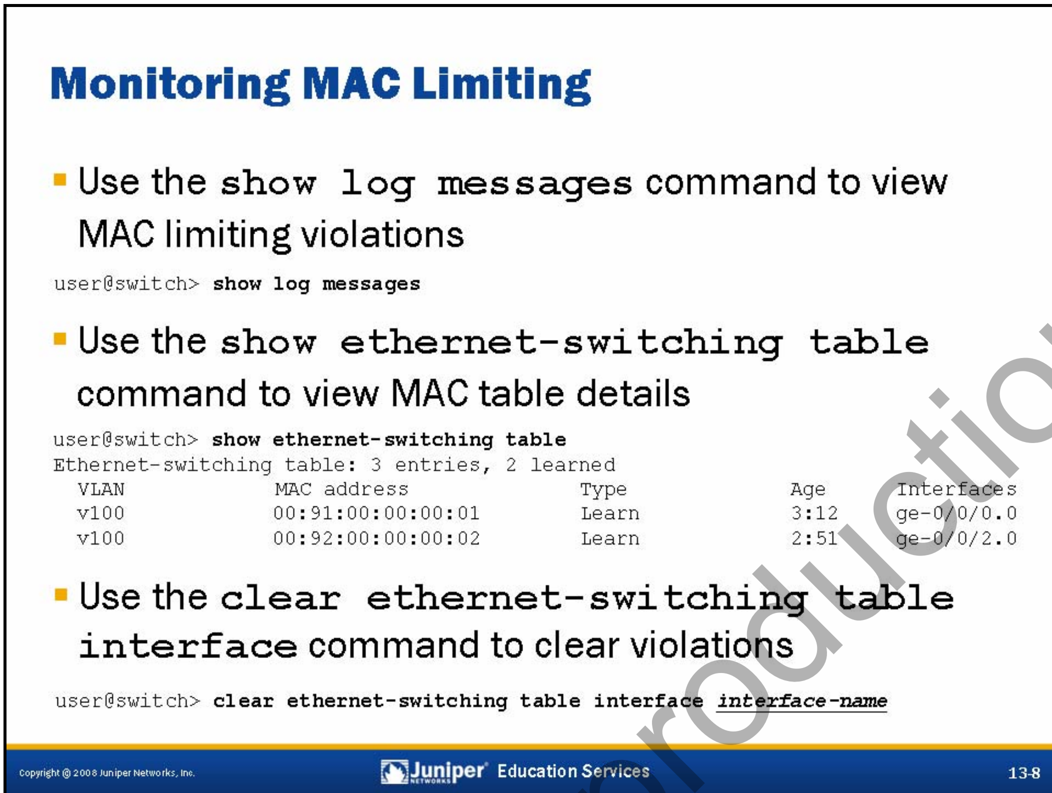


Configuring MAC Limiting

The slide illustrates a sample MAC limiting configuration. In this example, we see both enforcement methods, as well as the common actions invoked when a limit violation occurs. As mentioned previously, in addition to the actions of `log`, `drop`, and `shutdown`, a fourth action of `none` exists. The action `none` allows you to exclude individual interfaces from a MAC limiting configuration when the `interface all` statement is used. The following example illustrates this scenario:

```
[edit ethernet-switching-options]
user@switch# show
secure-access-port {
    interface ge-0/0/10.0 {
        mac-limit 1 action none;
    }
    interface all {
        mac-limit 1 action shutdown;
    }
}
```

As highlighted, when the `interface all` statement is used in conjunction with individual interface statements, JUNOS software considers the individual interface statements to be more specific, and they always take precedence.



Tracking Violation Events

Use the **show log messages** command to view MAC limiting violations. The actual log entry, shown within the log messages file, varies depending on the configured MAC limiting action. The following example shows the possible event messages:

```
user@switch> show log messages |match limit
...
Jan 11 12:45:34 switch eswd[746]: ESWD_MAC_LIMIT_EXCEEDED: MAC limit (1)
exceeded at ge-0/0/0.0
Jan 11 13:04:25 switch eswd[746]: ESWD_MAC_LIMIT_DROP: MAC limit (2) exceeded
at ge-0/0/5.0: dropping the packet
Jan 11 22:54:09 switch eswd[746]: ESWD_MAC_LIMIT_BLOCK: MAC limit (2) exceeded
at ge-0/0/10.0: shutting down the interface
...
```

You can also use the **show ethernet-switching interfaces detail** command to verify the current blocking state of an interface. Include the interface name to display the details for a specific interface. An example, which shows a blocked interface, is shown:

```
user@switch> show ethernet-switching interfaces ge-0/0/10.0 detail
Interface: ge-0/0/10.0 Index: 69
  State: up
  VLANs:
    default                untagged     blocked - MAC limit exceeded
```

Continued on next page.

MAC Table Verification

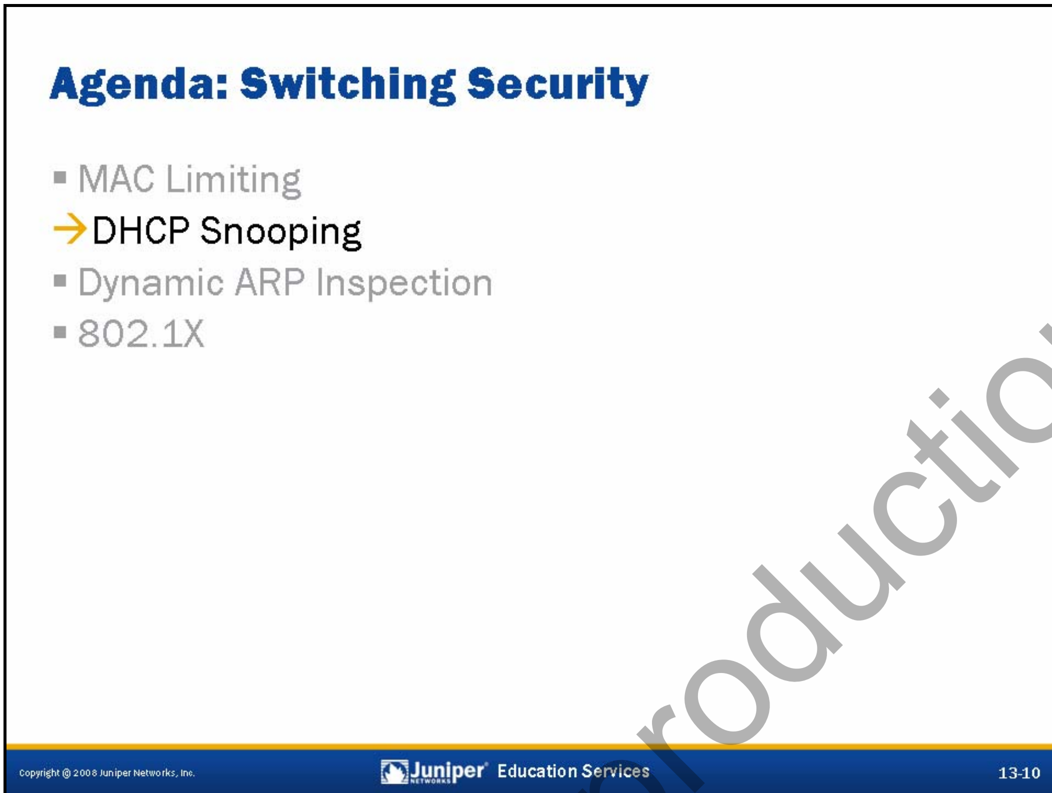
Use the **show ethernet-switching table** command to view the contents of the MAC table. The MAC table displays the learned MAC address entries. Each MAC address entry includes the associated virtual LAN (VLAN), type, age, and interface. If the entry is learned through an interface, the `Type` column displays `Learn`. If the MAC address belongs to a routed VLAN interface (RVI) (displayed as `Router` in the following example), the type `Static` is displayed. For all unknown MAC address within a given VLAN, a type of `Flood` is listed. The following example illustrates the various entry types:

```
user@switch> show ethernet-switching table
Ethernet-switching table: 6 entries, 3 learned
  VLAN              MAC address       Type         Age Interfaces
  default           *                 Flood          - All-members
  default           00:1b:c0:5e:53:a1 Learn          0 ge-0/0/2.0
  marketing         *                 Flood          - All-members
  marketing         00:19:e2:50:77:a1 Learn          0 ge-0/0/0.0
  marketing         00:19:e2:50:7c:00 Static         - Router
  marketing         00:1c:f9:cd:42:17 Learn          0 ge-0/0/0.0
```

Clearing MAC Table Entries

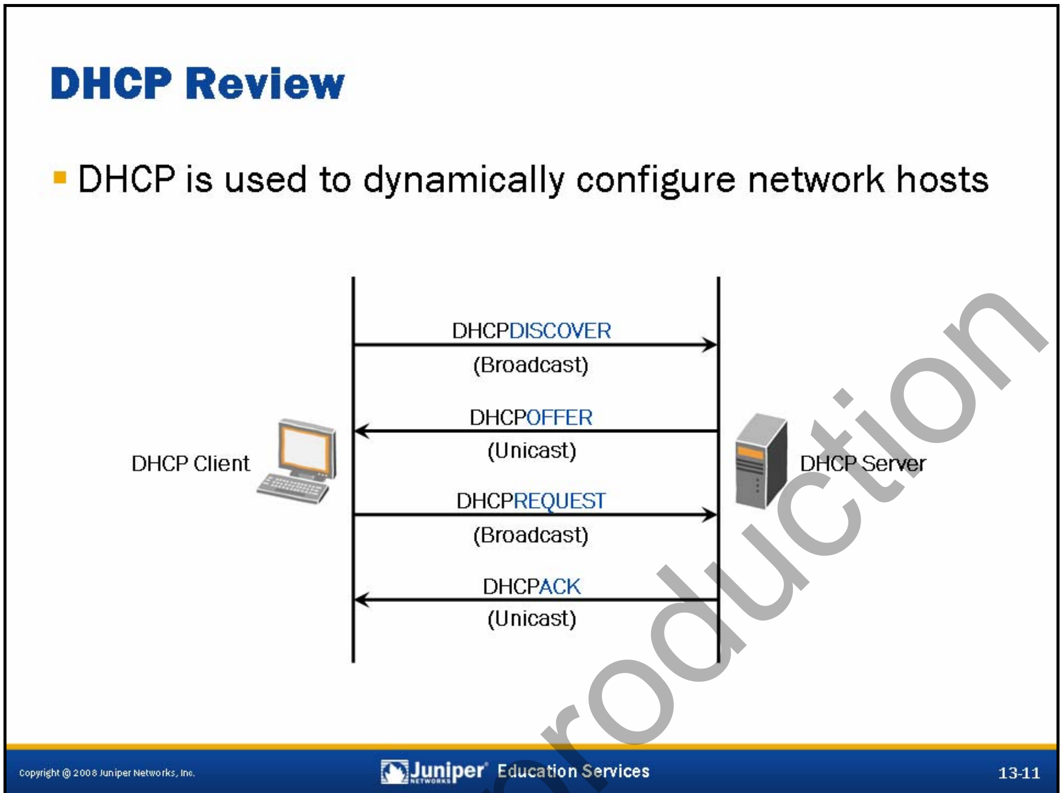
Use the **clear ethernet-switching table** or **clear ethernet-switching table interface *interface-name*** command to clear MAC table entries. The **clear ethernet-switching table** command clears all learned entries, whereas the **clear ethernet-switching table interface *interface-name*** command clears entries associated only with the referenced interface. The following example illustrates this operation:

```
user@switch> show ethernet-switching table
Ethernet-switching table: 6 entries, 4 learned
  VLAN              MAC address       Type         Age Interfaces
  marketing         *                 Flood          - All-members
  marketing         00:19:e2:50:3f:e1 Learn          0 ge-0/0/0.0
  marketing         00:19:e2:50:77:a1 Learn          0 ge-0/0/0.0
  marketing         00:19:e2:50:7c:00 Static         - Router
  marketing         00:19:e2:50:82:e1 Learn          0 ge-0/0/0.0
  marketing         00:1c:f9:cd:42:17 Learn          0 ge-0/0/0.0

user@switch> clear ethernet-switching table interface ge-0/0/0.0

user@switch> show ethernet-switching table
Ethernet-switching table: 3 entries, 1 learned
  VLAN              MAC address       Type         Age Interfaces
  marketing         *                 Flood          - All-members
  marketing         00:19:e2:50:7c:00 Static         - Router
  marketing         00:1c:f9:cd:42:17 Learn          0 ge-0/0/0.0
```



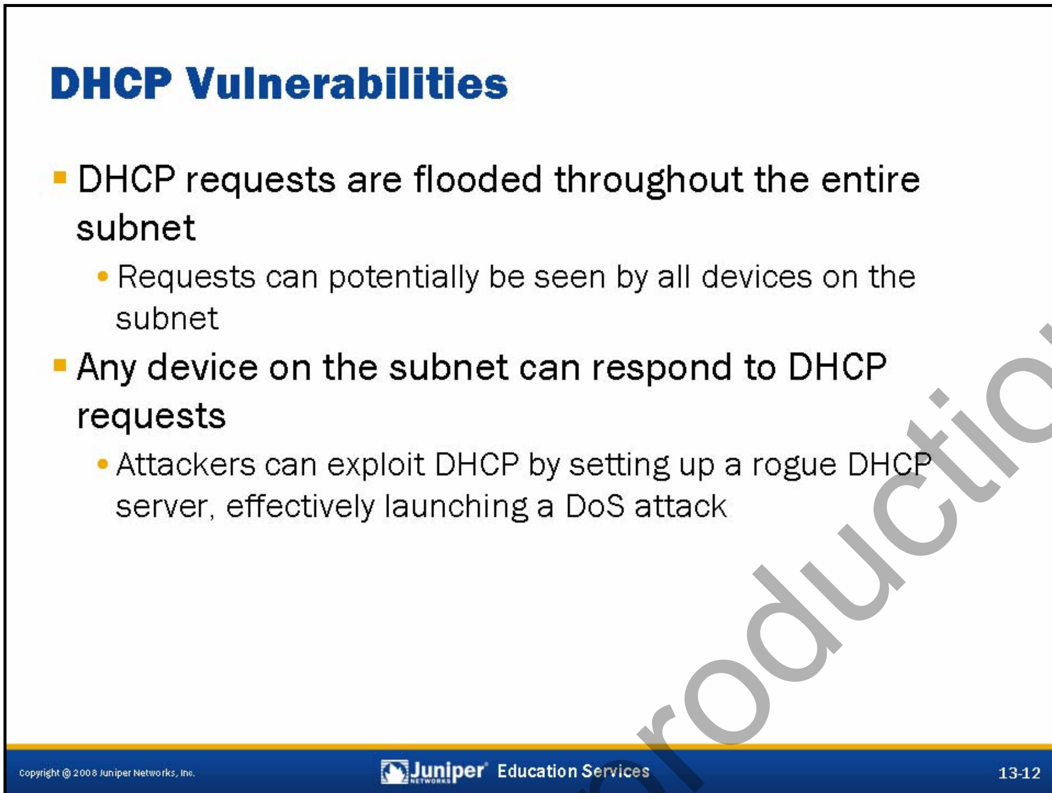
DHCP Snooping

The slide highlights the topic we discuss next.



DHCP Review

The Dynamic Host Configuration Protocol (DHCP) is used to dynamically configure hosts on a network. An administrator defines network parameters on a DHCP server. Based on individual requests from the DHCP clients, the DHCP server dynamically assigns network parameters that facilitate network access for the individual hosts, or DHCP clients. The slide illustrates the basic communication process between DHCP clients and a DHCP server, including the various messages types sent between clients and server.

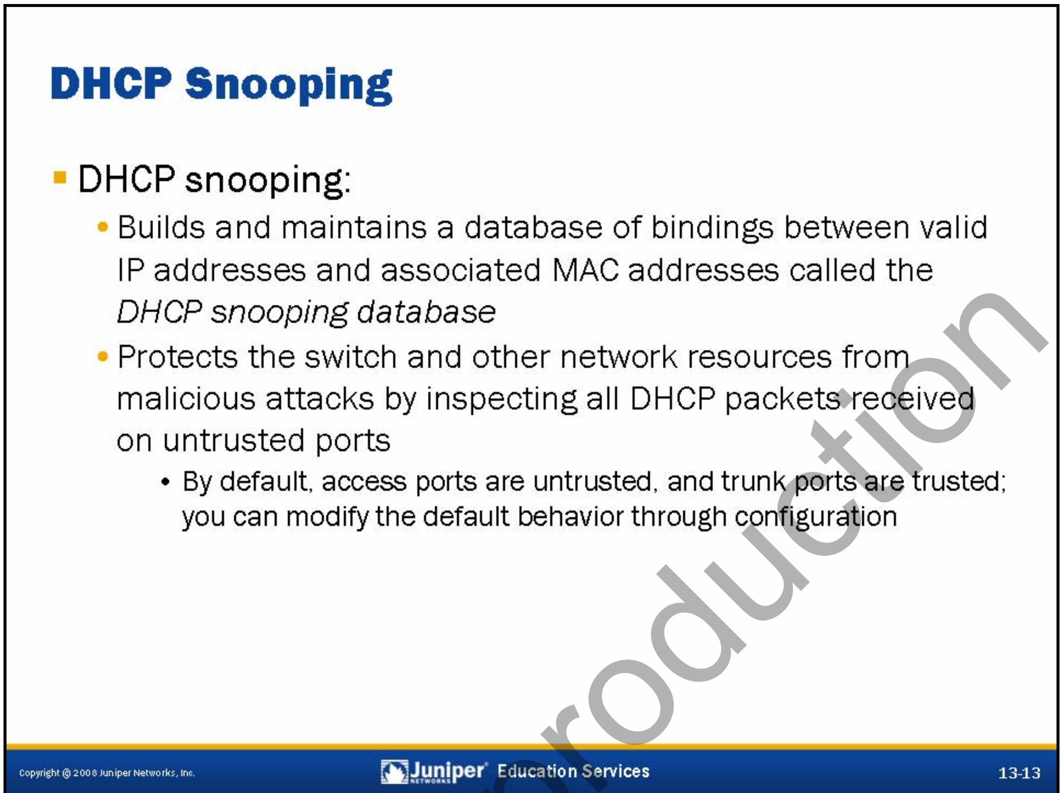


DHCP Requests

DHCP, like many other protocols, has inherent vulnerabilities, which can be exploited either intentionally or unintentionally. When a client sends a DHCP request, it is seen by all other devices participating on the subnet.

Who's Calling?

Because all DHCP requests can be viewed by any other device participating on the same subnet, it makes sense that any device on that subnet can also respond to that DHCP request. This inherent DHCP behavior facilitates opportunities for attackers to disrupt normal network operations and effectively launch a DoS attack. One such attack might include the use of a rogue DHCP server that responds to legitimate requests from authorized clients and provides bogus network parameters to those clients.



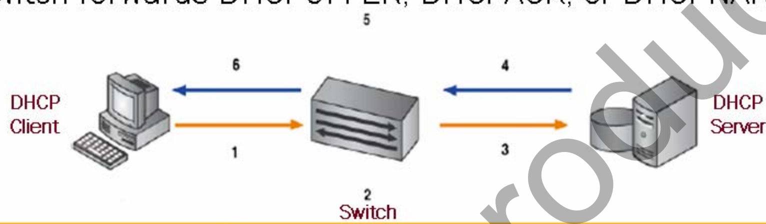
DHCP Snooping

DHCP snooping builds and maintains a database of valid IP addresses assigned to downstream network devices by a trusted DHCP server. DHCP snooping reads the lease information, which is sent from the DHCP server to the individual DHCP clients. From this information it creates the DHCP snooping database. This database is a mapping between IP address, MAC address, and the associated VLAN. When a DHCP client releases an IP address (by sending a DHCPRELEASE message), the associated mapping entry is deleted from the database. The switch also tracks the lease time, as assigned by the DHCP server, and purges all expired entries.

DHCP snooping protects the switch, as well as other network components, by inspecting all DHCP packets on untrusted ports. By default, JUNOS software treats access ports as untrusted and trunk ports as trusted. If a server is connected to a local access port, you must configure that port as a trusted port to accommodate the DHCP server traffic it receives. Note that DHCP snooping occurs only on interfaces for which an entry exists. If a switch port is connected to a device with a statically defined IP address, no inspection occurs.

DHCP Snooping Process

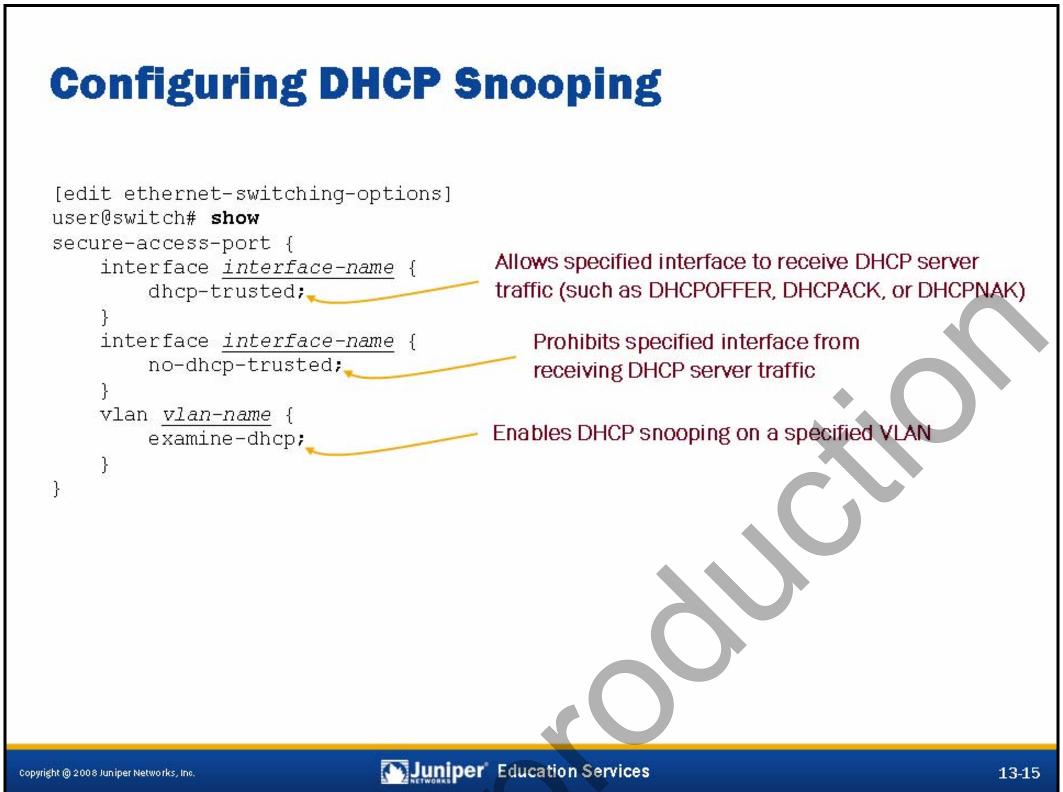
- DHCP snooping uses the following process:
  1. Device sends DHCPDISCOVER or DHCPREQUEST
  2. Switch snoops the packet and updates the snooping database
  3. Switch forwards DHCPDISCOVER or DHCPREQUEST
  4. Server sends DHCPOFFER, DHCPACK, or DHCPNAK
  5. Switch snoops the packet and updates the snooping database
  6. Switch forwards DHCPOFFER, DHCPACK, or DHCPNAK



 Juniper Education Services 13-14

DHCP Snooping Process

This slide illustrates the basic steps involved with the DHCP snooping process.



Configuring DHCP Snooping

This slide provides a basic DHCP snooping configuration example. This example shows the required configuration to enlist an access interface, which connects to a DHCP server as a trusted interface, as well as how to enable DHCP snooping on an individual VLAN.

Monitoring DHCP Snooping

- Use the `show dhcp snooping binding` command to view DHCP snooping database details:

```
user@switch> show dhcp snooping binding
DHCP Snooping Information:
MAC address            IP address   Lease    Type      VLAN    Interface
00:1B:C0:5E:53:A4      172.28.15.3  86271    dynamic   v100    ge-0/0/0.0
00:1B:C0:5E:53:A5      172.28.15.4  86271    dynamic   v100    ge-0/0/1.0
```

- Use the `clear dhcp snooping binding` commands to clear DHCP snooping database entries:

```
user@switch> clear dhcp snooping binding vlan vlan mac mac address

user@switch> clear dhcp snooping binding vlan vlan

user@switch> clear dhcp snooping binding
```

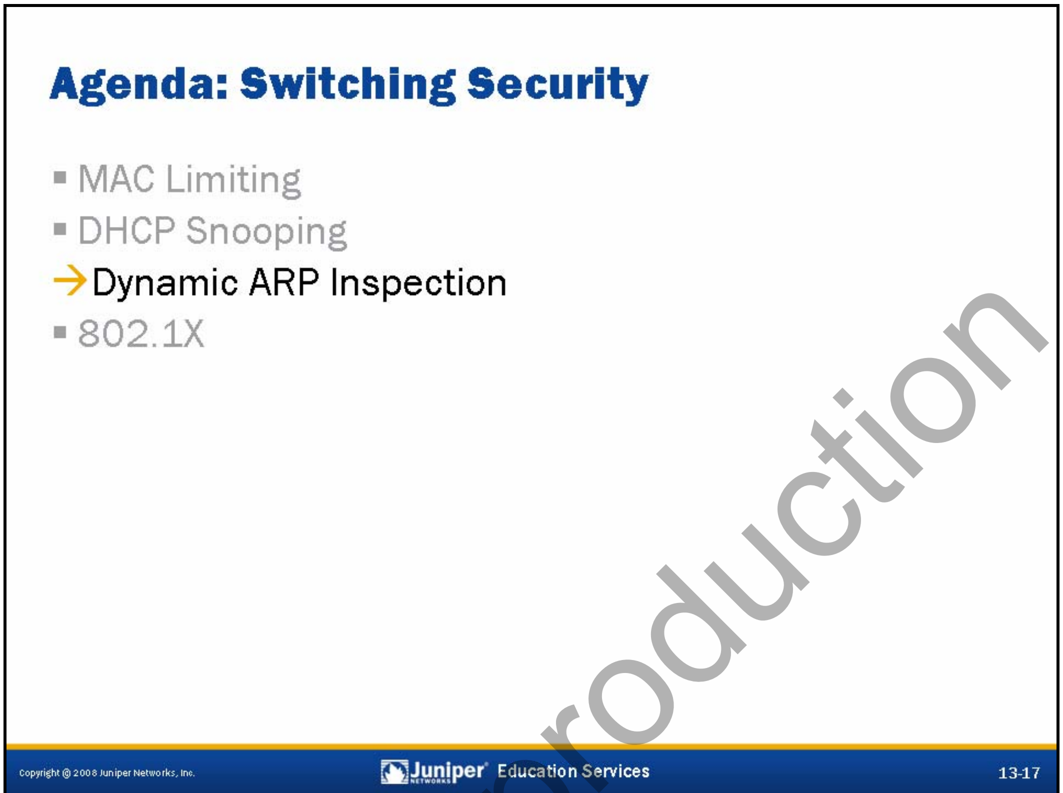
 Juniper Education Services 13-16

Viewing the DHCP Snooping Database

This slide displays the commands used to monitor DHCP snooping. Use the **`show dhcp snooping binding`** command to view the registered details within the DHCP snooping database.

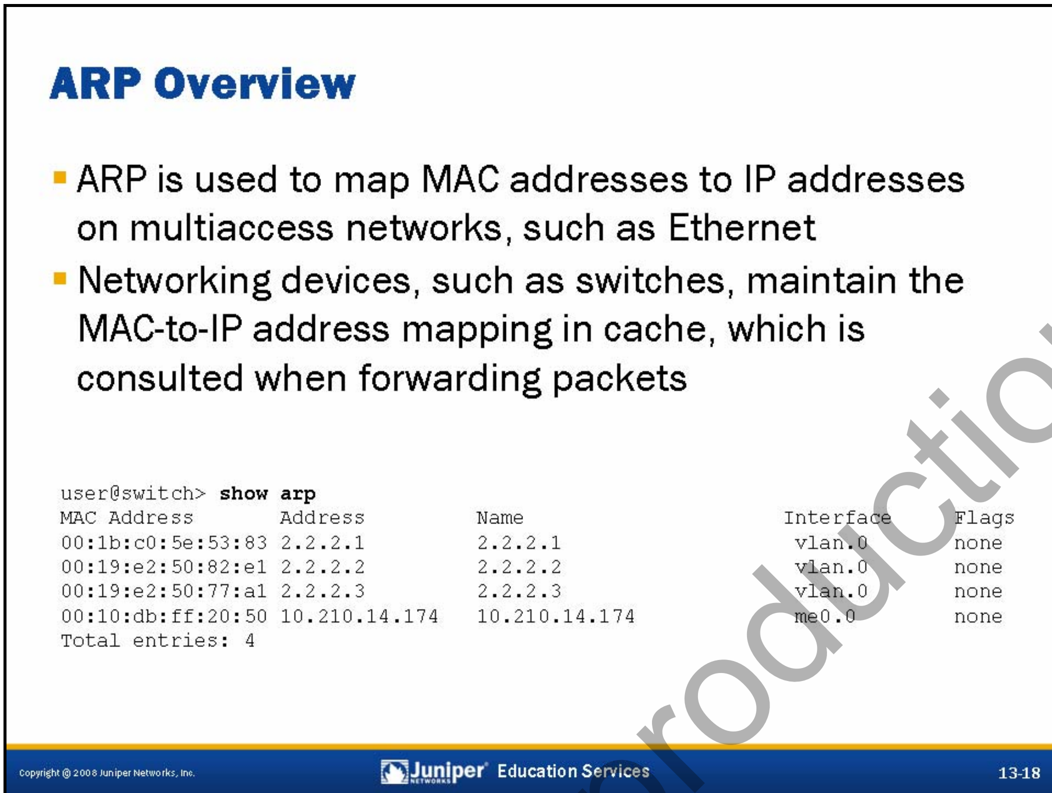
Clearing the DHCP Snooping Database

Use the **`clear dhcp snooping binding`** commands to clear entries within the DHCP snooping database. This command offers various options that allow the user to clear all entries, all entries for a particular VLAN, or individual entries within the DHCP snooping database.



Dynamic ARP Inspection

The slide highlights the topic we discuss next.

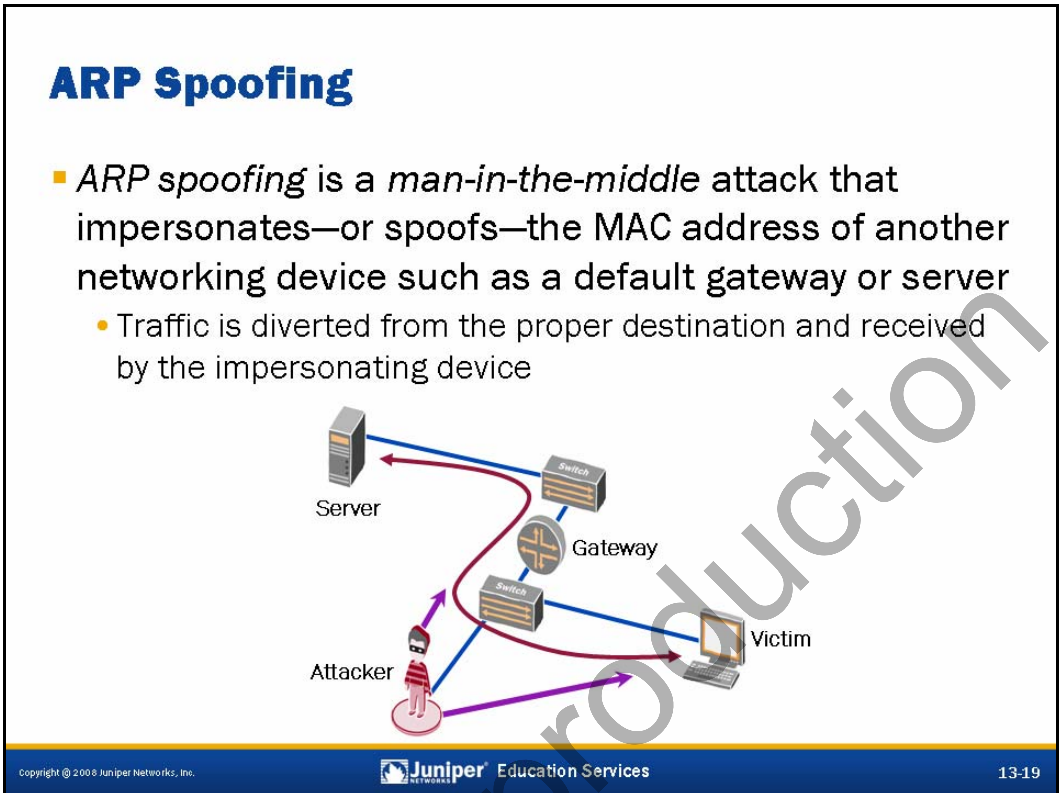


ARP Operation

Sending IP packets on a multiaccess network requires mapping an IP address to an Ethernet MAC address. Ethernet LANs use the Address Resolution Protocol (ARP) to map MAC addresses to IP addresses.

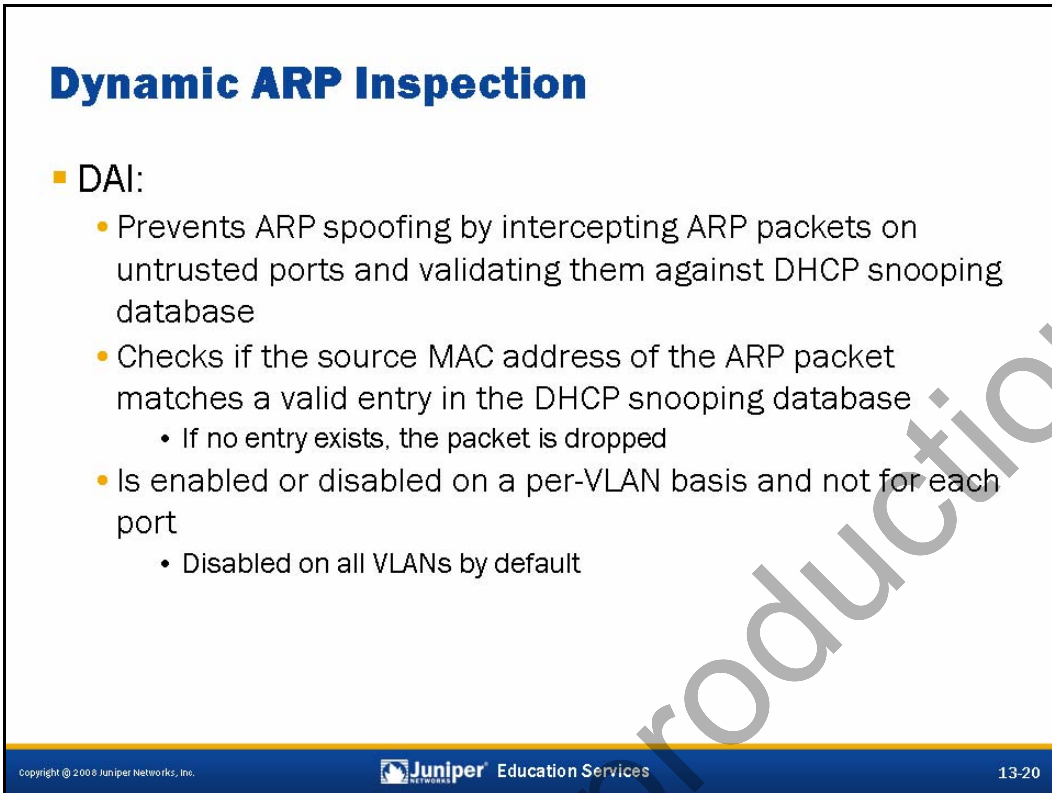
ARP Table

The switch, as well as other networking devices, maintains this mapping in a cache that it consults when forwarding packets to network devices. If the ARP cache does not contain an entry for the destination device, the host broadcasts an ARP request for that device's address and stores the response in the cache. This cache, detailing the ARP information, is often referred to as the ARP table. An example of an ARP table from an EX-series switch is shown on the slide.



ARP Spoofing

ARP spoofing, also known as *ARP poisoning*, is commonly used to initiate *man-in-the-middle* attacks. In these types of attacks, the attacker sends an ARP packet that spoofs the MAC address of another device on the LAN. Instead of the switch sending traffic to the proper network device, it sends the traffic to the impersonating device with the spoofed address. The result is that traffic from the switch is diverted from the proper destination and received by the impersonating device.



Dynamic ARP Inspection

Dynamic ARP Inspection (DAI) examines ARP requests and responses on the LAN. Each ARP packet received on an untrusted access port is validated against the DHCP snooping database. By validating each ARP packet received on untrusted access ports, DAI can prevent ARP spoofing.

If the DHCP snooping database does not contain an IP address-to-MAC address entry for the information within the ARP packet, DAI drops the ARP packet, thus preventing the propagation of invalid host address information. DAI also drops ARP packets when the IP address in the packet is invalid. Because DAI depends on the entries found within the DHCP snooping database, you must enable DHCP snooping. JUNOS software uses DAI for ARP packets received on access ports because these ports are untrusted by default. ARP packets bypass DAI on trunk ports because they are trusted interfaces.

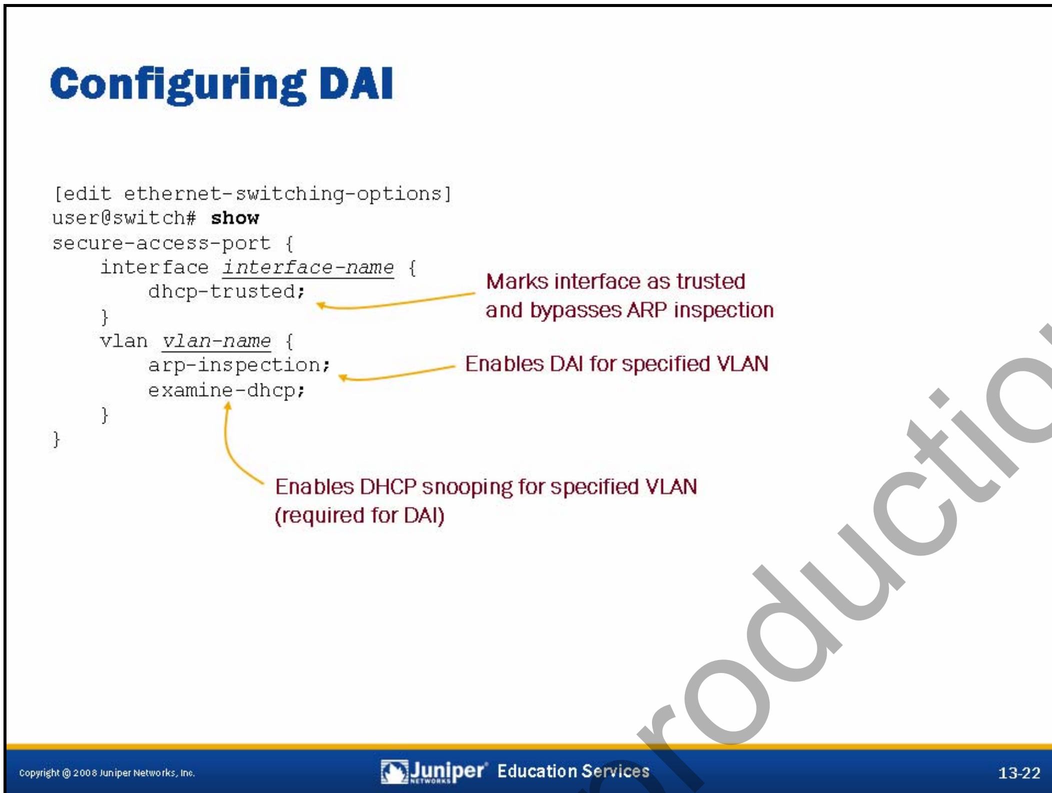
Continued on next page.

Dynamic ARP Inspection (contd.)

By default, DAI is disabled on EX-series switches. You enable DAI on individual VLANs and not for each port. If an access port is connected to a host with a statically defined IP address within a VLAN that has DHCP snooping and DAI enabled, you must configure that port as a trusted port to allow ARP packets to pass. You can set individual ports as trusted by adding the **dhcp-trusted** option on a given port, as shown in the following example:

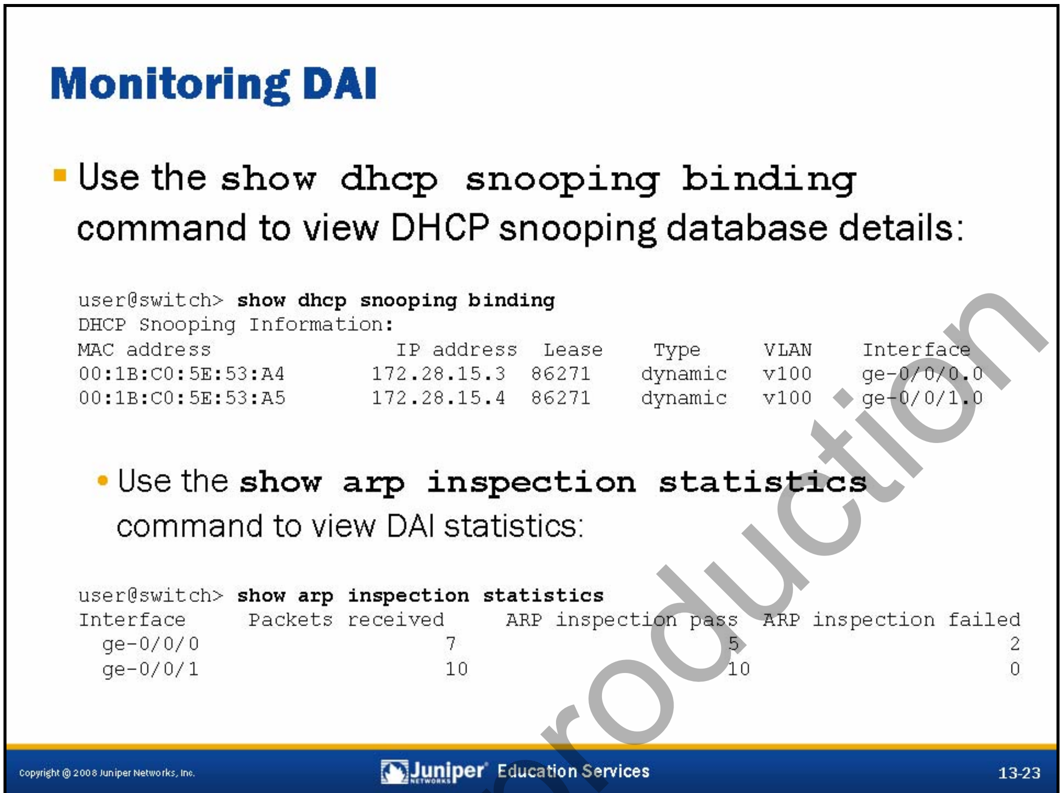
```
[edit ethernet-switching-options]
user@switch# show
secure-access-port {
    interface ge-0/0/20.0 {
        dhcp-trusted;
    }
}
```

JUNOS software broadcasts all ARP queries directed to the switch out all ports assigned to the associated VLAN. The software subjects ARP responses of those queries to the DAI check. ARP packets are sent to and reviewed by the RE. To prevent CPU overloading, JUNOS software rate-limits ARP packets destined for the RE.



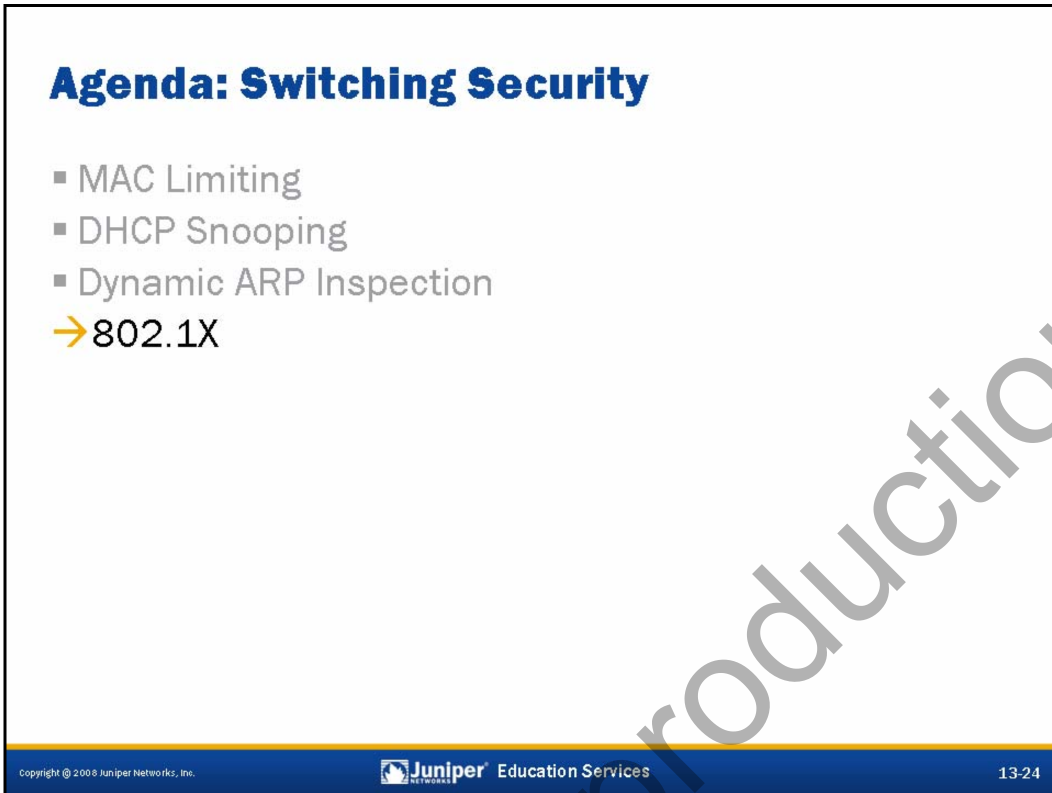
Configuring DAI

This slide illustrates a basic DAI configuration. As mentioned previously, DAI is configured on a per-VLAN basis.



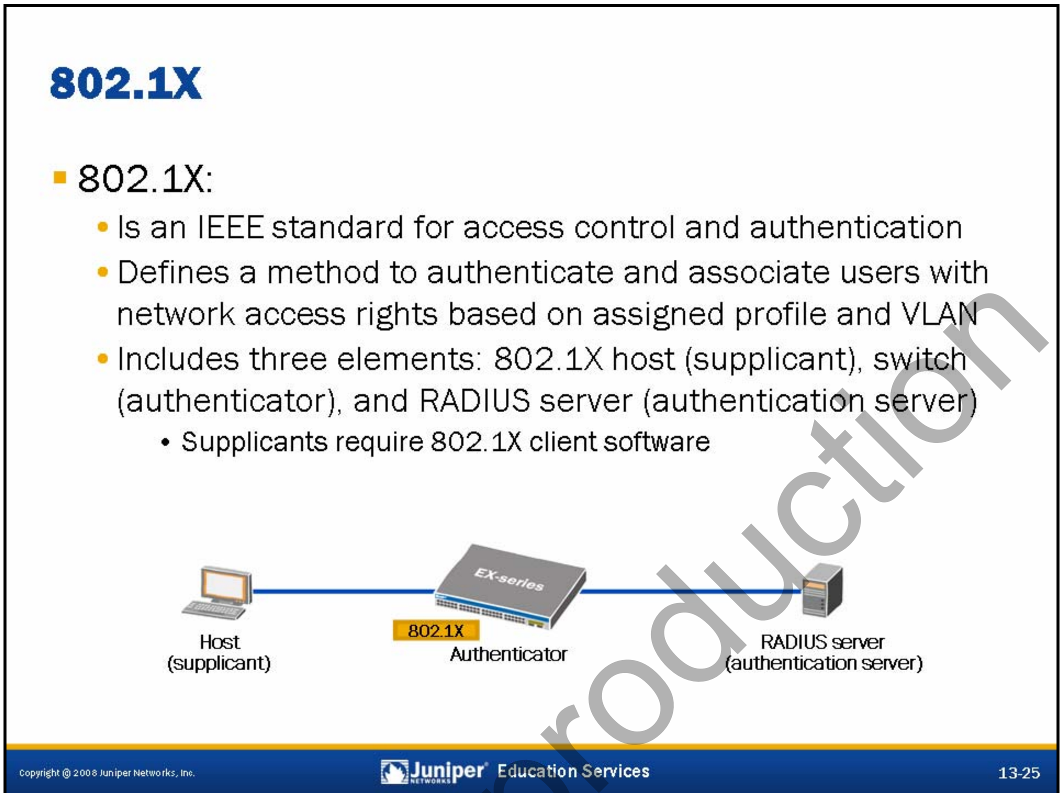
Monitoring DAI

This slide highlights some key commands used to monitor the operation of DAI. Use the **`show dhcp snooping binding`** command to view the recorded details within the DHCP snooping database. Use the **`show arp inspection statistics`** command to view DAI statistics.



802.1X

The slide highlights the topic we discuss next.



802.1X

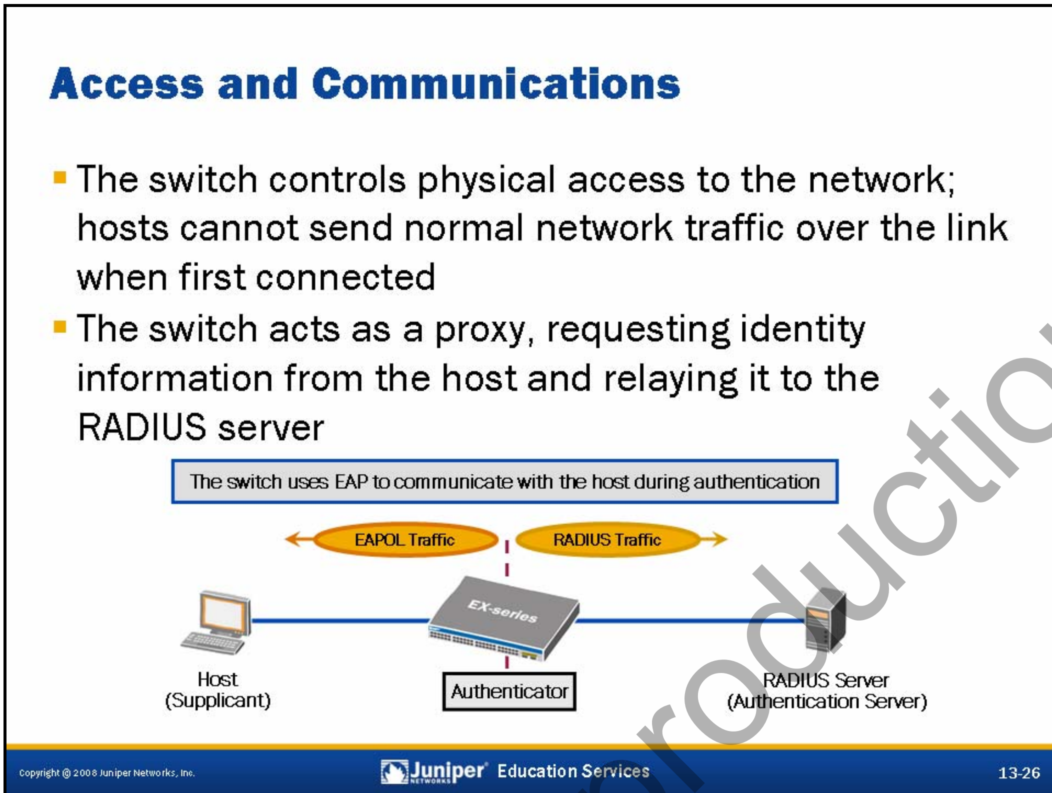
802.1X is an Institute of Electrical and Electronics Engineers (IEEE) standard used for port-level access control and authentication. 802.1X does not replace other security technologies; rather, it works together with port security features, such as DHCP snooping, DAI, and MAC limiting, to guard against DoS attacks and spoofing.

The 802.1X standard is based on the Extensible Authentication Protocol (EAP), a universal authentication framework. EAP is not an authentication mechanism by itself. Instead, EAP provides some common functions and a negotiation method to determine the authentication mechanism (EAP method) used between hosts and the authentication server. As individual hosts are authenticated, they can be associated with a specific profile and VLAN.

A LAN configured for 802.1X authentication contains three basic components:

- *Supplicant*: The device being authenticated. This device is typically a user's PC or an IP phone.
- *Authenticator*: The device that prevents a supplicant's access until it is authenticated. This device is a switch.
- *Authentication server*: The authenticating device. EX-series switches currently support RADIUS authentication servers for 802.1X.

To authenticate through 802.1X, supplicants require 802.1X client software. Some operating systems, such as Windows XP, include an 802.1X client by default.

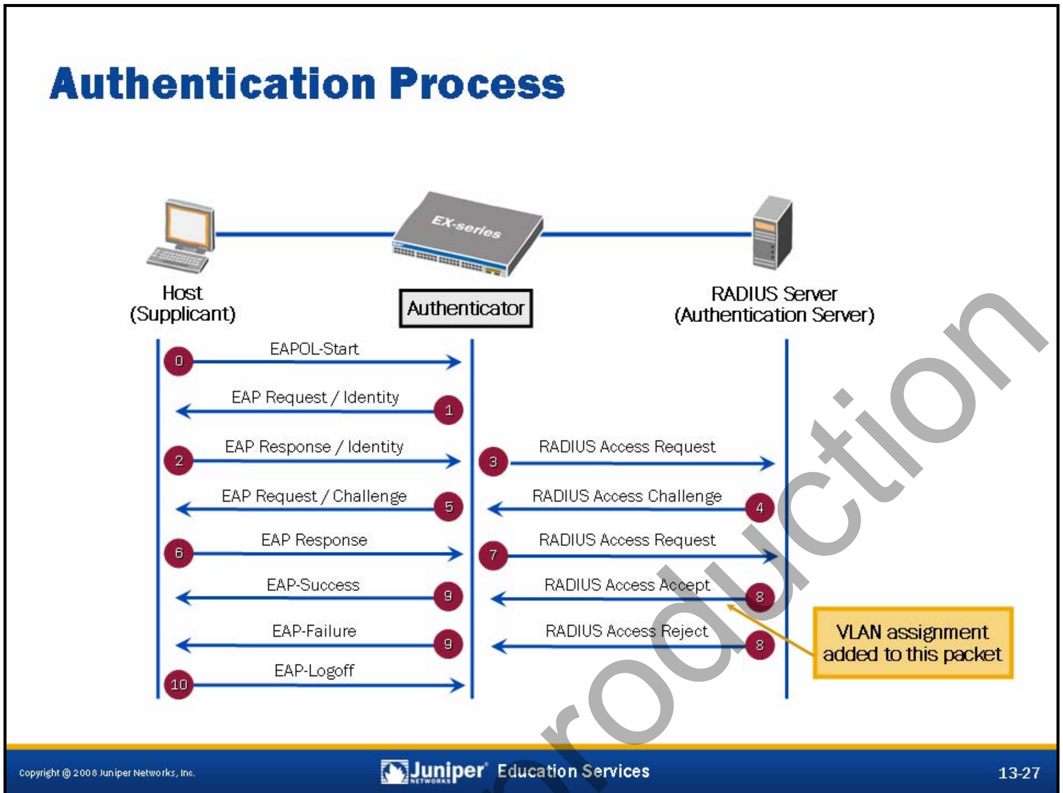


Controlling Access

EX-series switches, functioning as 802.1X authenticators, control network access by blocking all traffic to and from unauthorized supplicants. Access is granted only when the individual supplicants are properly authenticated. Supplicants request network access through their attached authenticator by sending and responding to EAP over LAN (EAPOL) messages. If an authenticated supplicant no longer requires access to the network, it notifies the authenticator, at which time the authenticator once again blocks network access through the associated network port.

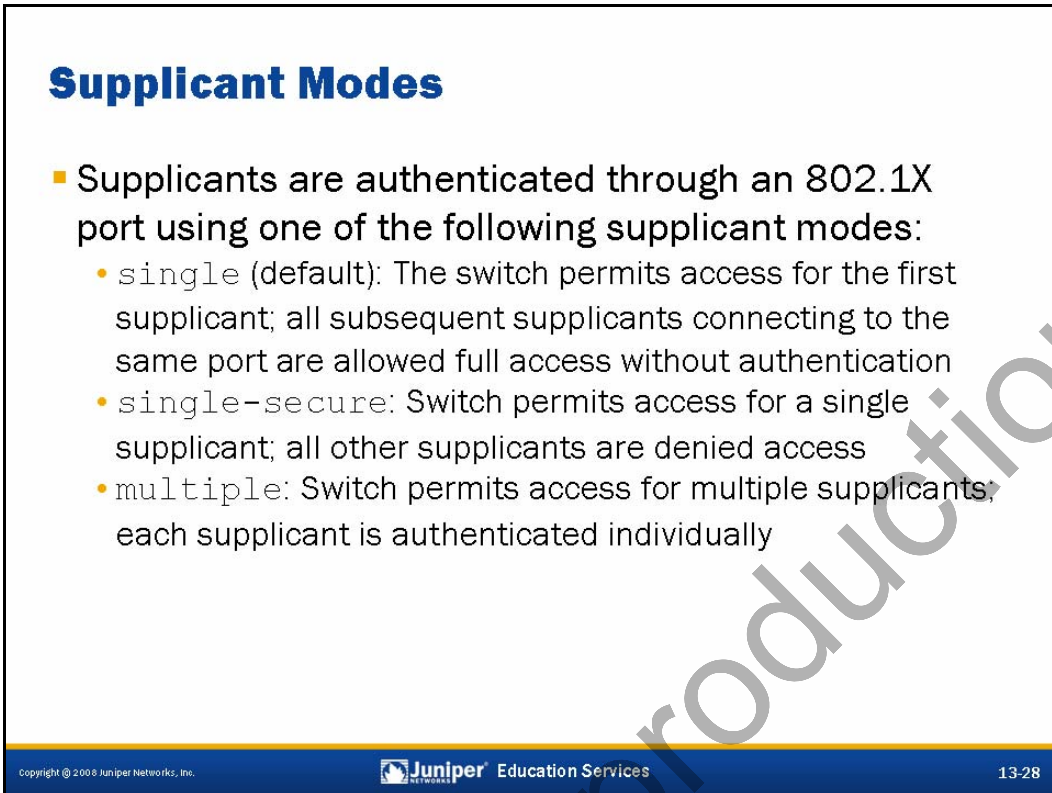
Relaying Information

When an authenticator receives authentication requests from a supplicant, those requests are received as EAPOL messages. The authenticator extracts and relays the identity information, found within the EAPOL message, to the authentication server as a RADIUS access request. The authenticator does not evaluate the supplicant's credentials but simply relays that information to the authenticating server in an understandable format.



Authentication Process

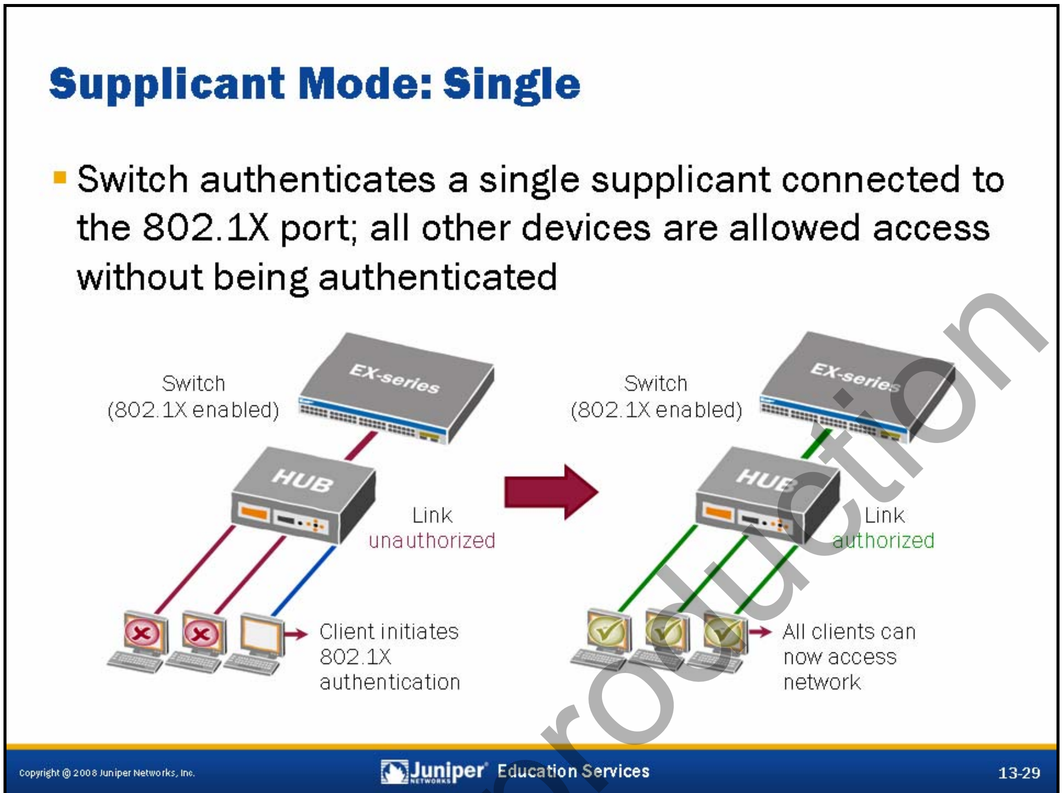
This slide shows the individual steps for the 802.1X authentication process.



Authenticating Supplicants

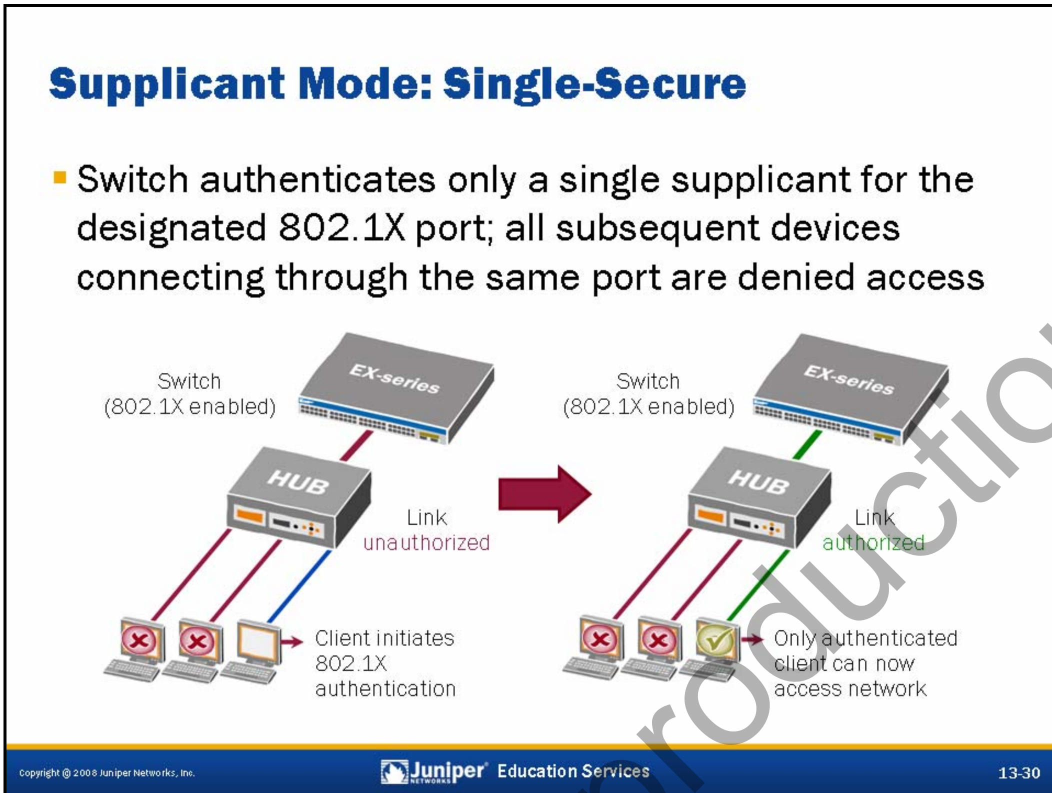
Although the authentication server performs the actual authentication process, it is up to the authenticator to facilitate network access for individual supplicants through the switch ports. Supplicants are authenticated in either `single` mode, `single-secure` mode, or `multiple` mode:

- `single`: Authenticates only the first supplicant. All other supplicants who connect later to the port are allowed full access without any further authentication. The subsequent supplicants effectively *piggyback* on the first supplicant's authentication. This is the default supplicant mode on EX-series switches. It is also the recommended mode when a user's PC and IP telephone use the same switch port and one of the supplicants does not support 802.1X.
- `single-secure`: Allows only one supplicant to connect to the port. No other supplicant is allowed to connect until the first supplicant logs out.
- `multiple`: Allows multiple supplicants to connect to the port. Each supplicant is authenticated individually.



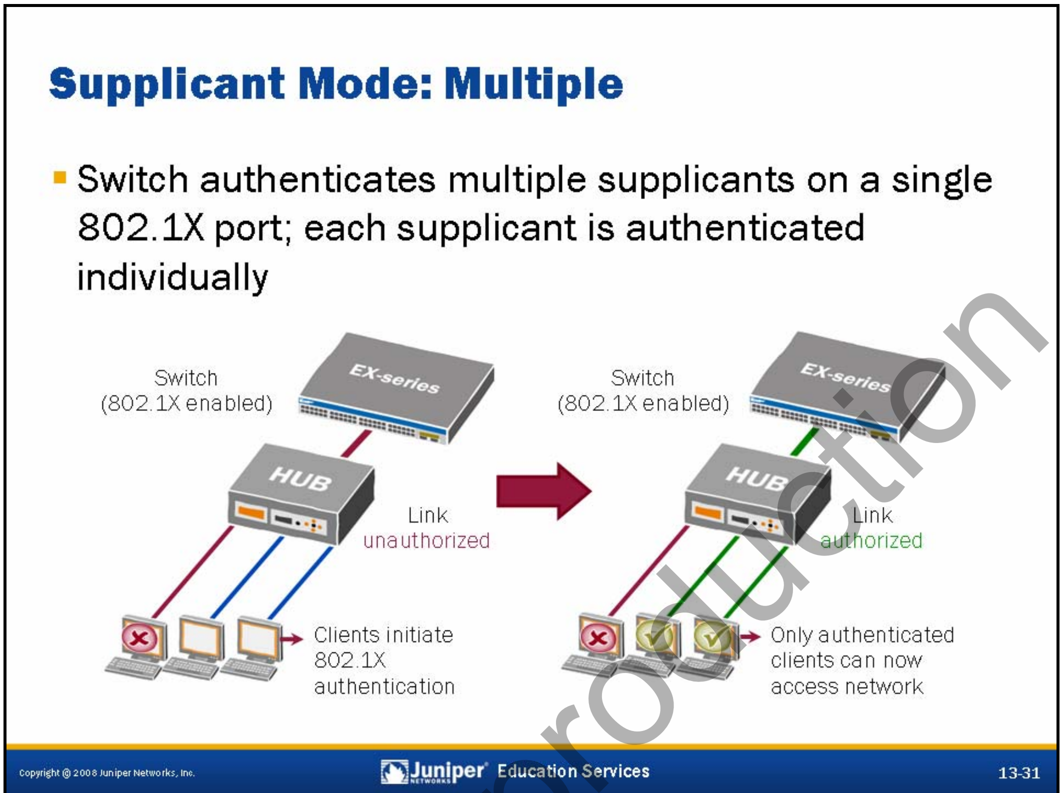
Single Supplicant Mode

This slide illustrates the `single` supplicant mode, which is the default supplicant mode for EX-series switches.



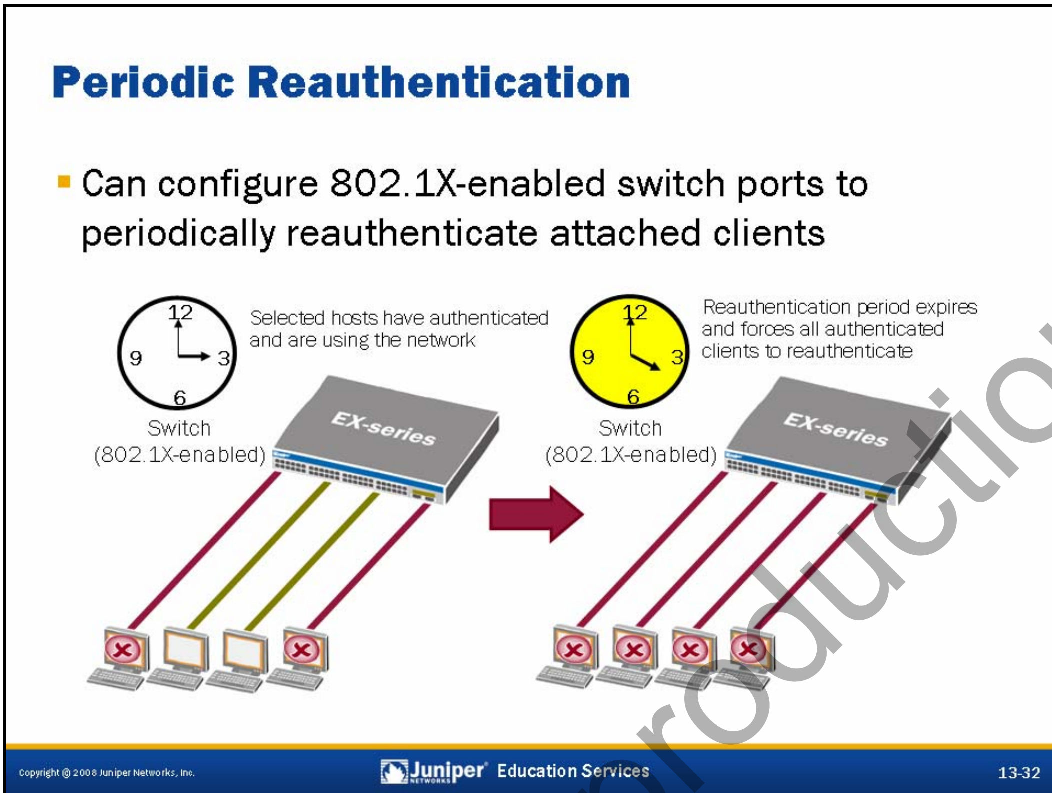
Single-Secure Supplicant Mode

This slide highlights the `single-secure` supplicant mode.



Multiple Supplicant Mode

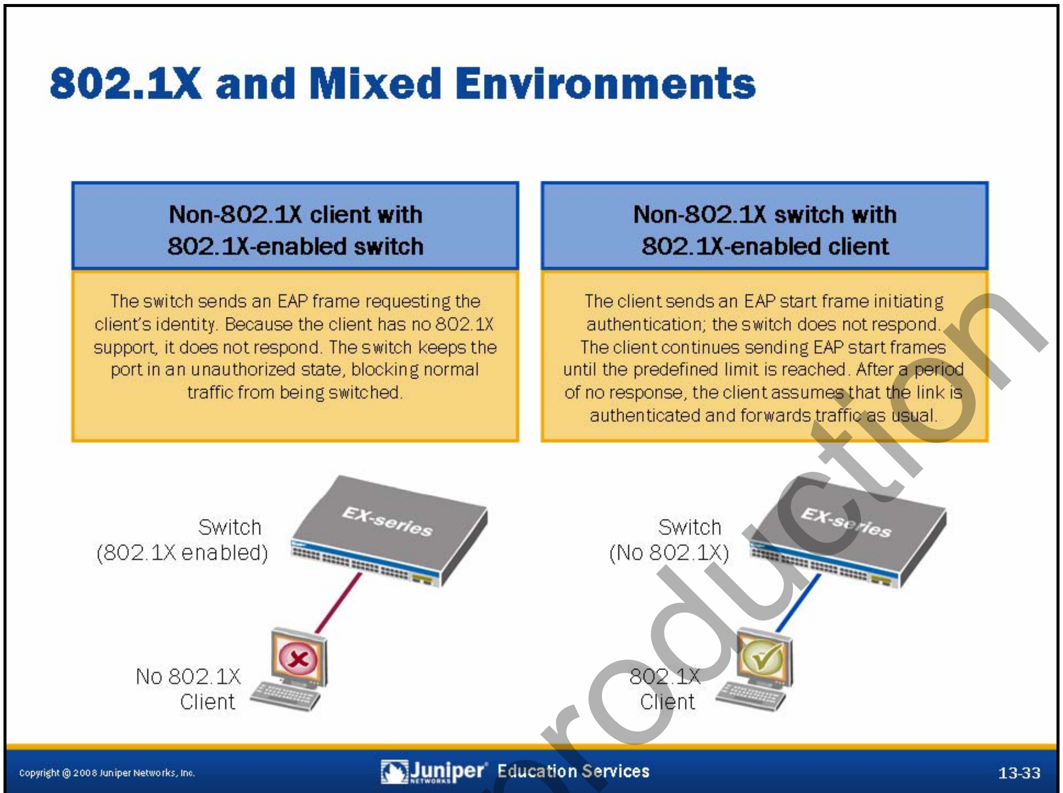
This slide highlights the `multiple` supplicant mode. The `multiple` supplicant mode overcomes the security concerns of the `single` supplicant mode while providing more flexibility than the `single-secure` mode.



Periodic Reauthentication

By default, EX-series switches functioning as 802.1X authenticators force all authenticated supplicants to periodically reauthenticate. The default reauthentication interval is 3600 seconds (1 hour). You can disable reauthentication or modify the reauthentication interval for individual ports at the `[edit protocols dot1x authenticator interface interface-name]` configuration hierarchy level. The reauthentication interval range is from 1 to 65535 seconds. A configuration example follows:

```
[edit protocols dot1x authenticator]
user@switch# show
interface {
    ge-0/0/0.0 {
        reauthentication 3600;
    }
    ge-0/0/1.0 {
        no-reauthentication;
    }
}
```

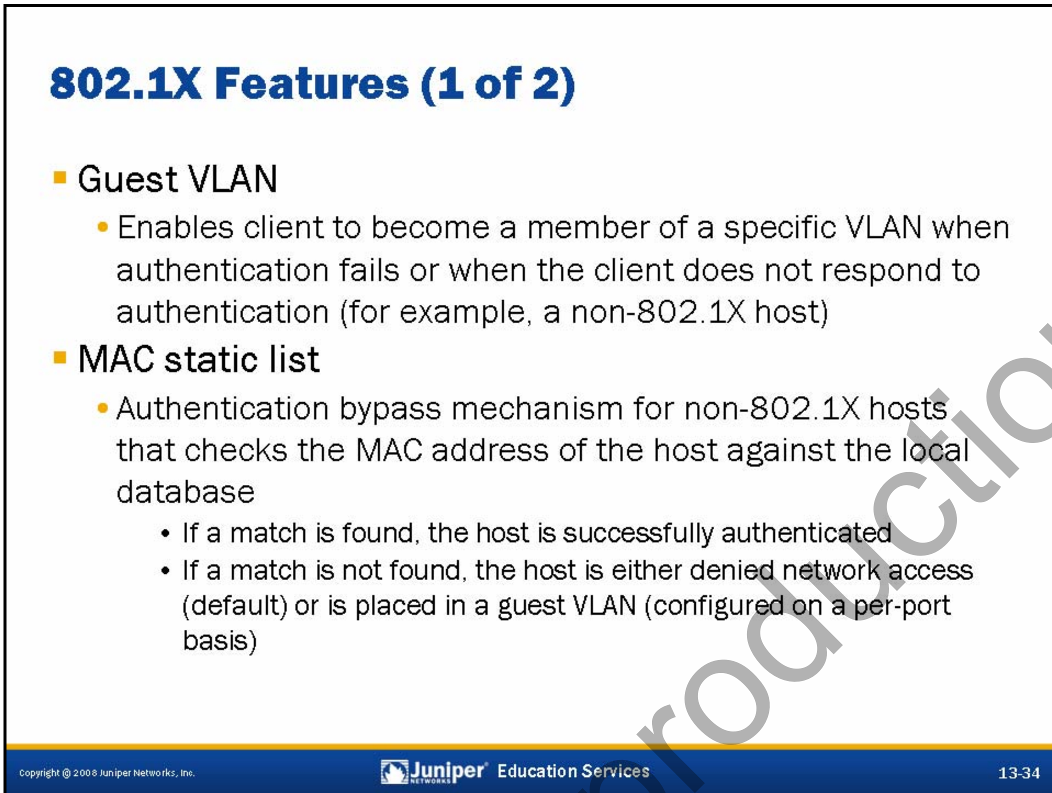


802.1X and Mixed Environments

The slide addresses two common scenarios in which either the supplicant (client) or the authenticator (switch) is not enabled for 802.1X, whereas the other component is enabled for 802.1X.

In the first scenario, the supplicant does not support 802.1X, but the authenticator does support 802.1X. In this case, the authenticator sends an EAP frame requesting the supplicant's identity. Because the supplicant has no 802.1X support, it does not respond. The authenticator keeps the port in an unauthorized state, blocking normal traffic from being switched.

In the second scenario, the authenticator does not support 802.1X, but the supplicant does support 802.1X. In this case, the supplicant sends an EAP start frame to initiate authentication. Because the authenticator is not configured for 802.1X, or does not support 802.1X, it does not respond. The supplicant continues sending EAP start frames until it reaches the predefined limit of attempts. The number of start attempts might vary depending on the supplicant client software. Once the predefined limit is reached, the supplicant assumes that the link is authenticated and begins forwarding traffic.



Guest VLAN

A guest VLAN provides limited access to a LAN—typically just to the Internet—for supplicants that fail 802.1X authentication. This feature is often configured on access ports on which visitors might connect. You enable the guest VLAN on individual network ports under the `[edit protocols dot1x authenticator interface]` hierarchy level. A sample configuration follows:

```
[edit protocols dot1x authenticator interface]
user@switch# show
ge-0/0/2.0 {
    guest-vlan free4all;
}
```

We defined the referenced *free4all* guest VLAN in this example under the `[edit vlans]` hierarchy level as shown in the following configuration:

```
[edit vlans]
user@switch# show
free4all {
    vlan-id vlan-id;
}
```

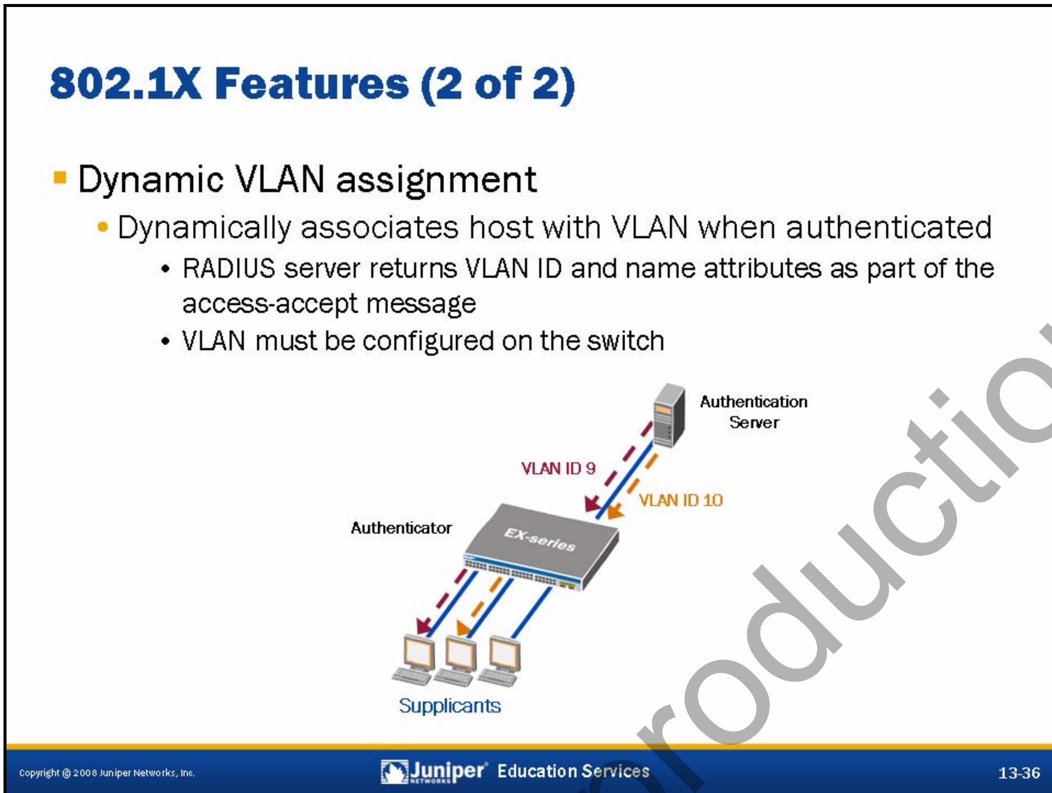
Continued on next page.

MAC Static List

You can choose to define a list of static MAC addresses as an 802.1X authentication bypass mechanism. This authentication option is typically used when a device does not support 802.1X. When a device does support 802.1X and is listed in the static MAC address bypass list, it will be authenticated by the switch, rather than through 802.1X by a RADIUS server.

When the static MAC bypass mechanism is used, the switch checks the supplicant's MAC address against the list of MAC addresses permitted through the static MAC address bypass list. If a match is found, the host is successfully authenticated. If a match is not found, the host is either denied network access (default) or is granted access through the guest VLAN, if configured. The MAC addresses can be bound to a specific network port or associated with any network port to facilitate user mobility. The following example configuration illustrates both scenarios:

```
[edit protocols dot1x authenticator static]
user@switch# show
00:1b:c0:5e:53:a1 {
    interface ge-0/0/2.0;
}
00:1b:c0:5e:01:b2;
```



Dynamic VLAN Assignment

802.1X provides the ability to dynamically associate supplicants with a designated VLAN during the authentication process. You can configure the RADIUS server to return VLAN attributes as part of the access-accept message. For proper operation, the same VLAN assigned to the supplicant's port by the RADIUS server must also be configured on the switch.

Configuring 802.1X (1 of 2)

```
[edit protocols dot1x]
user@switch# show
authenticator {
   authentication-profile-name profile-name;
   static {
        mac-address {
            vlan-assignment (vlan-name | vid);
            interface interface-name;
        }
    }
    interface {
        interface-name {
            disable;
            supplicant (single | single-secure | multiple);
            reauthentication seconds;
            no-reauthentication;
            guest-vlan (vlan-name | vid);
        }
    }
}
```

Referenced authentication profile is defined under the `[edit access]` hierarchy

Authentication bypass mechanism

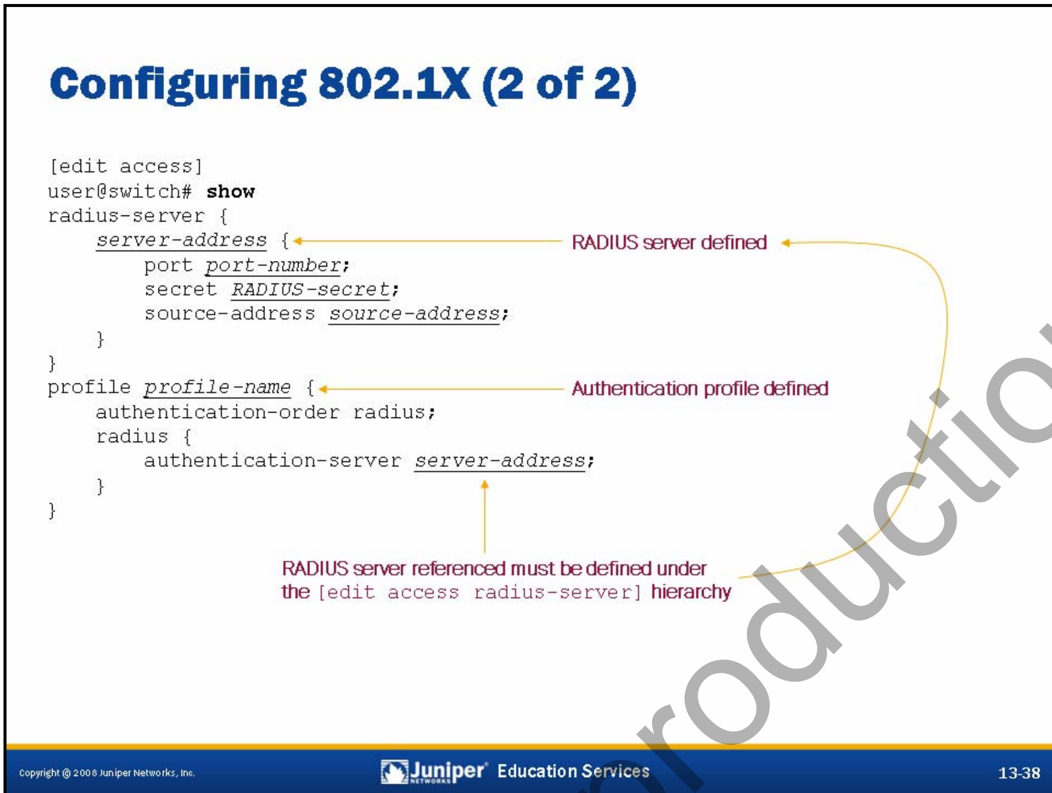
802.1X port settings

Juniper Education Services

13-37

Configuring 802.1X: Part 1

This slide illustrates some of the configuration options found under the `[edit protocols dot1x]` configuration hierarchy level. The authentication profile referenced in the sample configuration is defined under the `[edit access]` configuration hierarchy and is shown on the next slide.



Configuring 802.1X: Part 2

This slide shows the remainder of the 802.1X configuration example. In the sample configuration, we see the definition of a RADIUS server and an authentication profile. As shown on the previous slide, the authentication profile is referenced under the `[edit protocols dot1x authenticator]` configuration hierarchy level.

Monitoring 802.1X

- Operational-mode commands:
  - Use the **show dot1x interface** command to verify 802.1X status for individual interfaces:

```
user@switch> show dot1x interface
```

  - Use the **show dot1x static-mac-address** command to view static MAC address bypass details:

```
user@switch> show dot1x static-mac-address
```

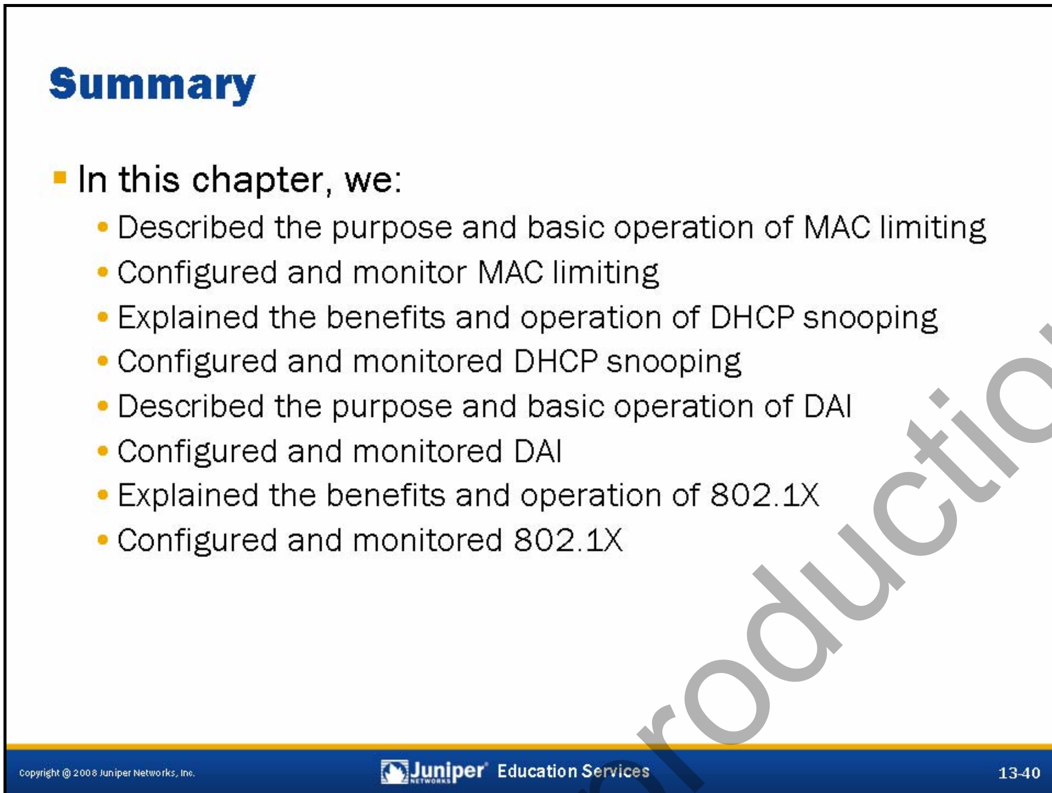
  - Use the **show dot1x authentication-failed-users** command to view failed user information:

```
user@switch> show dot1x authentication-failed-users
```


Juniper Education Services
13-39

Monitoring 802.1X

This slide displays the common operational-mode commands used to monitor 802.1X.



This Chapter Discussed:

- The purpose and basic operation of MAC limiting;
- Configuring and operational monitoring of MAC limiting;
- The benefits and operation of DHCP snooping;
- Configuring and monitoring DHCP snooping;
- The purpose and basic operation of DAI;
- Configuring and monitoring DAI;
- The benefits and operation of 802.1X; and
- Configuring and monitoring 802.1X.

Review Questions

1. How does MAC limiting protect against MAC flooding and MAC spoofing?
2. What is the purpose of DHCP snooping, and how does it work?
3. What is ARP spoofing, and what feature is available to protect against it?
4. Describe the 802.1X authentication process.
5. What supplicant modes are available? Describe each mode.

Juniper Networks Education Services
13-41

Review Questions:

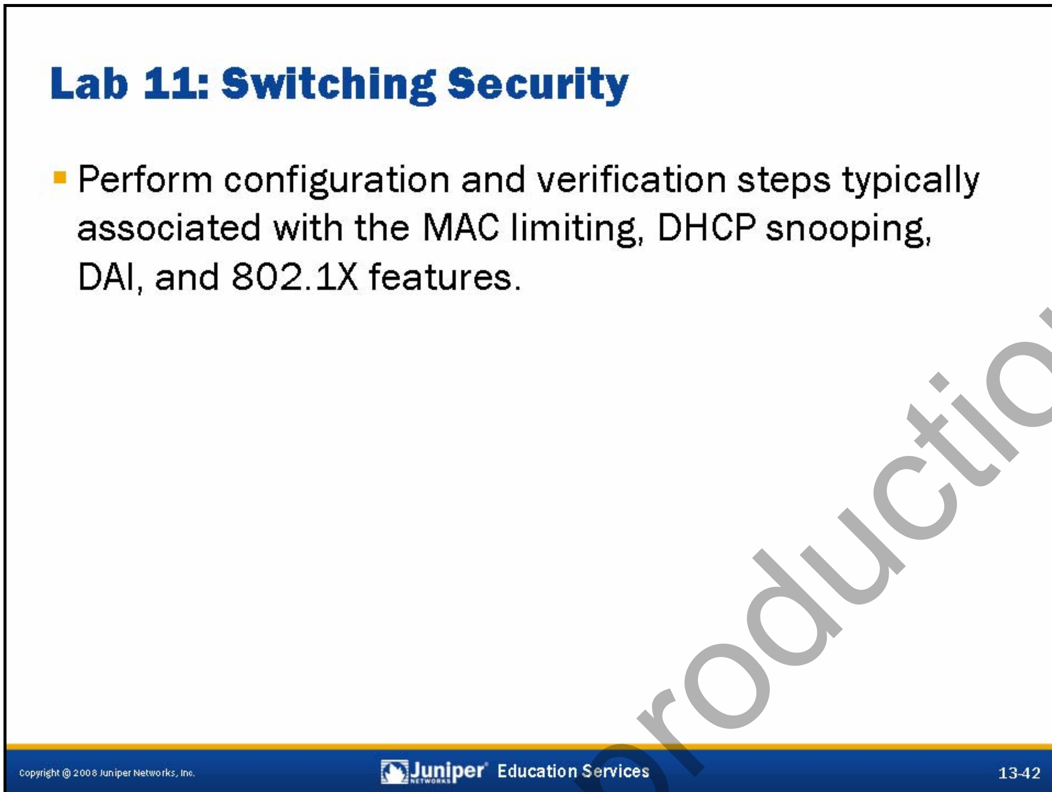
1.

2.

3.

4.

5.

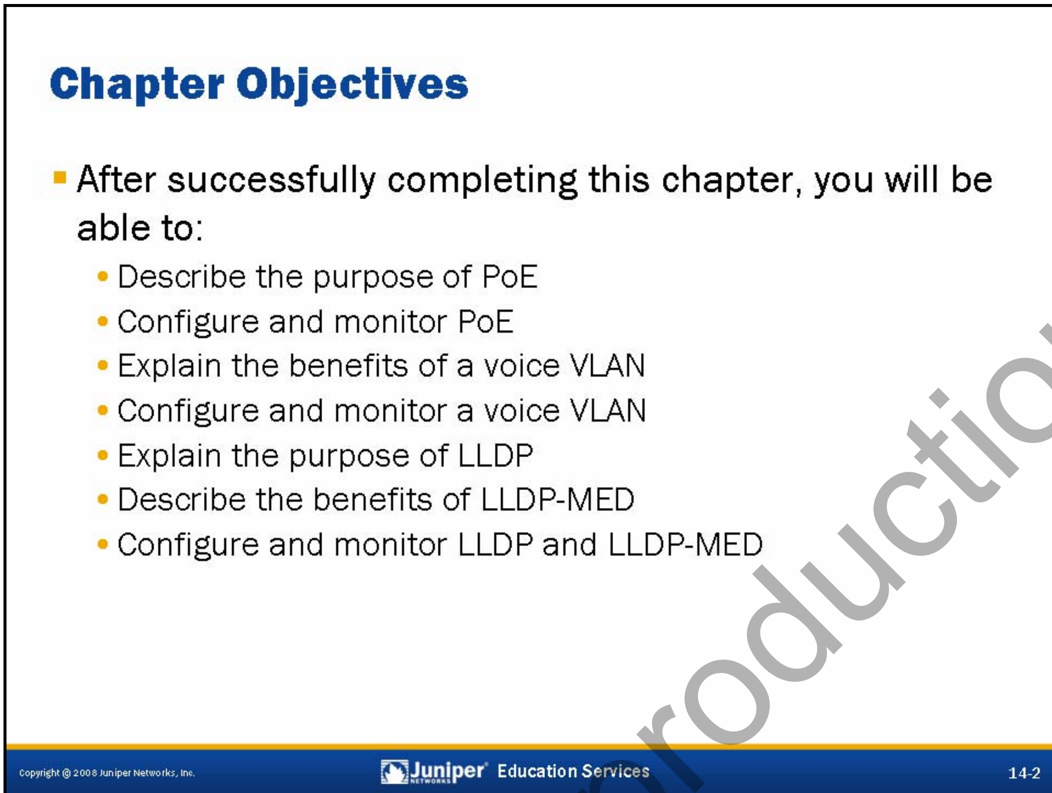


Lab 11: Switching Security

The slide provides the objective for this lab.

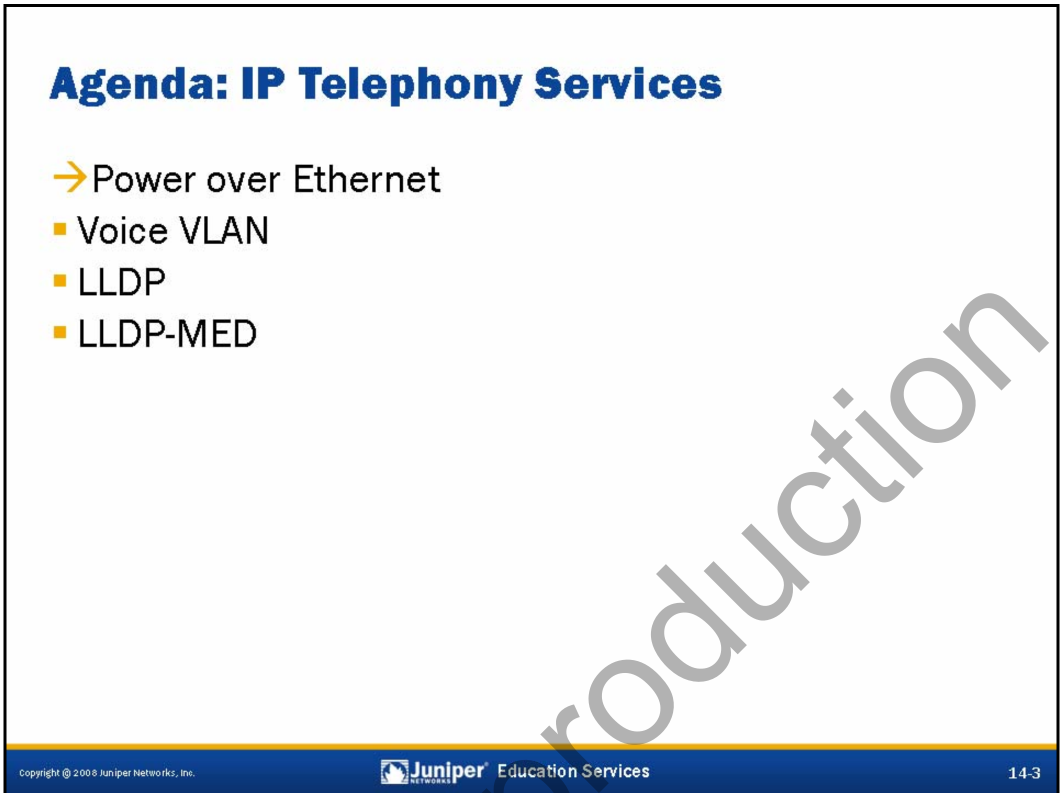
Operating Juniper Networks Switches in the Enterprise

Chapter 14: IP Telephony Services



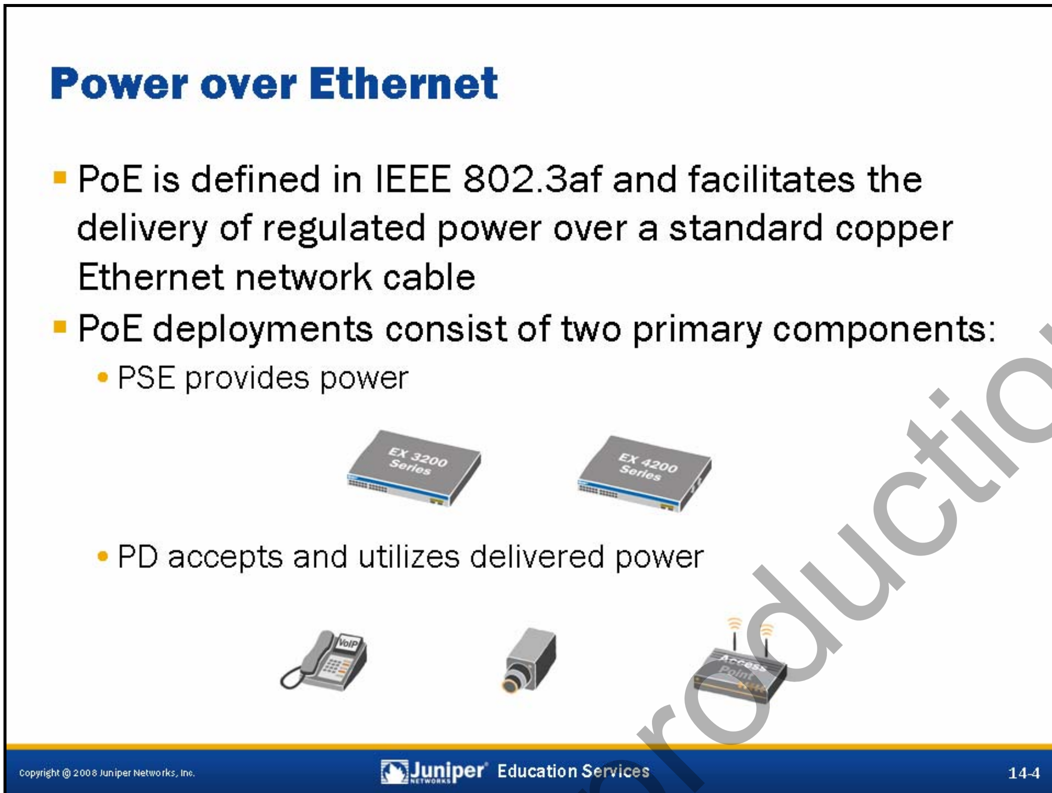
This Chapter Discusses:

- The purpose of Power over Ethernet (PoE);
- Configuring and monitoring PoE;
- The benefits of a voice virtual LAN (VLAN);
- Configuring and monitoring a voice VLAN;
- The purpose of the Link Layer Discovery Protocol (LLDP);
- The benefits of the Link Layer Discovery Protocol-Media Endpoint Discovery (LLDP-MED); and
- Configuring and monitoring LLDP and LLDP-MED.



Power over Ethernet

The slide lists the topics we cover in this chapter. We discuss the highlighted topic first.

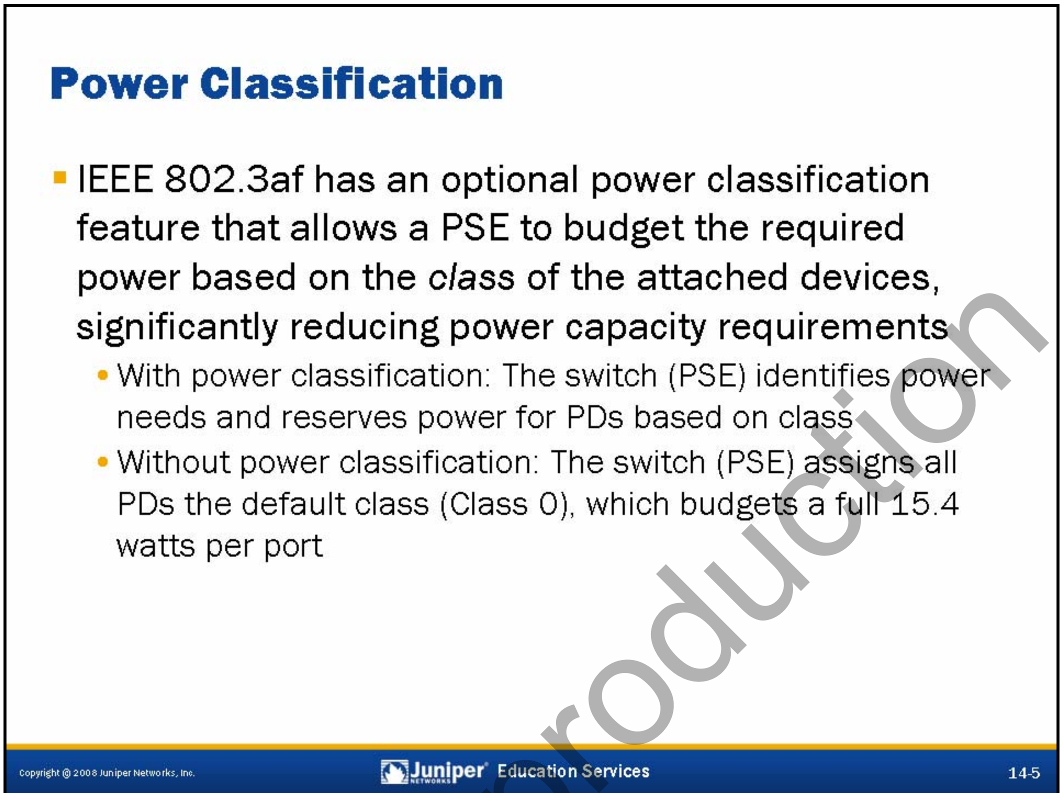


Power over Ethernet Defined

Power over Ethernet (PoE) is defined in the Institute of Electrical and Electronics Engineers (IEEE) 802.3af specification. PoE allows both data and electric power to pass over a copper Ethernet LAN cable. This technology allows voice over IP (VoIP) telephones, wireless access points, video cameras, and point-of-sale devices to safely receive power from the same access ports through which network access is provided. EX-series switches provide either 8, 24, or 48 PoE ports.

PoE Components

A PoE deployment consists of two primary components: the power sourcing equipment (PSE) and the powered device (PD). The PSE is any device that provides power in a PoE implementation. EX-series switches are considered PSE devices because they include PoE ports. A PD is a device powered by a PSE. These devices might include VoIP telephones, wireless access points, video cameras, and point-of-sale devices.



Power Classification

An optional power classification is included in the 802.3af specification. This power classification allows the PSE to allocate the needed power based on the attached device’s class. This classification option can greatly reduce the power capacity requirement for the PSE providing the power.

When power classification is used, the switch (which acts as the PSE) identifies the power requirements for each PD based on the device’s class. Once the device’s class is identified, the switch provides the required power. This approach efficiently allocates available power based on actual requirements. Without power classification, all PDs are assigned to the default class (Class 0) and allocated a full 15.4 watts, even when it is not required. This approach is less efficient and inevitably wastes power from the total available pool.

Power Class Definitions

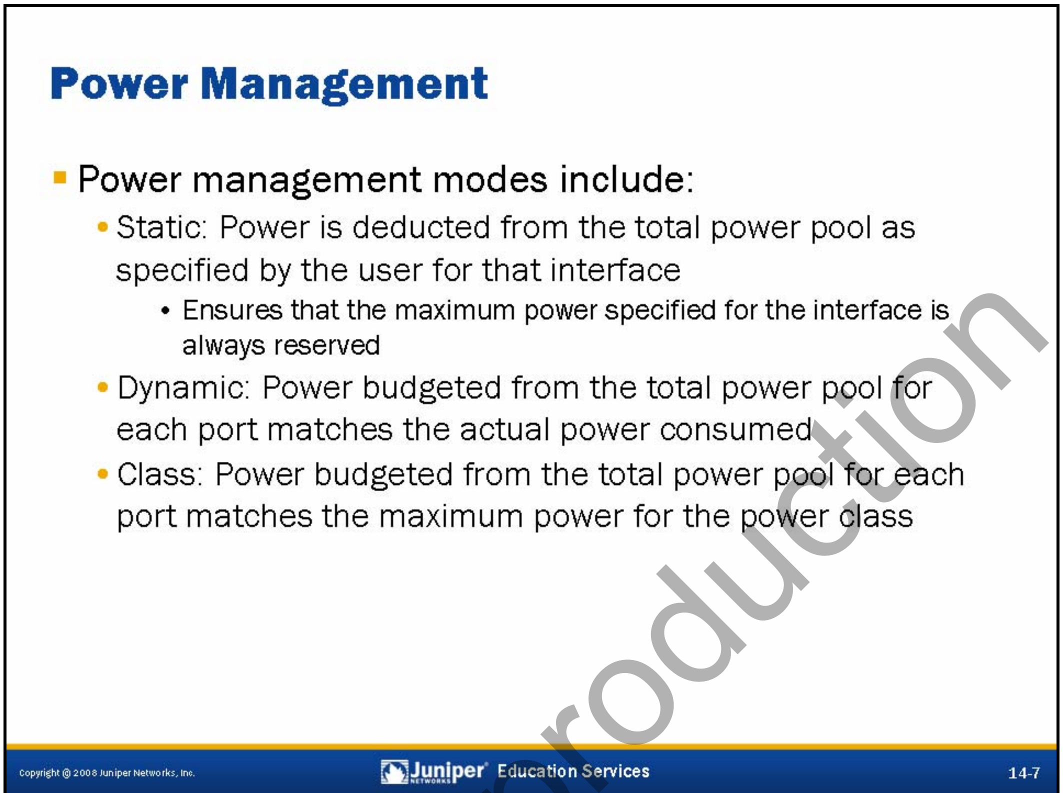
Class Number	Maximum Power at Output Port of PSE
0 (Default)	15.4 watts reserved (Actual device requirement can be less)
1	4 watts
2	7 watts
3	15.4 watts
4	Future expansion

 14-6

Power Class Definitions

A PD is classified based on the maximum power that it draws across all input voltages and operational modes. The most common class is Class 0, which is the default class for EX-series switches. With Class 0, the switch allows a maximum draw of 15.4 watts per port. The switch provides 15.4 watts at the port to guarantee enough power to run a device, after accounting for line loss. Line loss can reduce the total amount of received power by up to 16 percent, which leaves approximately 12.95 watts for the PD. This amount of power should be sufficient for all IEEE 802.3af-compliant devices requiring PoE service.

Note that Class 0 and Class 3 appear to be identical and offer the same maximum power level. However, IEEE created both classes to account for distinct situations. Class 0 can be used for all powered devices, including inexpensive, unsophisticated PDs that cannot be assigned to Classes 1, 2, or 3. Class 3 can be used only with 802.3af-compliant PDs capable of being assigned to Classes 1, 2, or 3. The slide lists all power classes and their associated power levels.



Power Management Modes

Three distinct power management modes are possible. These power management modes include static mode, dynamic mode, and class mode:

- The *static* power management mode allows you to specify the desired amount of power for individual ports. The power allocated is deducted from the overall pool of available power within the system. This power management mode ensures that the desired power level specify for each port is always available.
- The *dynamic* power management mode is more flexible than the static mode because the power budgeted matches the actual power consumed. As the name indicates, this mode is dynamic in nature and consumes only what is actually needed.
- The *class* power management mode uses the classification associated with the PD when determining the required power budget.

Currently, EX-series switches support the static power management mode with a default class of 0, which means all PDs receive a full 15.4 watts of power. As long as the appropriate power supply is installed in an EX-series switch, ample power should be available for all devices requiring PoE service.

Power Pool Usage Guidelines

- Three power supply capacities:
  - 320 W, 600 W, and 930 W
- Any power supply can be installed in any switch model
- Installing a higher capacity power supply does not increase the number of PoE ports on the switch

Power Supply Capacity	Switch Consumption	Power Budget for PoE Pool
320 W	190 W	130 W (8 PoE ports at 15.4 W each)
600 W	190 W	410 W (24 PoE ports at 15.4 W each)
930 W	190 W	740 W (48 PoE ports at 15.4 W each)

 Juniper Education Services 14-8

Power Supply Capacities

With the exception of the 24-port 100Base-FX/1000Base-X SFP fiber platform, all EX 3200 and EX 4200 switch models provide either 8, 24, or 48 PoE ports. To accommodate the required power for the PoE ports, three power supply capacities exist. The available power supply capacities are 320 watts, 600 watts, and 930 watts.

Power Supply and Switch Compatibility

EX 3200 and EX 4200 models can use the same power supplies. This design feature can prove to be a significant advantage when planning for spare field-replaceable units (FRUs).

Continued on next page.

Predetermined PoE Ports

All 802.3af-compliant PDs require no more than 12.95 watts. Thus, if you follow the recommended guidelines for selecting power supply units to support the number of PoE ports, the switch should be able to supply power to all connected PDs. If a higher capacity power supply unit is installed on a switch model that has only 8 PoE ports, it does not extend PoE capabilities to the non-PoE ports. If a lower capacity power supply unit is installed in a switch with 24 or 48 PoE ports, the total number of PoE ports is decreased. For example, if a 320 watt power supply unit is installed in a switch with 24 PoE ports, only the first 8 ports (port numbers 0 through 7) provide PoE power. If redundant power supplies of different capacities are installed in an EX 4200 switch—for example, a 600 watt power supply and a 930 watt power supply—the total available PoE pool is based on the lower of the two power supplies (600 watts). This design prevents power loss on already powered PoE devices in case of a power supply failure or a hot-swap of power supplies.

PoE Telemetries

- Switch provides telemetries support to keep history of power usage for each PD
  - Disabled by default; you enable PoE telemetries through configuration and specify duration and interval values

```
[edit poe interface interface-name telemetries]
user@switch# set ?
Possible completions:
+ apply-groups          Groups from which to inherit configuration data
+ apply-groups-except   Don't inherit configuration data from these groups
  disable               Disable telemetries
  duration              Duration to continue recording of data (1...24 hours)
  interval              Interval at which data should be recorded (1...30 minutes)
```


PoE Telemetries

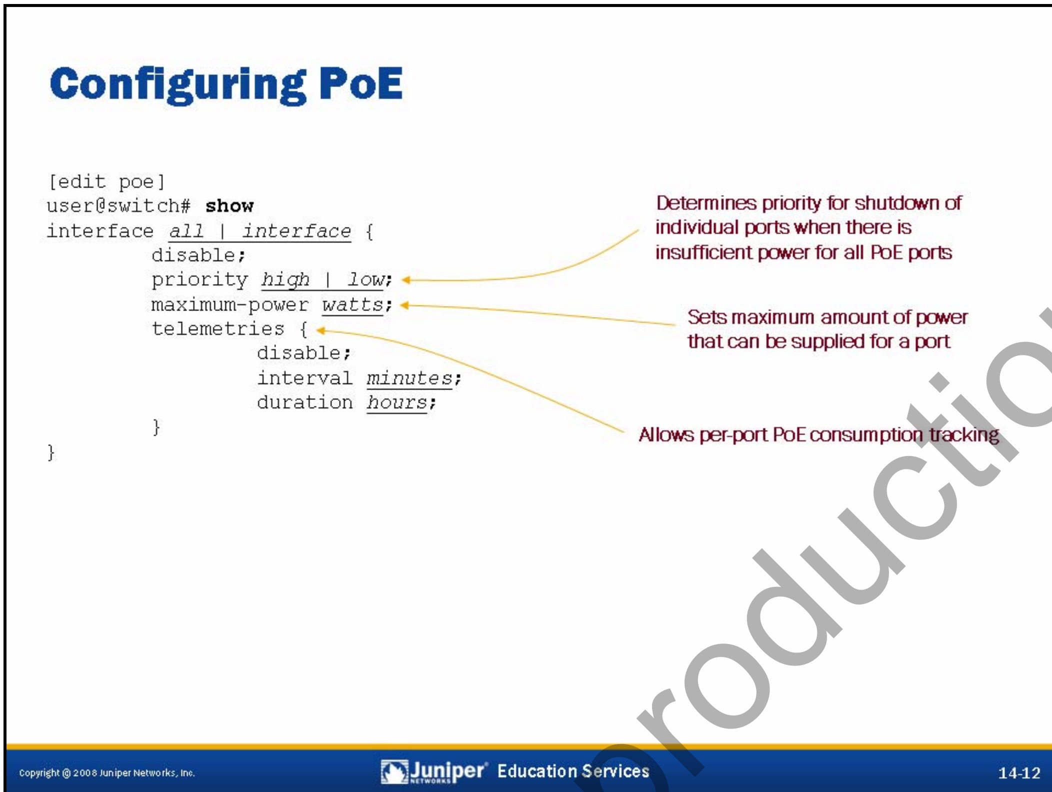
To maintain a history of power usage, EX-series switches support PoE telemetries. This feature is disabled by default, but you can easily enable it through the configuration. You enable PoE telemetries for individual interfaces or all interfaces using the keyword **all** under the `[edit poe interface]` hierarchy. You can also define custom `interval` and `duration` settings, which determine how often and for how long the usage data is tracked. The default duration is 1 hour and has a configurable range of 1 to 24 hours. The default interval is 5 minutes and has a configurable range of 1 to 30 minutes.

Continued on next page.

PoE Telemetries (contd.)

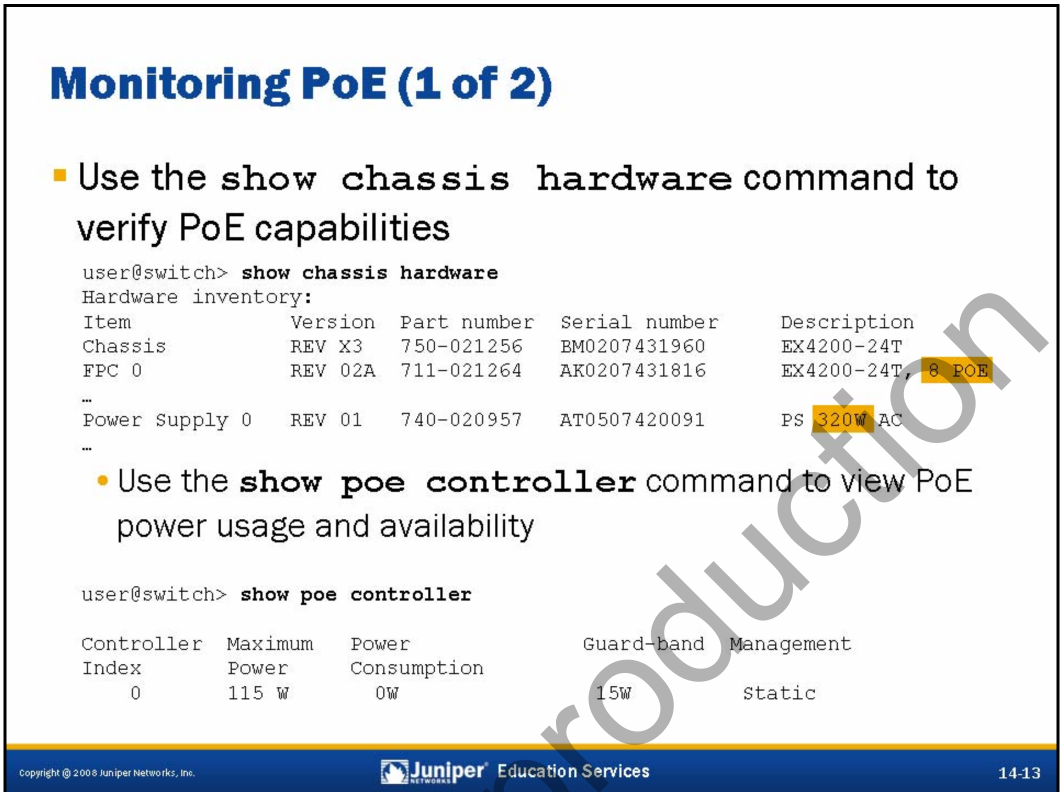
The following sample configuration shows the usage of individual interfaces, the interface **all** keyword, as well as mixed `interval` and `duration` settings:

```
[edit poe]
user@switch# show
interface all {
    telemetries {
        interval 5;
        duration 1;
    }
}
interface ge-0/0/0 {
    telemetries {
        disable;
    }
}
interface ge-0/0/5 {
    telemetries {
        interval 1;
        duration 1;
    }
}
interface ge-0/0/7 {
    telemetries {
        interval 30;
        duration 24;
    }
}
```



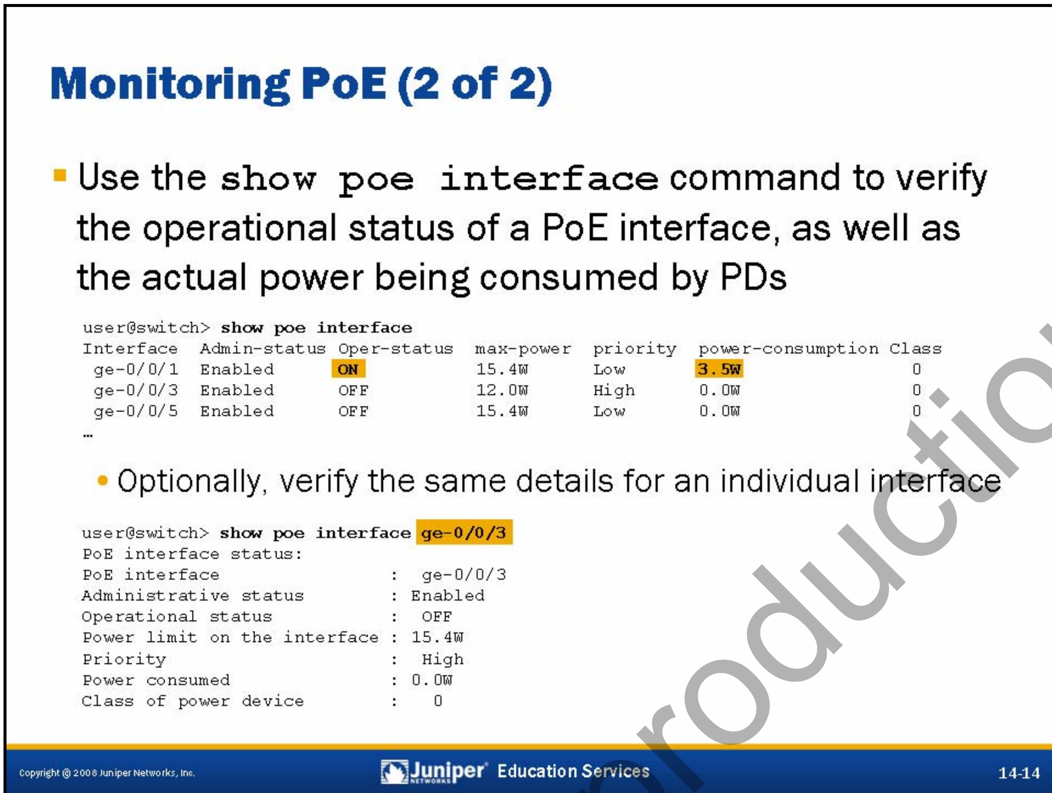
Configuring PoE

The slide illustrates the basic hierarchy and configuration options for PoE.



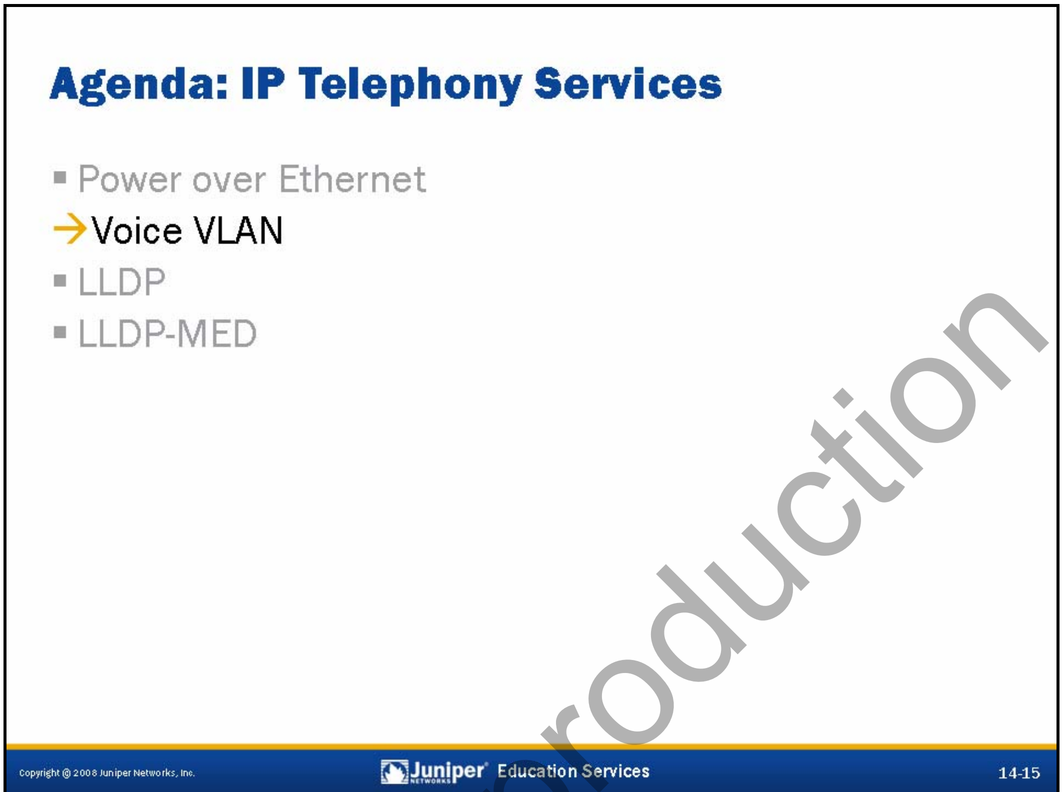
Monitoring PoE: Part 1

This slide and the next highlight some key operational-mode commands used to monitor PoE. Use the **`show chassis hardware`** command to determine the PoE capabilities of an EX-series switch. Use the **`show poe controller`** command to view PoE power usage and availability details.



Monitoring PoE: Part 2

This slide illustrates the **show poe interface** command, which you use to verify the operational state of PoE interfaces. The output of this command also displays the power consumption details for PDs attached to the PoE interfaces. As shown on the bottom of the slide, you can choose to add the interface name to the **show poe interface** command to limit the generated output and display PoE details for only the specified interface.

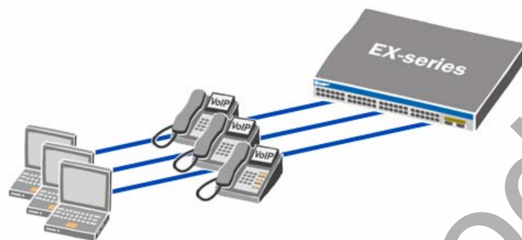


Voice VLAN

The slide highlights the topic we discuss next.

Voice VLAN

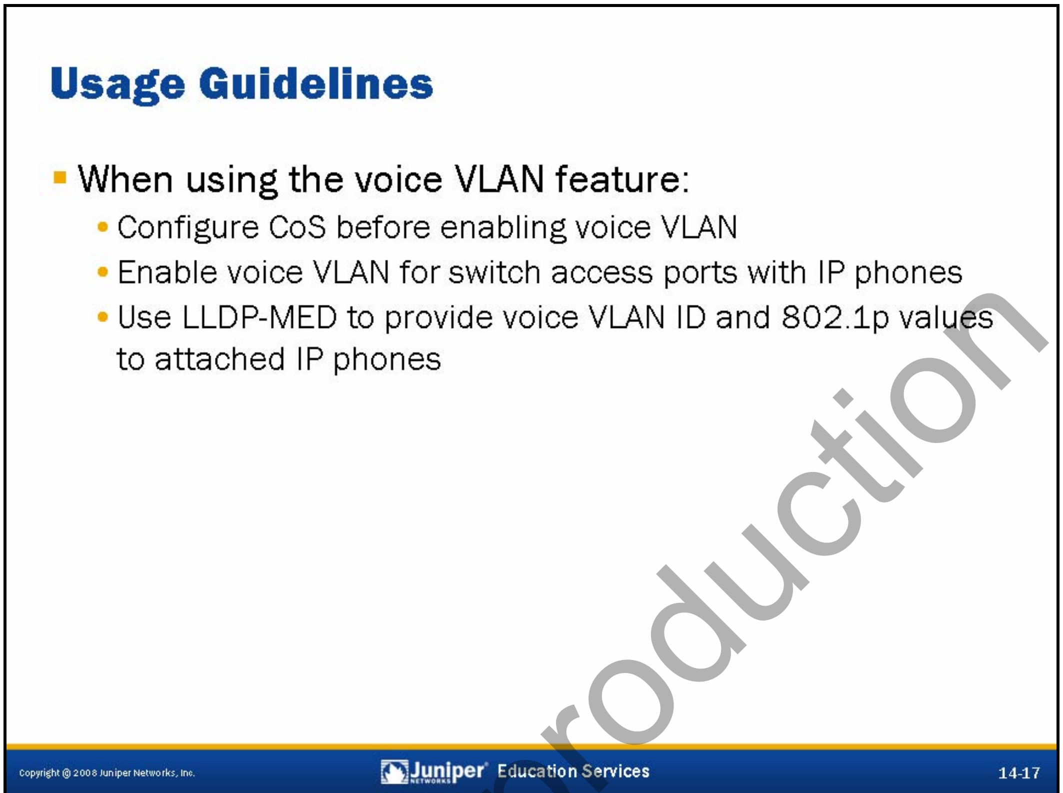
- The voice VLAN feature enables access ports to accept both untagged (data) and tagged (voice) traffic and separate that traffic into different VLANs



 Juniper Education Services 14-16

Voice VLAN

EX-series switches accommodate implementation scenarios that include an IP phone and a user's PC connected to a single switch port. Typically, administrators choose to treat VoIP traffic differently from user data traffic. To treat this traffic differently, a mechanism must be able to separate common user data traffic from voice traffic. The voice VLAN is used for this purpose. The voice VLAN enables a single access port to accept untagged data traffic as well as tagged voice traffic and associate each type of traffic with distinct and separate VLANs. By doing this, a network's class-of-service (CoS) implementation can treat voice traffic differently, generally with a higher priority than common user data traffic.



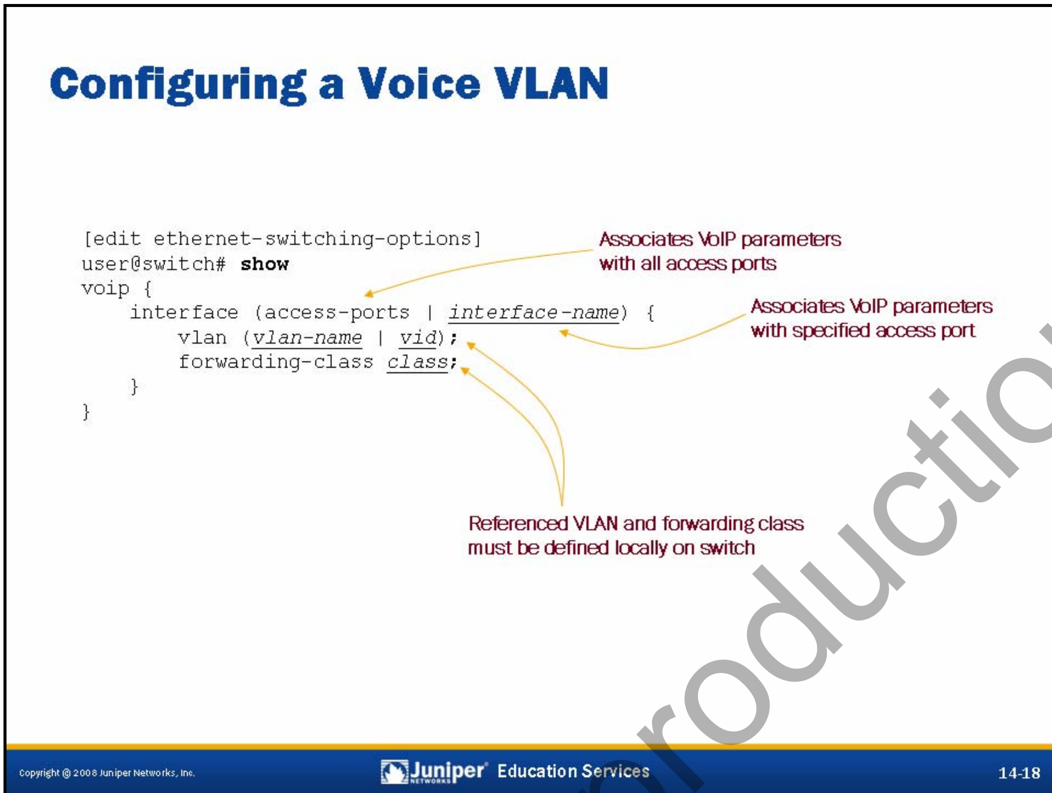
Voice VLAN Usage Guidelines

When implementing the voice VLAN feature, keep in mind the following:

1. We highly recommend that you configure CoS prior to enabling the voice VLAN feature. CoS is used to provide differentiated treatment for the various traffic types flowing throughout the network. Typically, voice traffic is treated with a higher priority than common user traffic. Without differentiated treatment through CoS, all traffic, regardless of the type, is subject to the same delay during times of congestion.

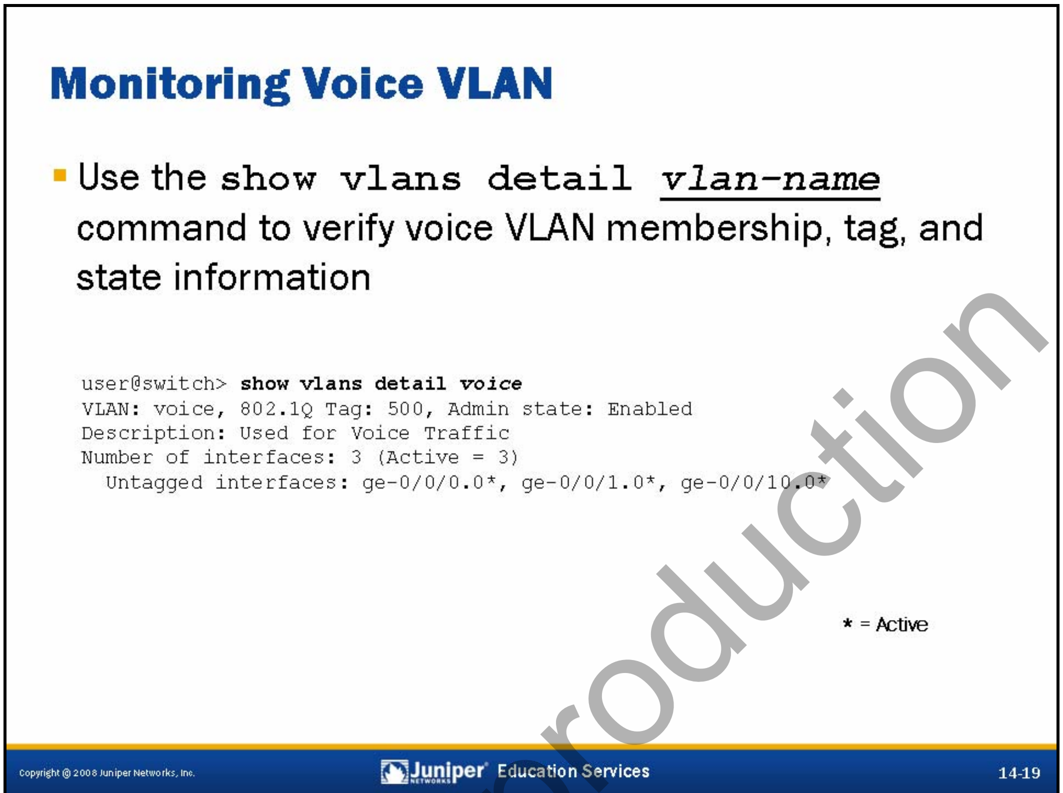
2. You should enable the voice VLAN only on access ports on which IP phones are actually connected. It makes little sense to enable an IP telephony feature on ports that do not carry IP voice traffic.

3. Use LLDP-MED to provide the voice VLAN ID and 802.1p values to the attached IP phones. This dynamic method associates each IP phone with the appropriate voice VLAN and assigns the necessary 802.1p values, which are used by CoS, to differentiate service for voice traffic within a network.



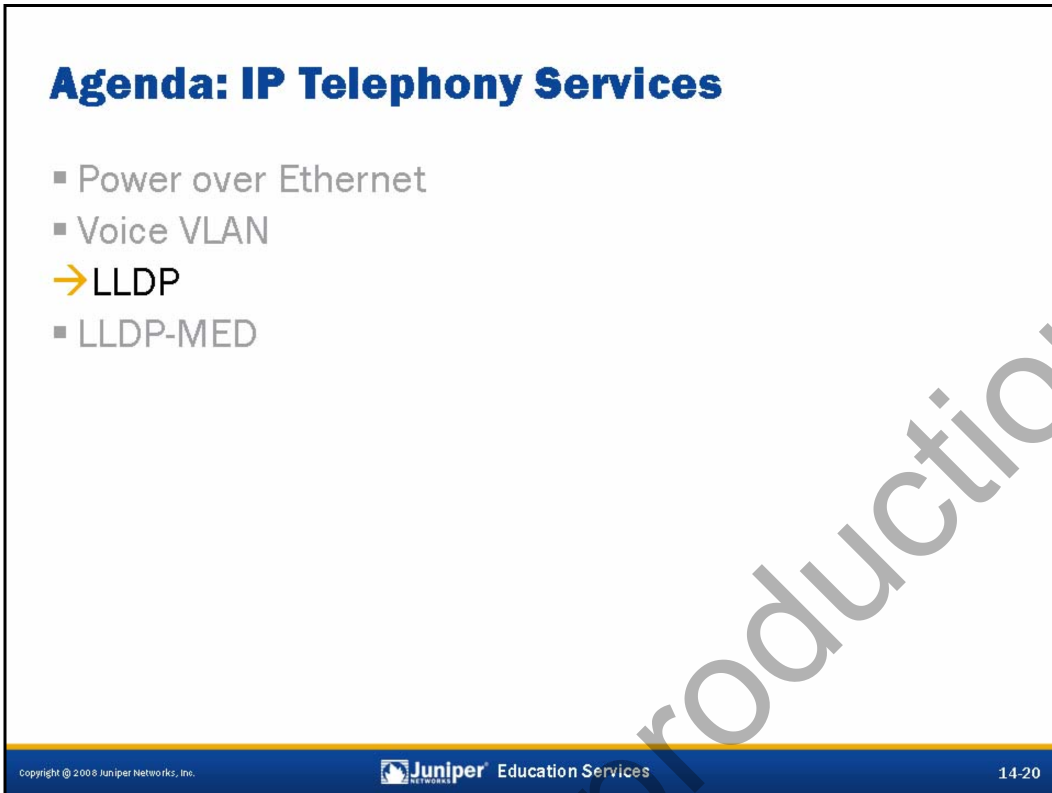
Configuring a Voice VLAN

This slide illustrates the basic hierarchy structure along with the available configuration options associated with the voice VLAN feature.



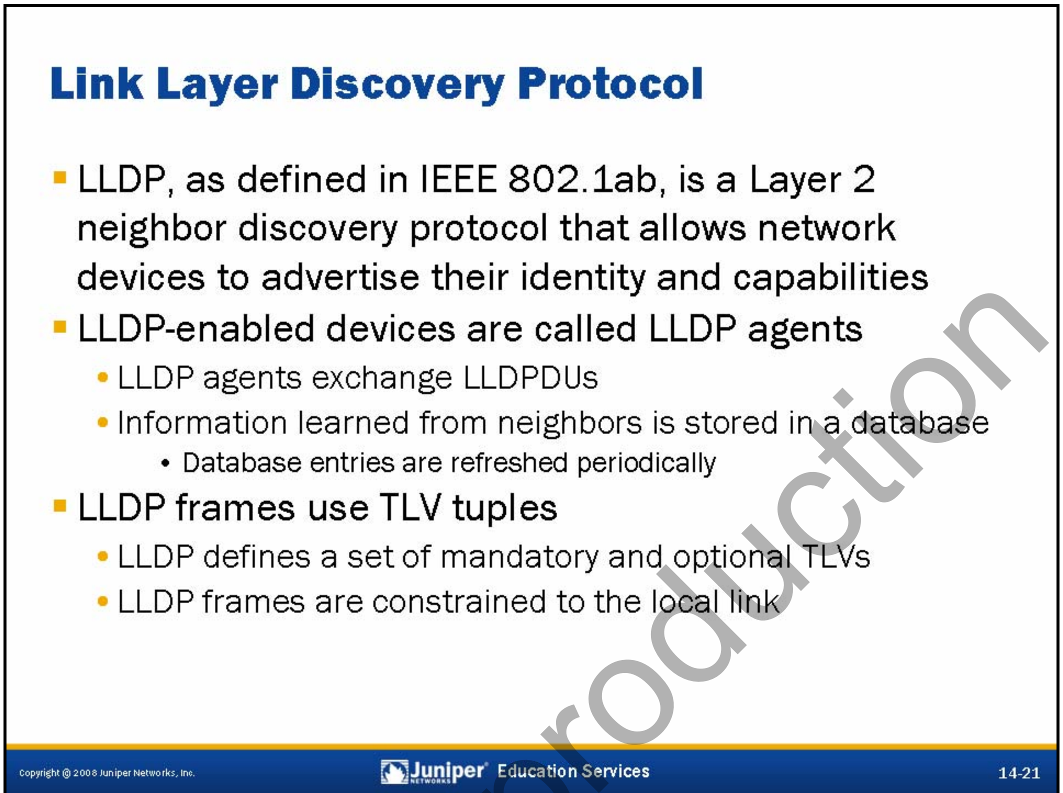
Monitoring a Voice VLAN

This slide displays a sample output from the **show vlans detail *vlan-name*** command. This command shows voice VLAN details such as membership, tag, and state information.



LLDP

The slide highlights the topic we discuss next.



LLDP Defined

The Link Layer Discovery Protocol (LLDP) is defined in IEEE 802.1ab as a Layer 2 protocol that facilitates network and neighbor discovery. Neighbor discovery is made possible through advertisements sent by each network device participating in LLDP. Advertisements are sent by LLDP-enabled devices to identify themselves and to announce their capabilities to neighboring devices. LLDP is somewhat comparable in purpose to the Cisco Discovery Protocol (CDP). LLDP operates on both Layer 2 and Layer 3 interfaces. Also, for operability of the protocol, it does not matter whether the port is a trunk port or an access port because the LLDP frames are untagged. This behavior helps the protocol build the network topology, regardless of specific configuration parameters assigned to the port.

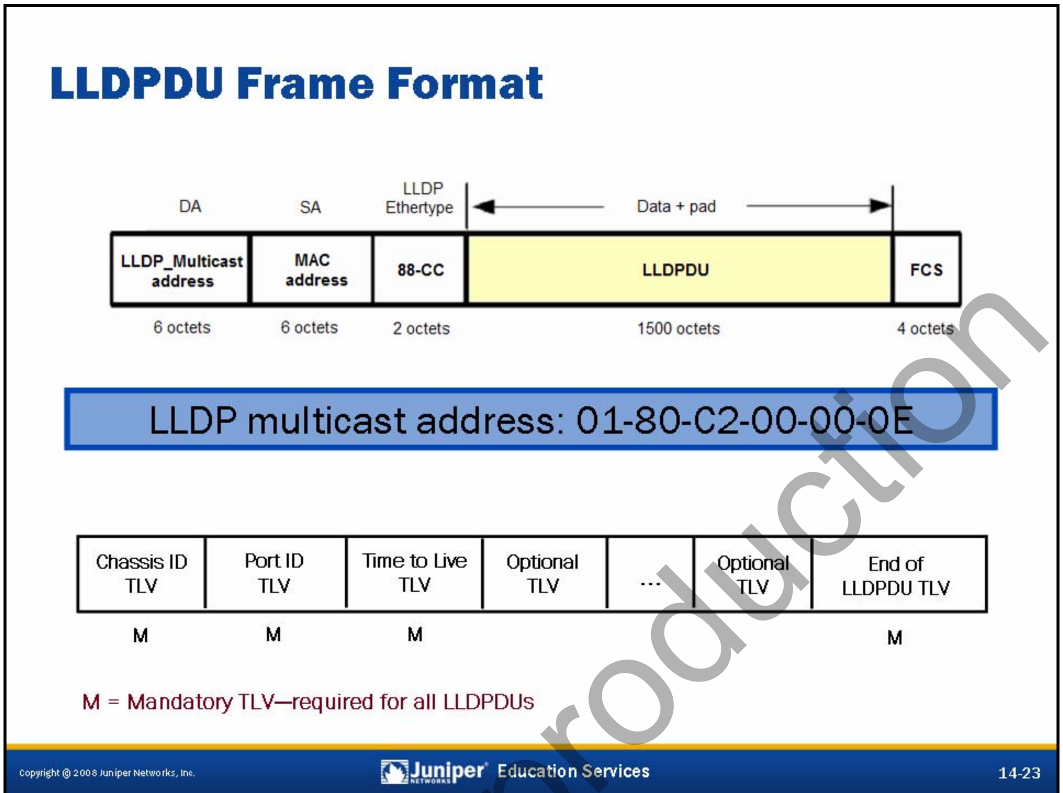
LLDP Agents

Any LLDP-enabled device is known as an LLDP *agent*. Each LLDP agent exchanges LLDP data units (LLDPDUs) with all other neighboring LLDP agents. LLDP agents store the information learned from neighbors in a local database. LLDP periodically refreshes the local database to maintain accurate information for all neighboring LLDP agents.

Continued on next page.

Type/Length/Value Messages

LLDP-capable devices transmit information in type/length/value messages (TLVs) to neighbor devices. Device information can include specifics such as chassis and port identification, system name, and system capabilities. LLDP defines some TLVs as mandatory, whereas others are listed as optional. The TLVs leverage this information from parameters that are already configured in the JUNOS software. All LLDP frames carrying TLVs are constrained to the local link, which means LLDP frames are never relayed or passed beyond a directly connected neighbor.



LLDPDU Frame Format

This slide illustrates the LLDPDU frame format. This illustration highlights the LLDP multicast address as well as the TLVs that are required in all LLDPDUs. A basic description of the mandatory TLVs follows:

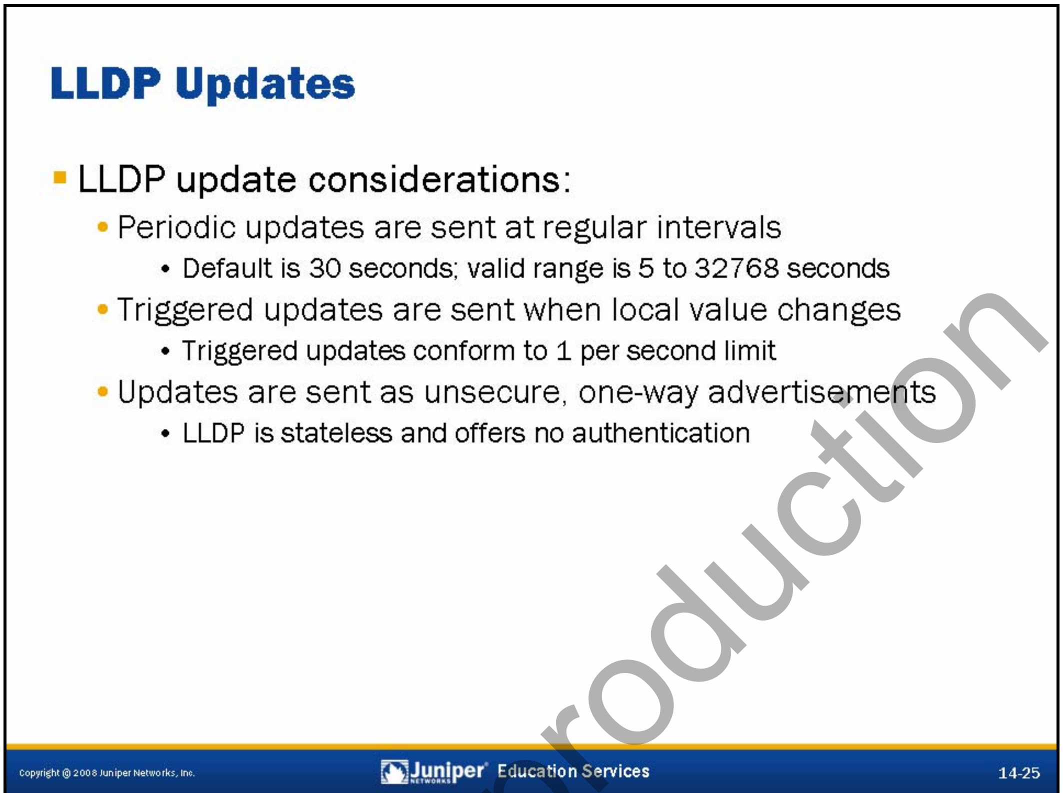
- *Chassis ID*: This TLV identifies the MAC address associated with the local system.
- *Port ID*: This TLV identifies the port from which the LLDPDU is transmitted.
- *Time to live (TTL)*: This TLV identifies how long the device's information is valid. A nonzero value indicates that the information is to be updated. A value of 0 indicates that the information is no longer valid and should be removed from the receiver's database.
- *End of LLDPDU*: This TLV identifies the end of TLVs in the LLDPDU.

Continued on next page.

LLDPDU Frame Format (contd.)

In addition to the mandatory TLVs, EX-series switches support the following set of basic TLVs:

- *Port description*: This TLV provides the user-configured port description. The port description can be a maximum of 256 characters.
- *System name*: This TLV identifies the user-configured name of the local system. The system name can be a maximum of 256 characters.
- *System description*: This TLV provides the system description containing information about the software and current image running on the system. This information is not configurable but taken from the software.
- *System capabilities*: This TLV identifies the primary function performed by the system. The capabilities that the system supports are defined—for example, bridge or router. This information is not configurable but based on the model of the product. An EX-series switch is capable of both switch and router operations.
- *Management address*: This TLV identifies the IPv4 management address of the local system.



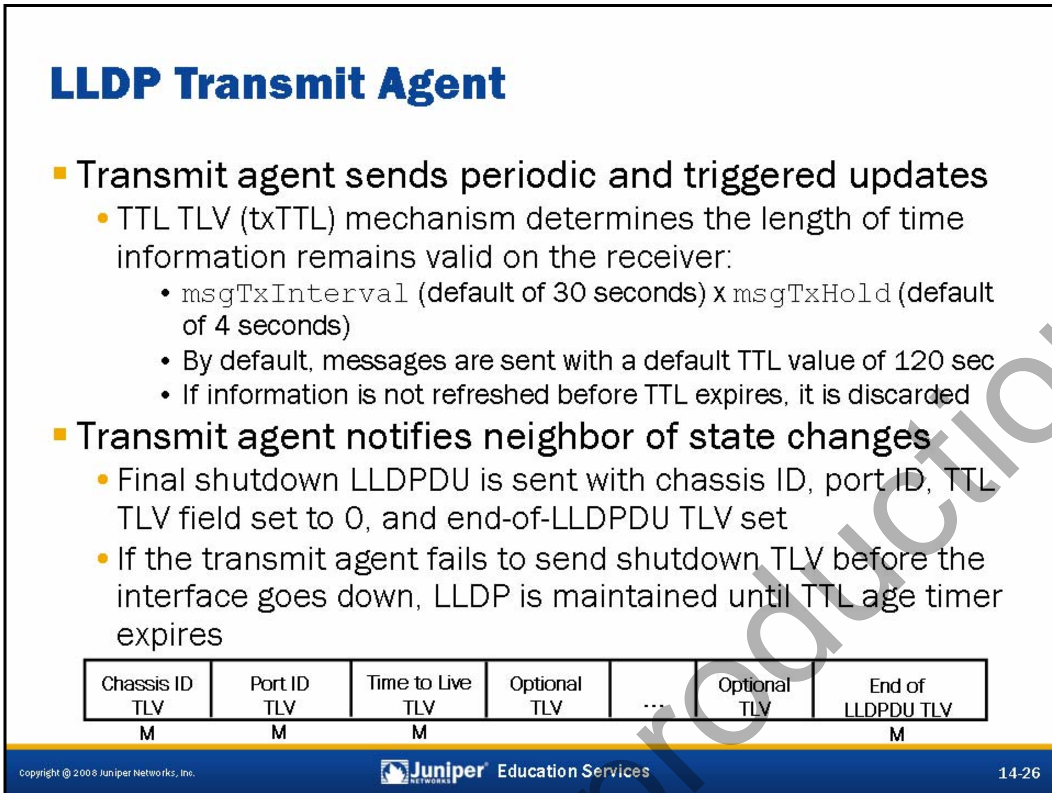
LLDPDU Update Considerations

EX-series switches send periodic LLDP updates to neighboring devices. These updates are advertised every 30 seconds by default. The `advertisement-interval` range is 5 to 32768 seconds and is configured at the `[edit protocols lldp]` hierarchy level. A sample configuration follows:

```
[edit protocols lldp]
user@switch# set ad?
Possible completions:
  advertisement-interval  Transmit interval for LLDP messages (5..32768 secs)
```

EX-series switches also send LLDP updates as needed based on local changes. These updates are often referred to as *triggered updates* because a value or state change triggers the update, as opposed to the regularly scheduled updates. Triggered updates cannot occur more than once per second.

LLDP updates are always sent as unsecured, one-way advertisements. Because LLDP is a stateless protocol, there is no verification or guarantee that the neighboring devices are actually receiving the transmitted advertisements. There is no acknowledgement that the advertisements or updates sent by one device are received by the neighboring devices. LLDP does not offer any authentication mechanism; therefore, all LLDP advertisements are unsecured. If a switch port connects to an untrusted boundary, such as a customer's network, we highly suggest that LLDP be disabled on that port.



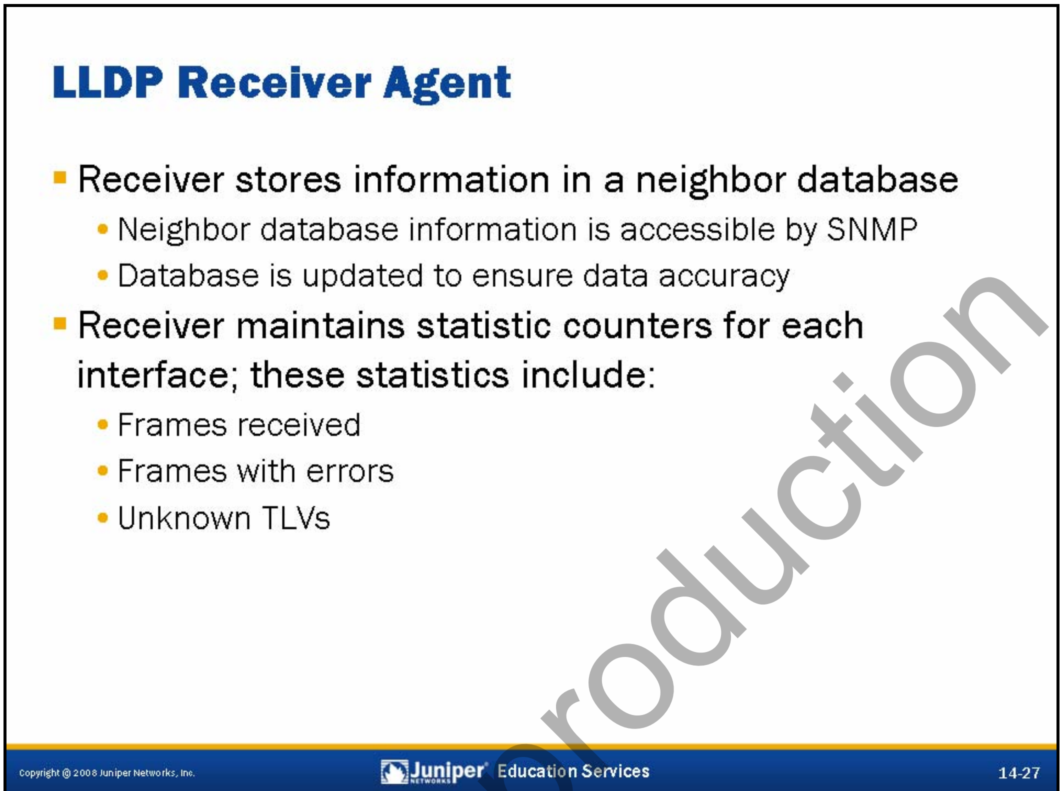
Transmit Agent Sends Updates

LLDP transmit agents send periodic and triggered updates to neighboring devices. The TTL TLV mechanism is used to ensure that only valid information is stored on the receiver agent. The TTL value is determined by multiplying the `msgTXInterval` value with the `msgTXHold` value. These values are defined on the transmit agent using the `advertisement-interval` and `hold-multiplier` settings.

As shown on the slide, the default `msgTXInterval` is 30 seconds and the default `msgTXHold` value is 4 seconds. These default values produce a default TTL value of 120 seconds. Thus, if a receiver agent does not receive an updated or refreshed message within 120 seconds from a given transmit agent, the information associated with that transmit agent is discarded.

Transmit Agent Notifies Neighbors

LLDP transmit agents notify their neighbors when an interface is about to become nonoperational or when LLDP is disabled on that interface. A final shutdown LLDPDU is sent with the chassis ID, port ID, TTL TLV fields set to zero (0), and the end-of-LLDPDU TLV set. If a transmit agent fails to send a shutdown TLV for an interface before it actually goes down, the receiver agent maintains the learned information on that interface until the TTL age timer expires (default of 120 seconds).



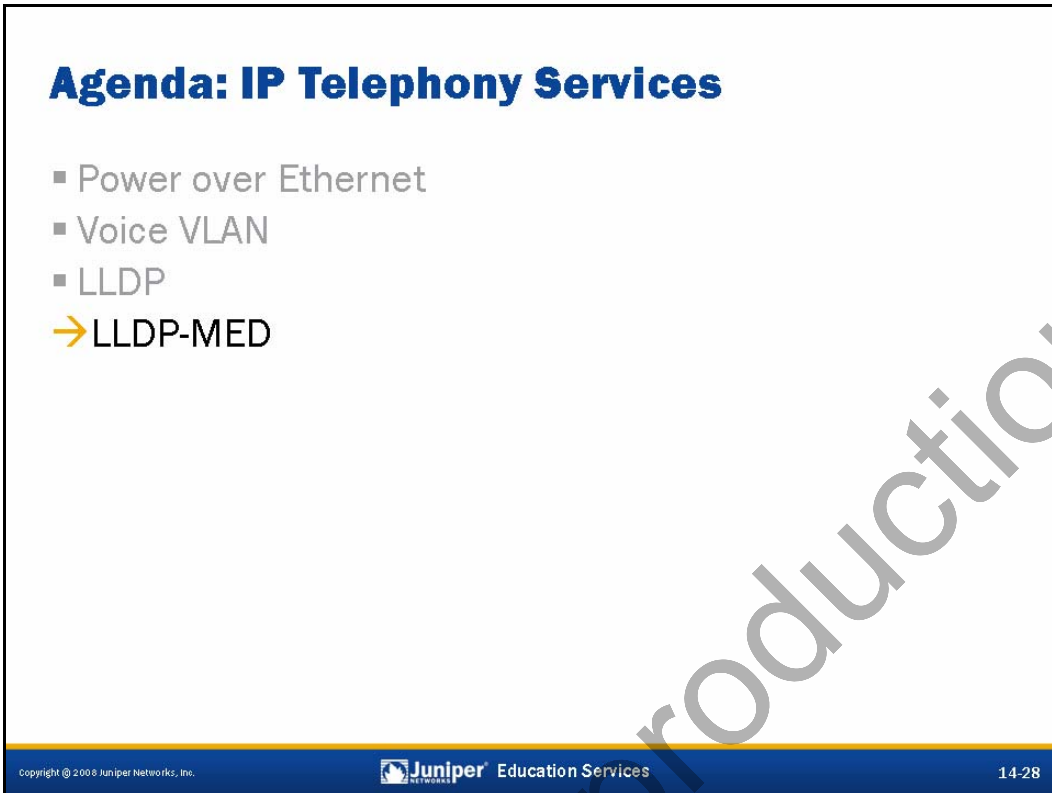
Receiver Agent Stores Information

The LLDP receiver agent stores information received from all neighboring LLDP-enabled devices (LLDP transmit agents) in a neighbor database. The information found within the neighbor database is accessible through SNMP. With the help of the TTL TLV mechanism, the contents within the neighbor database are refreshed or purged regularly to ensure that only accurate data is maintained.

Receiver Agent Maintains Statistics

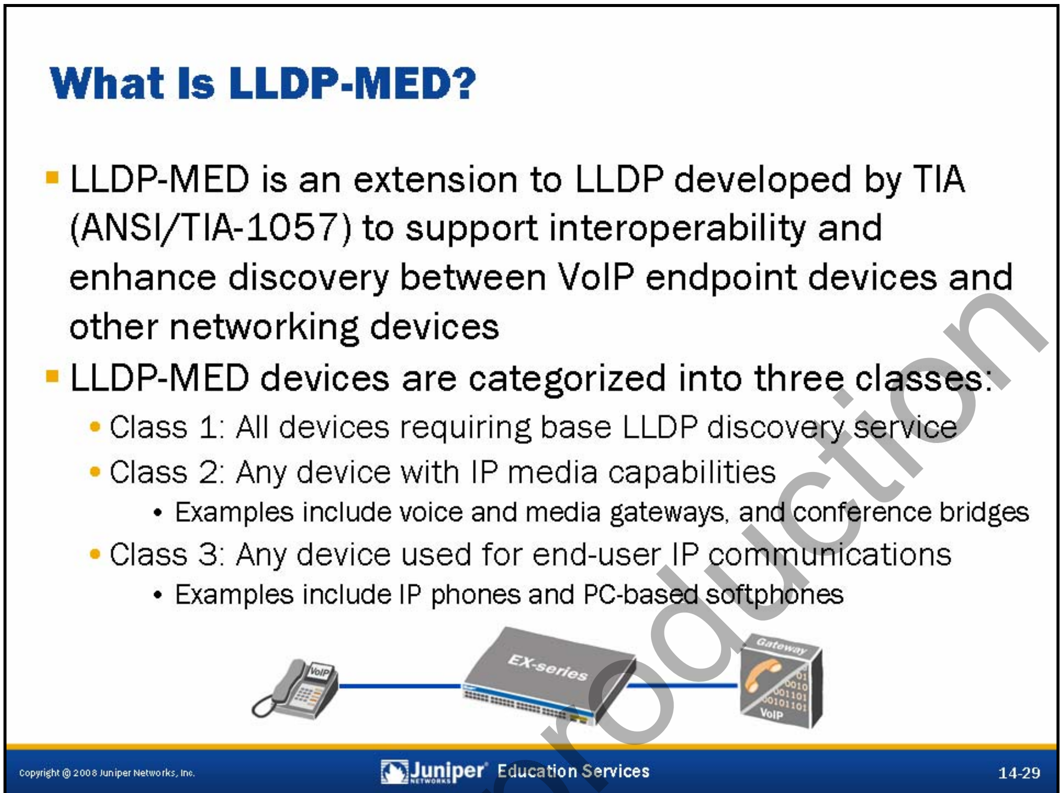
The LLDP receiver agent maintains detailed statistic counters for all LLDP-enabled interfaces. The statistic counters include frames received, frames with errors, and unknown TLVs for a given interface. The following example illustrates the statistic counters for both received and transmitted LLDP traffic for all participating interfaces:

```
user@switch> show lldp statistics
Interface     Received    Unknown TLVs  With Errors  Transmitted  Untransmitted
ge-0/0/13.0  2665        0             0            2666         0
ge-0/0/10.0  17106       0             0            17115        0
ge-0/0/2.0   0           0             0            17111        4
```



LLDP-MED

The slide highlights the topic we discuss next.



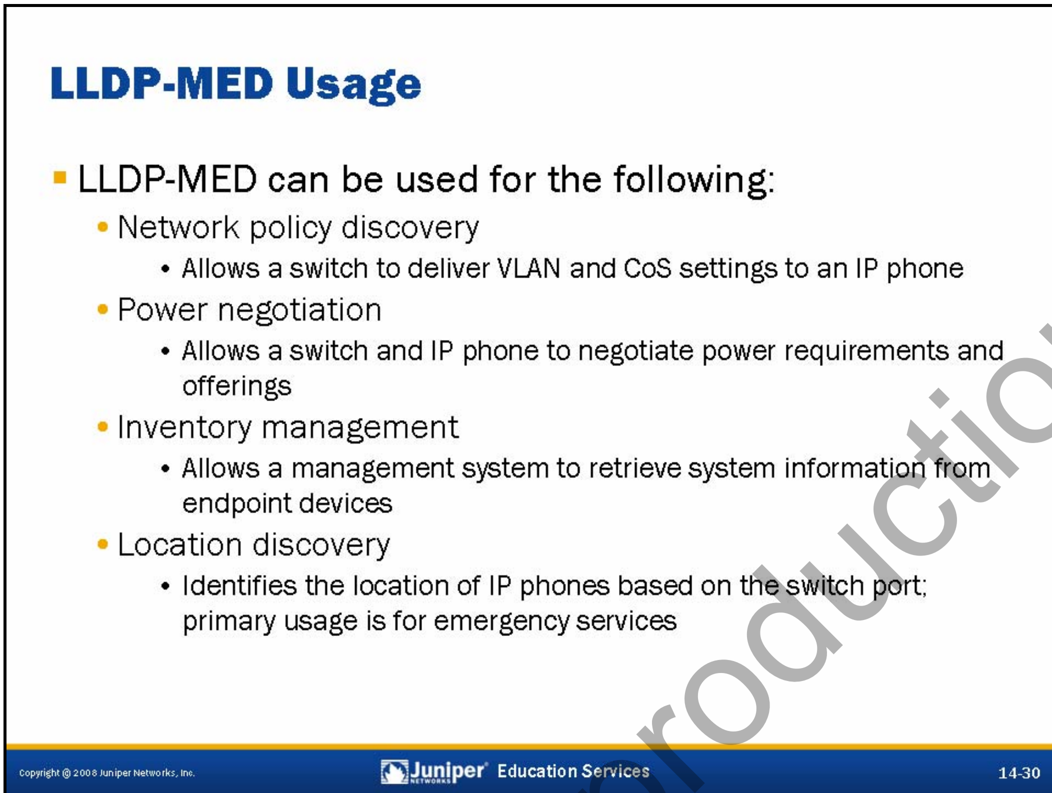
LLDP-MED Defined

The Link Layer Discovery Protocol-Media Endpoint Discovery (LLDP-MED) was developed by the Telecommunications Industry Association (TIA) and is defined in the American National Standards Institute (ANSI)/TIA-1057 standard. LLDP-MED is an extension to LLDP (IEEE 802.1AB) and was developed to support interoperability and enhance discovery capabilities between VoIP endpoint devices, such as IP phones, and other networking devices, such as EX-series switches.

LLDP-MED Device Categories

LLDP-MED-enabled devices are categorized into three classes. Every class defines a set of mandatory and optional TLVs to which the LLDP-MED implementation of the device should conform. The three classes are the following:

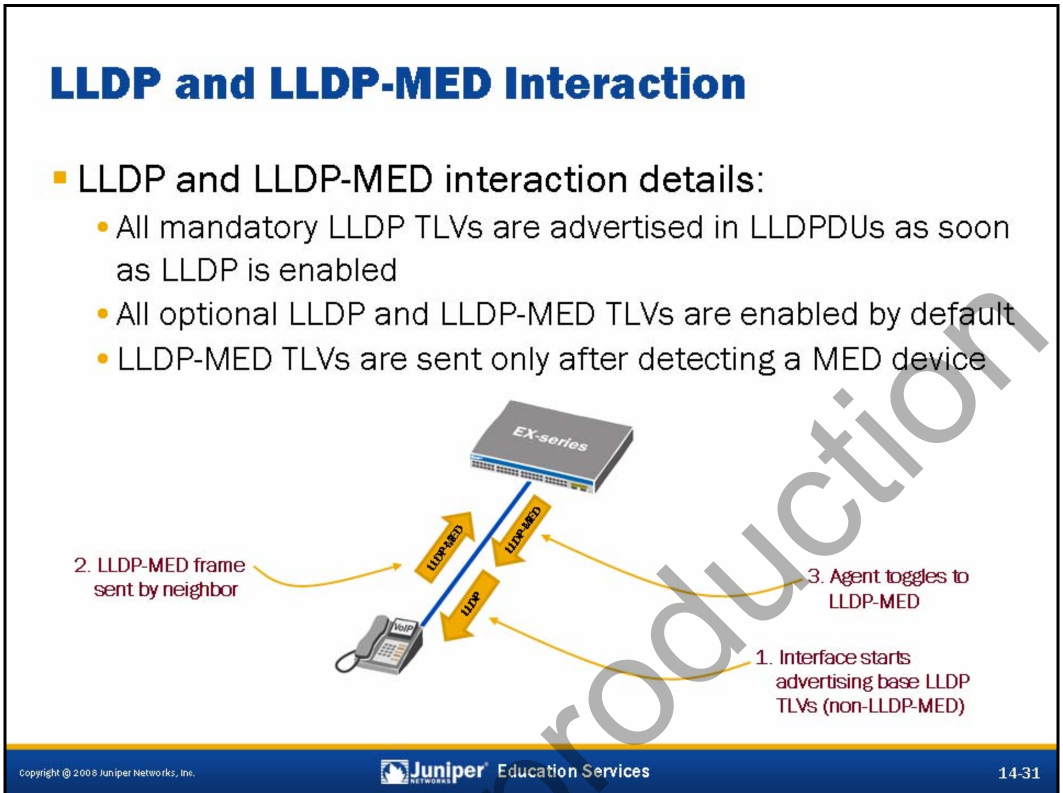
- *Class 1* (generic endpoints): This class definition is applicable to all endpoints that require the base LLDP discovery service.
- *Class 2* (media endpoints): This class includes endpoints that have IP media capabilities. Some examples of devices that belong to this class include voice and media gateways, and conference bridges.
- *Class 3* (communication endpoints): This class includes devices acting as end-user communication appliances that support IP media. Some examples of devices that belong to this class include IP phones and PC-based softphones.



LLDP-MED Usage

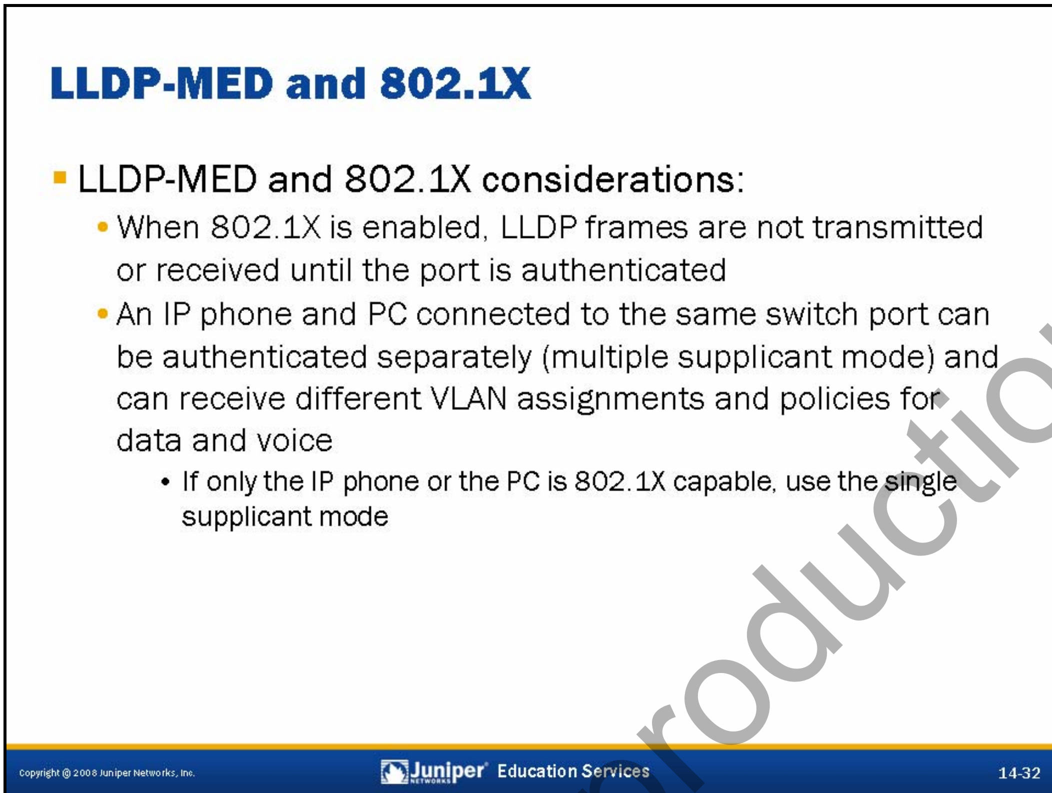
LLDP-MED capabilities are advertised within TLVs. The slide lists LLDP-MED capabilities.

As of JUNOS software Release 9.0, EX-series switches support only the network policy discovery capability. EX-series switches do not currently support the power negotiation, inventory management, or location discovery capabilities, although they are being considered for future versions of JUNOS software. The power negotiation capability should not be a concern on EX-series switches because all PoE ports support a full 15.4 watts when the proper power supply unit is installed.



Interaction Between LLDP and LLDP-MED

An EX-series switch participating in LLDP and LLDP-MED initially advertises the base LLDP TLVs (that is, non-LLDP-MED TLVs). If the EX-series switch detects a neighbor device that requires LLDP-MED, it toggles to the LLDP-MED mode and begins advertising LLDP-MED TLVs. All LLDP and LLDP-MED TLVs are enabled by default. An EX-series switch determines the required mode of operation based on the TLV type within the first LLDPDU received from an LLDP neighbor.

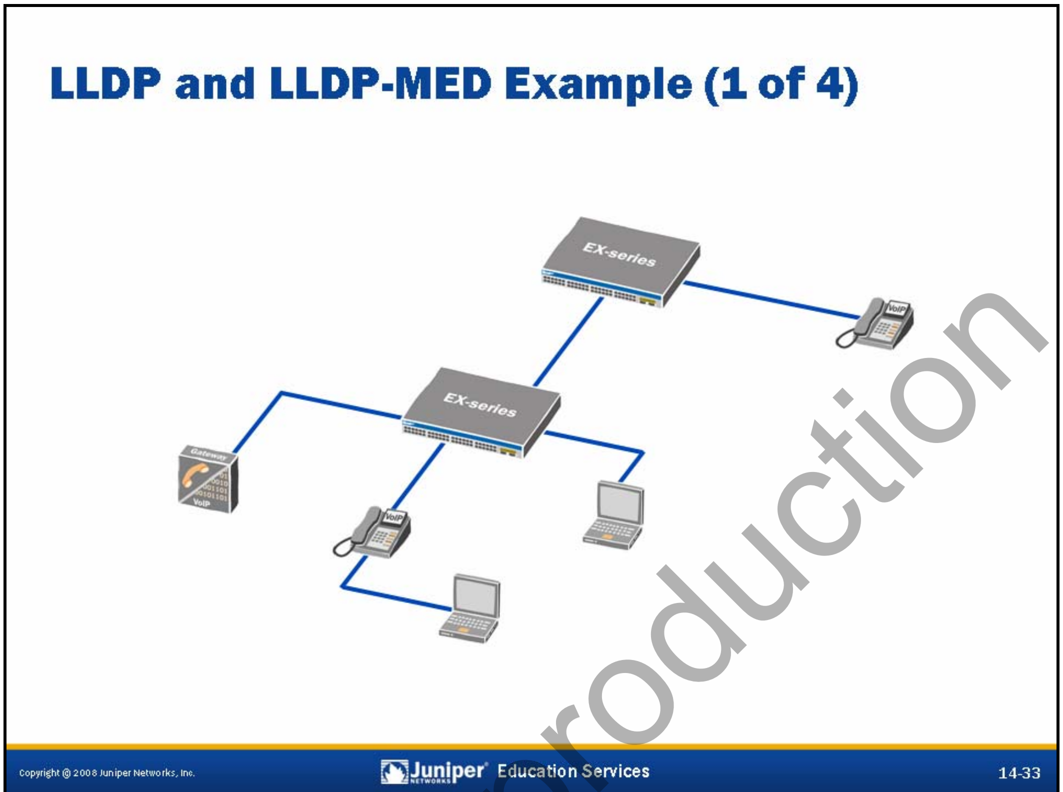


LLDP-MED and 802.1X

When 802.1X is used on an interface, LLDP frames are advertised and processed only when a secure port is authenticated. In other words, when 802.1x is enabled on a given switch port, LLDP frames are not transmitted or received until that port becomes authenticated.

In the case where an IP phone and a user's PC are connected to the same switch port, both devices can be authenticated separately when multiple-supplicant mode is enabled on that port. Using multiple-supplicant mode allows both devices to receive their appropriate VLAN assignments and individual policies for voice and data traffic. Voice traffic is commonly associated with a voice VLAN, which is treated with a high priority through CoS. User data traffic is often treated as a lower priority and might be dropped sooner than voice traffic during times of congestion.

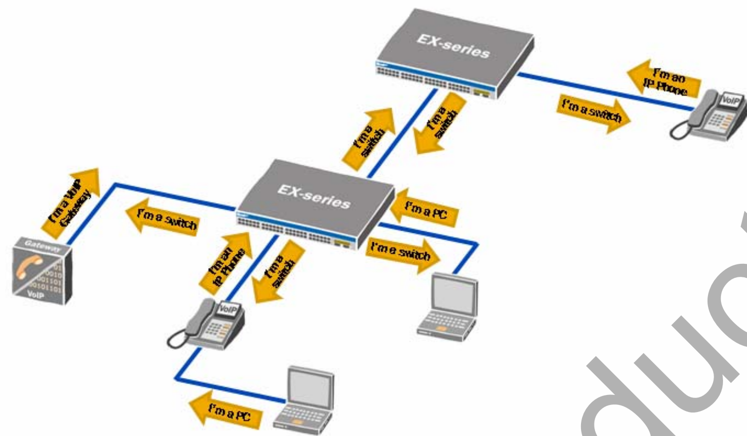
If an IP phone and a user's PC are connected to the same switch port and only one of the two devices is 802.1X capable, use single-supplicant mode. Single-supplicant mode authenticates the 802.1X-capable device and freely permits the other device without further authentication. In this situation, both devices are permitted network access, and LLDP or LLDP-MED messages can be exchanged.



LLDP and LLDP-MED Example: Part 1

This slide and the next several slides highlight the basic operation of LLDP and LLDP-MED. This slide introduces a typical topology that includes various network devices attached to some EX-series switches.

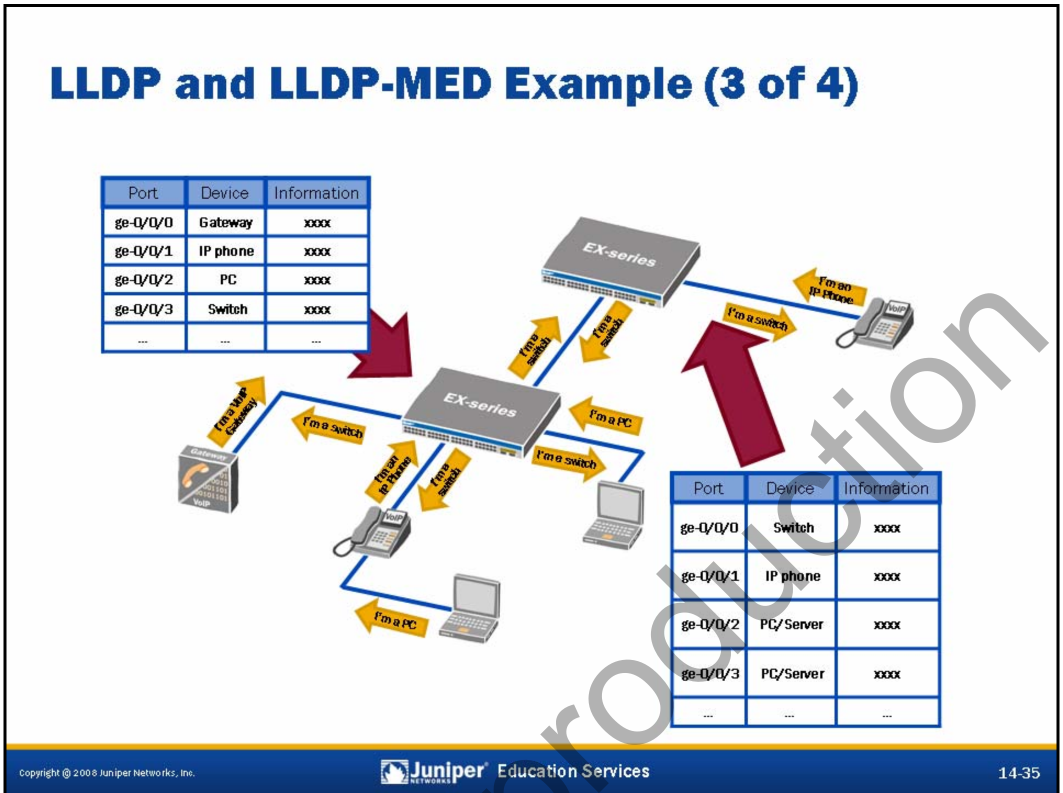
LLDP and LLDP-MED Example (2 of 4)



 Juniper Education Services 14-34

LLDP and LLDP-MED Example: Part 2

This slide illustrates the initial advertisement from each device running LLDP or LLDP-MED. Here we see that all devices announce their identity and share various other details by way of LLDPDUs.

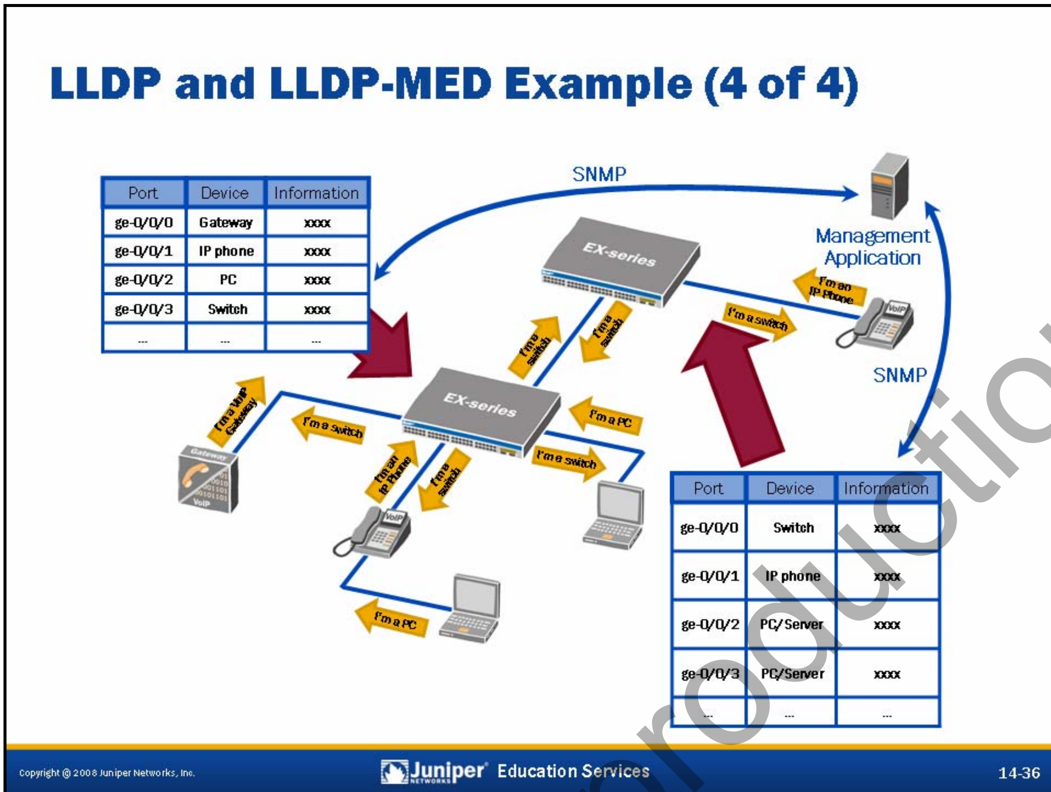


Port	Device	Information
ge-0/0/0	Gateway	xxxx
ge-0/0/1	IP phone	xxxx
ge-0/0/2	PC	xxxx
ge-0/0/3	Switch	xxxx
...	...	...

Port	Device	Information
ge-0/0/0	Switch	xxxx
ge-0/0/1	IP phone	xxxx
ge-0/0/2	PC/Server	xxxx
ge-0/0/3	PC/Server	xxxx
...	...	...

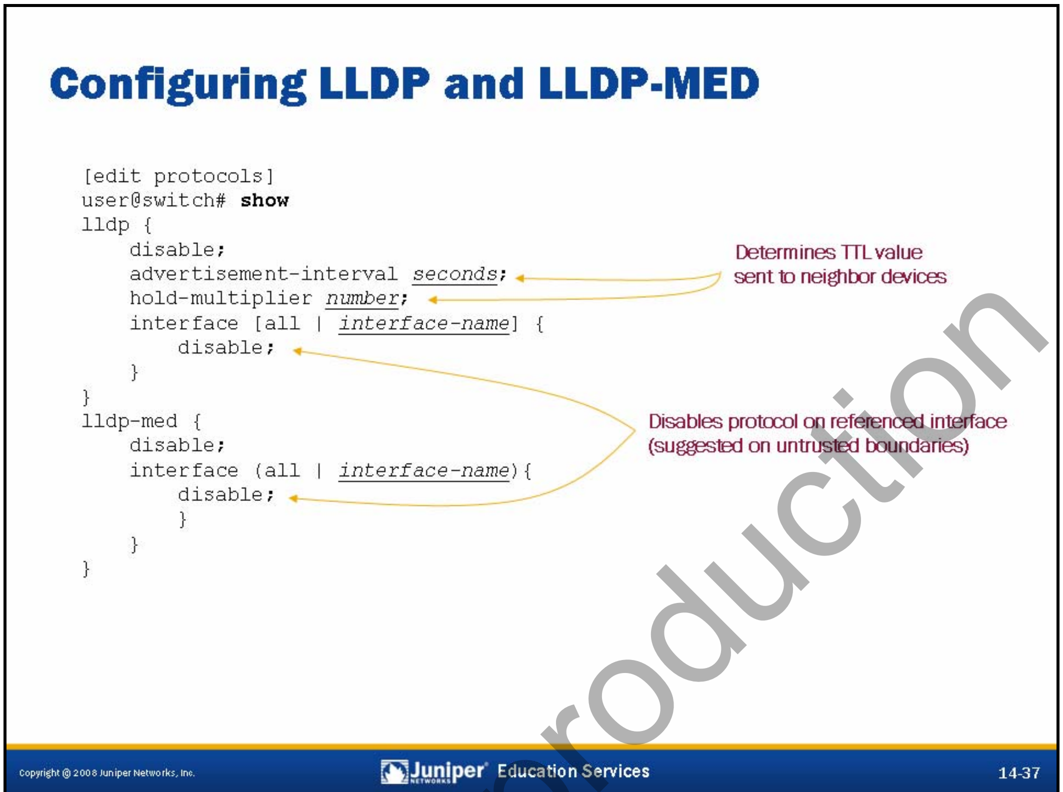
LLDP and LLDP-MED Example: Part 3

This slide displays sample LLDP neighbor tables, which are built on the EX-series switches based on the exchanged LLDPDUs.



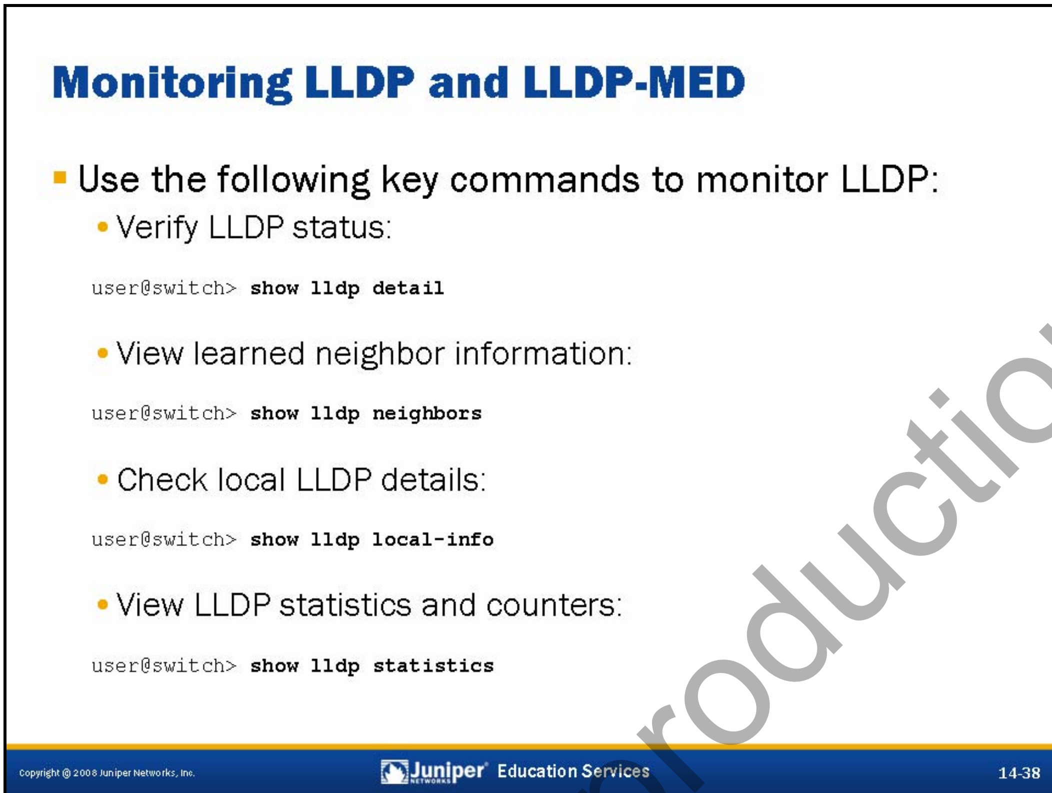
LLDP and LLDP-MED Example: Part 4

This slide illustrates the retrieval of LLDP neighbor information through SNMP. Allowing a network management system to retrieve information from the LLDP neighbor database through SNMP can aid in inventory management and topology mapping efforts.



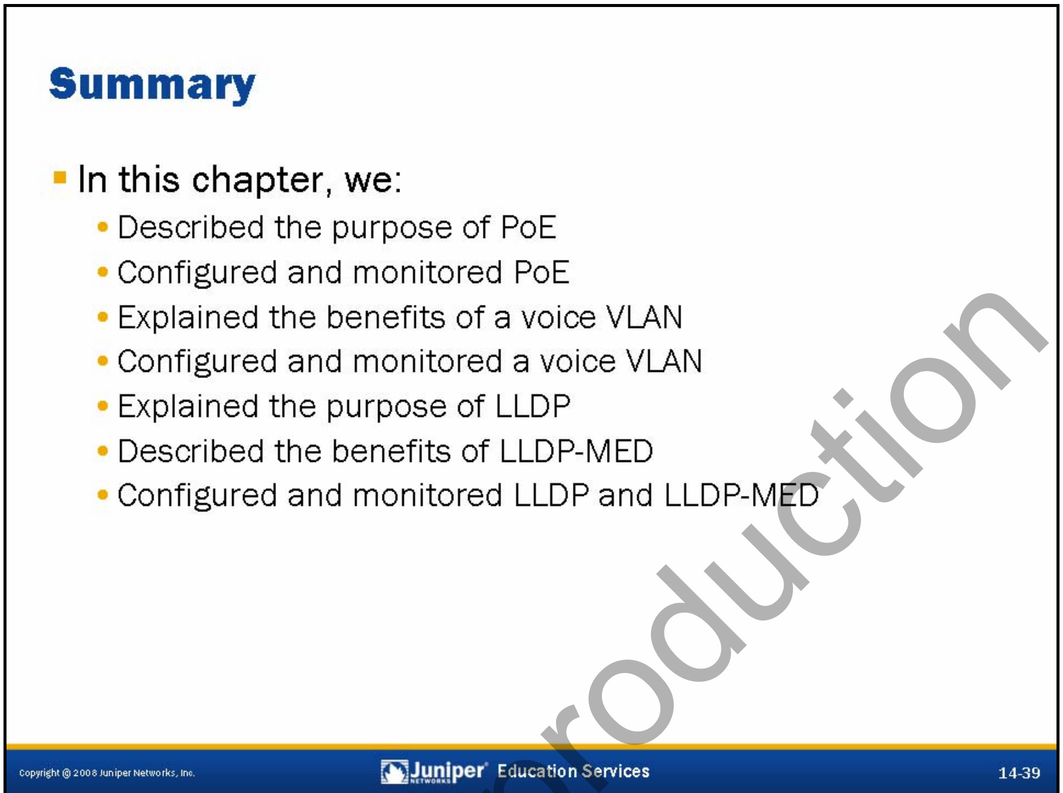
Configuring LLDP and LLDP-MED

The slide illustrates the basic hierarchy and configuration options for LLDP and LLDP-MED.



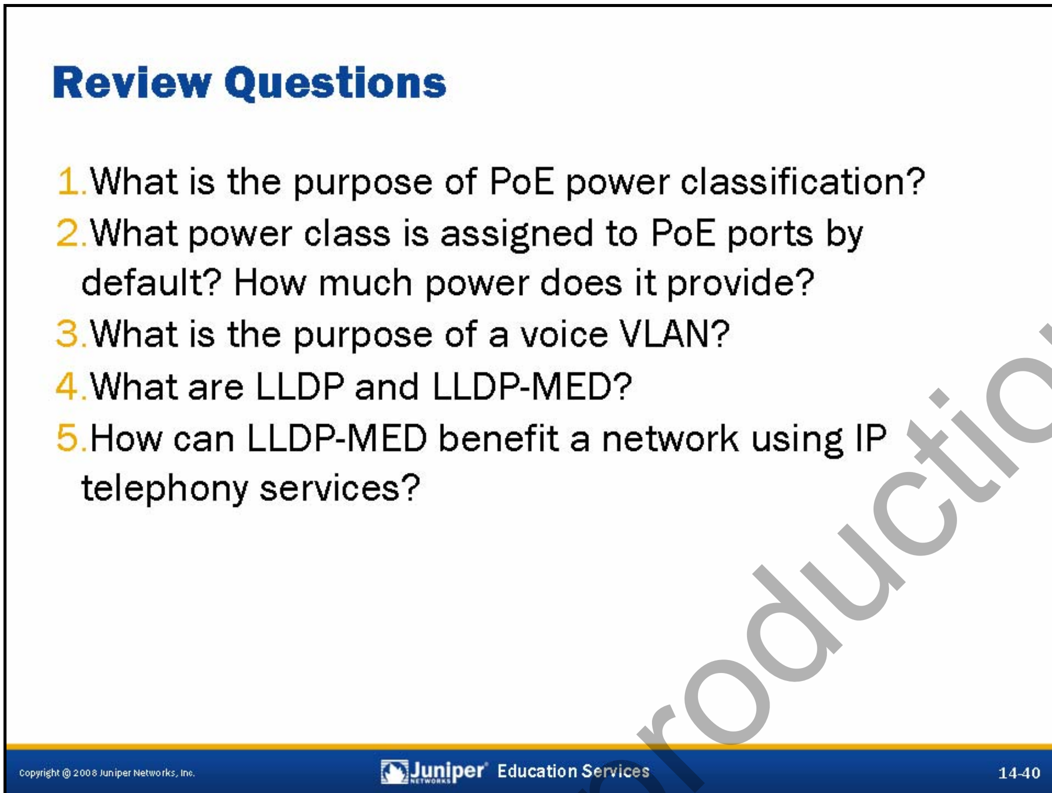
Monitoring LLDP and LLDP-MED

This slide highlights some key operational-mode commands used to monitor LLDP and LLDP-MED. Use the **`show lldp detail`** command to determine the status of LLDP or LLDP-MED. Use the **`show lldp neighbor`** command to view information learned from neighboring LLDP devices. Use the **`show lldp local-info`** command to view local LLDP details. The output from this command essentially represents much of the information that is sent from the local device to neighboring LLDP-enabled devices. Use the **`show lldp statistics`** command to view LLDP statistics and counters.



This Chapter Discussed:

- The purpose of PoE;
- Configuring and monitoring PoE;
- The benefits of a voice VLAN;
- Configuring and monitoring a voice VLAN;
- The purpose of the LLDP;
- The benefits of LLDP-MED; and
- Configuring and monitoring LLDP and LLDP-MED.



Review Questions

1.

2.

3.

4.

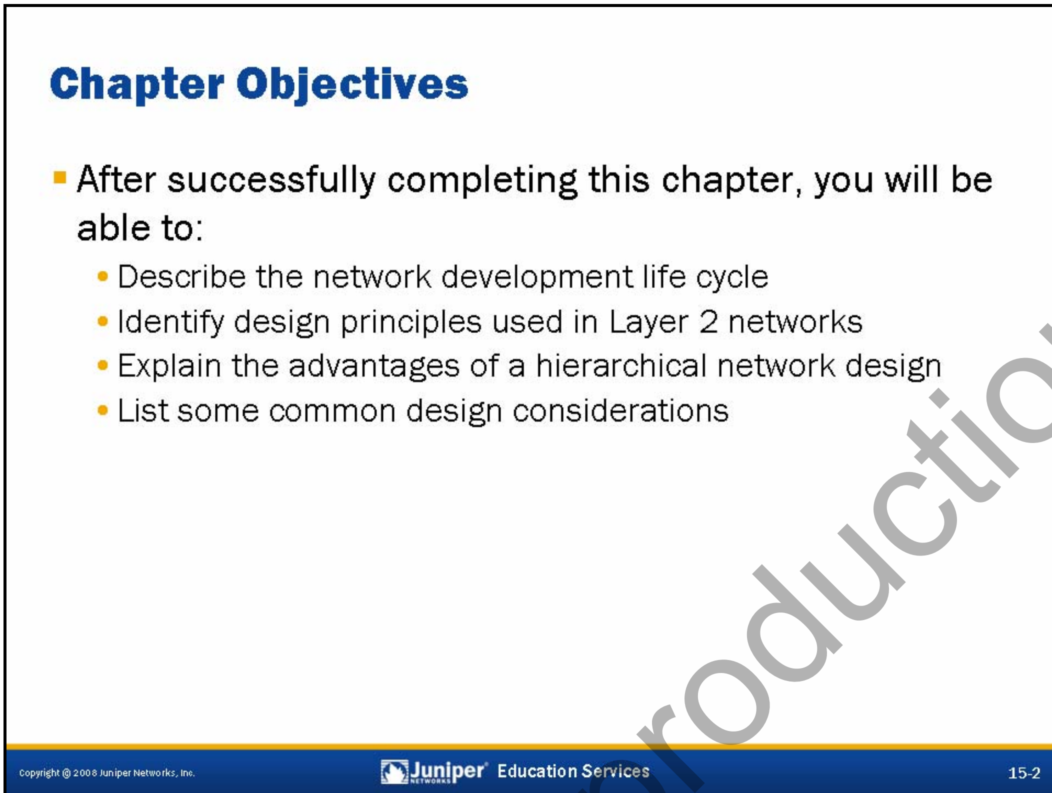
5.

Lab 12: IP Telephony Services

The slide provides the objective for this lab.

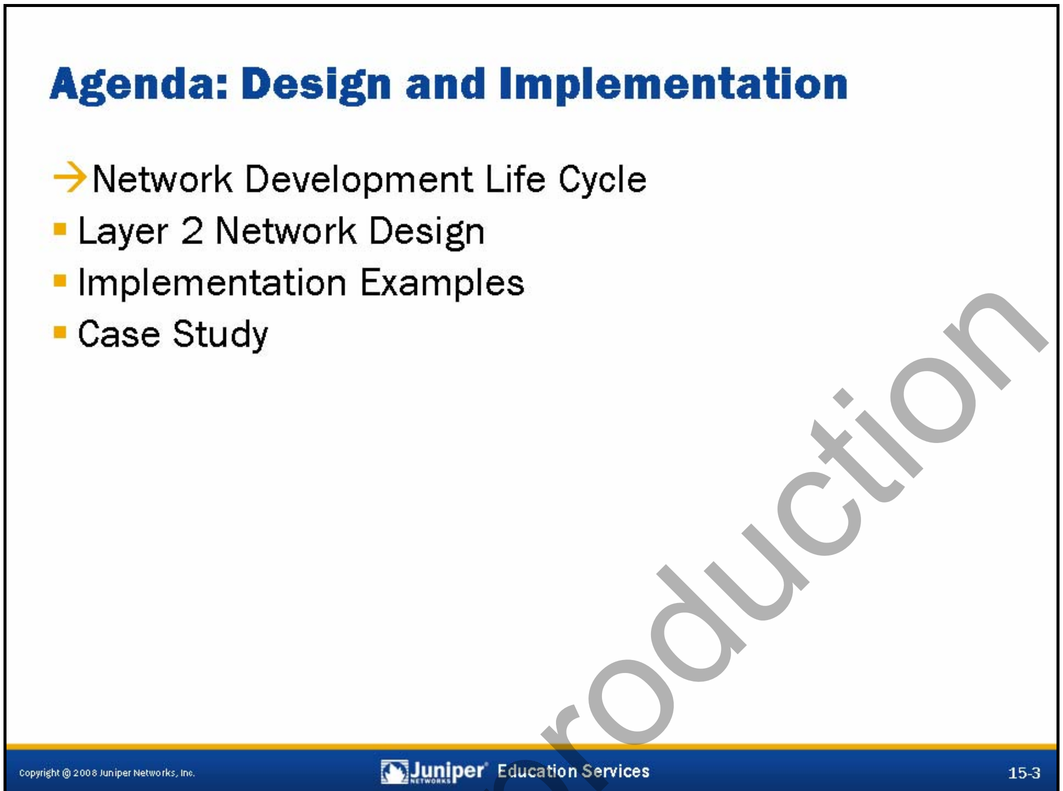
Operating Juniper Networks Switches in the Enterprise

Chapter 15: Design and Implementation of Layer 2 Networks



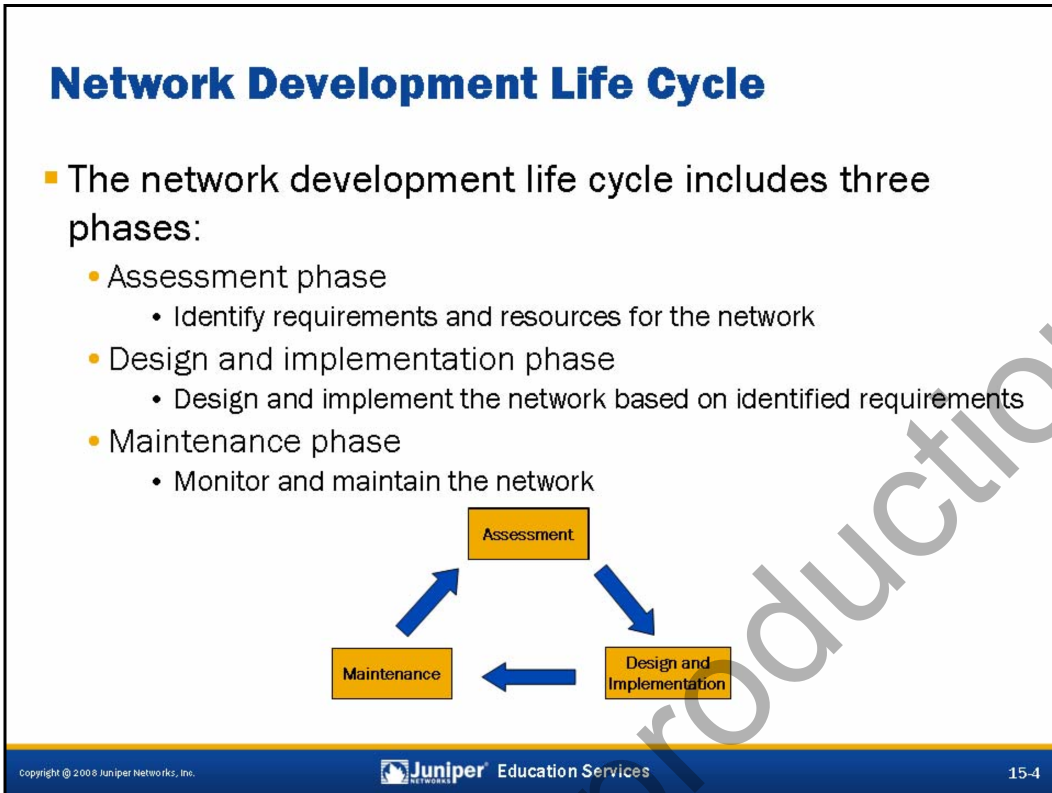
This Chapter Discusses:

- The network development life cycle;
- Basic design principles used when designing Layer 2 networks;
- Advantages of implementing a hierarchical network design; and
- Common design considerations.



Network Development Life Cycle

The slide lists the topics we cover in this chapter. We discuss the highlighted topic first.

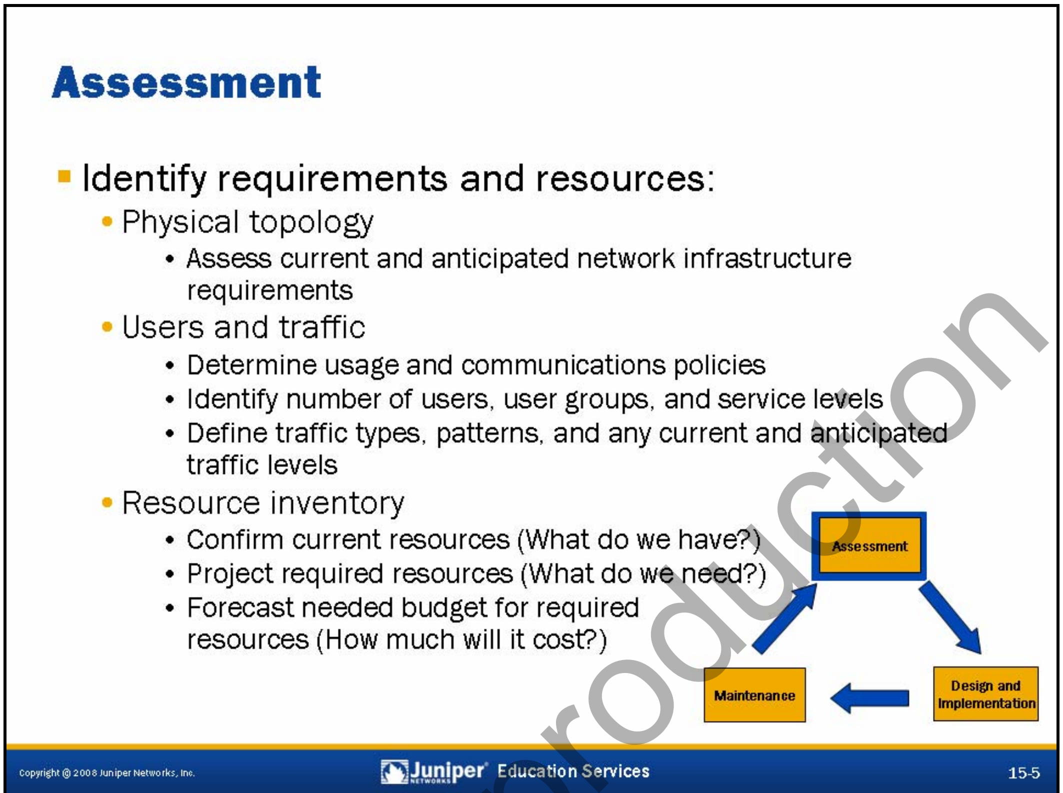


Phases of the Network Development Life Cycle

This slide introduces the three phases of the network development life cycle. At a high level, the three phases and associated tasks are the following:

- *Assessment*: In this phase appropriate individuals collectively assess and identify the required resources for the network.
- *Design and implementation*: In this phase select engineers design and implement the network based on the requirements identified in the assessment phase.
- *Maintenance*: In this phase select employees monitor and maintain the network as defined in the design and implementation phase. This phase is also key in identifying needed changes and enhancements.

These three individual phases, which collectively constitute the network development life cycle, are covered in greater detail on subsequent slides.



Identifying Requirements and Resources

To successfully develop an effective network design, you must first gain a comprehensive understanding of the requirements and available resources for a given network. The assessment phase provides managers and networking professionals the opportunity to become unified in this understanding. Unless both groups are unified and work through this process together, the likelihood of successfully implementing a network that truly meets the requirements and needs of a business is minimal. In the assessment phase, managers and those responsible for designing, implementing, and managing the network should identify and document all applicable requirements.

One applicable requirement is the physical topology. The current and anticipated physical infrastructure must be assessed. This assessment helps identify connectivity requirements along with the available connectivity options.

Many companies implement policies that outline resource usage and communication guidelines. Company policies might effectively determine network usage polices. For example, some workgroups might be forbidden to exchange work-related information because of confidentiality. A network enforcement policy should be in place to substantiate and help enforce this company communications policy.

Continued on next page.

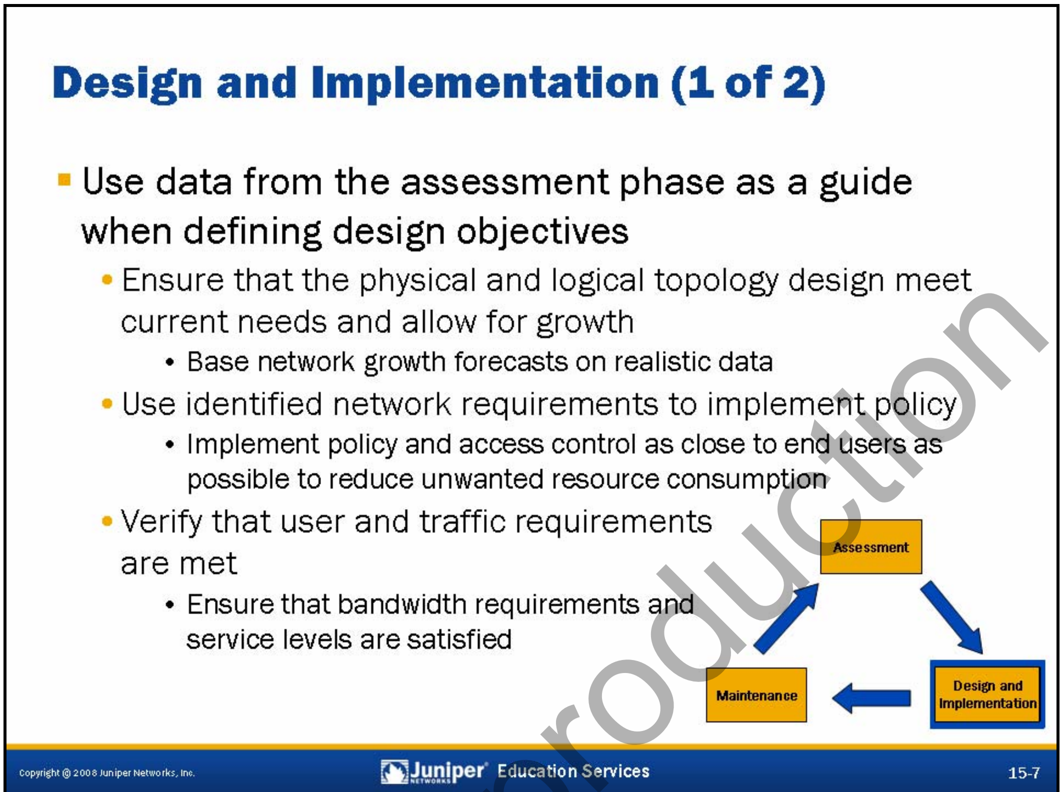
Identifying Requirements and Resources (contd.)

To properly design and implement a network, certain capacity-related questions must be answered. At a basic level the number of users requiring network access must be identified, as well as any workgroup associations and variations in service levels. Variations in service levels might be based on workgroup, traffic type, or both.

Defining the types of traffic, expected traffic patterns, and any current and anticipated traffic levels for a network might be especially helpful. You can use this information to properly structure network paths and provide ample bandwidth where necessary. By implementing a sound network infrastructure with sufficient bandwidth, the overall network performance increases, while the amount of congestion decreases. Determining the expected and acceptable traffic types aids in the development of user access control policy for individual users or groups of users.

It is said that the best things in life are free; if so, we can quickly eliminate high-end networks from the list. Put simply, this stuff costs money. With this in mind, all relevant parties must understand the financial impact that goes along with designing and implementing a high-performing network. Without management's commitment and support, the desired network infrastructure will not be implemented.

To fully identify the expense of a network, a complete list of required resources must be compiled. Start by creating a preliminary list of the anticipated resources required for the network infrastructure. Next, identify what resources are currently available. Once a list of available resources is created, identify the resources on that list that can actually be used. The list of existing, usable resources helps identify what additional resources must be acquired. After identifying the list of resources that must be purchased, pricing information is typically compiled and evaluated. Based on the pricing evaluation, you can determine a forecast of the required budget.



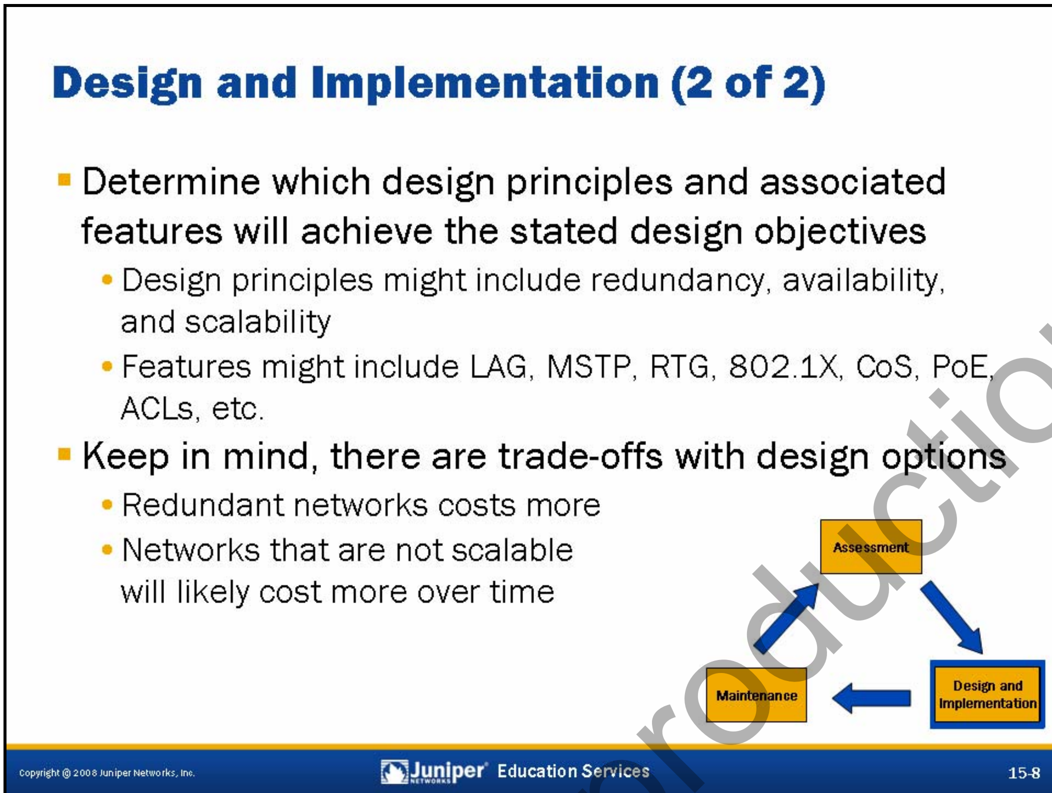
Defining Design Objectives

The assessment phase can produce a number of stated requirements along with a significant amount of facts and figures. Most of this information is used in the design and implementation phase to aid in the definition of design objectives.

One key design objective is to ensure that the physical and logical topology design satisfies the current needs of the company and accommodates growth. Any growth forecasts should be based on realistic data. Later in this chapter we discuss the benefits of implementing a hierarchical design, which facilitates growth within a company.

Another example of a design objective might include network and access control policies. Any network usage or communication guidelines determined in the assessment phase can be used to identify and implement network and access control policies. We recommend that the required policies be implemented as close to the end user as possible. This approach helps reduce unnecessary resource consumption within a network.

You can define a series of design objectives that center around user and traffic requirements. The number of users, classes of users, expected traffic types, and expected traffic patterns are used within the design and implementation phase to properly structure a network and account for high-usage connections within that network. Implementing sufficient bandwidth for high-usage connections increases network performance and decreases the likelihood of congestion. Many of today's networks implement differentiated service levels through class of service (CoS) to manage congestion for the various types of traffic.



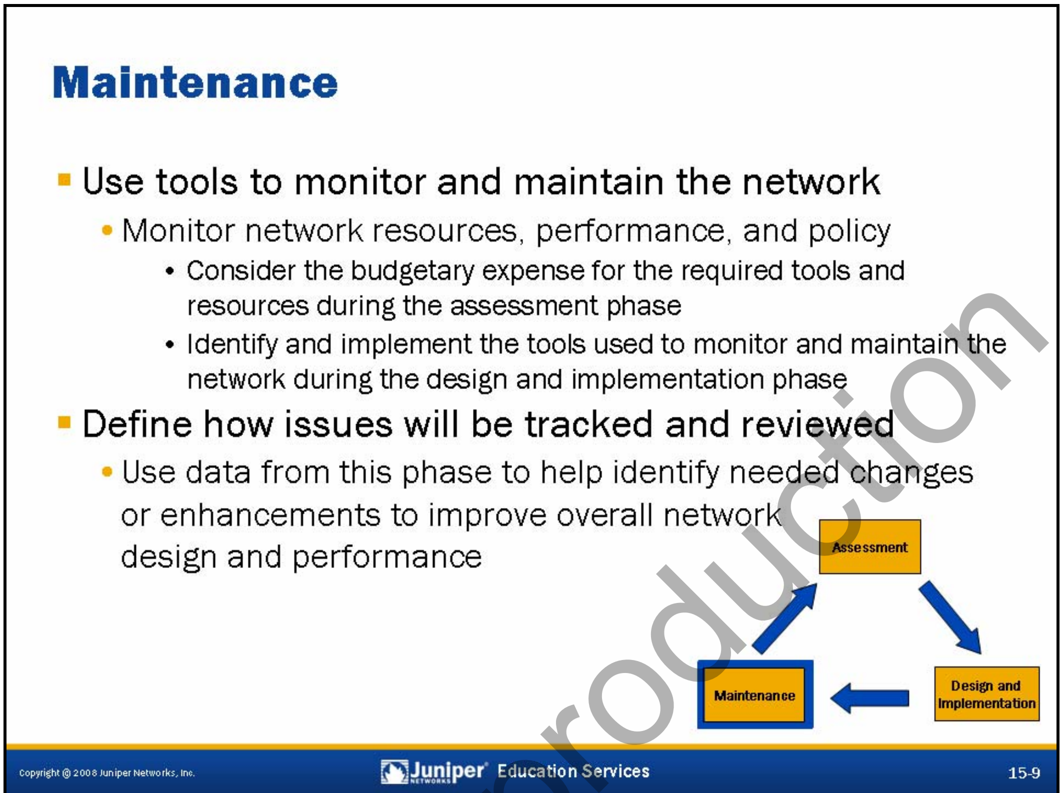
Design Principles and Features

Once a design objective is identified, the people responsible for designing and implementing the network must determine which design principles and associated features accomplish those objectives and meet the underlying requirements. The slide lists some of the possible design principles and associated features that can be used to satisfy the stated design objectives.

Considering the Trade-offs

When determining the best design option, there are a number of trade-offs to consider. If a requirement states that the network must be fully redundant and always available, the cost associated with that network increases significantly. The cost increases significantly because redundant, high-available networks typically consist of backup devices with secondary paths as well as redundant components, such as power supplies, within the individual devices.

On a similar note, a scalable network with hierarchical layers might initially cost more than a nonscalable, flat network. Although the cost associated with a flat network might be lower initially, that design option might cost more over time. Nonscalable design options often require major changes to the network infrastructure during times of growth. A scalable network, accommodating the same amount of growth, generally requires less changes and is less expensive. A scalable network design often includes other benefits as well, such as increased performance and ease of maintenance.

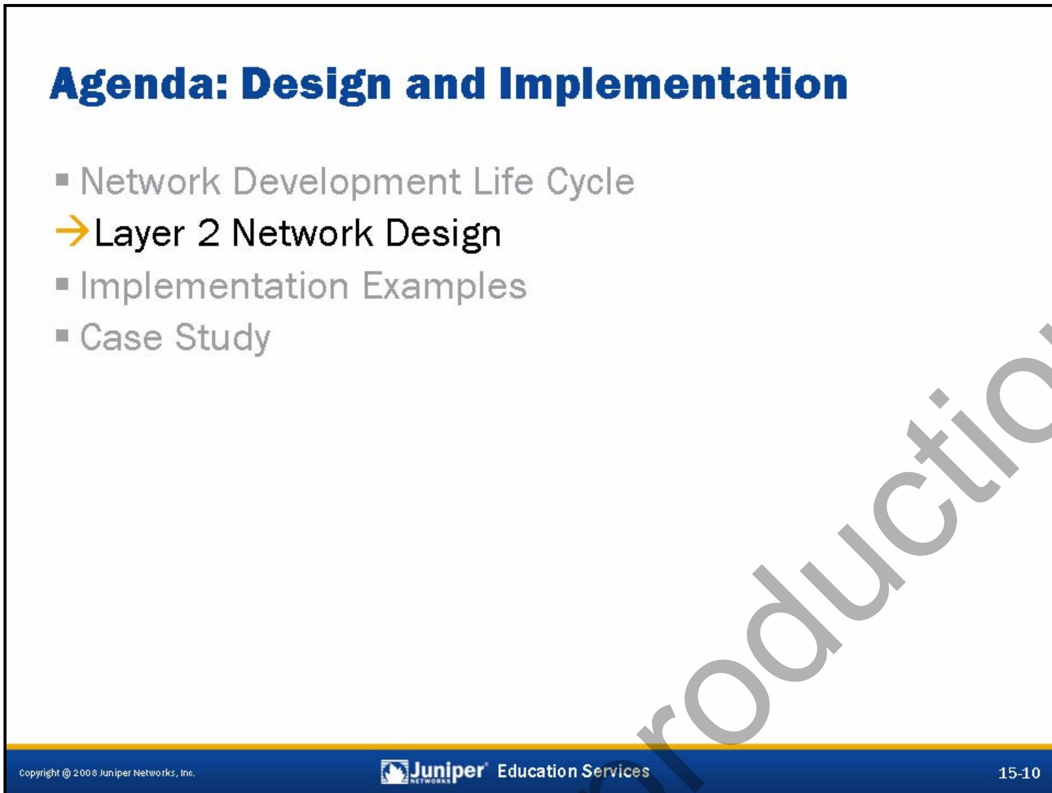


The Right Tool for the Job

Once the network is designed and implemented, it must be monitored and maintained. Selecting and using the appropriate tools for monitoring and maintaining a network often determines an organization's ability to get the most out of its investment. The proper tools allow administrators to monitor the network's resources, performance, and policies. Ideally, the tools required during the maintenance phase should be considered during the assessment phase because they will incur a budgetary expense. All monitoring and maintenance tools should be identified and implemented during the design and implementation phase.

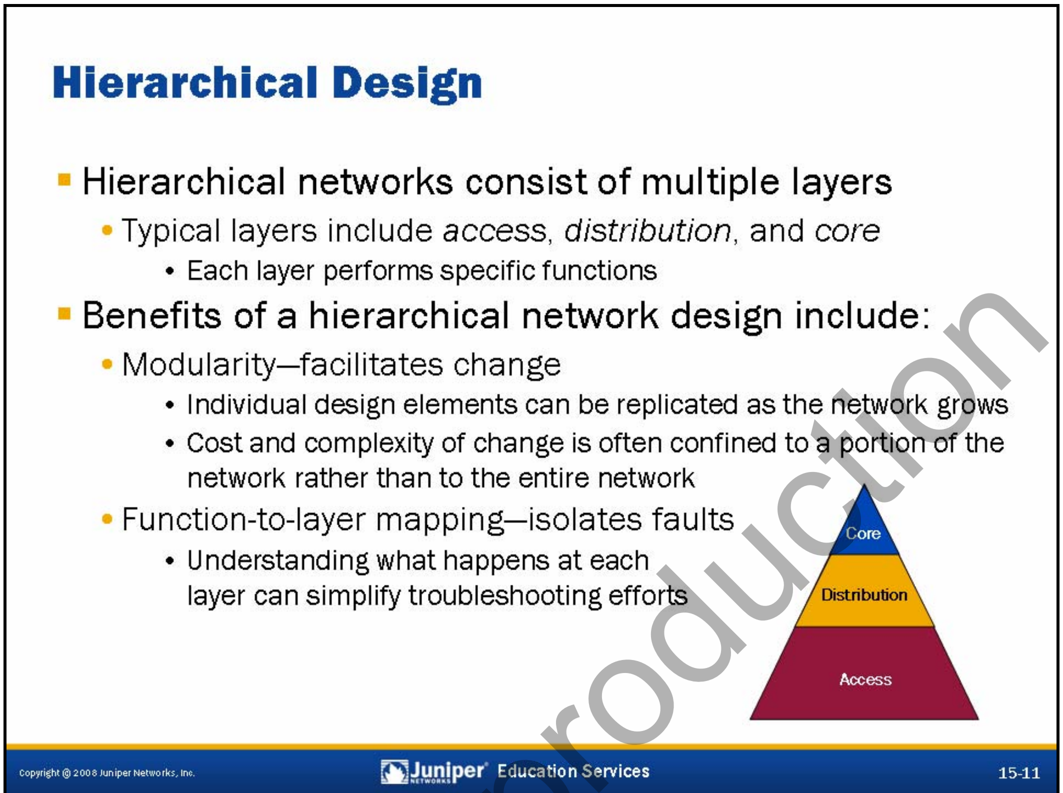
Tracking Issues

When issues are identified in the maintenance phase, they should be tracked. Some issues might reflect design flaws or point out opportunities for enhancing the current network implementation. The identified issues and opportunities should be collected and reviewed. If the issue or opportunity is significant, it should be reviewed or assessed by the same individuals who participated in the original assessment phase. Following this process helps identify needed changes or enhancements and encourages constant improvement within a network.



Layer 2 Network Design

The slide highlights the topic we discuss next.



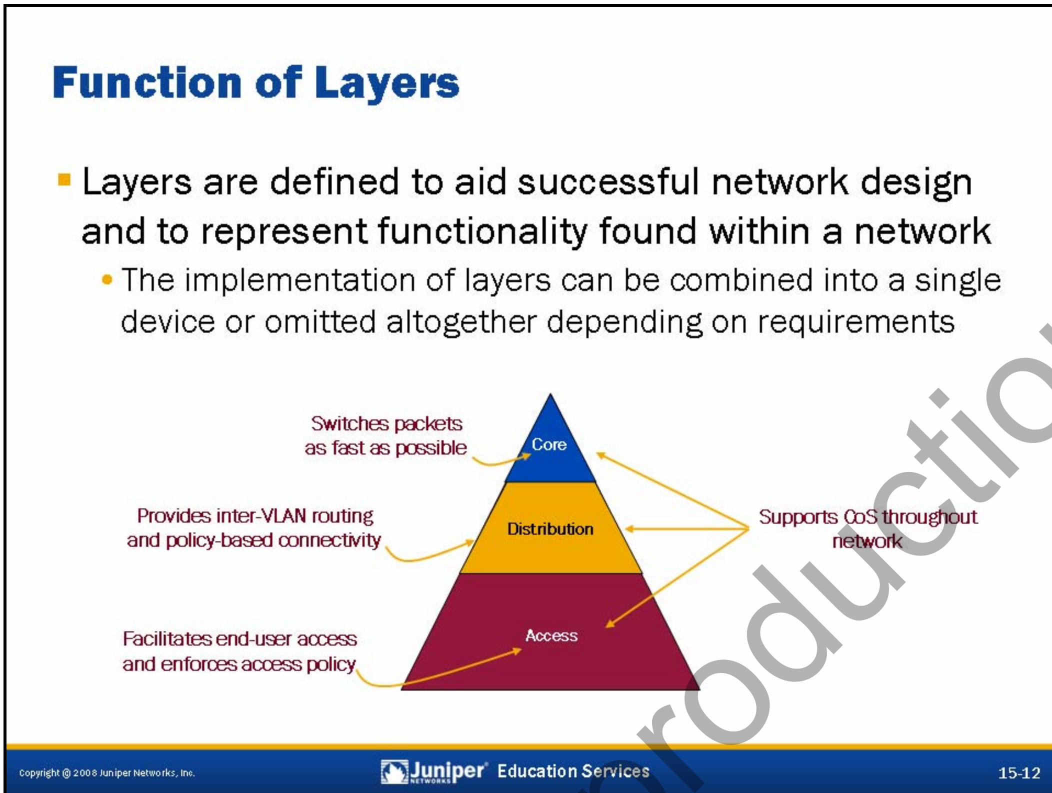
Multiple Layers

A hierarchical network consists of multiple layers. The image on the slide illustrates the typical layers, which include access, distribution, and core. Each of these layers performs unique responsibilities.

Benefits of a Hierarchical Design

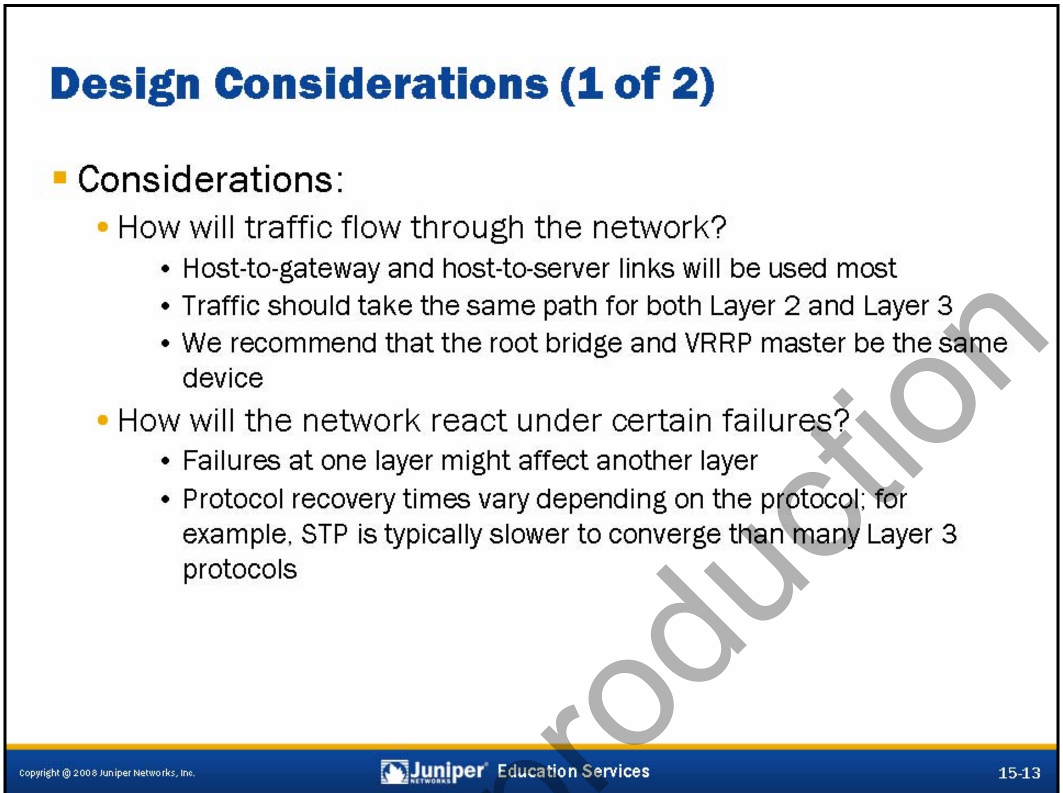
Hierarchical networks are designed in a modular fashion. This inherent modularity facilitates change and makes this design option quite scalable. When working with a hierarchical network, the individual elements can be replicated as the network grows. The cost and complexity of network changes is generally confined to a specific portion of the network rather than to the entire network.

Because functions are mapped to individual layers, faults relating to a specific function can be isolated to that function's corresponding layer. The ability to isolate faults to a specific layer can greatly simplify troubleshooting efforts.



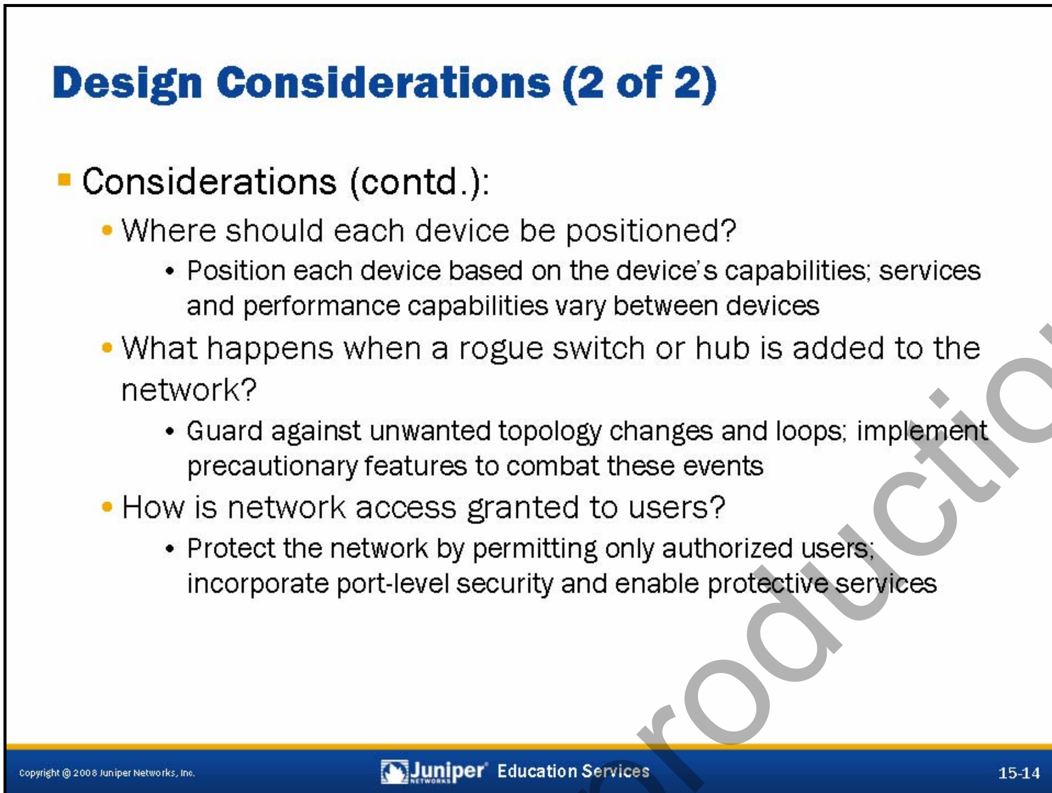
Network Design Assistance

When designing a hierarchical network, individual layers are defined and represent specific functions found within a network. It is often mistakenly thought that the access, distribution, and core layers must exist in clear and distinct physical devices, but this is not a requirement, nor does it make sense in some cases. The layers are defined to aid successful network design and to represent functionality that exist in many networks. The implementation of the three layers can be in distinct switches, can be combined in a single switch, or can be omitted altogether. The manner in which the layers are implemented should always depend on the network requirements and the design objectives. The slide highlights the access, distribution, and core layers and provides a brief description of the functions commonly implemented in those layers. If CoS is used in a network, it should be incorporated consistently in all three layers.



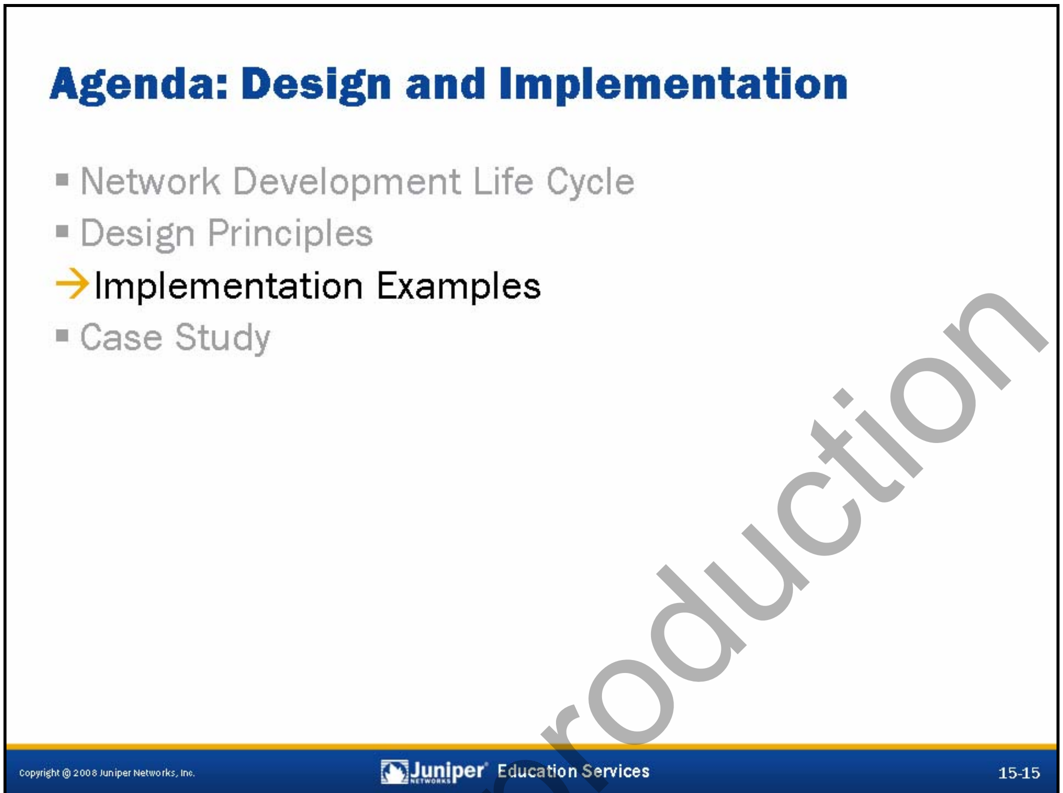
Design Considerations: Part 1

This slide and the next highlight some common considerations and recommendations when designing and implementing a network.



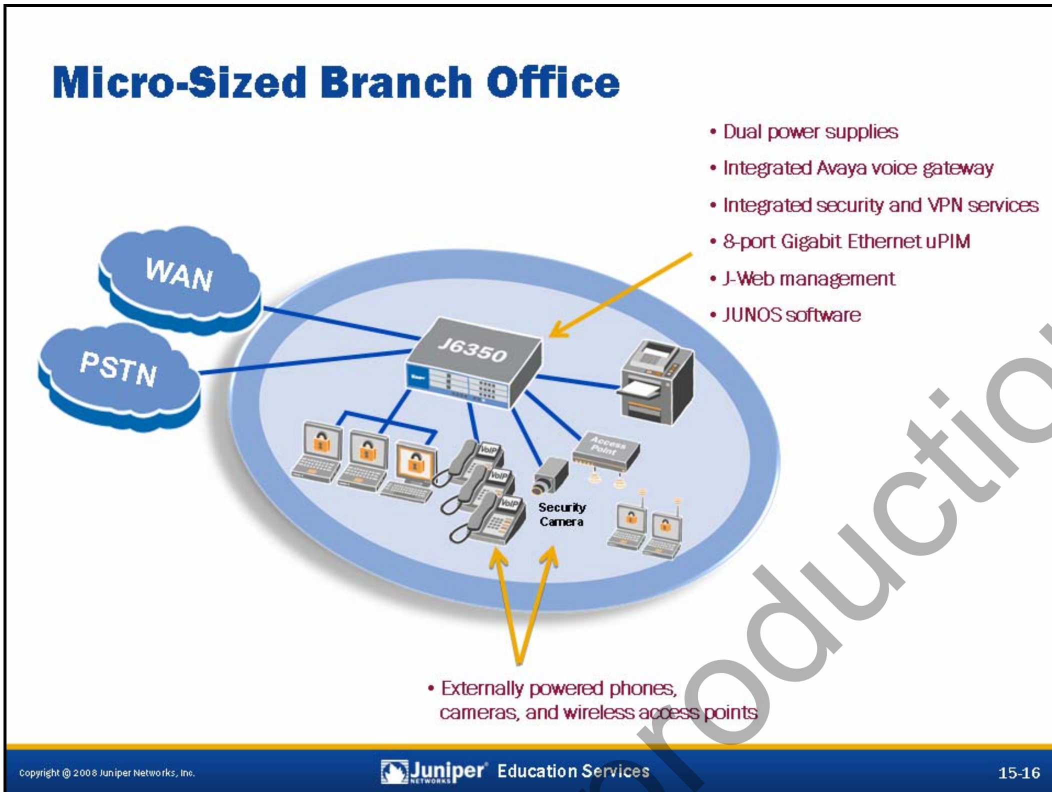
Design Considerations: Part 2

This slide illustrates the remainder of the highlighted considerations and recommendations when designing and implementing a network.



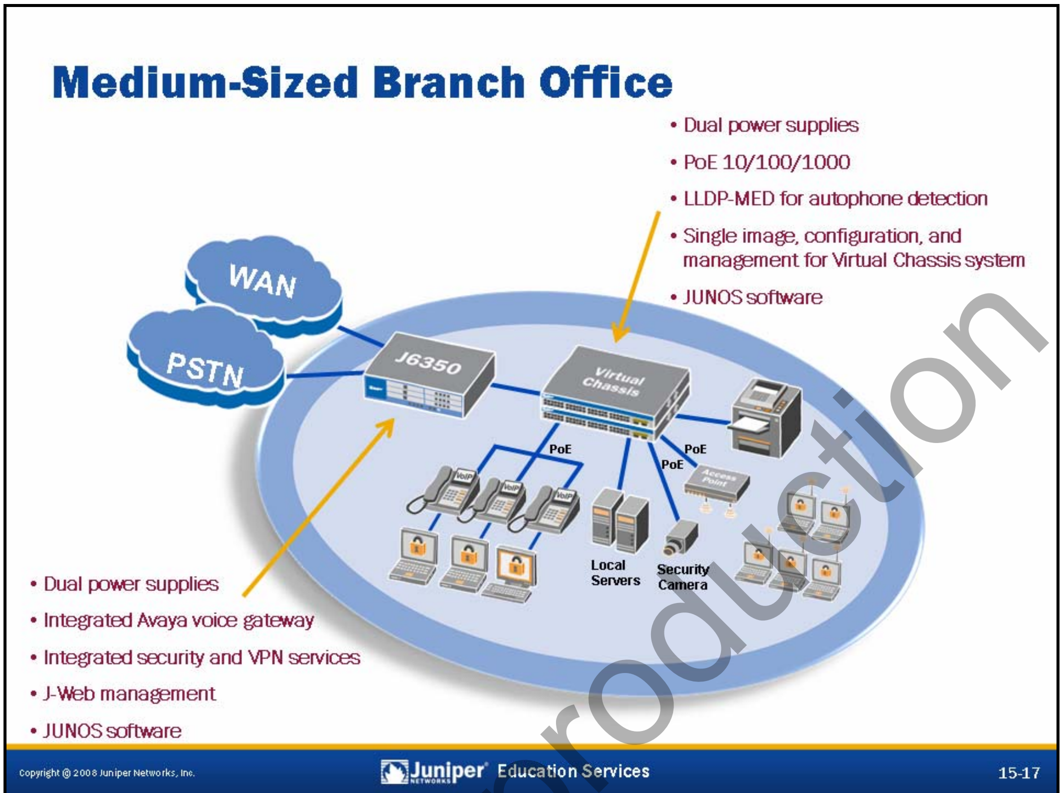
Implementation Examples

The slide highlights the topic we discuss next.



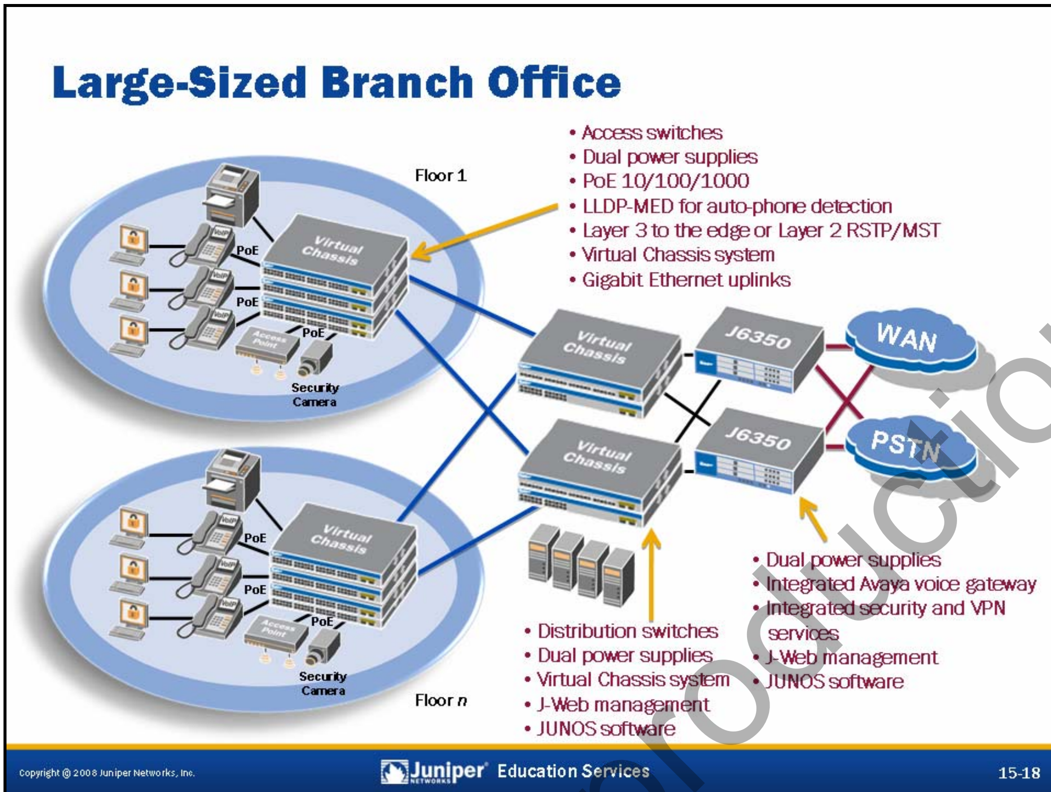
Micro-Sized Branch Office Implementation Example

This slide highlights a micro-branch office implementation example.



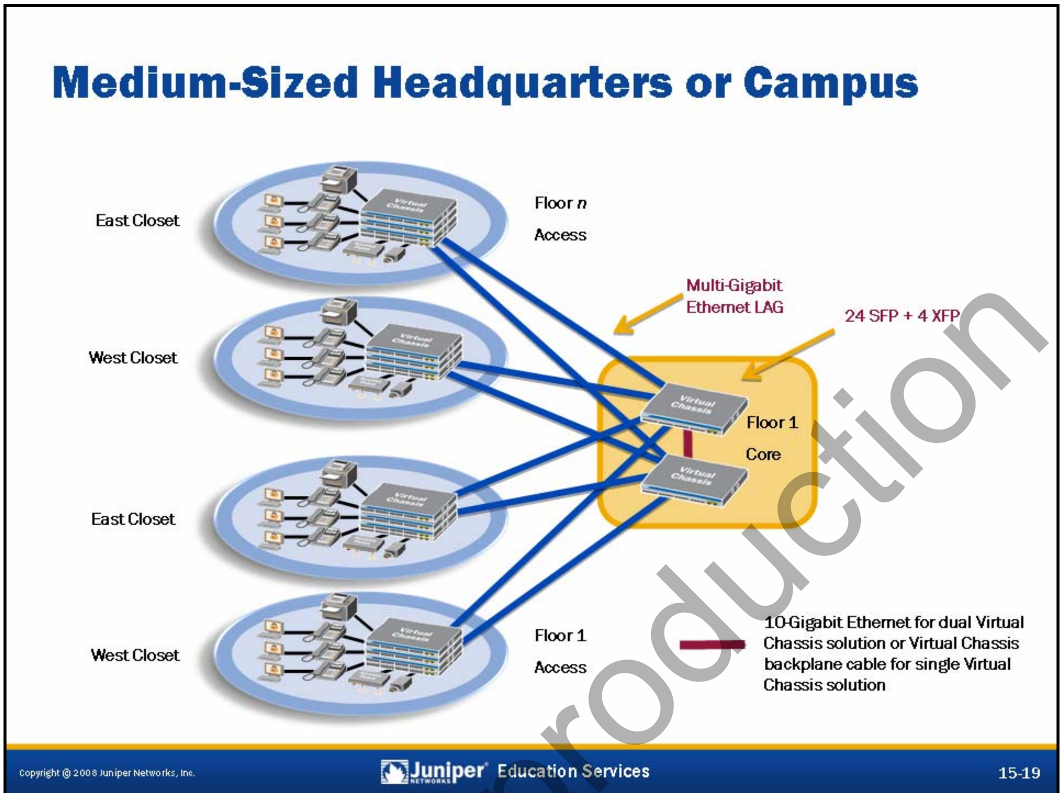
Medium-Sized Branch Office Implementation Example

This slide highlights a medium-sized branch office implementation example.



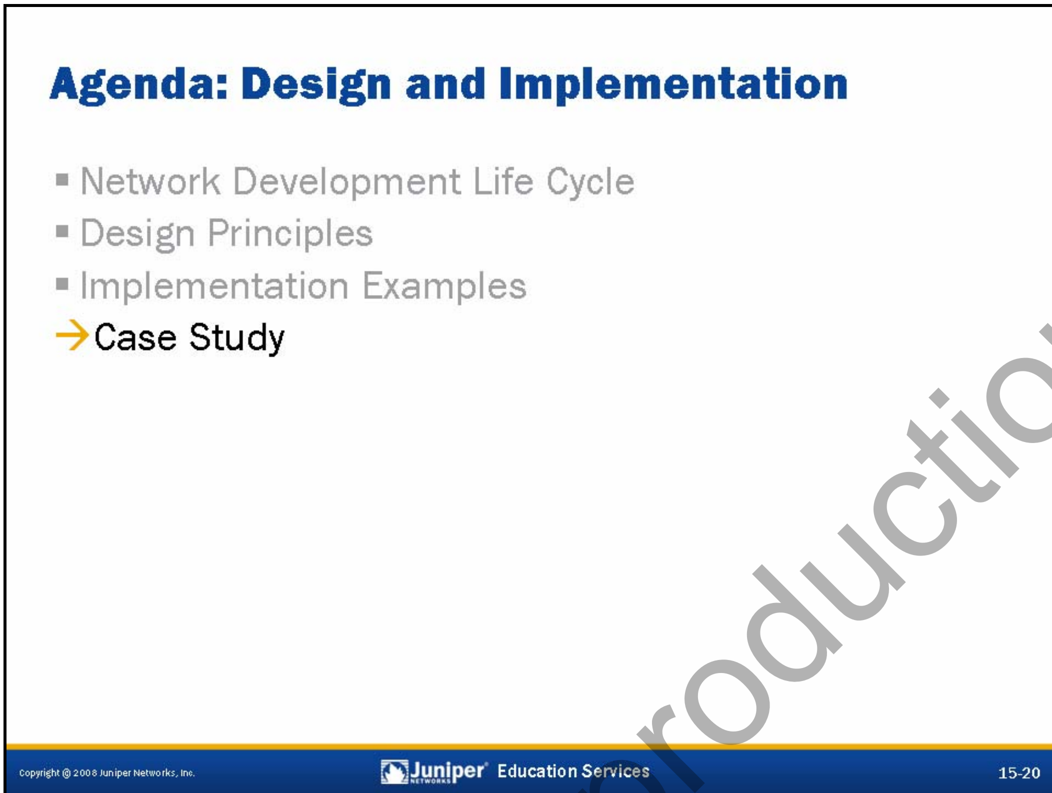
Large-Sized Branch Office Implementation Example

This slide highlights a large-sized branch office implementation example.



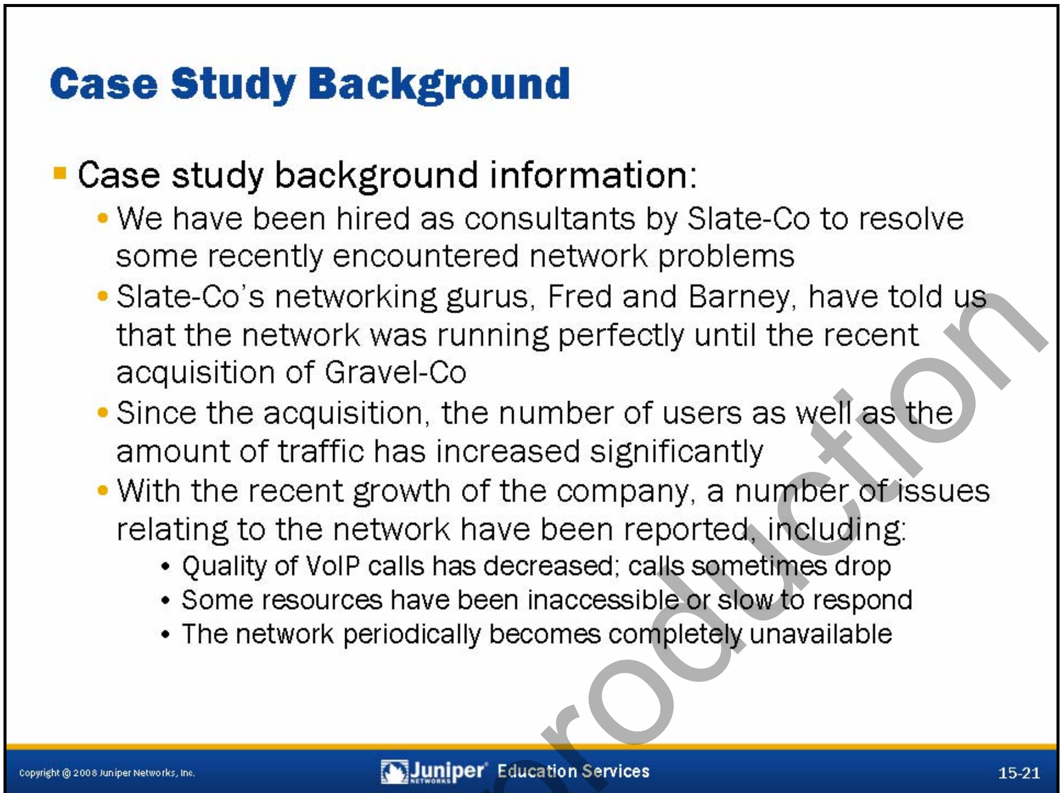
Medium-Sized Headquarters or Campus Implementation Example

This slide highlights a medium-sized headquarters or campus implementation example.



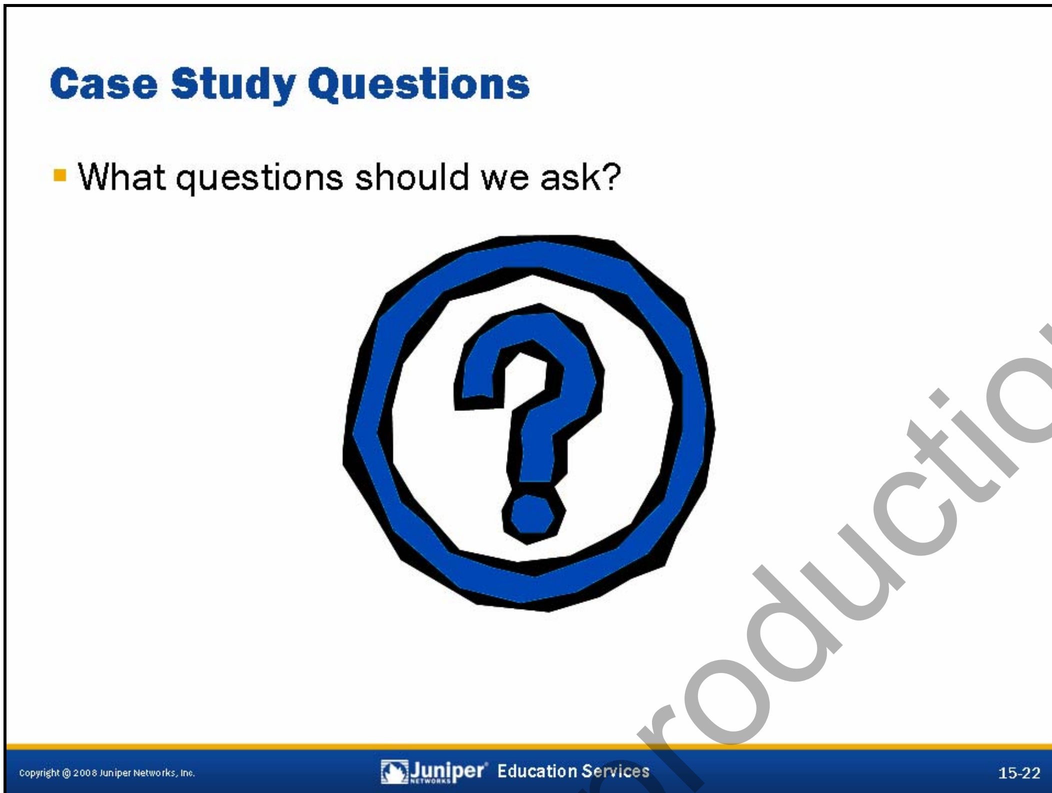
Case Study

The slide highlights the topic we discuss next.



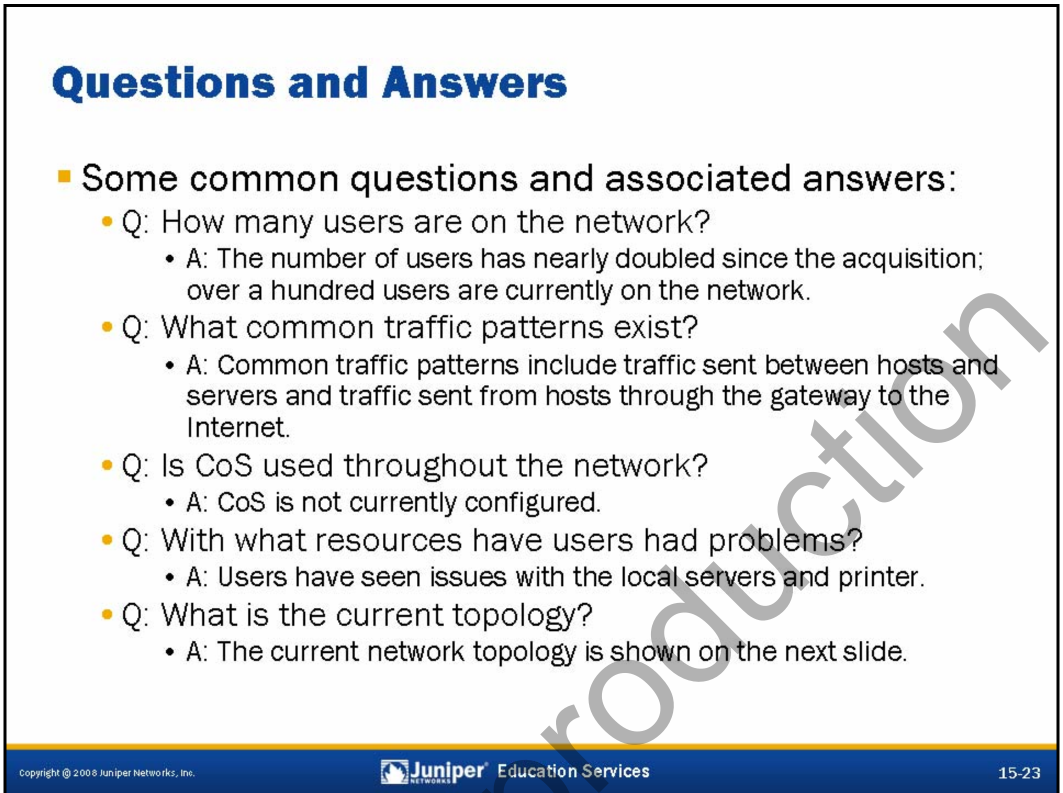
Case Study Background

This slide introduces a hypothetical situation and provides various details for the situation.



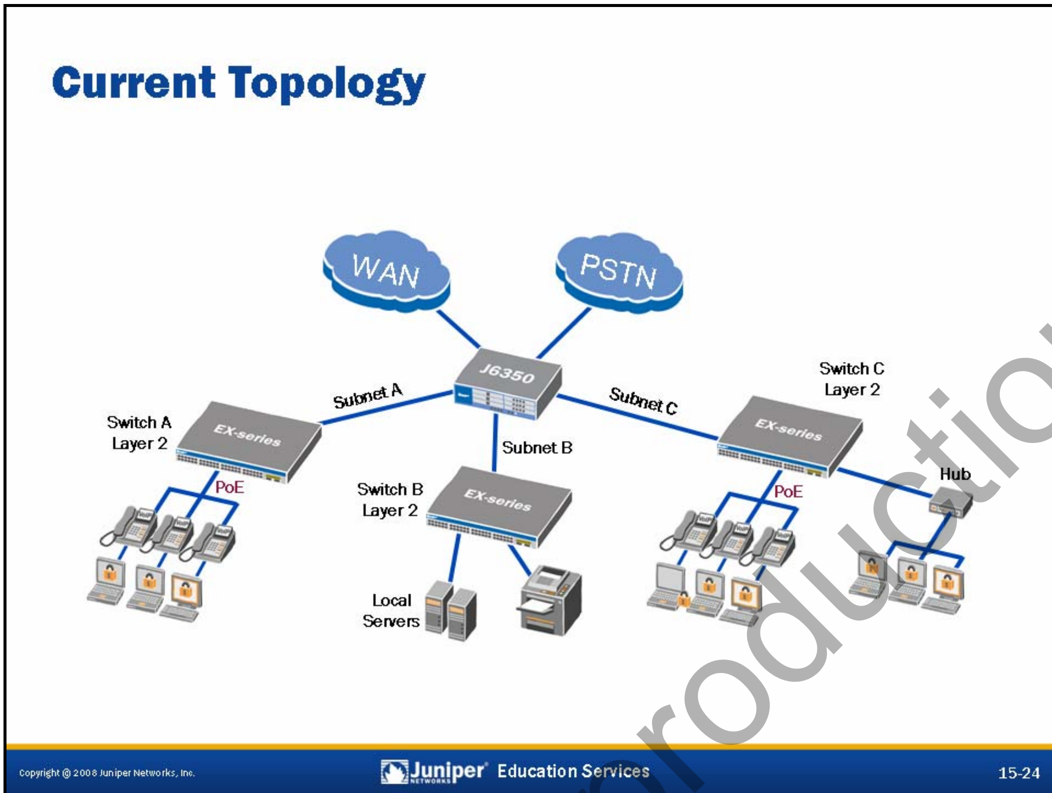
Additional Questions

What additional information would help identify issues with Slate-Co's current network implementation?



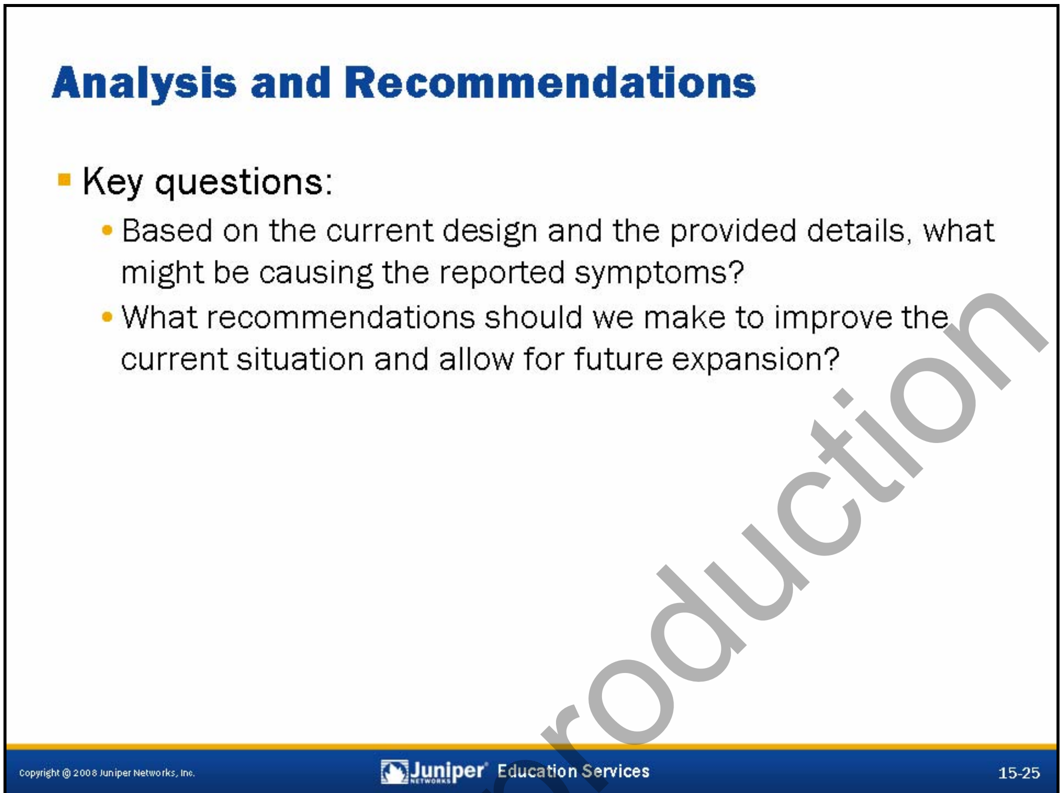
Did I Answer Your Question?

This slide illustrates some of the anticipated questions along with the answers for those questions. Other insightful questions might exist.



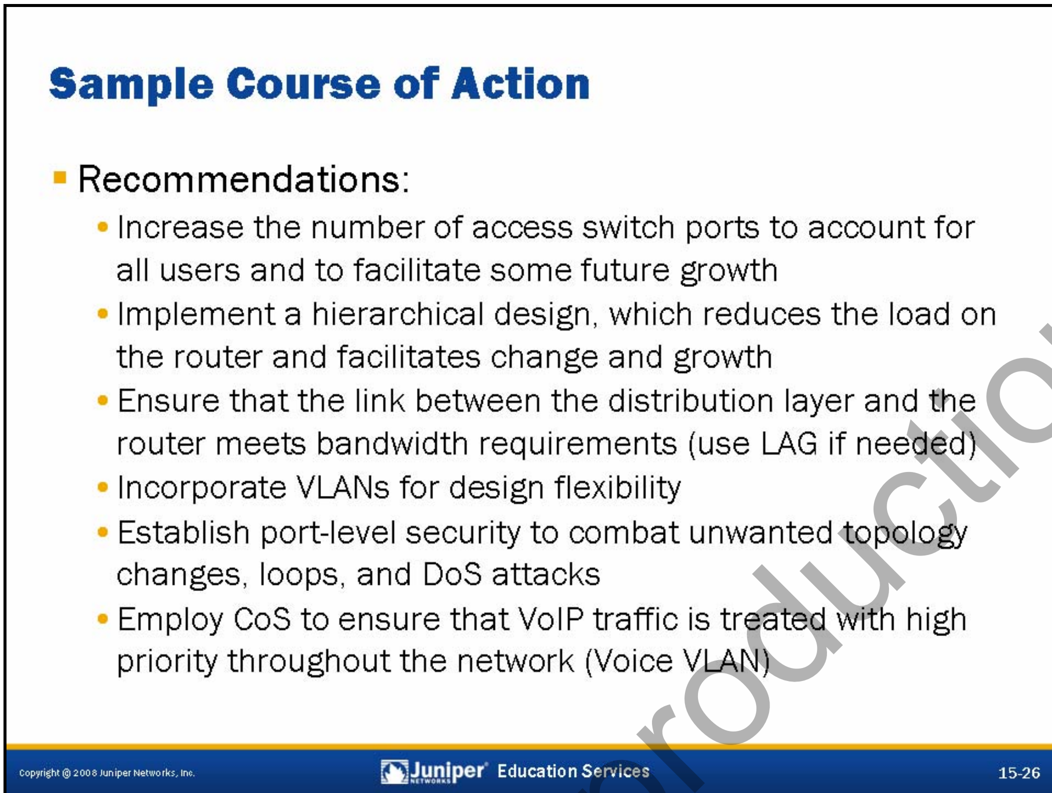
Current Topology

This slide displays Slate-Co's current topology.



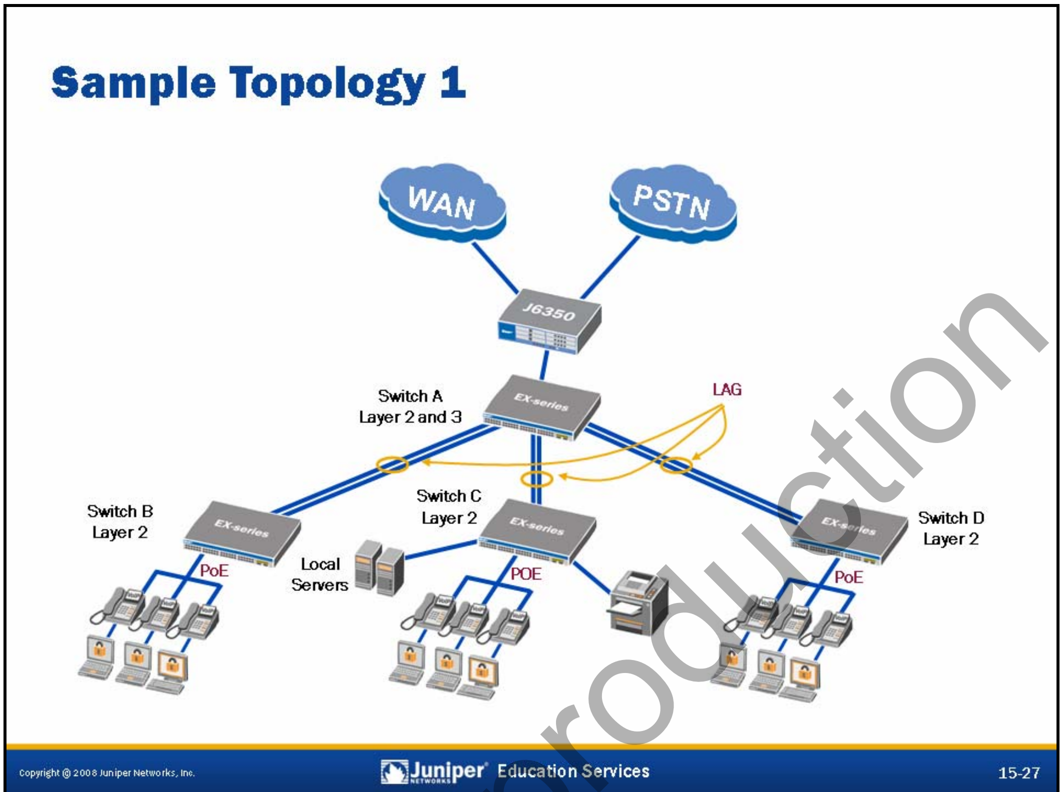
Analysis and Recommendations

This slide lists some key questions to aid in the analysis process. Based on the analysis, recommendations for improving the network can be made.



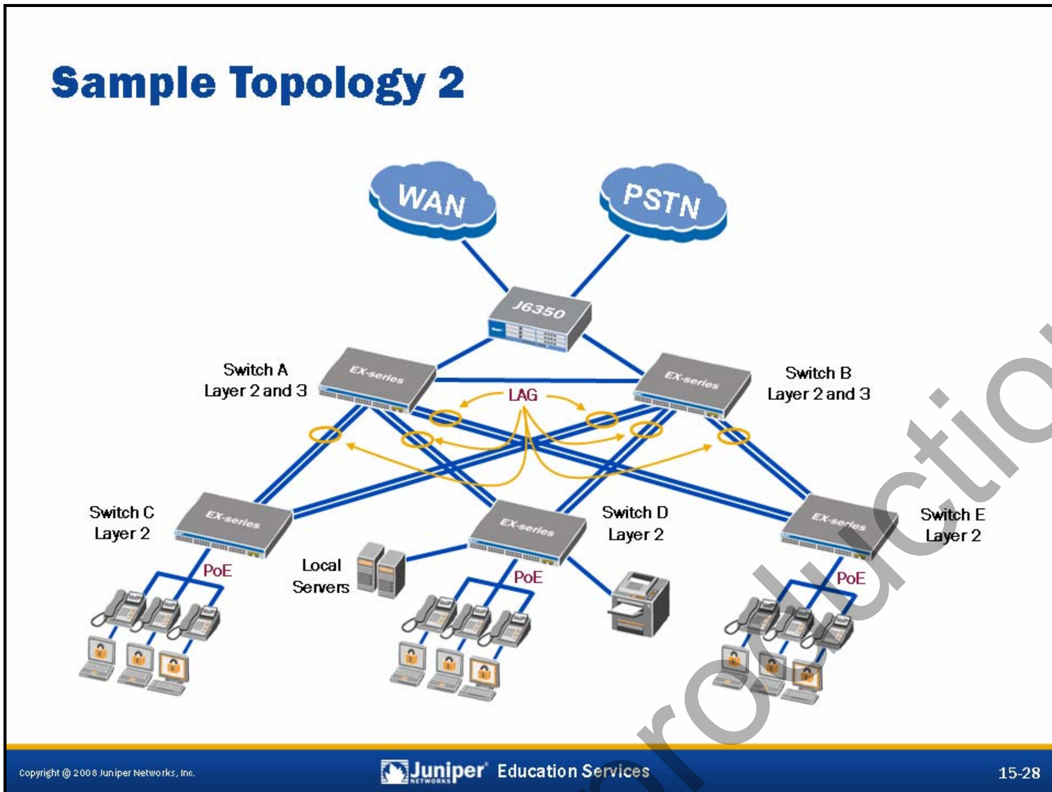
Sample Course of Action

This slide lists some recommendations on how we might proceed to improve Slate-Co's network. Other valid recommendations might exist.



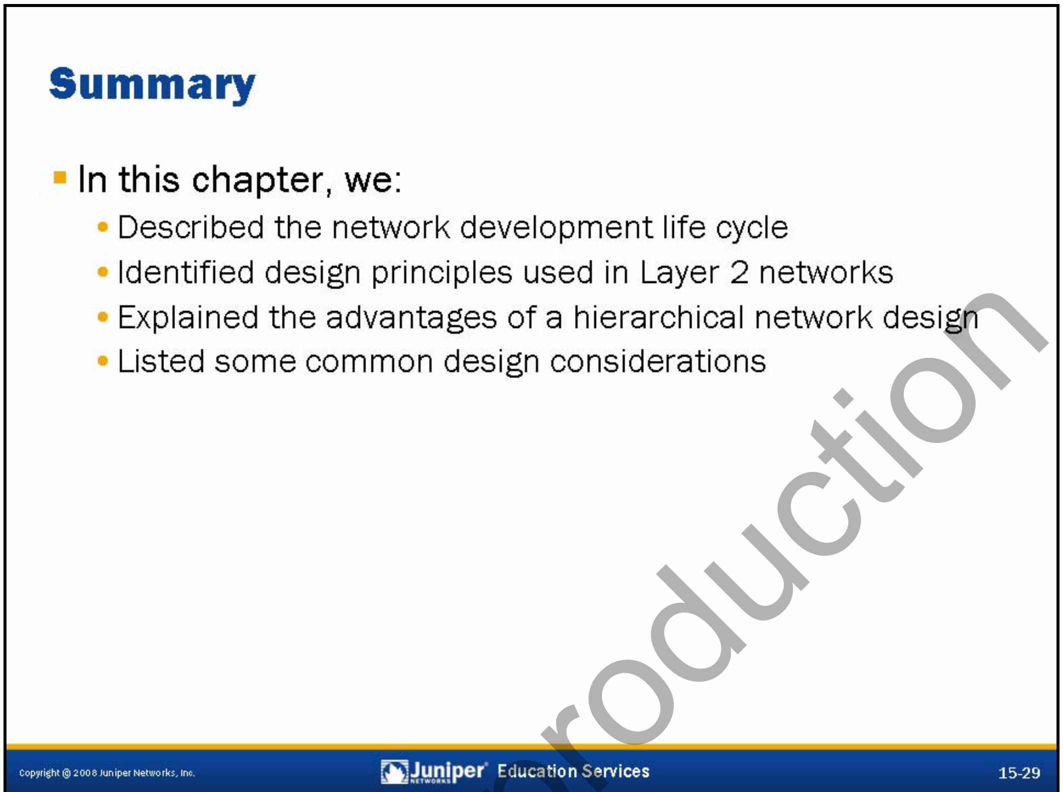
Sample Topology 1

This slide illustrates one possible design that might improve performance and allow for future growth in Slate-Co's network.



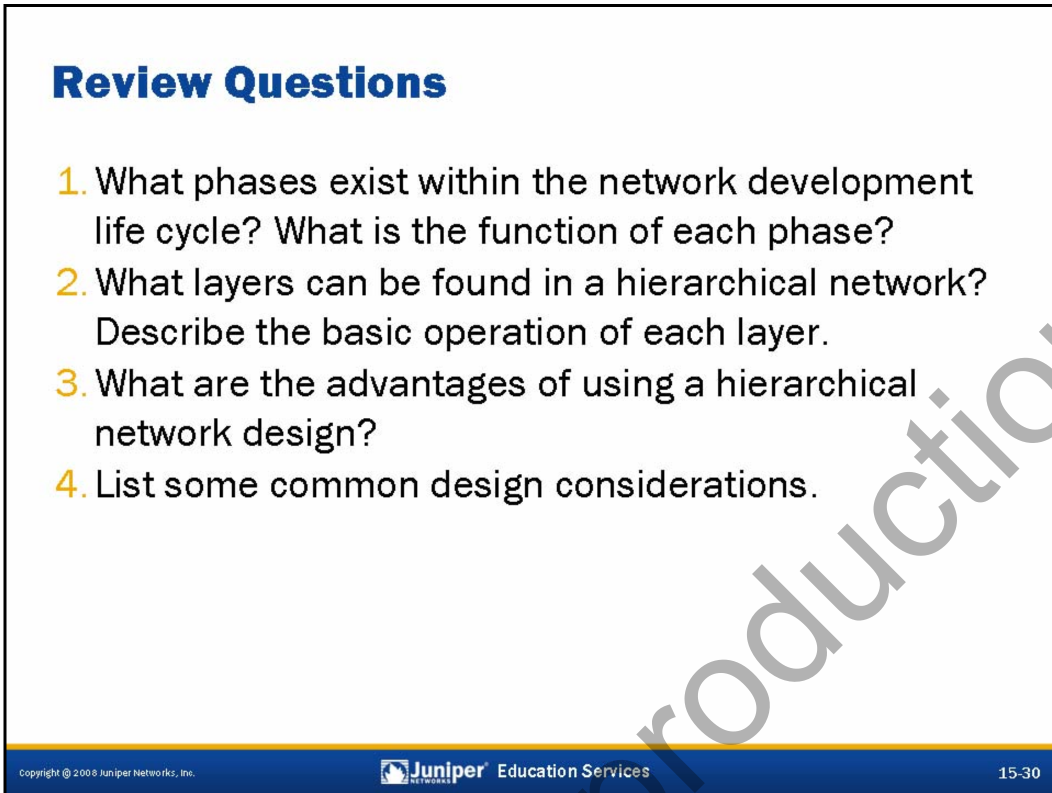
Sample Topology 2

This slide illustrates a second possible design that might also improve performance and allow for future growth in Slate-Co's network. This sample topology also incorporates some redundancy and costs more to implement.



This Chapter Discussed:

- The network development life cycle;
- Basic design principles used when designing Layer 2 networks;
- Advantages of implementing a hierarchical network design; and
- Common design considerations.



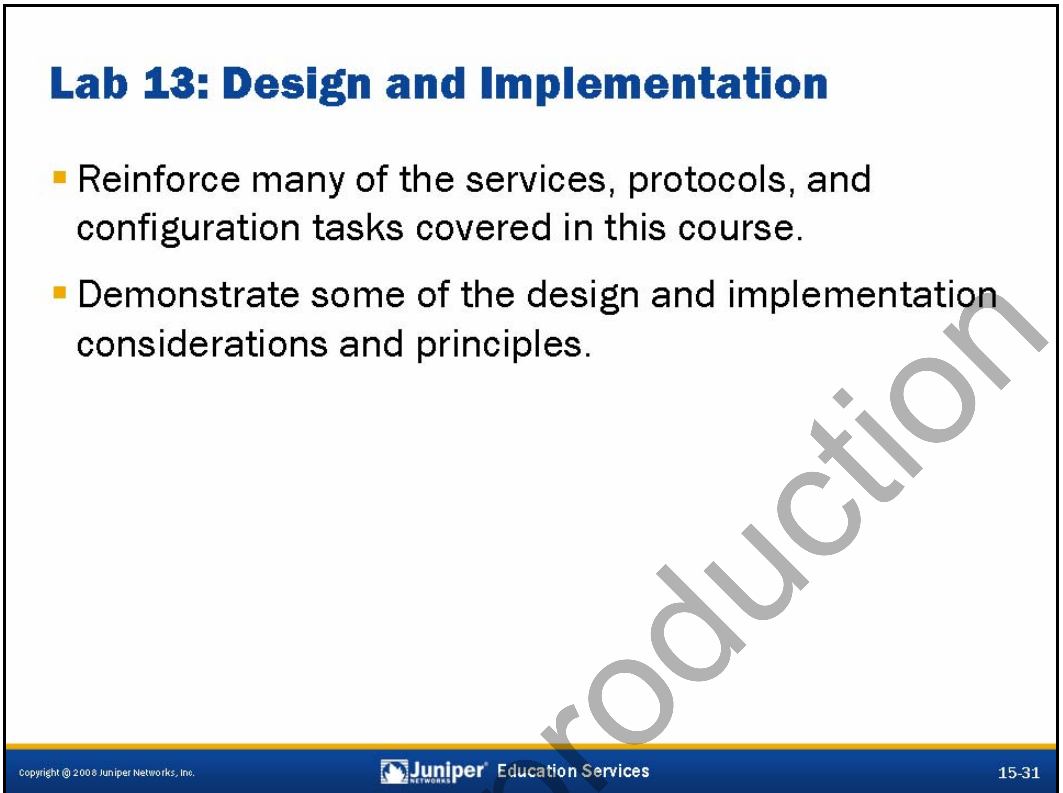
Review Questions

1.

2.

3.

4.



Lab 13: Design and Implementation

The slide provides the objectives for this lab.

Appendix A: Acronym List

ABR	area border router
ACK	acknowledgement
ANSI	American National Standards Institute
ARP	Address Resolution Protocol
AS	autonomous system
ASBR	autonomous system boundary router
ASIC	application-specific integrated circuit
BDR	backup designated router
BOOTP	Bootstrap Protocol
BPDU	bridge protocol data unit
BT	bridging table
CDP	Cisco Discovery Protocol
CE	customer edge
CIST	common and internal spanning tree
CLI	command-line interface
CoS	class of service
CSMA/CD	carrier-sense multiple access with collision detection
CST	common spanning tree
DAI	Dynamic ARP Inspection
dcd	device control process
DHCP	Dynamic Host Configuration Protocol
DNS	Domain Name System
DoS	denial of service
DR	designated router
DTE	data terminal equipment
EAP	Extensible Authentication Protocol
EAPOL	EAP over LAN
ESD	electrostatic discharge
FDDI	Fiber Distributed Data Interface
FRU	field-replaceable unit
FT	forwarding table
GB	gigabyte
GUI	graphical user interface
HSRP	Hot Standby Routing Protocol
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol over Secure Sockets Layer
ICMP	Internet Control Message Protocol
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force
IGP	interior gateway protocol
JNTCP	Juniper Networks Technical Certification Program
KB	kilobytes
LACP	Link Aggregation Control Protocol
LAG	link aggregation group
LLDP	Link Layer Discovery Protocol
LLDP-MED	Link Layer Discovery Protocol-Media Endpoint Discovery
LSA	link-state advertisement
MAC	media access control
MB	megabytes

MIB . . . . . Management Information Base
MSTI . . . . . multiple spanning tree instance
MSTP . . . . . Multiple Spanning Tree Protocol
NMS . . . . . network management system
NTP . . . . . Network Time Protocol
OID . . . . . object identifier
OoB . . . . . out-of-band
OSI . . . . . Open Systems Interconnection
PACL . . . . . port-based access control list
PD . . . . . powered device
PDU . . . . . protocol data unit
PFE . . . . . Packet Forwarding Engine
PoE . . . . . Power over Ethernet
pps . . . . . packets per second
PSE . . . . . power sourcing equipment
PSU . . . . . power supply unit
RACL . . . . . router-based access control list
RE . . . . . Routing Engine
RID . . . . . router ID
RMON . . . . . Remote Monitoring
RPS . . . . . redundant power supply
RSTP . . . . . Rapid Spanning Tree Protocol
RU . . . . . rack unit
RVI . . . . . routed VLAN interface
SFP . . . . . small form-factor pluggable transceiver
SPF . . . . . shortest path first
STP . . . . . Spanning Tree Protocol
TCAM . . . . . ternary content addressable memory
TCN . . . . . topology change notification
TIA . . . . . Telecommunications Industry Association
TLV . . . . . type/length/value
TTL . . . . . time to live
USB . . . . . universal serial bus
USM . . . . . user-based security model
VACL . . . . . VLAN-based access control list
VACM . . . . . view-based access control model
VCB . . . . . Virtual Chassis backplane
VCEP . . . . . Virtual Chassis extender port
VCP . . . . . Virtual Chassis port
VIP . . . . . virtual IP
VLAN . . . . . virtual LAN
VoIP . . . . . voice over IP
VRID . . . . . virtual router identifier
VRRP . . . . . Virtual Router Redundancy Protocol
vty . . . . . virtual terminal
XFP . . . . . 10-gigabit small form-factor pluggable transceiver