TSHOOT

Troubleshooting and Maintaining Cisco IP Networks

Volume 1

Version 1.0

Student Guide

Text Part Number: 97-2818-01

Americas Headquarters
Cisco Systems, Inc.
San Jose, CA

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

Europe Headquarters
Cisco Systems International BV
Amsterdam, The Netherlands

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

Students, this letter describes important course evaluation access information!

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

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

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

Sincerely,

Cisco Systems Learning

Table of Contents

Volume 1

Course Introduction

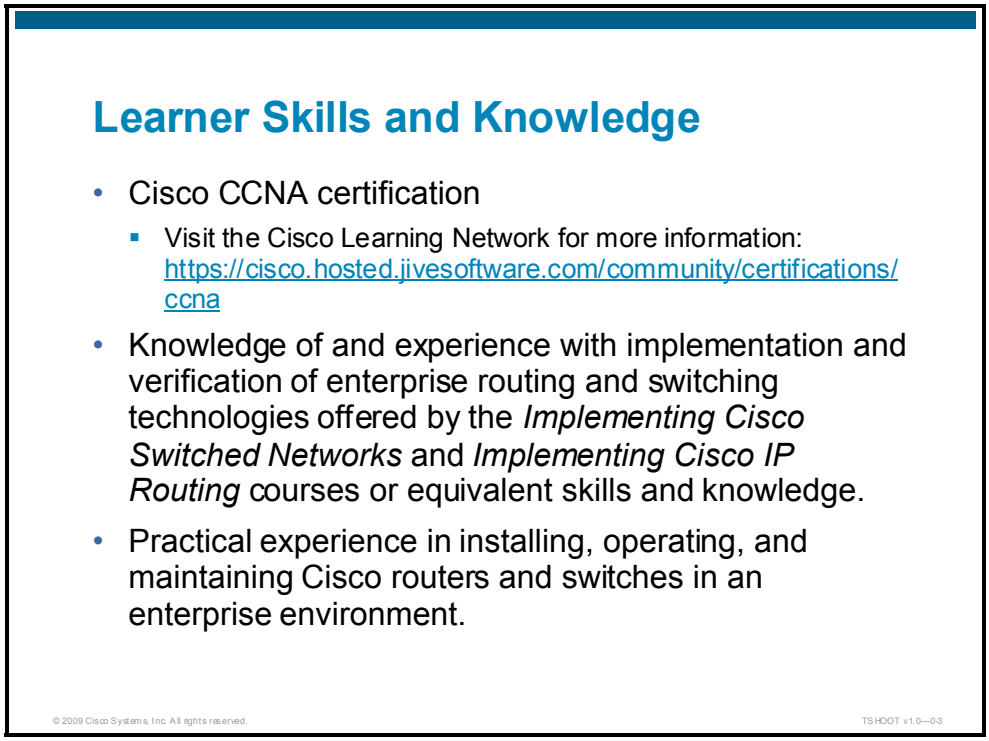
Overview

Troubleshooting and Maintaining Cisco IP Networks (TSHOOT) v1.0 is an instructor-led training course that is presented by Cisco training partners to customers who use Cisco products. This five-day course is designed to help network professionals improve the skills and knowledge that they need to maintain their network and to diagnose and resolve network problems quickly and effectively. It also assists the network professional in preparing for Cisco CCNP® certification. This course is a component of the CCNP curriculum.

The course is designed to teach professionals who work in complex network environments the skills they need to maintain their networks and to diagnose and resolve network problems quickly and effectively. The course will provide information about troubleshooting and maintaining particular technologies, as well as procedural and organizational aspects of the troubleshooting and maintenance process. A large part of the training will consist of practicing these skills and reinforcing the concepts by putting them to use in a controlled environment. At the end of the course, you should have increased your skill level and developed a set of best practices that are based on your own experience and the experiences of other students and that you can take back to their organizations.

Student Skills and Knowledge

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



In addition to CCNA® certification, it is recommended that the student have practical experience in installing, operating, and maintaining Cisco routers and switches in an enterprise environment.

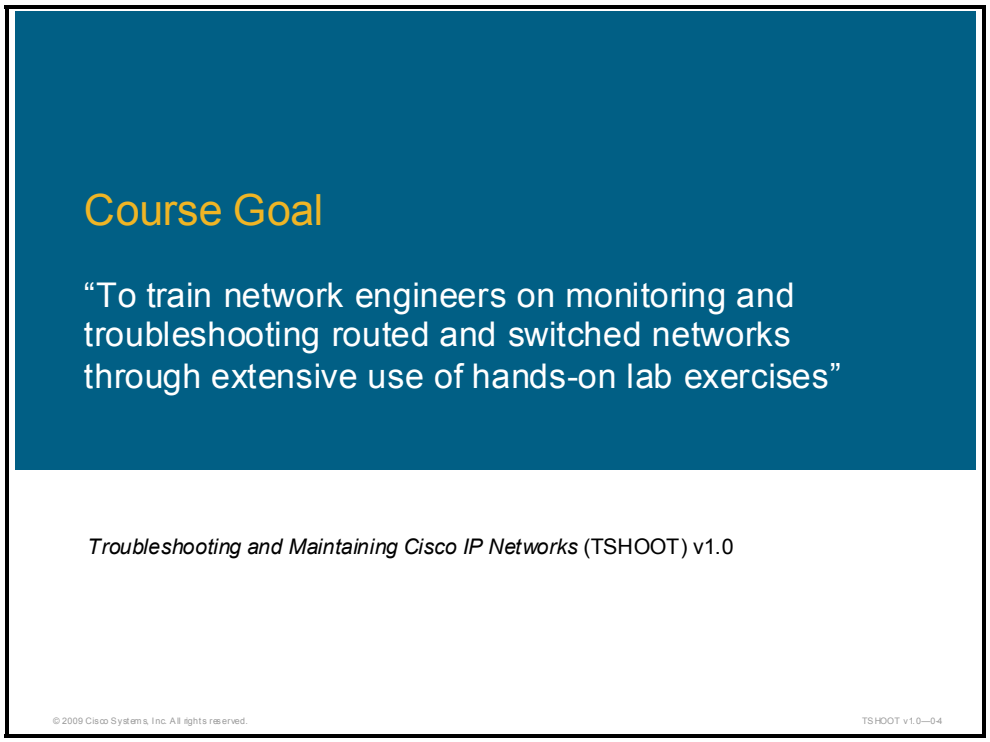
Students should also have knowledge of and experience with the implementation and verification of enterprise routing and switching technologies as offered by the *Implementing Cisco Switched Networks* (SWITCH) and *Implementing Cisco IP Routing* (ROUTE) courses or equivalent skills and knowledge. This includes knowledge and experience of the following technologies:

- Layer 2 switching
  - Private VLANs, VLAN access control lists, port security
  - Switch security issues
- Link aggregation protocols
- Spanning Tree Protocol (STP)
  - Multiple Spanning Tree (MST)
  - Per VLAN Spanning Tree (PVST)
  - Per VLAN Rapid Spanning Tree (PVRST)
- Inter-VLAN routing solutions
- First Hop Redundancy Protocols (FHRPs)
  - Hot Standby Router Protocol (HSRP)
  - Virtual Router Redundancy Protocol (VRRP)
  - Gateway Load Balancing Protocol (GLBP)
- Infrastructure support of wireless, VoIP and video
- Branch office operations

- Enhanced Interior Gateway Routing Protocol (EIGRP)
- Open Shortest Path First (OSPF)
- Layer 3 path control
- Redistribution
- External Border Gateway Protocol (EBGP)
- IP version 6 (IPv6) migration

Course Goal and Objectives

This topic describes the course goal and objectives.

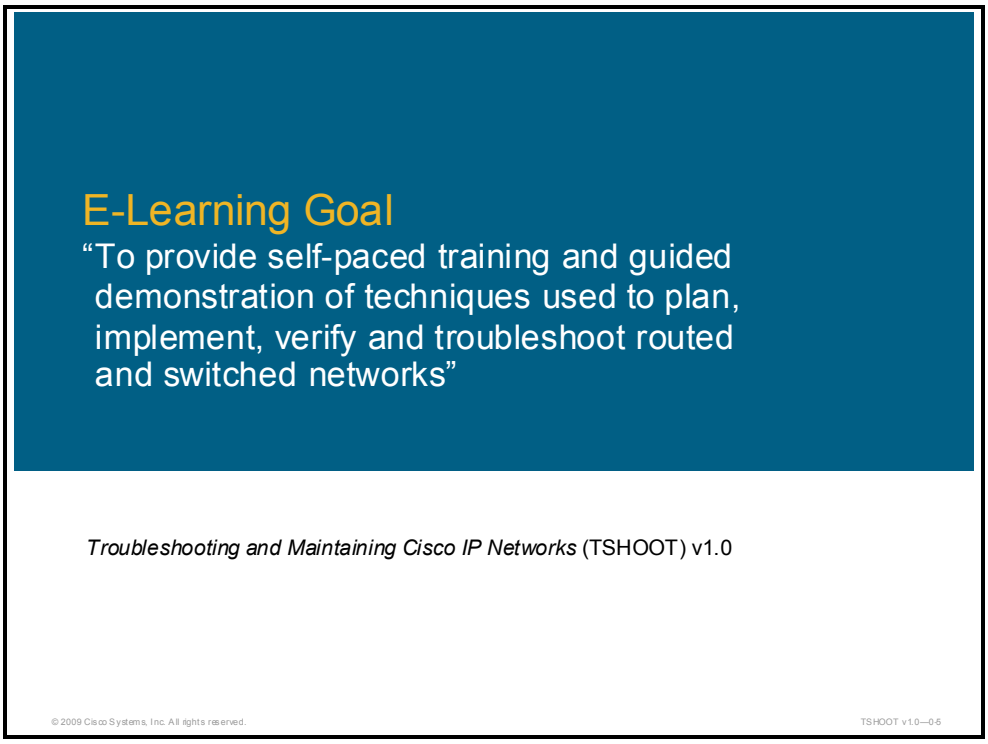


Upon completing this course, you will be able to meet these objectives:

- Plan and document the most commonly performed maintenance functions in complex enterprise networks
- Develop a troubleshooting process to identify and resolve problems in complex enterprise networks
- Select tools that best support specific troubleshooting and maintenance processes in large, complex enterprise networks
- Practice maintenance procedures and fault resolution in switching-based environments
- Practice maintenance procedures and fault resolution in routing-based environments
- Practice maintenance procedures and fault resolution in a secure infrastructure
- Troubleshoot and maintain integrated, complex enterprise networks

E-Learning Goal and Objectives

This topic describes the course goal and objectives of the e-learning course *Troubleshooting and Maintaining Cisco IP Networks* (TSHOOT) v1.0.



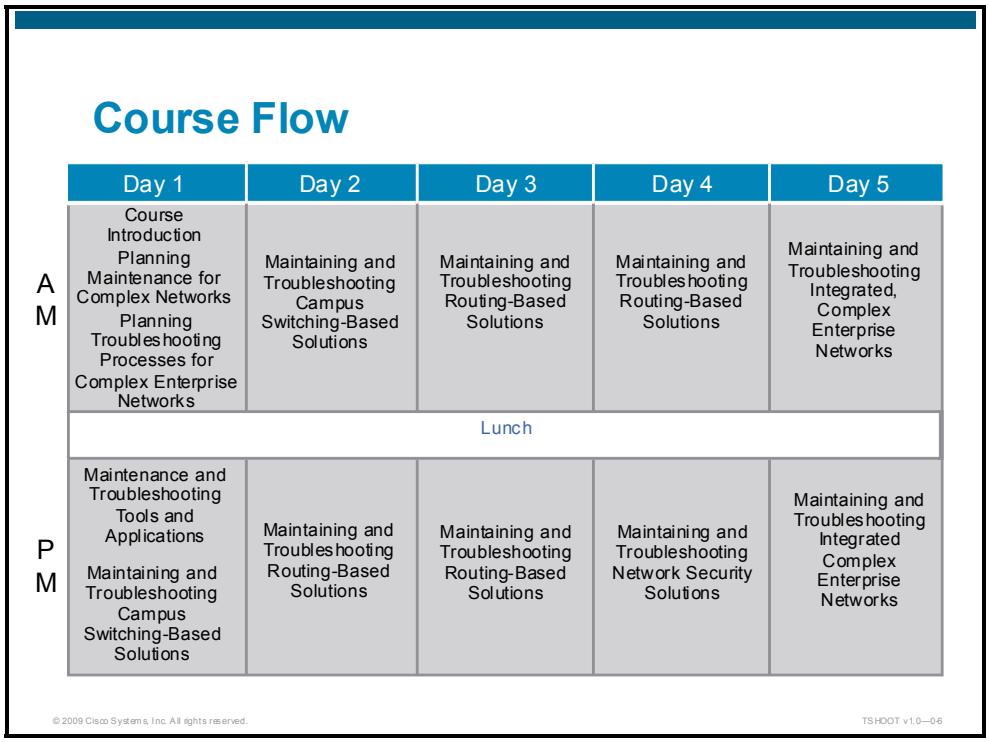
The e-learning modules provide self-paced training and guided demonstrations of the technology topics that are covered on the TSHOOT exam. The e-learning modules, along with the instructor-led training (ILT) course, form the complete curriculum on troubleshooting routed and switched networks that is recommended by Cisco.

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

- Troubleshoot performance problems on switches
- Diagnose wireless connectivity problems
- Diagnose problems in transporting VoIP
- Diagnose problems in transporting video
- Troubleshoot NAT
- Troubleshoot DHCP
- Troubleshoot IPv6, OSPFv3, and RIPng
- Maintain and Troubleshoot Network Applications Services
- Troubleshoot Branch Office and Remote Worker Problems

Course Flow

This topic presents the suggested flow of the course materials.

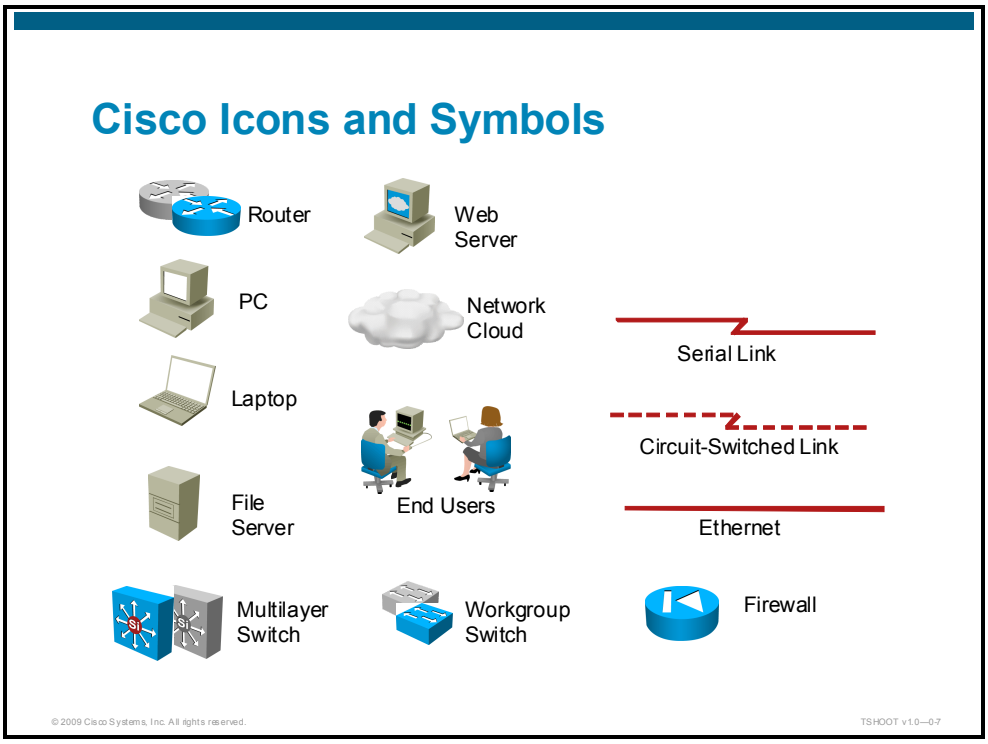


The schedule reflects the recommended structure for this course. This structure allows enough time for the instructor to present the course information and for you to work through the lab activities. The exact timing of the subject materials and labs depends on the pace of your specific class.

Much of the time in this class is spent on labs. In addition to teaching you about network troubleshooting and maintenance processes, tools, and techniques, this course offers many opportunities for you to apply the knowledge that is gained and practice using these processes, tools, and techniques to improve your skills. Each lab consists of a number of trouble tickets that allow you to practice troubleshooting particular protocols and technologies. Although sample solutions to the problems will be provided, the biggest value in these labs is in the process itself and the opportunity that it offers to learn from the instructor as well as your peers in the industry. To accommodate the discussion of best practices for processes, procedures, and tools, each lab will be followed by a lab debrief where the instructor will lead a group discussion to allow you to compare your solutions and processes to those of other students and together establish a set of best practices for network troubleshooting and maintenance.

Additional References

This topic presents the Cisco icons and symbols that are used in this course, as well as information on where to find additional technical references.

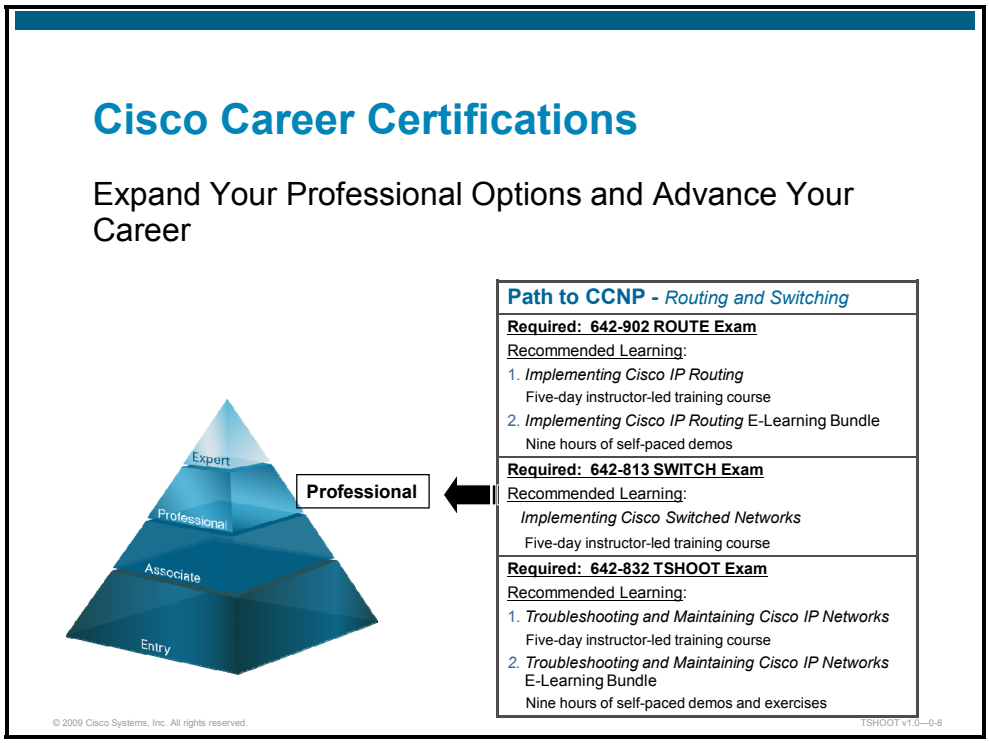


Cisco Glossary of Terms

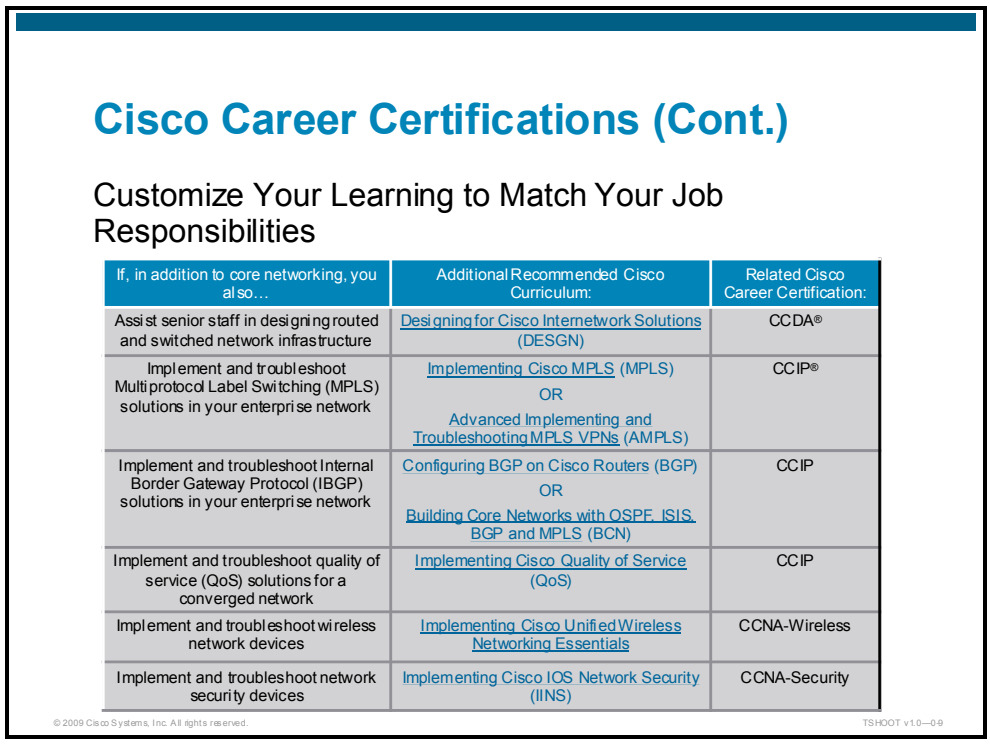
For additional information on Cisco terminology, refer to the *Cisco Internetworking Terms and Acronyms* glossary of terms at http://www.cisco.com/en/US/docs/internetworking/terms_acronyms/ita.html.

Your Training Curriculum

This topic presents the training curriculum for this course.



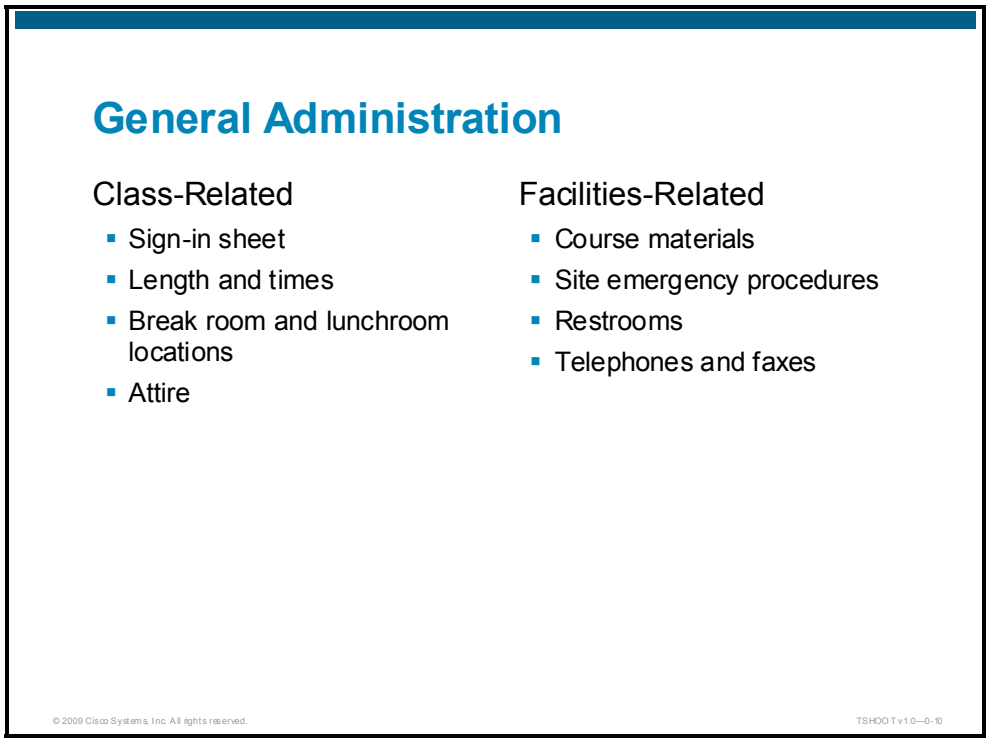
You are encouraged to join the Cisco Learning Network, a social learning network designed to enhance and advance your IT career. Browse technical content, connect, and share insights, opinions, and knowledge with the community. The network is open to any registered participant. Visit http://www.cisco.com/go/learningnetwork.



If you require additional skills and knowledge to complement the skills and knowledge acquired as part of your CCNP training, Cisco offers many additional curricula and certifications to help you obtain skills suitable to your job role.

General Administration

This topic presents the general administration for this course.



The instructor will discuss the following administrative issues so that you know exactly what to expect from the class:

- Sign-in process
- Start and anticipated end times of each class day
- Class break and lunch facilities
- Appropriate attire during class
- Materials you can expect to receive during class
- What to do in the event of an emergency
- Location of the restrooms
- How to send and receive telephone and fax messages

Learner Introductions

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

TSHOOT v1.0—0-11

Prepare to share this information:

- Your name
- Your company
- Your job responsibilities
- The prerequisite skills that you have
- A profile of your experience
- What you would like to learn from this course

Module 1

Planning Maintenance for Complex Networks

Overview

Smooth network operation and network high availability have become crucial to organizations because they depend on their network infrastructure for more and more of their business processes. Unplanned downtime can quickly lead to loss of productivity, loss of reputation, or even direct loss of revenue.

To maximize the availability of the network, while at the same time controlling the associated costs, it is important to plan network maintenance processes and procedures carefully.

In this module, you will identify the tasks that network engineers perform as part of the job of maintaining large, complex networks. You will evaluate the methodologies, models, and processes that help in structuring these tasks. In addition, you will identify and evaluate organizational aspects, tools, and resources that engineers could use to execute these tasks more efficiently.

Module Objectives

Upon completing this module, you will be able to plan and document the most commonly performed maintenance functions in complex enterprise networks. This ability includes being able to meet these objectives:

- Evaluate and rate commonly practiced models and methodologies for network maintenance
- Identify the processes and procedures that are a fundamental part of any network maintenance methodology
- Identify, evaluate, and select the tools, applications, and resources that are needed to support network maintenance processes

Lesson 1

Applying Maintenance Methodologies

Overview

The tasks of maintaining and supporting networks and network equipment are among the core job tasks of a network engineer. Network maintenance essentially consists of all the tasks that need to be done to keep the network running smoothly and minimize any disruptions to the business processes that depend on the network.

Network maintenance includes both interrupt-driven tasks, such as responding to device and link failures and supporting users, and regularly scheduled tasks, such as making backups and upgrading devices or software.

By applying a structured approach to network maintenance, you can reduce unplanned outages and minimize network downtime. This lesson presents a typical approach. Your skills, past experience, and most importantly, the policies and procedures of your company will govern your approach.

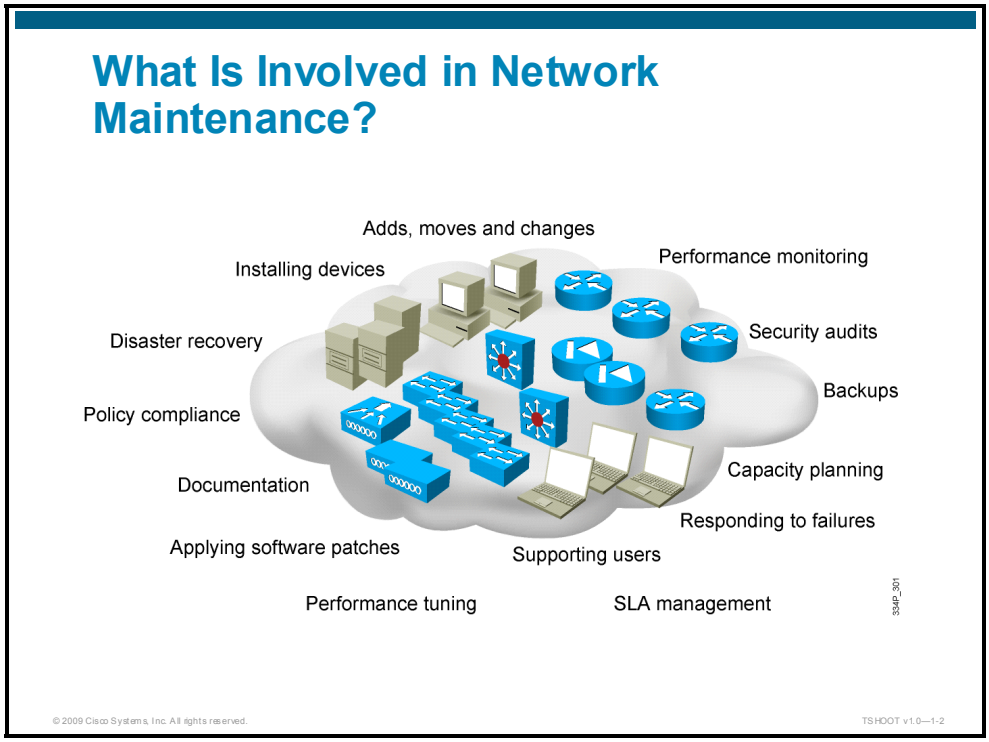
Objectives

Upon completing this lesson, you will be able to evaluate and rate commonly practiced network maintenance models and methodologies. This ability includes being able to meet these objectives:

- Evaluate the models and methodologies that are commonly used for network maintenance and identify the benefits that these models bring to an organization
- Select generalized maintenance models and planning tools that fit your organization

Maintenance Models and Methodologies

This topic describes commonly used models and methodologies that help structuring network maintenance tasks.



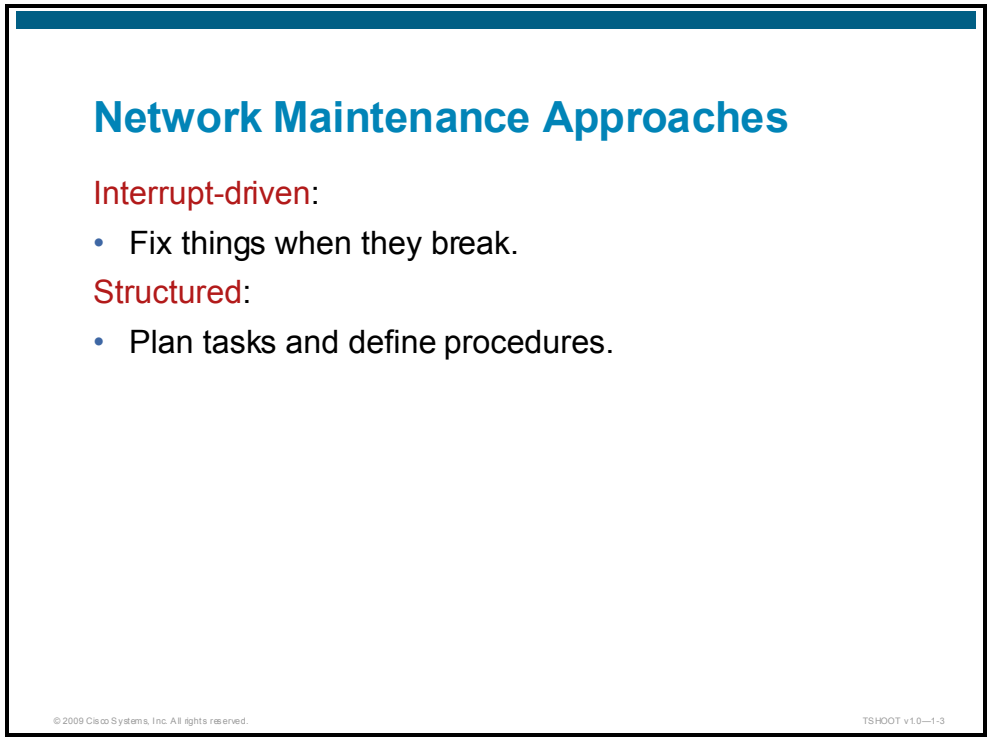
What is involved in maintaining networks?

The job description of a typical network engineer includes elements such as installing, implementing, maintaining, and supporting network equipment. However, what do these tasks exactly entail? What do those high-level task descriptions include?

Depending on the size and type of organization, the job will likely include some or all of the following:

- Tasks related to device installation and maintenance, such as installing devices and software, creating configurations, and backing up configurations and software
- Tasks related to failure response, such as supporting users who experience network problems, troubleshooting device or link failures, replacing equipment, and restoring backups
- Tasks related to network performance, such as capacity planning, performance tuning, and usage monitoring
- Tasks related to business procedures, such as documenting, compliance auditing, and service level agreement (SLA) management
- Tasks related to security, such as following and implementing security procedures and security auditing.

The exact set of tasks that is considered part of network maintenance is different for each organization. What does your organization consider network maintenance? What are the policies and procedures at your company?

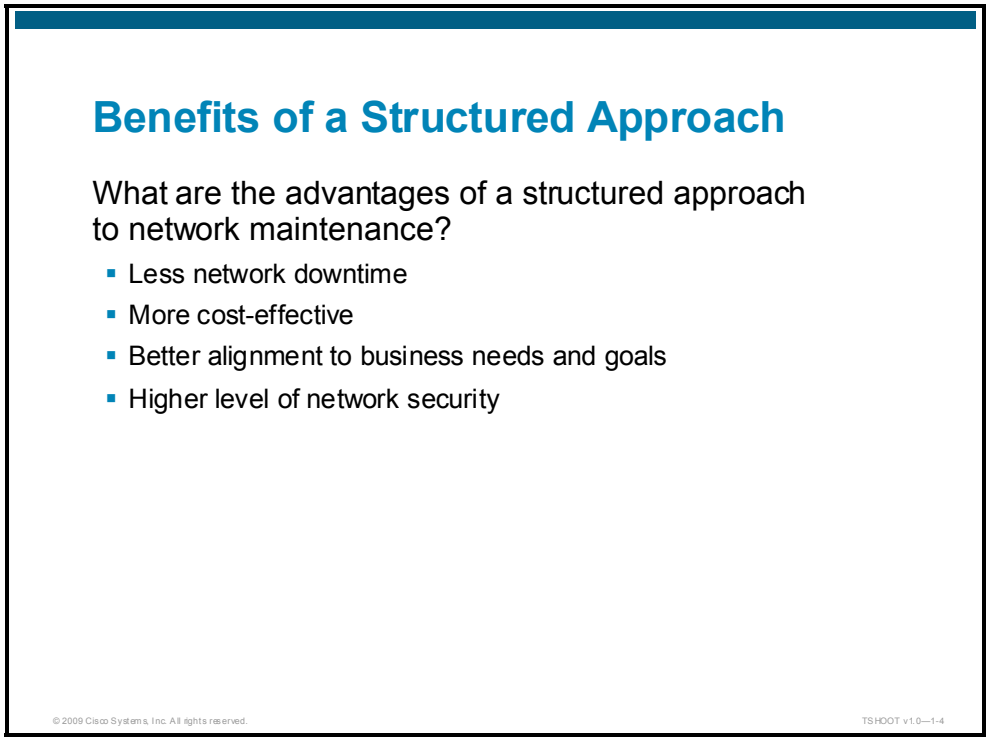


How do you execute all of these network maintenance tasks?

In many smaller networks, the process is often largely interrupt-driven. When users have problems, you start helping them; when applications start experiencing performance problems, you upgrade links or equipment; when you have a security incident, you review and improve the security of the network.

Although this method is obviously the most basic method of performing network maintenance, it clearly has some disadvantages. Tasks that are beneficial to the long-term health of the network may be ignored, postponed, or forgotten. Tasks may not be executed in order of priority or urgency, but are done based on the timing of the requests instead. The network may experience more downtime than necessary because problems are only handled, not prevented.

Obviously, you can never avoid interrupt-driven work entirely because failures and incidents do happen and you cannot plan them. However, you can reduce the amount of incident-driven work by proactively monitoring and managing your systems. You can also structure the processes and procedures that you use to respond to such incidents in ways that allow you to be more efficient.

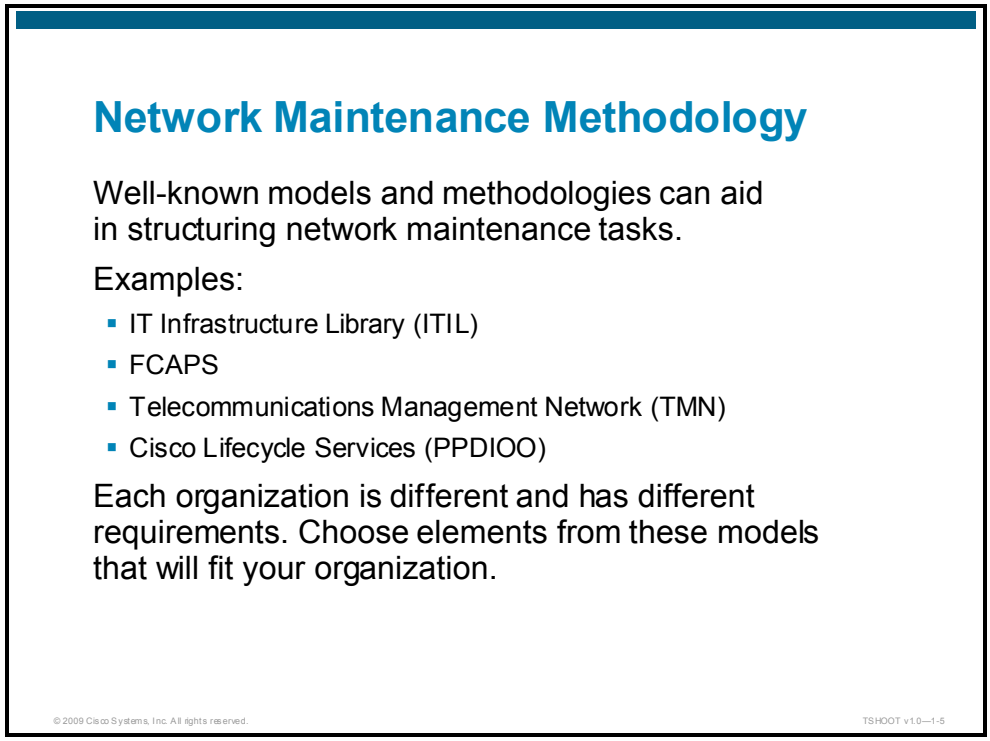


The more important your network is to the business, the higher the price of downtime. Therefore, from a business perspective, it often makes sense to follow a more structured approach. By discovering and preventing problems before they happen, you can prevent downtime. Even if you cannot prevent the problems, you can reduce the amount of time it takes to fix a problem by having the right tools available.

This result is not the only advantage to a structured approach. By incorporating processes for performance monitoring and capacity planning, you can achieve a better ratio of price to performance over the lifetime of your equipment. In addition, proactive network management can decrease the total maintenance and support costs, because you will spend less time responding to emergencies.

Another benefit of a structured approach is that it becomes much easier to align the maintenance processes with the business needs and objectives. Instead of prioritizing tasks and assigning budgets based on incidents, you can allocate time and resources to processes based on the importance of the process to the business.

Generally, security also benefits from a structured approach. It becomes less likely that incidents will go unnoticed or that vulnerabilities will not be addressed when security is treated as an integral part of a structured network maintenance plan.



Although each organization is different and has different requirements, you can define generic models that categorize maintenance tasks or define procedures to aid in structuring network maintenance tasks.

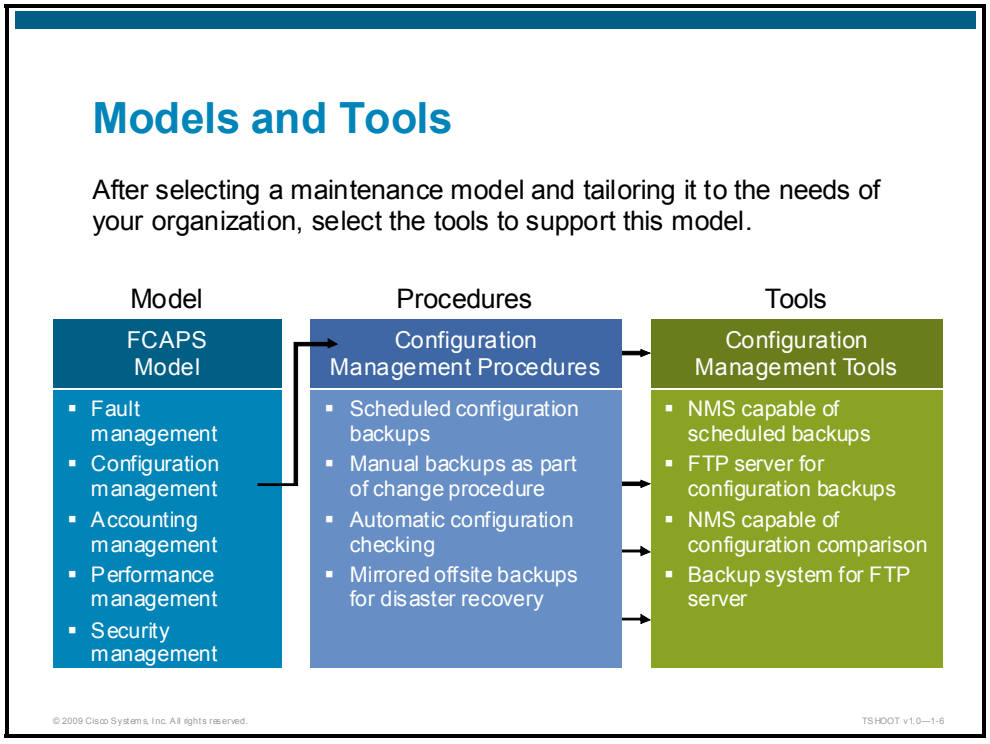
A few well-known models and methodologies that are related to network maintenance are as follows:

- **IT Infrastructure Library (ITIL):** This model is a framework of best practices for IT Service Management. ITIL describes best practices that can help IT professionals provide high-quality IT services that are aligned with business needs and processes.
- **Fault, Configuration, Accounting, Performance, and Security (FCAPS):** This model, developed by the ISO, divides network management tasks in five different categories: Fault management, Configuration management, Accounting management, Performance management, and Security management.
- **Telecommunications Management Network (TMN):** The ITU-T integrated and refined the FCAPS model to create this approach, which defines a framework for the management of telecommunications networks.
- **Cisco Lifecycle Services:** This approach is a model that helps businesses to successfully deploy, operate, and optimize Cisco technologies in their network. This model is sometimes also referenced to as the PPDIOO model, based on the names of the six phases of the network lifecycle: Prepare, Plan, Design, Implement, Operate and Optimize. Network maintenance tasks are usually considered part of the Operate and Optimize phases of the cycle.

These are just some of the most common examples of maintenance models. Many other models could be used to cover the various aspects of network maintenance and IT service management, in general.

Models, Procedures, and Supporting Tools

This topic describes the decision criteria that should be used to select a model and supporting tools that fit your organization.



When you have decided to use a structured network maintenance approach, you can select the model that best fits your organization or take elements from different models and combine them to suit your needs. As an example, assume that you have selected the FCAPs model as the foundation for your network maintenance procedures.

FCAPS is an ITU standard model that defines a structured network maintenance approach. It consists of the following five domains:

- Fault management
- Configuration management
- Accounting management
- Performance management
- Security management

Fault Management

Fault management is the domain where network problems are discovered and corrected. While some fault management is event-based, much of the effort should be focused on preventive maintenance. Steps can then be taken to prevent the problems from occurring or recurring. By taking these steps, the network remains operational and downtime is minimized.

Configuration Management

The focus of configuration management is the installation, identification, inventory, removal, and configuration of hardware (including components such a cards, modules, memory, and software), software, firmware, and services. Configuration management also includes monitoring and managing the deployment status of a device and the functional areas of software management, change control, and inventory management.

Accounting Management

Accounting management is focused on how to distribute resources optimally among enterprise subscribers. This type of management helps to minimize the cost of operations by making the most effective use of the systems that are available. This level of management also involves ensuring that users are billed correctly.

Performance Management

Performance management involves managing the overall performance of the enterprise network. The focus of this domain is identifying potential problems. The network engineer uses performance management to maximize throughput, identify bottlenecks, and determine what improvements are needed to yield the best enterprise network performance.

Security Management

The focus of security management is the protection of the network from unauthorized users and physical and electronic sabotage. It includes user authentication and authorization, and it is used to maintain the confidentiality of user information.

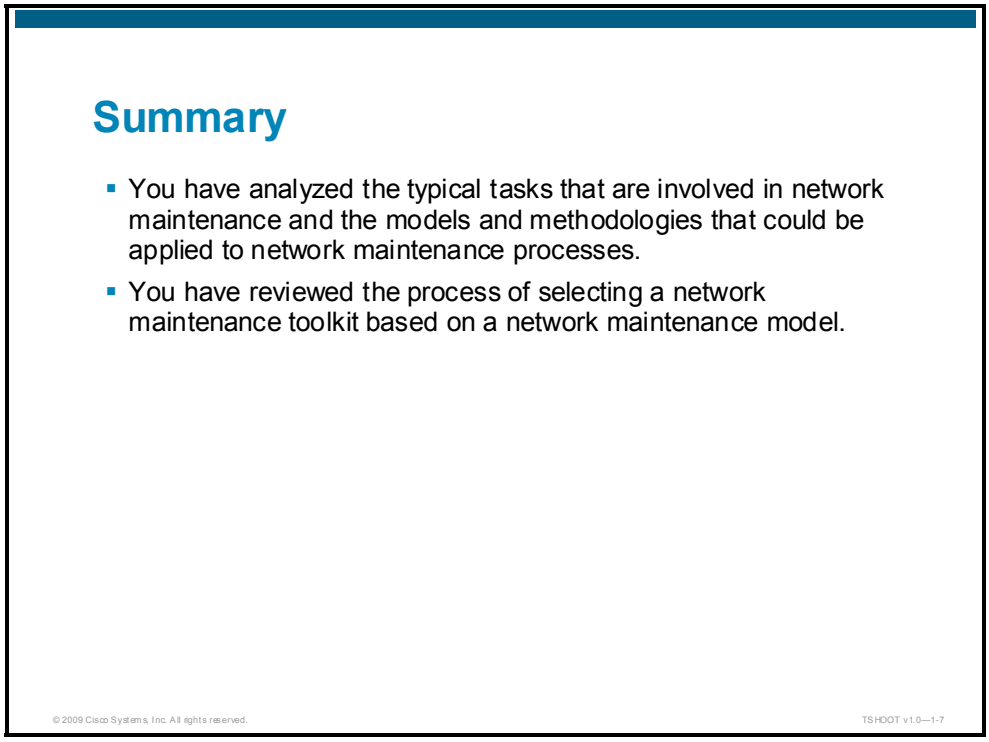
After you have selected a model, you need to translate the theoretical model to practical procedures that structure the network maintenance processes for your network. When you have defined your processes and procedures, it is much easier to see what functionality you need to have in your network management toolkit to support these processes.

By first defining the model and processes, you will be able to select an efficient and cost-effective network management and support toolkit.

Compared to the structured approach, an interrupt-driven approach usually results in a more fragmented toolkit. This toolkit becomes fragmented because tools are acquired on an on-demand basis to provide specific functions. The network engineer needs to consider the toolkit as a whole and build it to support all aspects of the network maintenance processes.

Summary

This topic summarizes the key points that were discussed in this lesson.



Lesson 2

Common Maintenance Processes and Procedures

Overview

Network maintenance involves many tasks. Those tasks will be different in each organization, depending on the needs and requirements of the business.

However, processes such as maintenance planning, change control, documentation, disaster recovery, and network monitoring are common elements of all network maintenance plans.

In this lesson, you will analyze the common elements of these processes and determine how to establish procedures that fit the needs of your organization.

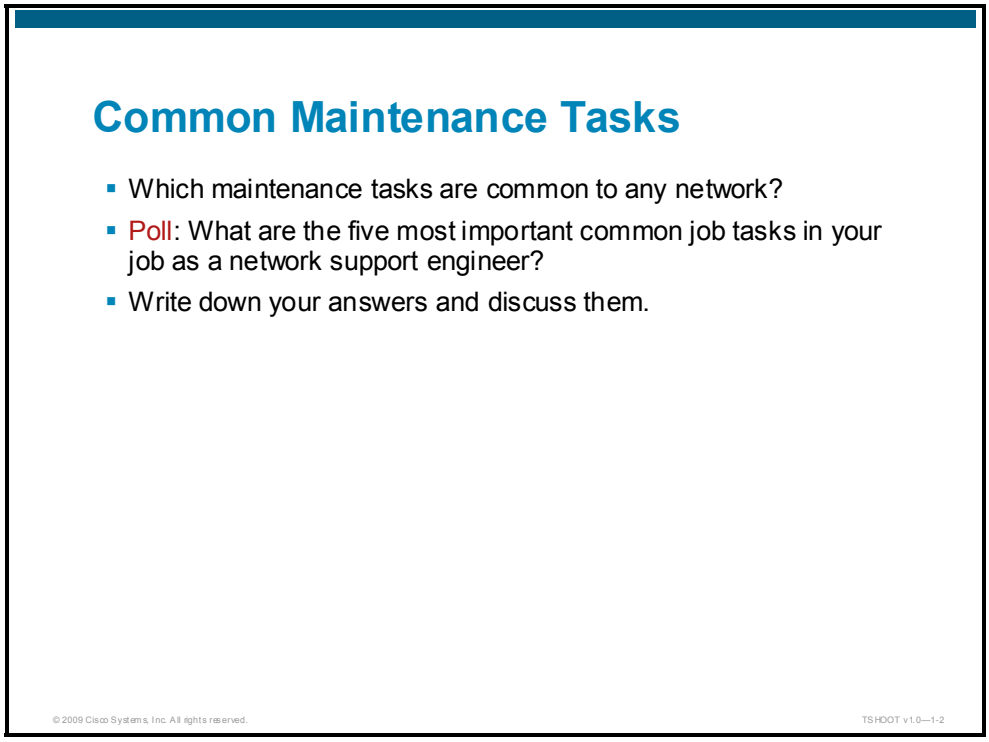
Objectives

Upon completing this lesson, you will be able to identify the processes and procedures that are a fundamental part of any network maintenance methodology. This ability includes being able to meet these objectives:

- Identify essential network maintenance tasks
- Describe the advantages of scheduled maintenance
- Evaluate the key decision factors that affect change control procedures in order to create procedures that fit the needs of your organization
- Describe the essential elements of network documentation and its function
- Plan for efficient disaster recovery
- Describe the importance of network monitoring and performance measurement as an integral element of a proactive network maintenance strategy

Essential Network Maintenance Tasks

This topic describes the most fundamental network maintenance tasks.



No matter what model you decide to use, what methodology you follow, or what size your network is, a certain set of tasks has to be included in any network maintenance plan. However, the amount of resources, time, and money that an organization spends on these tasks will vary depending on the size and type of the organization.

What are the essential tasks that any engineer involved in network maintenance has to manage on a day-to-day basis?

Write down the top five tasks or categories of tasks that you perform in your work as a network support engineer.

1. _____
2. _____
3. _____
4. _____
5. _____

Common Maintenance Tasks (Cont.)

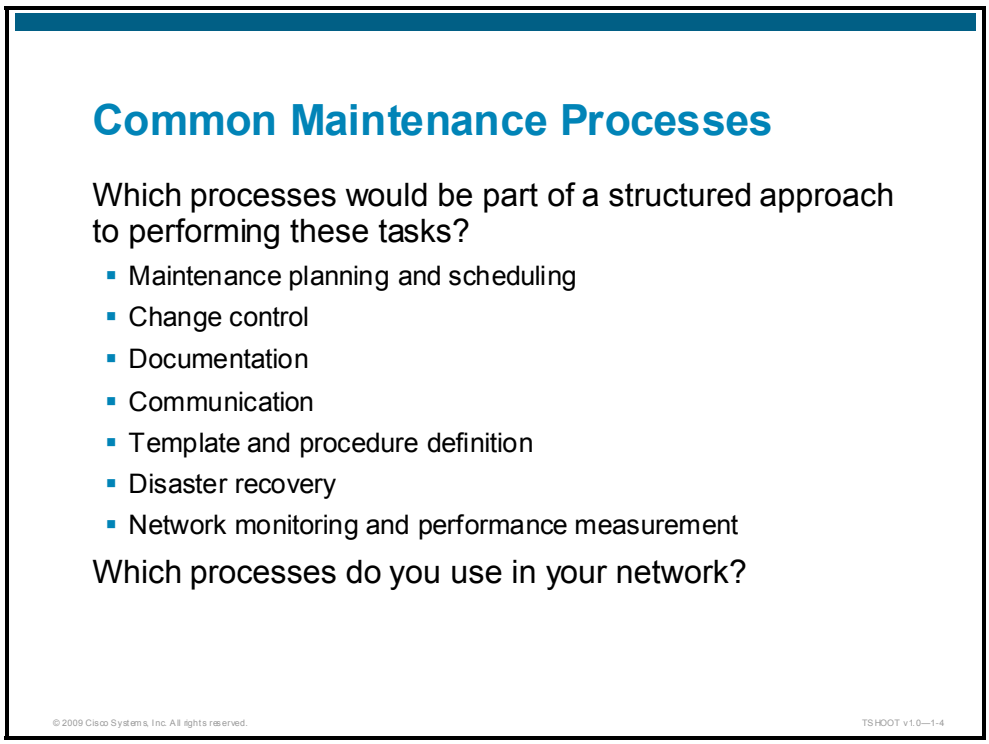
Some maintenance tasks that are common to all networks:

- Adds, moves, and changes
- Installation and configuration of new devices
- Replacement of failed equipment
- Configuration and creation of software backups
- Network failure diagnosis and resolution
- Software upgrades and patches
- Network monitoring
- Performance measurement and capacity planning
- Writing and updating documentation

Essentially, all network maintenance plans need to include procedures that are used to manage the following tasks:

- **Minor configuration and cabling changes:** Networks are always undergoing changes. As people move and offices are changed and restructured, network devices such as PCs, printers, and servers may need to be moved. These tasks are often referred to as “adds,” “moves,” and “changes,” and are a normal part of network maintenance.
- **Installation and configuration of new devices:** Even if the implementation of new technologies is handled by a different group within your organization or by an external party, adding ports, link capacity, or otherwise upgrading the existing network is usually considered a part of network maintenance.
- **Replacement of failed devices:** Whether you manage this task via service contracts or by having spare equipment on a shelf, as part of your maintenance plan you have to determine how you will handle equipment replacement when devices fail.
- **Backup of device configurations and software:** In a way, this item is linked to the previous item of replacing failed devices. Without good backups of both software and configurations, the time that it takes to replace failed equipment or recover from other severe device failures will be much longer.
- **Troubleshooting link and device failures:** Inevitably, network components, links, or service provider connections experience failures. Diagnosing and resolving these failures are an essential part of the job of a network engineer.
- **Software upgrading or patching:** At a minimum, you need to stay informed of critical security vulnerabilities and patch the devices of your organization when the devices are at risk..

- **Network monitoring:** Monitoring operation of the devices and user activity on the network is generally part of a network maintenance plan. This task can require the use of very simple mechanisms, such as the collection of router and firewall logs, or the use more complex methods, such as the use of specialized network monitoring applications.
- **Performance measurement and capacity planning:** Since the demand for bandwidth is continually increasing, you have to perform at least some basic measurements to decide when it is time to upgrade links or equipment and to justify the cost of the corresponding investments.
- **Writing and updating documentation:** It is important to maintain documentation that describes the current state of the network for reference during implementation, administration, and troubleshooting tasks.

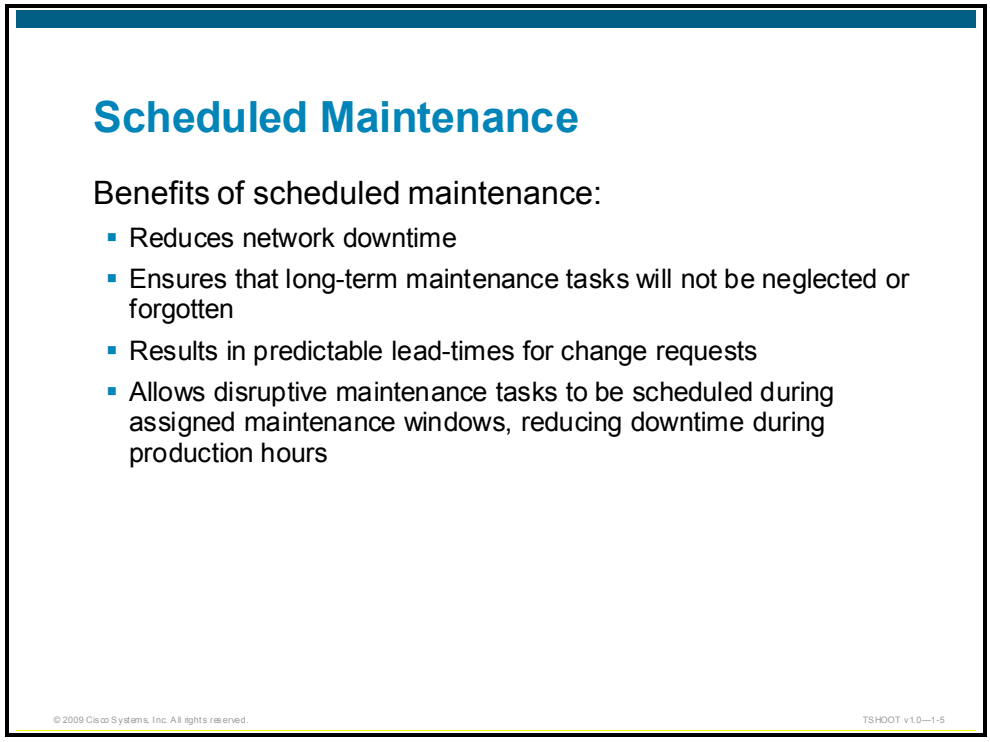


When you look at this list of tasks, you see that a structured approach would include organizing these tasks through the use of processes and procedures. Which common processes would help in structuring these tasks?

Obviously, it is difficult to create a complete list of processes and to rank the importance of each of these processes in this course because the nature and size of the business determine what is needed. The processes that are needed will vary for each person taking this course For instance, communication is more important if you are working in a network operation center that uses shifts to provide 24/7 support than it is if you are the only network engineer in your company. Network monitoring and performance measurement become much more important when you are using service level agreements (SLAs) that require you to maintain a certain level of service by contract. Most of the processes in the preceding list apply to most networks. In this lesson, you will examine some of these processes and learn more about what they entail..

Maintenance Planning

This topic describes the benefits of scheduled maintenance.



When you have determined the tasks and processes that are part of network maintenance, you can assign priorities to each of the tasks. You can also determine which of these tasks are interrupt-driven by nature (hardware failures, outages, and so on), and which tasks are part of a long-term maintenance cycle (software patching, backups, and so on). For the long-term tasks, you have to work out a schedule that guarantees that these tasks will be done regularly and will not get lost in the busy day-to-day work schedule.

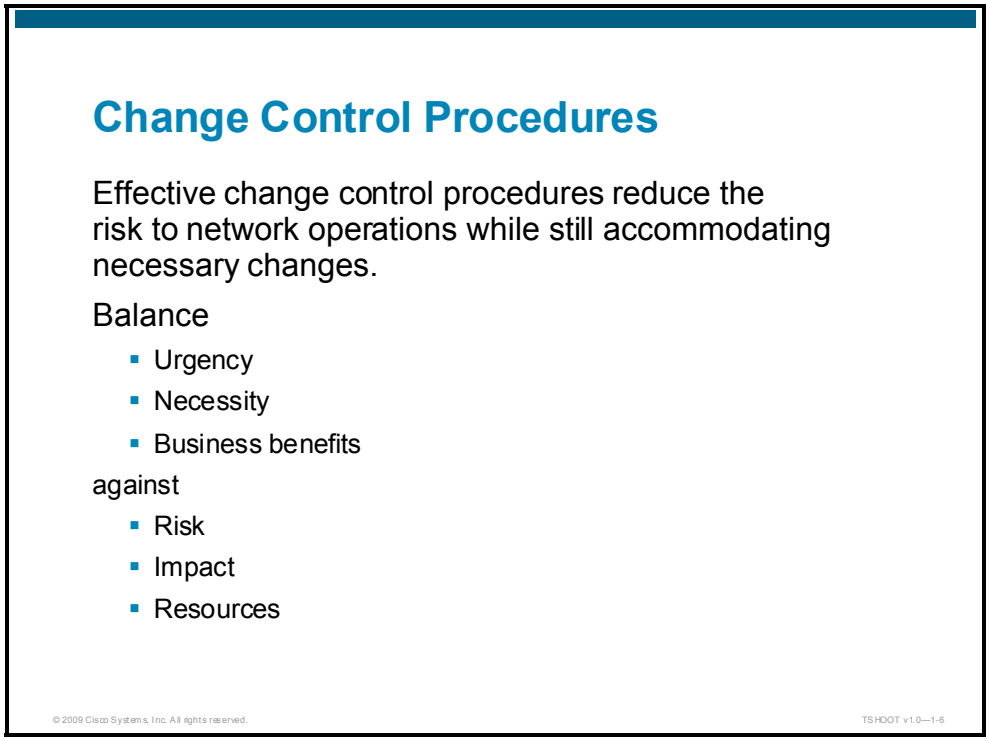
For some tasks, like adds, moves and changes, you can adopt a procedure that is partly interrupt-based (incoming change requests) and partly scheduled: Change requests need not be handled immediately, but can be handled during the next scheduled time frame. This practice allows you to prioritize tasks properly, while still providing a predictable lead-time during which the requesting party can expect the problem to be fixed.

Another advantage of a scheduled maintenance cycle is that you can schedule the tasks that are disruptive to the network during off hours. You can select maintenance windows during evenings or weekends when outages will be acceptable, thereby reducing unnecessary outages during office hours.

The result of this type of scheduling is that the overall uptime of the network will be increased, because both the number of unplanned outages, as well as their duration will be reduced.

Change Control

This topic describes the elements of effective change control procedures.



Many network maintenance tasks involve making changes to such things as configuration, software, and cabling. Any change that you make has an associated risk of mistakes or failures. Keep in mind the popular saying: “If it ain’t broke, don’t fix it.”

For any changes, always consider the possible impact of the change on the network first and balance that against the urgency of the change. If the possible impact is very high, you may need to justify the need for the change and obtain authorization to go forward. High-impact changes generally need to be done during maintenance windows that are specifically scheduled for that purpose.

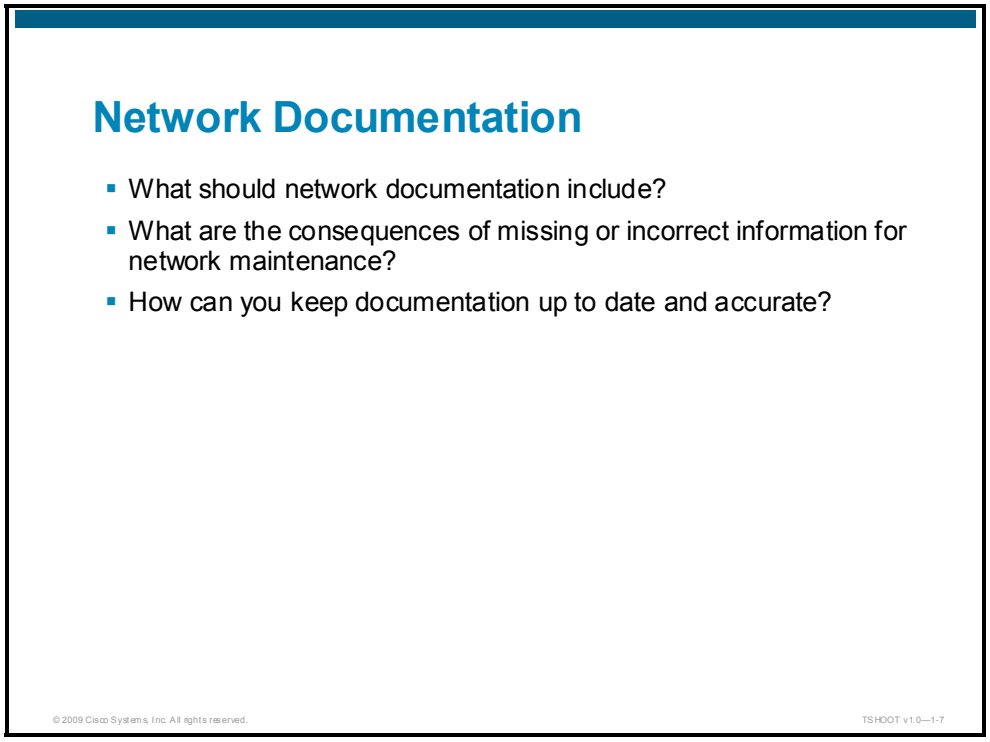
On the other hand, you also need a process for emergency changes. For instance, if you have a broadcast storm on your LAN and you need to disconnect some links to break the loop and allow the network to stabilize, you may not be able to wait for authorization and the next maintenance window before you start taking action.

In many companies, change control is explicitly described by procedures that define these parameters. Typically, the procedures would include answers to the following questions:

- Which types of changes require authorization and who is responsible for authorizing them?
- Which changes have to be done during a maintenance window and which changes can be done immediately?
- What kind of preparation needs to be done before executing a change?
- What kind of verification needs to be done to confirm that the change was effective?
- What other actions, such as updating documentation and completing administrative tasks, need to be taken after a successful change?
- What actions do you take when the change has unexpected results or causes problems?
- What conditions allow you to override the normal change procedures? Which elements of the procedures should still be followed?

Documentation

This topic describes the necessity and benefits of up-to-date documentation.



An essential part of any network maintenance job is creating and updating documentation about the network. Without up-to-date network documentation, it is difficult to correctly plan and implement changes, and as a result, troubleshooting is more difficult and time-consuming. Usually documentation is created during the design and implementation of the network, but keeping it up-to-date is part of network maintenance. Therefore, any good change control procedure will include updating the relevant documentation after a change has been completed.

Documentation could be as simple as having a couple of network drawings, equipment and software lists, and information on the current configurations of all devices. Alternatively, it could include an extensive set of documents that describe all implemented features, design choices that were made, service contract numbers, change procedures, and so on.

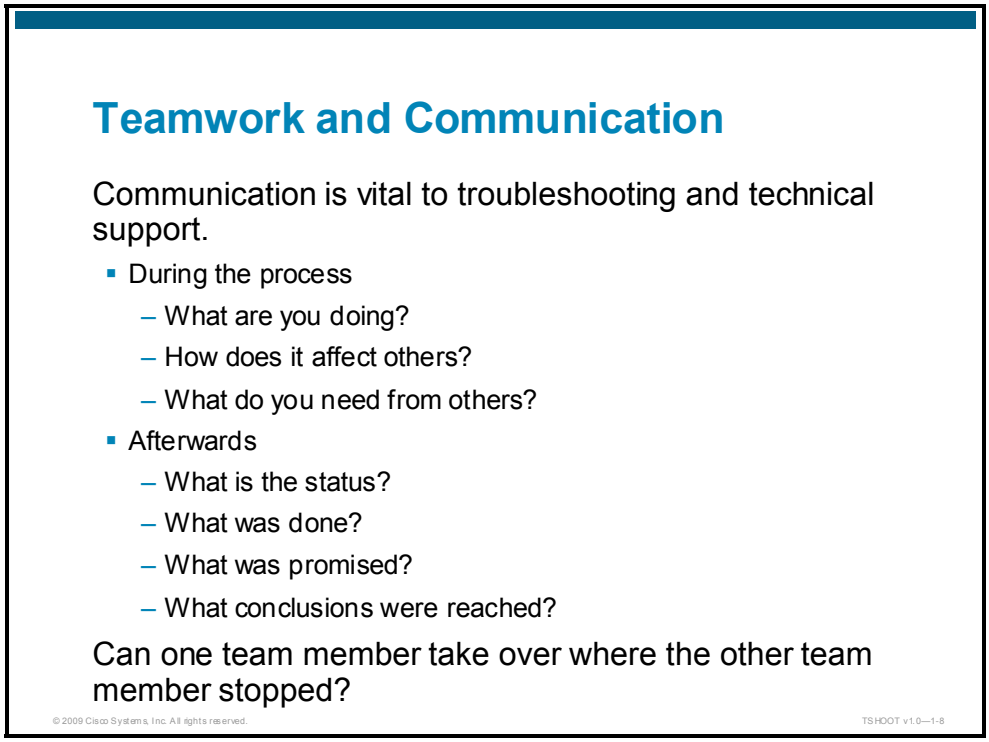
Typical elements of network documentation include the following:

- **Network drawings:** Diagrams of the physical and logical structure of the network
- **Connection documentation:** A document, spreadsheet, or database listing all relevant physical connections, such as patches, connections to service providers, and power circuits
- **Equipment lists:** A document, spreadsheet, or database listing all devices, part numbers, serial numbers, installed software versions, and (if applicable) licenses for the software

- **IP address administration:** A document, spreadsheet, or database that lists the subnetting scheme and all IP addresses that are in use
- **Configurations:** A set of all current device configurations or even an archive that contains all previous configurations
- **Design documentation:** A document describing the motivation behind certain implementation choices

Communication

Communication is an essential part of the network maintenance process, and it should be considered a part of the documentation.



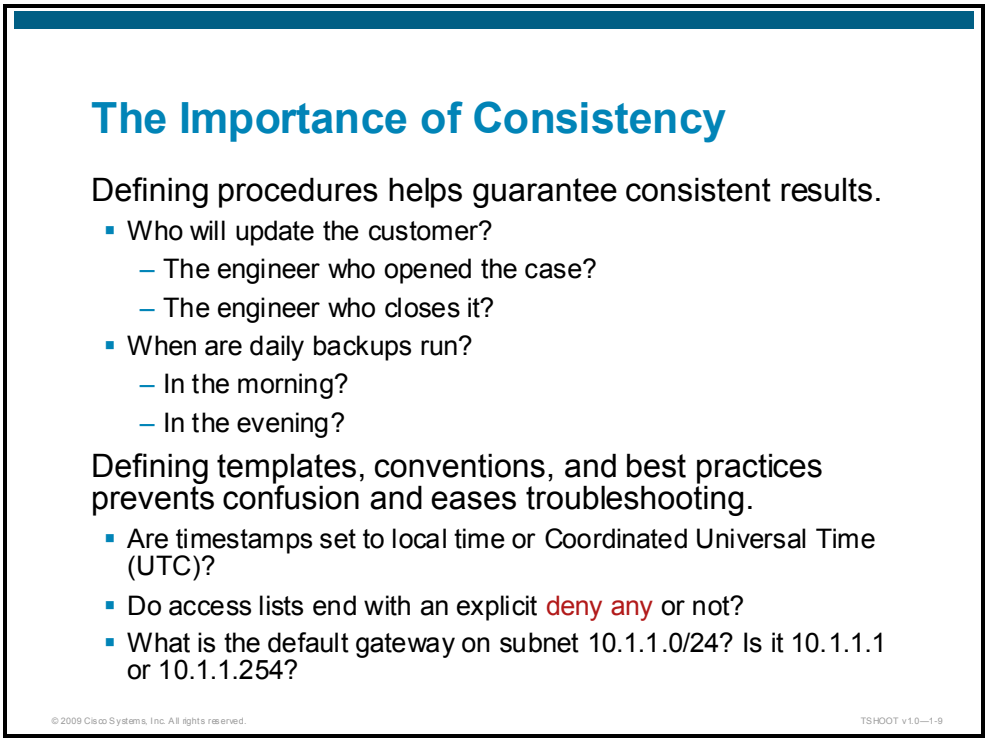
Network maintenance is typically a job that is performed by a team of people and cannot easily be divided into sets of tasks that do not affect each other. Even if you have specialists who are responsible for a particular technology or set of devices, they will always have to communicate with team members who are responsible for different technologies or other devices.

The best means of communication depends on the situation and organization, but a major consideration for choosing a communication technology is how easily it is logged and shared with the network maintenance team.

It is vital to have a well-structured communication process, especially during troubleshooting procedures. Who makes changes and when? What are the results of tests that were done? What conclusions can be drawn? If actions, test results, and conclusions are not communicated among team members, the process of one of the team members may be disruptive to the process of another member. Instead of solving problems, new problems may be introduced. In addition, there may be cases where the diagnosis and resolution must be done by multiple people or during multiple sessions. In those cases, it is important to have a log of the actions, tests, communication, and conclusions that can be passed on to another engineer.

Template and Procedure Definition

In a larger organization, standardization can improve the network maintenance and troubleshooting processes.



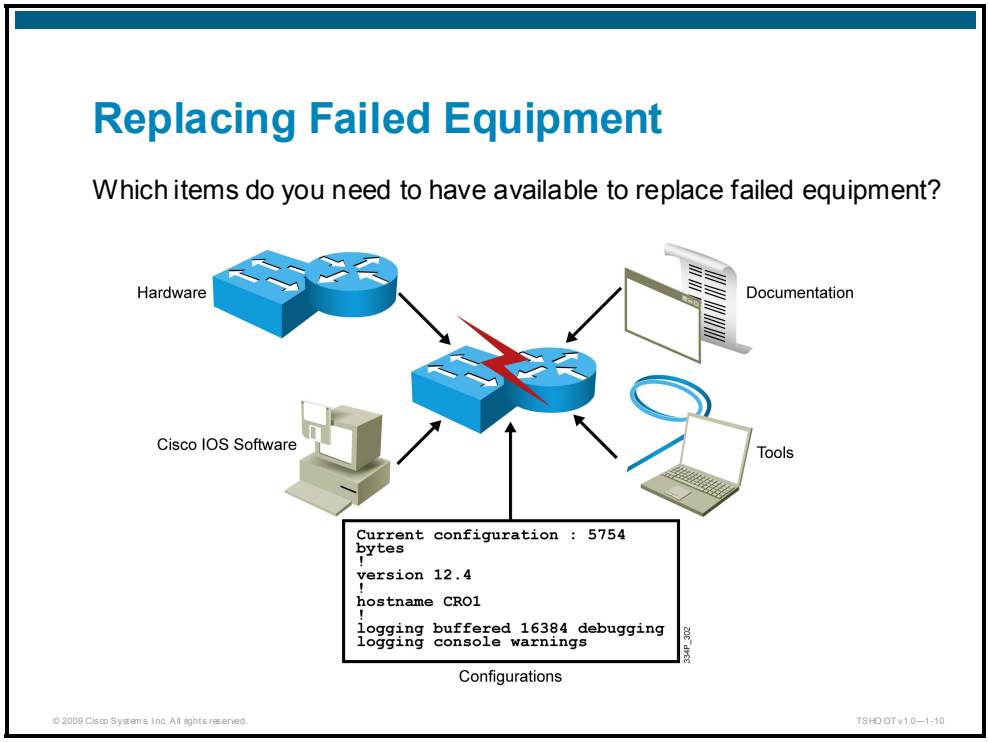
When a team of people execute the same tasks, it is important that the tasks be done in a consistent way. Although different people may have different working methods, styles, and backgrounds, it is important for the results to be consistent. Even if two approaches to the same task are both valid, the effect of two team members using different approaches may lead to inconsistent results.

One of the ways to streamline the processes and help ensure that tasks are executed in a consistent manner is to define and document procedures. This way, both the person performing the task and those receiving the results of the task will know what to expect.

Another aspect of network documentation and processes is defining and using templates. Templates can help create a consistent network maintenance process. In many cases, you can configure a device in several different ways to achieve the same results. However, using different methods to achieve the same results in the same network can easily lead to confusion, especially during troubleshooting. Under pressure, valuable time can be wasted in verifying configurations simply because they are configured differently.

Disaster Recovery

This topic describes the essential procedures that will help recover from device failures in a quick and efficient way.



Although the mean time between failures (MTBF) for network devices is usually 10 years or more, you always have to consider the possibility that a device will fail. If you have a plan for such occasions and know what to do when a failure occurs, you will be able to significantly reduce the amount of downtime that you incur when the failure happens.

Of course, the first step that should be taken to reduce the impact of outages is to build redundancy into the network at critical points and ensure that a single device or link failure can never cause your whole network to go down. Still, it is hard to make every single connection and device redundant.

In addition, you need to prepare for the possibility that you could be struck by a disaster like flooding, a fire in the server room, or other circumstances where you will have to replace failed equipment, and the quicker you can replace the equipment and restore functionality, the quicker your network will be running again.

To replace a failed device, you will need the following items:

- Replacement hardware
- The current software version for the device
- The current configuration for the device
- The tools to transfer the software and configuration to the device
- Licenses (if applicable)
- Knowledge of the procedures to install software, configurations, and licenses

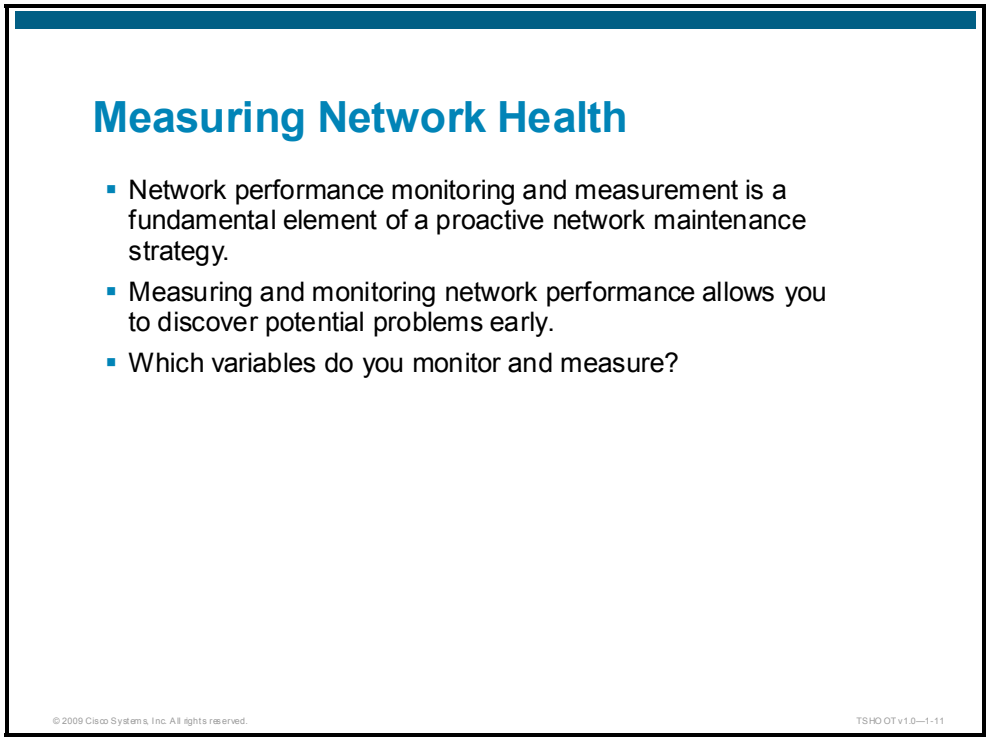
Missing any of these items will severely affect the time it takes to replace the device. To ensure that you have these items available when you need them, follow these guidelines:

- **Replacement hardware:** You need to have either spare devices or a service contract with a distributor or vendor that will replace the failed hardware. Typically, you need documentation of the exact hardware part numbers, serial numbers, and service contract numbers for the devices.
- **Current software:** Usually devices are delivered with a particular version of software, which is not necessarily the same version as the one that you are running on the device at the time of a failure. Therefore, you should have a repository where you store all current software versions that you are using in your network.
- **Current configuration:** In addition to creating backups of your configurations any time you make a change, you need to have a clear versioning system so that you know which configuration is the most recent.
- **Tools:** You need to have the appropriate tools to transfer software and configurations to the new device, which you should be able to do even if the network is unavailable.
- **Licenses:** If your software requires a license, you need to have that license or know the procedure to obtain a new license.
- **Knowledge:** Because these procedures are used infrequently, you may not have them committed to memory. Having all necessary documentation ready, however, saves time in executing the necessary procedures and decreases the risk of making mistakes.

Therefore, we see that the most important factors in a successful disaster recovery are defining and documenting recovery procedures and making sure that your regular maintenance cycle incorporates actions that ensure that you always have the necessary elements available when disaster strikes.

Network Monitoring and Performance Measurement

This topic describes the network monitoring and performance measurement process.



Another process that helps you transform your network maintenance process into a less interrupt-driven, more methodical approach is the implementation of network and performance monitoring.

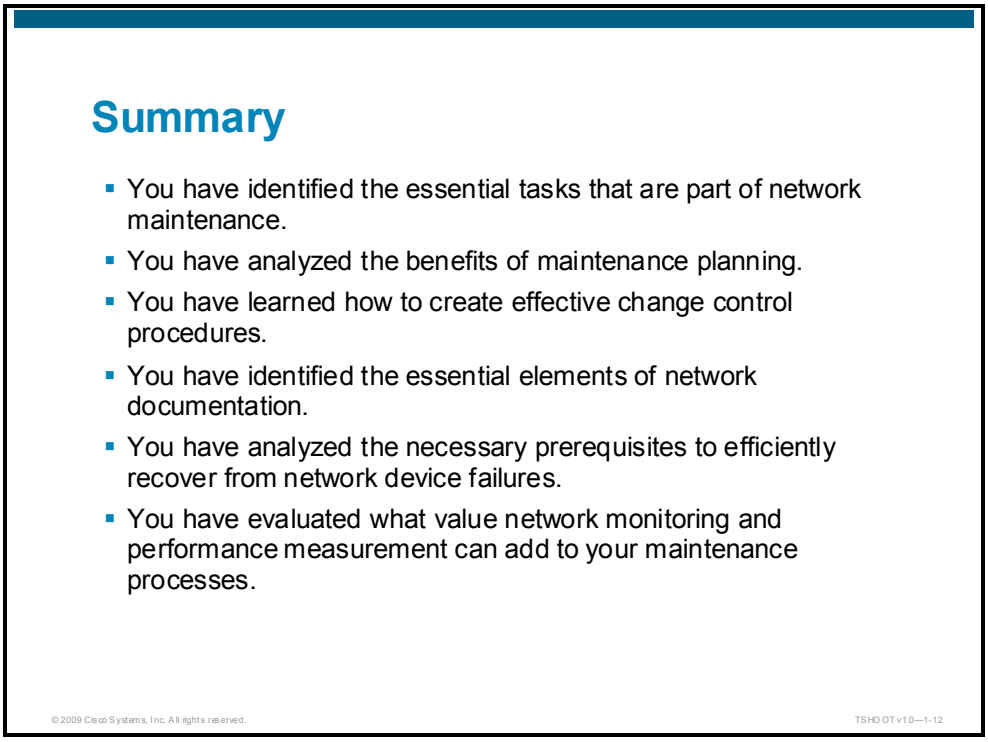
Knowing the overall health of your network helps you catch potential issues before they develop into problems and helps you isolate problems faster. Gathering performance data allows you to upgrade before a lack of resources develops into a performance problem. In addition, the data you gather helps build a business case to support the need for investments in network upgrades.

When you are committed to meeting the service level agreements (SLAs) for the performance of your network or if your service provider is guaranteeing you a certain level of service via an SLA, monitoring network performance will also help you make sure that those SLAs are met.

One of the essential steps in network performance measurement and monitoring is choosing the variables that need to be monitored and measured. These variables can include interface status, interface load, CPU load, and memory usage of your devices. The more sophisticated metrics, such as measurements of network delay, jitter, or packet loss, can be included in a network monitoring and performance measurement policy. This policy and the corresponding choice of metrics will be different for each organization and needs to be aligned to the business requirements. Which variables do you monitor and measure in your network?

Summary

This topic summarizes the key points that were discussed in this lesson.



Lesson 3

Network Maintenance Tools, Applications, and Resources

Overview

When you have defined the processes and procedures you want to implement in your organization, you need to select the tools, applications, and resources that best support these processes and procedures.

The tools that you select should enable you to execute your network maintenance tasks in an efficient manner. In addition, the tools should be cost- and resource-effective in the sense that they support all the tasks that are part of your maintenance plan, but do not have a huge amount of unneeded extra functionality.

In this lesson, you will analyze what should be contained in a minimal network maintenance toolkit and which additional components could be considered if your organization has specific network maintenance needs.

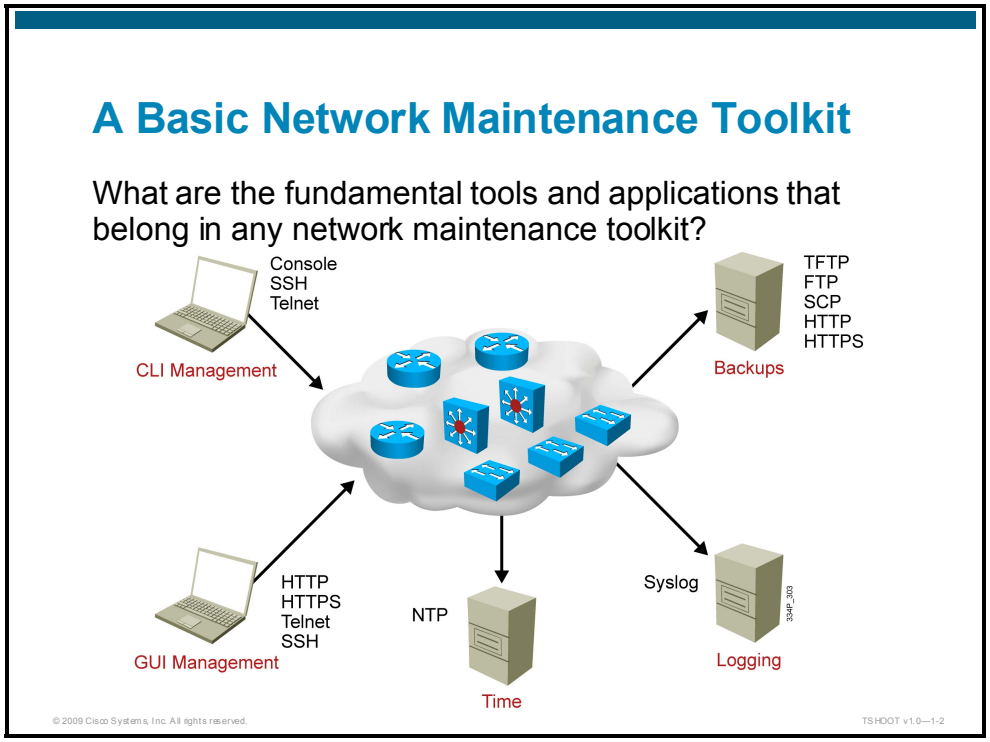
Objectives

Upon completing this lesson, you will be able to identify, evaluate, and select the tools, applications, and resources that you need to support network maintenance processes. This ability includes being able to meet these objectives:

- Identify, evaluate, and implement the elements of a basic network maintenance toolkit
- Evaluate tools that support the documentation process and select the tools that are appropriate to your organization
- Describe how configuration, software, and hardware resource management software can improve disaster recovery procedures
- Describe how network monitoring software benefits the maintenance process
- Analyze the metrics that could be used to measure network performance and the primary elements of the performance measurement process in order to create a performance measurement plan that is appropriate to your organization

Fundamental Tools

This topic describes the fundamental tools, applications, and resources that you can use in any network using Cisco routers and switches.



There are many tools, applications, and resources that you can use to support network maintenance processes. These tools can range from simple, freely available tools to complex and powerful applications that can manage thousands of devices.

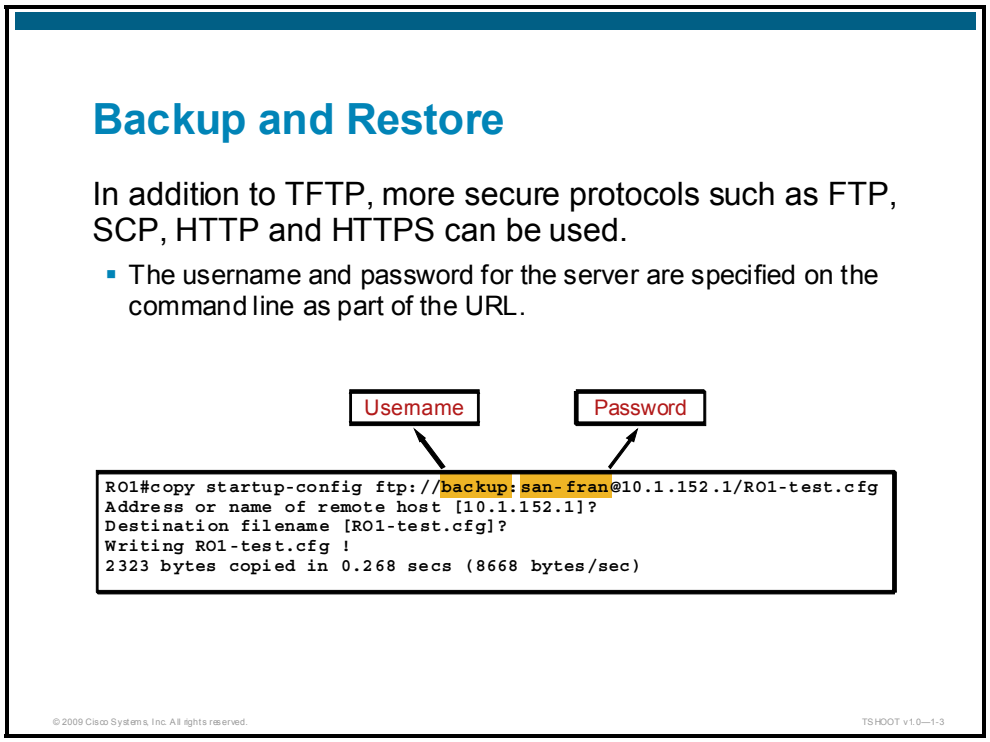
What does a basic network maintenance toolkit consist of?

- **Command-line device management:** The Cisco IOS Software includes a powerful command-line interface (CLI) that you can use to configure and monitor individual routers and switches. This software includes commands such as the show commands, the debug commands, Cisco IOS Embedded Event Manager (EEM) commands, and IP service level agreement (SLA) commands. After an initial configuration through the serial console of the device, the command line is commonly accessed remotely through use of the Telnet or Secure Shell (SSH) protocols. To be able to manage the devices during network outages, an out-of-band (OOB) management solution can be implemented to allow access to the CLI via the serial console at all times.
- **Graphical user interface (GUI)-based device management**: Cisco provides free GUI-based device management tools for many Cisco routers and switches: Cisco Configuration Professional, Cisco Router and Security Device Manager (Cisco SDM), Cisco Configuration Assistant, and Cisco Network Assistant.
- **Backup server:** To create backups of the software and configurations of your routers and switches you will need to provide a TFTP, FTP, HTTP, or Secure Copy Protocol (SCP) server. Many operating systems include these services as an optional add-on, while there are also many other widely available software packages that can provide this functionality.

- **Log server:** Basic logging functionality can be provided by sending the log messages of the router or switch to a syslog server using the syslog protocol. Syslog is a standard service on most UNIX-based operating systems, or it could be provided by installing additional software on the operating system of your choice.
- **Time server:** To synchronize clocks on all of your network devices it is useful to have a Network Time Protocol (NTP) server on your network. In addition, many public timeservers are available on the Internet to which you could synchronize the clocks on your routers and switches.

Implementing Backup and Restore Services

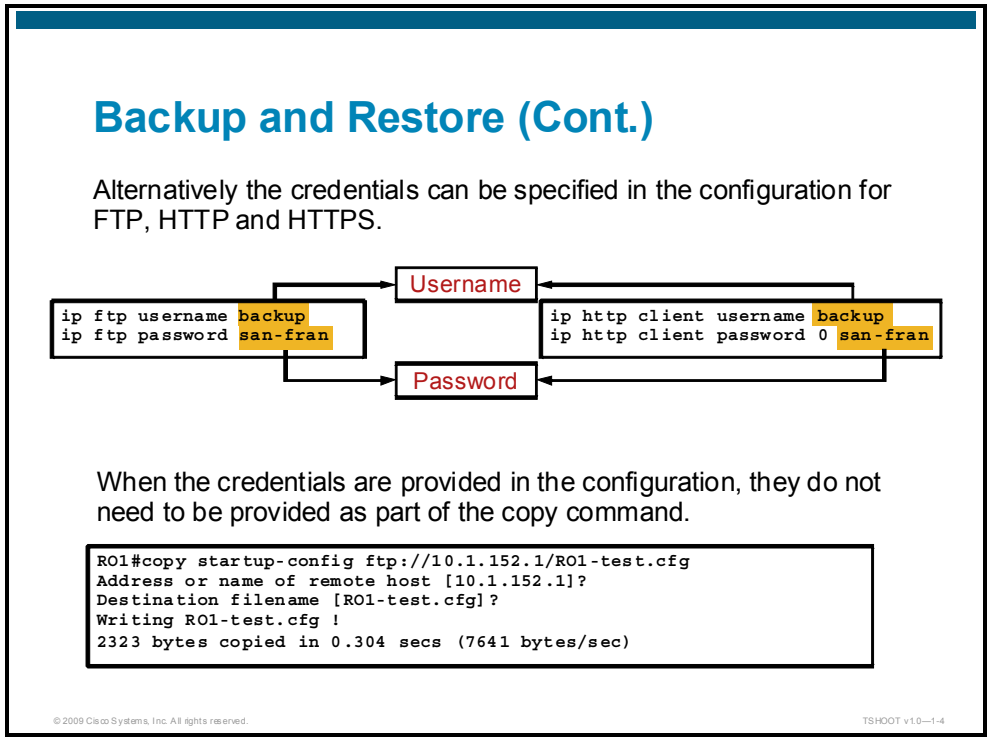
An essential element of any network maintenance toolkit is a backup server that device configurations and Cisco IOS Software can be copied to and restored from. The simplest and most commonly implemented service is TFTP, which does not require any configuration on the network devices. The server is set up to serve and receive files without any need for authentication or identification, other than specifying the name of the configuration or software file itself. The fact that the protocol does not require any authentication and that all content is sent across the network in clear text, makes it a relatively insecure mechanism. More secure protocols such as FTP, SCP and HTTP or HTTPS can also be used as a means of transferring configurations and software.



To use any of these more secure protocols you have to specify a username and password that are used to authenticate to the server. For all of these protocols, the credentials can be specified as part of the URL that is used as part of the **copy** command. The username and password are specified by placing the username and password as ***username*:*password*@** before the server name or IP address in the URL.

For example, to copy the startup configuration through FTP to a server with the IP address 10.1.152.1 and a file named *RO1-test.cfg* using the user name "backup" and password "san-fran", you would issue the command **copy startup-config ftp://backup:san-fran@10.1.152.1/RO1-test.cfg**

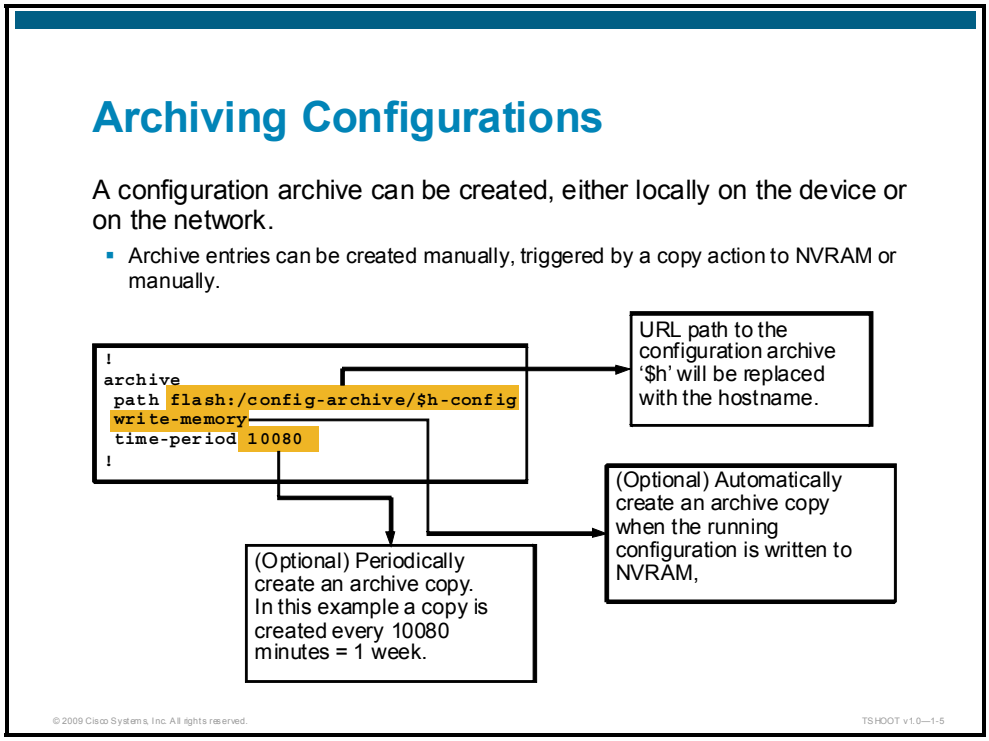
For SCP, HTTP, and HTTPS you would use a similar syntax, replacing the URL prefix **ftp://** with **scp://**, **http://** or **https://** respectively.



Specifying the username and password on the command line is somewhat cumbersome and suffers from the fact that the password is displayed in clear text on the screen, which is not desirable from a security standpoint. To circumvent this issue, the username and password can be specified in the configuration, instead of on the command line for the FTP, HTTP, and HTTPS protocols.

The same configuration commands are used for both HTTP and HTTPS. The only difference is the protocol identifier in the URL.

Note	Although FTP and HTTP require authentication, these protocols send credentials in clear text. HTTPS and SCP use encryption to ensure confidentiality of both the transmitted credentials and the content of the transferred file.



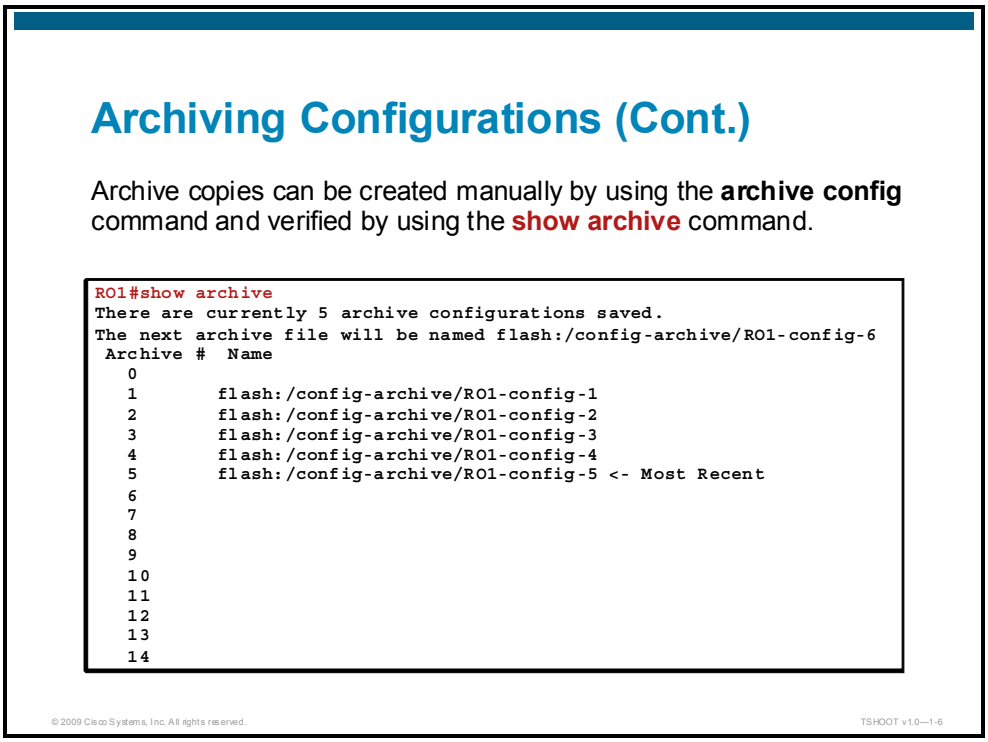
Creating backups of configurations should be an integral part of your network maintenance routines. After any change, you should create backups, copying the configuration file to NVRAM on the device as well as to a network server. If you have sufficient flash storage space on the device, it can be very useful to not only build a configuration archive on the server, but in the flash memory of the device as well.

A feature that can be helpful in the creation of configuration archives, either locally on the device or remotely on a server, is the configuration archiving feature that is part of the "Configuration Replace and Configuration Rollback" feature that was introduced in Cisco IOS Release 12.3(7)T.

The configuration archive is set up by entering the **archive** command in global configuration mode, which puts you into the **config-archive** configuration mode. In this configuration sub mode, you can specify the parameters for the archive. The only mandatory parameter is the base file path. This path will be used as the base filename and is appended with a number for each subsequent archived configuration. The path is specified in URL notation and can either be a local or a networked path supported by the Cisco IOS Software file system. Not all types of local flash storage are supported, so check the flash type of your device for support of this feature if you want to store your configuration archive locally on a device instead of on a server.

The configuration path can include the variables **$h** for the hostname of the device and **$t** to include a time and date stamp in the filename.

After you specify the location of the archive, it is ready to be used, and archive copies of the configuration can be created manually by issuing the **archive config** command. However, the biggest advantage of this feature is the way you can use it to create and update a configuration archive automatically. By adding the **write-memory** option to the **archive** configuration section, you can trigger an archive copy of the running configuration to be created any time the running configuration is copied to NVRAM. It is also possible for you to generate archive copies of the configuration periodically by specifying the **time-period** option followed by a time period, specified in minutes. Each time the configured time period elapses, a copy of the running configuration will be archived.



You can verify the presence of the archived configuration files by issuing the **show archive** command. In addition to showing you the files themselves, the output will also show you the most recent archived file and the filename for the next archive to be created.

Restoring Configurations

Configurations can be restored by using the **configure replace** command.

```
RO1#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
RO1(config)#hostname TEST
TEST(config)#^Z
TEST#configure replace flash:config-archive/RO1-config-5 list
This will apply all necessary additions and deletions
to replace the current running configuration with the
contents of the specified configuration file, which is
assumed to be a complete configuration, not a partial
configuration. Enter Y if you are sure you want to proceed. ? [no]: yes
!Pass 1

!List of Commands:
no hostname TEST
hostname RO1
end


Total number of passes: 1
Rollback Done

RO1#
```


When you create backups, either by manually copying the files or through use of configuration archiving, you have something to fall back to when disaster strikes. If the configuration of a device is lost through human error, hardware failure, or when a device needs to be replaced, you can copy the last archived configuration to the NVRAM of the device and boot it to restore it to the exact same configuration that you had stored in your archive.

Another common event that might necessitate restoring the device to its last archived configuration is when you have made a change or a series of changes and they did not work out as expected. If these changes were made during a regularly scheduled maintenance window, you can often perform the same procedure as when you have lost a configuration entirely. You copy the last archived configuration that you know was good to the NVRAM of the device and reload it. However, if you made these changes during normal network operation, for instance while troubleshooting a problem, then reloading the device could be a disruptive operation and not acceptable unless there is no other option.

This situation is what the configuration replace feature was designed to manage. The **configure replace** command allows you to replace the currently running configuration on the router with a saved configuration. It does so by comparing the running configuration with the configuration file appointed by the **configure replace** command and then creating a list of differences between the files. Based on these differences the command then generates a set of Cisco IOS configuration commands that will change the existing running configuration to the replacement configuration. The advantage of this method is that only parts of the configuration that are different will be changed. The device does not need to be reloaded and existing commands are not reapplied. This manner of rolling back to an existing archived configuration is the least disruptive method that could be used.

Note	In the documentation on http://www.cisco.com, the use of the **configure replace** command is sometimes referred to as “configuration rollback,” although the command itself does not include “rollback” as a keyword.

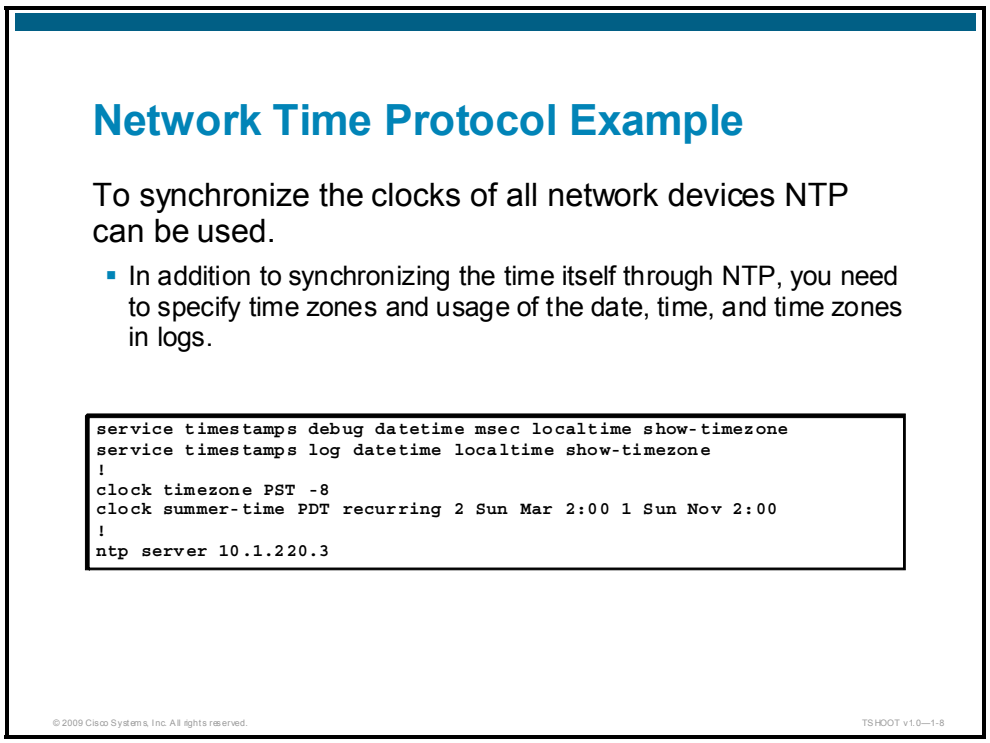
In the example in the figure, the hostname of a device is changed and then the configuration is rolled back to the most current archived configuration. The command option **list** is added to the **configure replace** command, in order to show the configuration commands that are being applied by the configuration replacement. As you can see from the example, the change that was made is undone, without affecting any other parts of the configuration.

Although this command was designed to complement the configuration archiving feature, the **configure replace** command can be used with any complete Cisco IOS configuration file.

For more detailed information about this feature consult:
http://www.cisco.com/en/US/docs/ios/fundamentals/configuration/guide/cf_config-rollback_ps6350_TSD_Products_Configuration_Guide_Chapter.html

Implementing Time Services

To ensure correct timestamps on logging and to support other time-based features such as the use of certificates or time-based access, it is vital for you to properly set and synchronize the clocks of the network devices.



The Network Time Protocol (NTP) can be used to synchronize the clock of a device to the clock of an NTP server, which in turn is synchronized to another server higher up the NTP hierarchy. The position of a device in the NTP hierarchy is determined by its stratum, which serves as an NTP hop count. A stratum 1 server is a server that is directly connected to an authorative time source such as a radio or atomic clock. A server that synchronizes its clock to a stratum 1 server will become a stratum 2 time source, and so on.

As a primary clock source, you can either set up multiple stratum 1 servers on your own network that get their clocks from an authorative source like a GPS clock. As an alternative, you can use publicly available stratum 1 and stratum 2 servers found on the Internet to synchronize the clocks on your devices to. Internet service providers offer NTP service to their customers as well.

It is customary to have a redundant set of servers in the core of the network that are synchronized to an authoritative source and to configure your other devices to synchronize their clocks to these central sources. In large networks, this hierarchy could even consist of multiple levels.

Timeservers are configured using the **ntp server** command. If you want to use multiple timeservers for redundancy, you configure multiple **ntp server** commands. If you use multiple timeservers, the NTP protocol will decide which server is most reliable and synchronize to that server. Alternatively, a preferred server can be appointed by use using the **prefer** command option on the **ntp server** command.

In addition to defining timeservers, you most likely want to define your local time zone and configure the device to adapt to daylight savings time. Finally, when you have the time synchronized and set to the correct time zone, you have to configure the router to reflect the time in its logging time stamps properly.

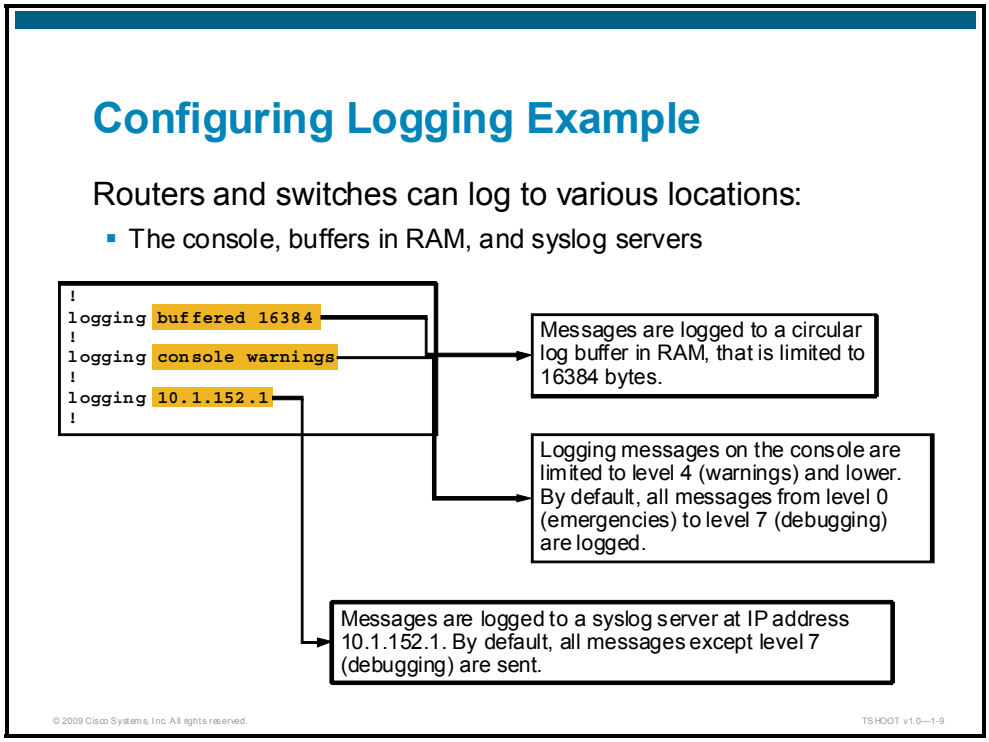
In the example in the figure, the clock of the device is synchronized to a single timeserver with IP address 10.1.220.3. The time zone is configured to Pacific Standard Time (PST), which has an -8 hour offset to Coordinated Universal Time (UTC). The clock is configured to change to daylight savings time on the second Sunday in March at 2:00 a.m. and back to standard time on the first Sunday in November at 2:00 a.m. The logging for system logging is configured to use the local date and time in the time stamps and to include the time zone in the time stamp. For log entries generated by debugs, the settings are similar, but milliseconds are included in the timestamps for greater accuracy.

For more details about the configuration of NTP, consult the “Configuring NTP” section of the Cisco IOS Network Management Configuration Guide:

http://www.cisco.com/en/US/docs/ios/netmgmt/configuration/guide/nm_basic_sys_manage_ps6350_TSD_Products_Configuration_Guide_Chapter.html#wp1001170

Implementing Logging Services

For troubleshooting, having access to device event logs is essential. When you implement logging services, the availability of these logs can be ensured.



Events that happen on the router, such as interfaces going up and down, configuration events, routing protocol adjacencies being established, and so on, are logged only to the console of the device by default. However, because the console is in many cases not easily accessible, let alone monitored, it is worthwhile to collect and store the logs on a server, or at least in a separate piece of memory of the router, so they can be accessed during troubleshooting procedures.

Logging the messages to buffers on the router or switch is a minimal step that guarantees that logs are available on the device, as long as it is not rebooted. On some devices and Cisco IOS Software versions, logging to buffers is turned on by default. To enable buffer logging manually, you can use the **logging buffered** command to specify that messages should be logged to a buffer in the device's RAM. The amount of RAM that should be allocated to this buffer can be specified as an option. The buffer is circular, meaning that when the buffer has reached its maximum capacity, the oldest messages will be discarded to allow the logging of new messages. The content of this logging buffer can be displayed by using the **show logging** command. In addition, the severity level, which ranges from level 0 (emergencies) to level 7 (debugging), can be specified as an option. This setting will cause the device to only log messages with a severity of the configured level or lower to the buffer. By default, messages of all severity levels are logged to the buffers.

In a similar way, you can adjust the logging severity level of the console. By default, all messages from level 0 to 7 will be logged to the console, but similar to buffer logging, the severity level can be configured as an optional parameter on the **logging console** command.

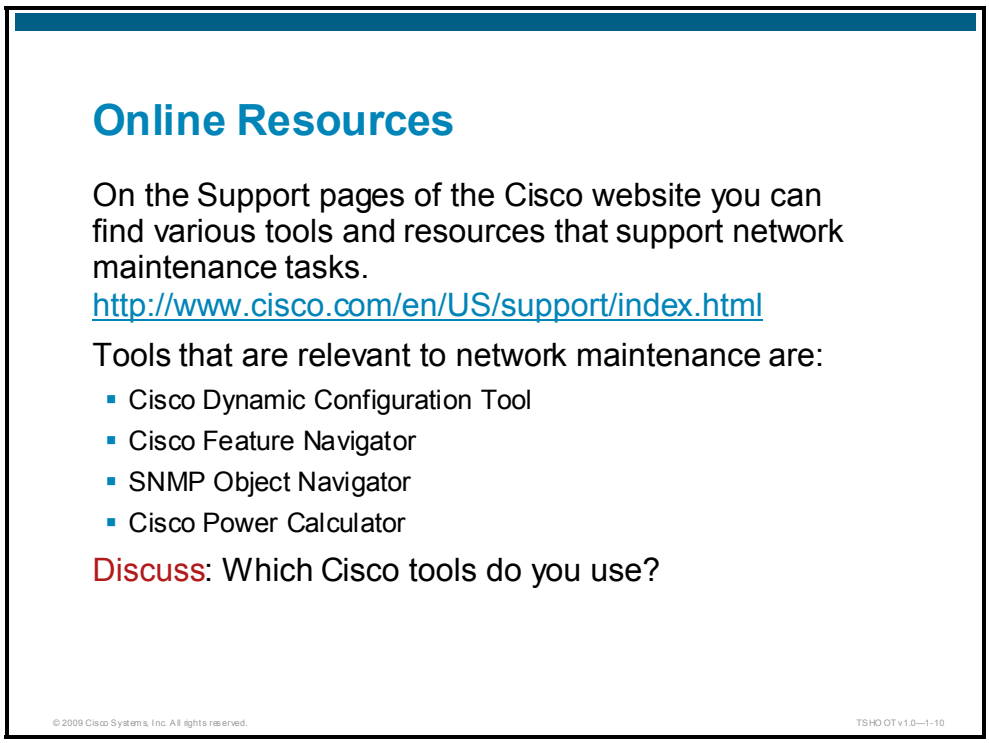
Finally, log messages can be sent to a syslog server. This setting allows you to store the logs of all your network devices centrally. One or more syslog servers can be configured by using the **logging *host*** command. By default, only messages of severity level 6 or lower will be logged to the syslog server. This setting can be changed, similar to buffer or console logging, but unlike these other commands, the severity is configured by use of a separate command **logging trap *level***. This command applies to all configured syslog hosts.

For more detail about the configuration of logging and syslog services, consult the “Logging System Messages” section of the Cisco IOS Network Management Configuration Guide:

http://www.cisco.com/en/US/docs/ios/netmgmt/configuration/guide/nm_troubleshooting_ps6350_TSD_Products_Configuration_Guide_Chapter.html#wp1054847

Web-Based Maintenance Tools

In addition to tools and applications that you run on your network, many additional online tools and resources can be helpful during the planning and implementation of network maintenance procedures.



The Cisco website provides numerous tools to support your network maintenance processes. You can find an overview of these tools at the Support Tools and Resources page:

http://www.cisco.com/en/US/partner/support/tsd_most_requested_tools.html

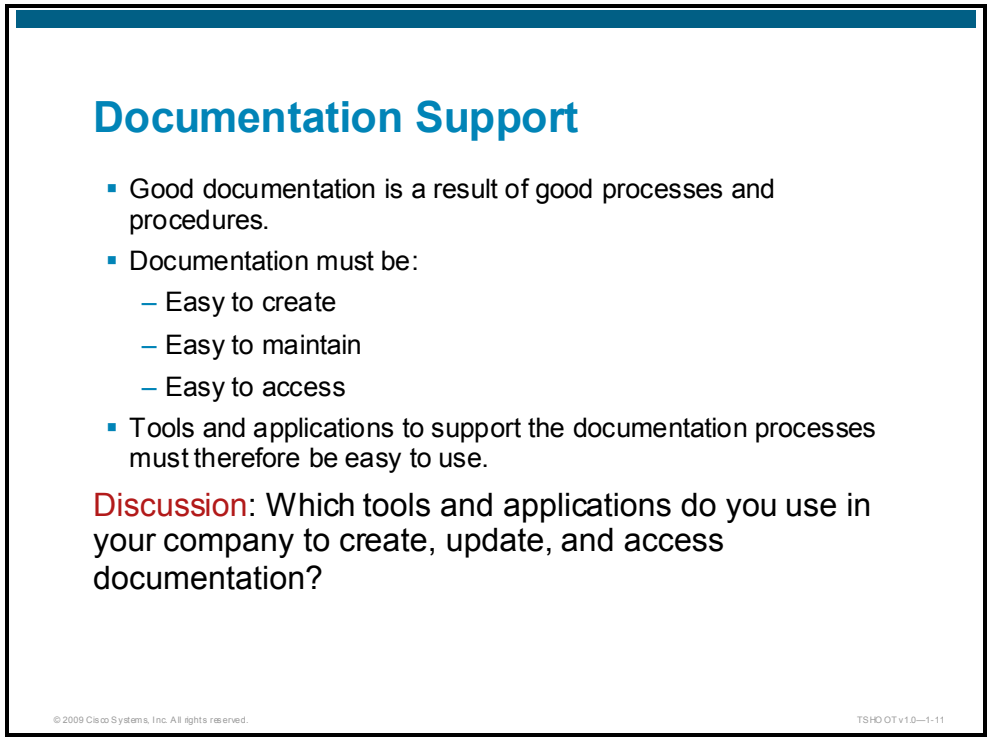
Note	Many of the tools are only available to registered users with a valid Cisco service contract or to Cisco Channel Partners.

Some useful freely available tools are:

- **Cisco Dynamic Configuration Tool:** This tool aids you in creating hardware configurations. It will verify if the hardware and software that you select is compatible and it will give you a complete Bill of Materials (BOM) that lists all the necessary part numbers.
- **Cisco Feature Navigator:** This tool allows you to find the right Cisco IOS Software release quickly for the features that you want to run on your network.
- **SNMP Object Navigator:** This tool helps translate Simple Network Management Protocol (SNMP) object IDs (OIDs) into object names, download SNMP MIB files, and allows you to verify the supported MIBs in a particular Cisco IOS Software version.
- **Cisco Power Calculator:** This tool enables you to calculate the power supply requirements for a particular Power over Ethernet (PoE) hardware configuration.

Documentation Tools

This topic describes tools that can help create an efficient documentation process.



The value of documentation heavily depends on two factors: It needs to be easily accessible and it needs to be up-to-date. If you cannot find or get access to documentation when you need it, or if you cannot trust the information that you find, the documentation and information lose almost all their value. Therefore, any tool or application used to support the creation, retrieval, or updating of documentation must be easy to access and use.

The following list gives examples of tools that can be used to create, update, and access documentation:

- **Wiki**: A wiki combines easy web-based access with intuitive editing capabilities. These qualities make it very suitable as a base documentation system. You can use it as a framework to link various other existing documentation systems.
- **Issue tracking system:** Other names for issue tracking systems include trouble ticket, support ticket, or incident ticket system. An issue tracking system allows incoming support requests, problems, or other incidents to be logged, tracked, and documented. By documenting progress, communication, and escalation of incidents, an issue tracking system allows a team of people to work on the same incidents in an efficient manner. It also helps build a historical database of problems, their treatment, and resolution.

Good documentation is mostly a result of a good process, so although good tools can be very helpful in supporting the documentation process, it is most important that creating and updating documentation is an integral part of your maintenance processes.

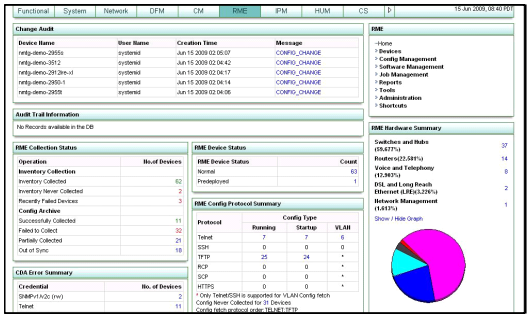
Disaster Recovery Tools

This topic describes tools and applications that help to create and restore backups.

Resource Management

Efficient disaster recovery is dependent on the following:

- Up-to-date configuration backups
- Up-to-date software backups
- Up-to-date hardware inventories
- Configuration and software provisioning infrastructure



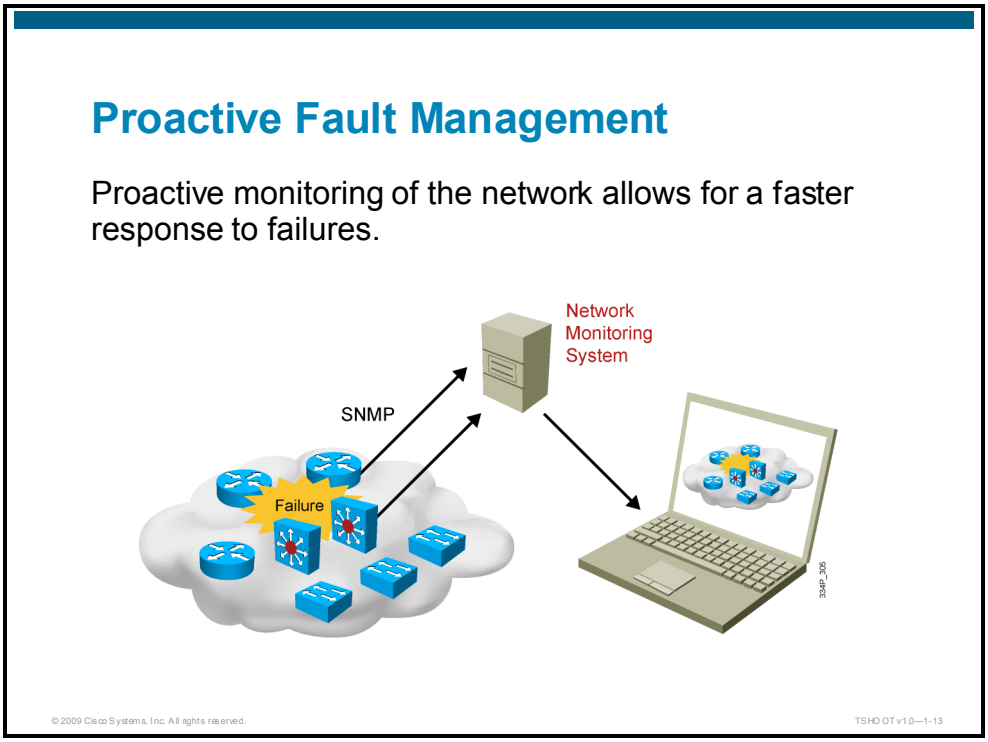
As part of the fundamental toolkit, we discussed the use of a TFTP, FTP, SCP, HTTP, or HTTPS server to create or restore backups of the configuration and operating system software of a Cisco router or switch. You can use additional software to extend this functionality and include some or all of the following functions:

- Automatic backup scheduling
- Configuration file comparison and change tracking
- Template creation and editing
- Pushing configurations to multiple devices
- Hardware inventory tracking

CiscoWorks Resource Manager Essentials (CiscoWorks RME), which is part of the CiscoWorks LAN Management Solutions (CiscoWorks LMS), is a prime example of software that provides this type of functionality.

Network Monitoring Tools

This topic describes the benefits that network-monitoring software can bring to your maintenance processes.



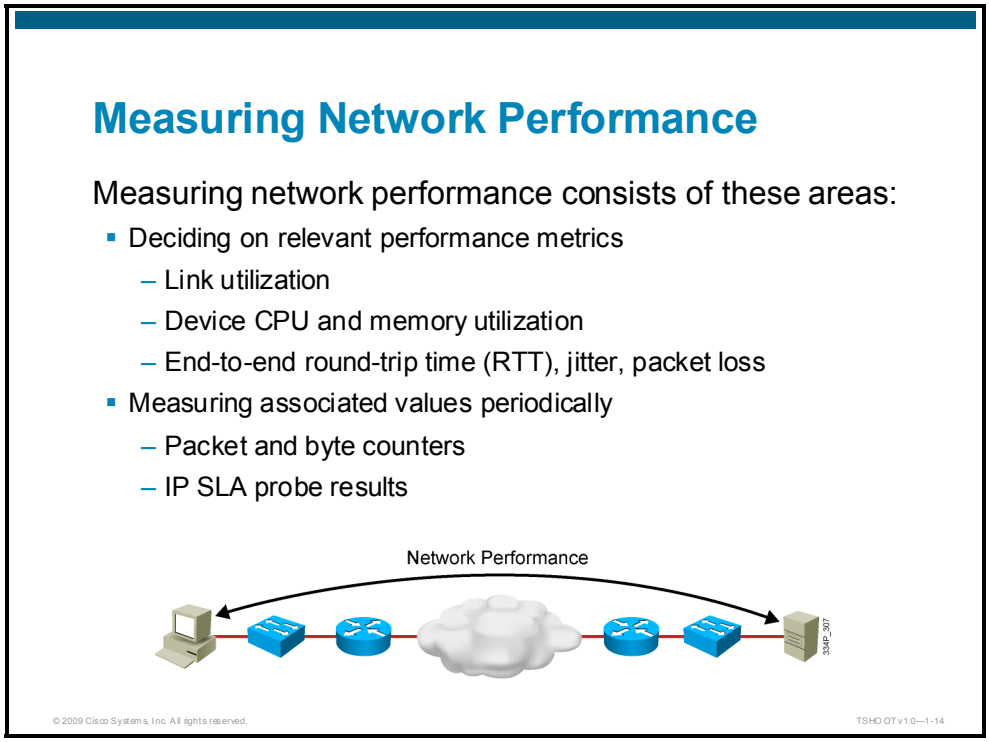
GUI-based and CLI-based device management tools give you the ability to examine individual systems when you suspect that there is a problem, but you would not detect the fact that there is a problem until you are notified through means of a user complaint. At that point, users have noticed the failure and it has had at least some impact on the business. A network monitoring system continuously checks the availability and status of your network devices. This continuous monitoring allows you to detect possible problems as soon as they occur and may even allow you to diagnose and resolve these problems before they even become apparent to end users.

Most network-monitoring software uses a combination of SNMP, Internet Control Message Board (ICMP), and syslog to monitor devices and network events. Additionally, Cisco IOS NetFlow technology may be used, not only to monitor the devices, but also to monitor the actual traffic on the network as it is flowing through these devices.

Some network monitoring tools may also incorporate performance-monitoring capabilities. Essentially, there is some overlap between those two functions, since you could be monitoring performance to find potential performance problems, but also as input for capacity planning, SLA compliance measurements, or accounting purposes.

Performance Measurement Tools

This topic describes the typical elements of performance measurement processes and the tools that you can use to support these processes.



What are some reasons to measure the performance of your network?

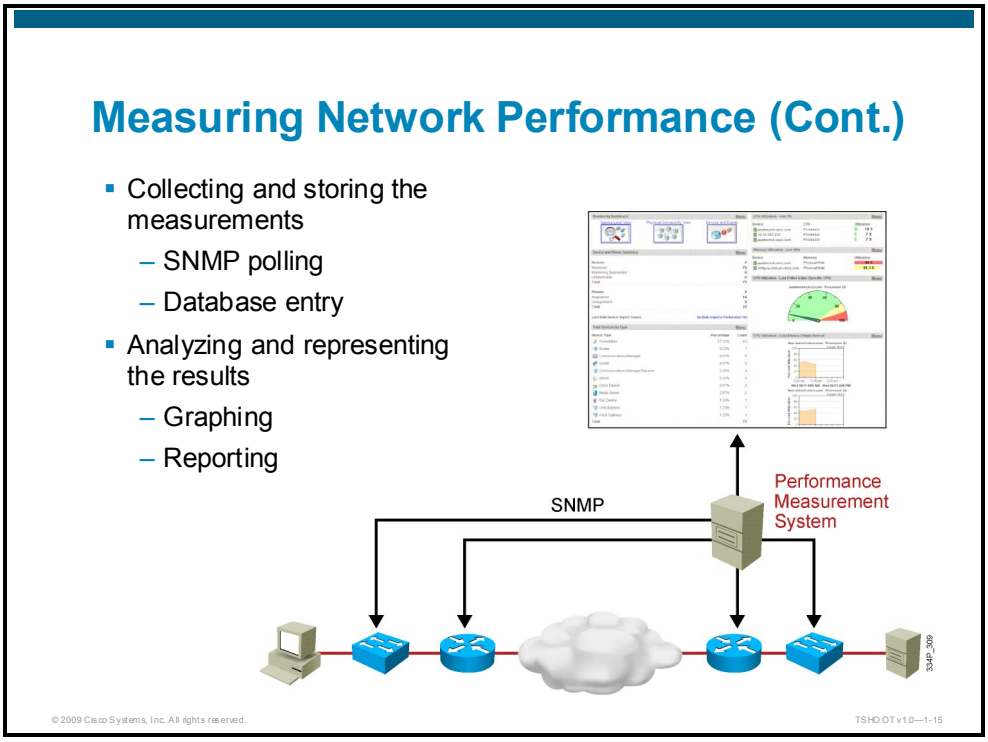
- **Capacity planning:** By measuring average and peak loads on the network, you can create a baseline of the traffic on your network. This baseline allows you to know the utilization levels of your network. By repeating measurements over time, you will also be able to recognize trends in the growth of the traffic and predict when you need to upgrade links or equipment before the growth starts causing congestion and performance problems.
- **Diagnosing performance problems:** Performance problems are amongst the hardest problems to troubleshoot because they are hard to quantify and very often intermittent in nature. A user might say, “Application X is really slow lately.” But what does that mean? What is slow? When is it slow? And what is causing this slowness? Is it the client, the server, the network in between? Having a good insight into the load on the network, specifically on the path between the client and the server, helps you to determine if network congestion might be causing these problems.
- **SLA compliance:** Whether you are guaranteeing a level of service to others through an SLA or whether you have been promised a certain level of service by a provider, you need to have a method to measure whether the service guarantees defined in the SLA are met or not.

How Do You Measure Network Performance?

You can gather many statistics from the routers and switches using the SNMP protocol. These statistics can then be stored in a database and graphed over time or analyzed to get an insight into device and link utilization and performance. The typical statistics you would gather are the packet and byte counters on interfaces and device CPU and memory utilization.

For capacity planning and some performance troubleshooting, these steps are a good start. However, for SLA compliance and a detailed insight into performance issues, you need to measure the key indicators that could give you an insight into application performance such as round-trip time (RTT), jitter, and packet loss.

The IP SLA feature that is available on many Cisco routers allows you to set up probes that measure these key indicators along particular paths through your network.



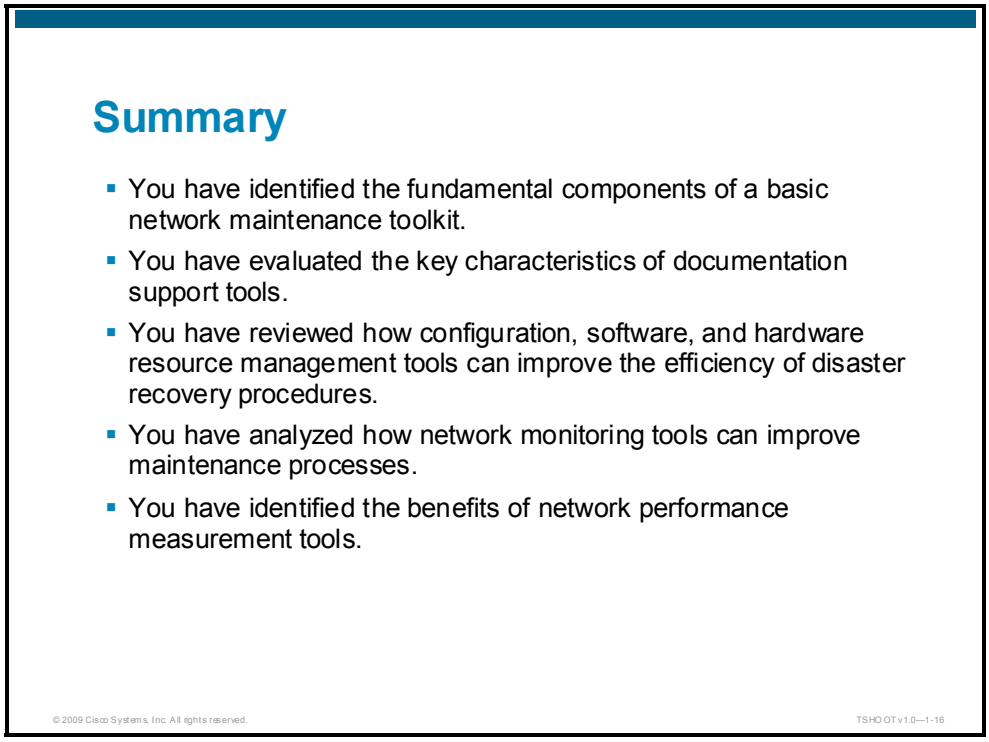
The statistics can then be read using Cisco IOS Software commands or collected via the Simple Network Management Protocol (SNMP). The raw data is stored in a database and can then be analyzed or graphed.

Cisco Internet Performance Monitor (CiscoWorks IPM), which is part of the CiscoWorks LMS, can use the IP SLA functionality in the routers to provide detailed performance graphs.

Additionally, various other network management solutions include the capability to collect statistics via SNMP and graph the results. A well-known example of this type of software is the open source Multi Router Traffic Grapher (MRTG) and other products that are based on it.

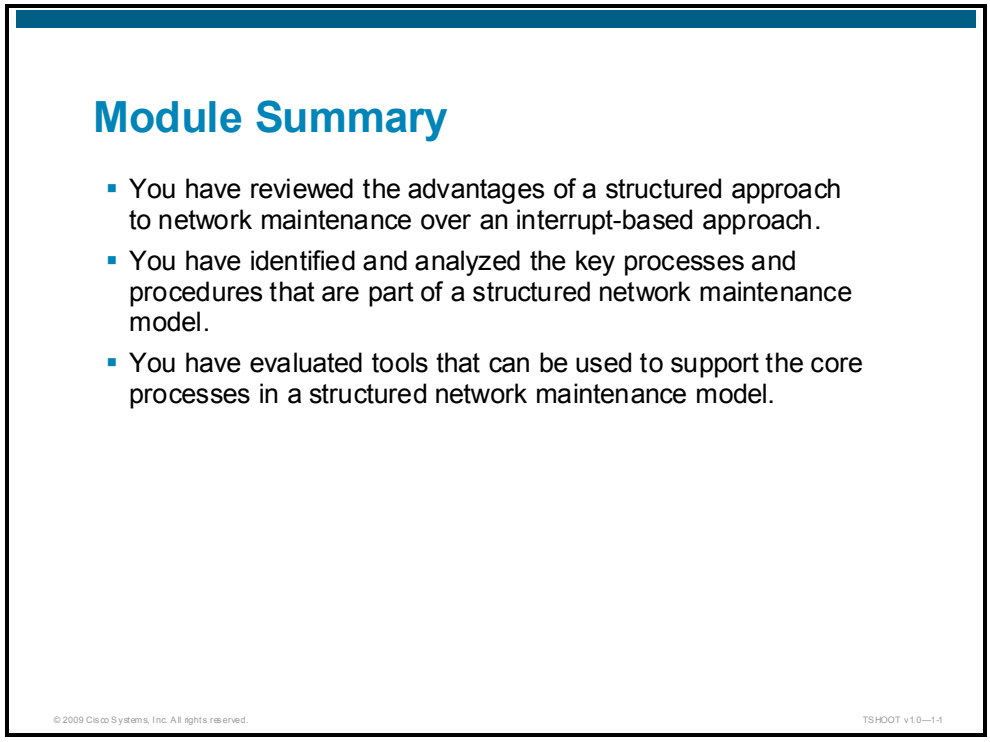
Summary

This topic summarizes the key points that were discussed in this lesson.



Module Summary

This topic summarizes the key points that were discussed in this module.



In this module, the benefits of a methodical approach to network maintenance were investigated.

The core processes and procedures that are part of such a network maintenance methodology were analyzed. This included processes such as maintenance scheduling and planning, change control, documentation, communication, disaster recovery, network health, and performance monitoring.

The essential elements that are part of a network maintenance toolkit to support these core processes were identified. The implementation of some of the important services that are part of this toolkit, such as backup and restore services, time services, and logging services, was also reviewed.

References

For additional information, refer to these resources:

- Office of Government Commerce (OGC). The Official ITIL Website*:* http://www.itil-officialsite.com/home/home.asp
- Cisco Systems, Inc. *Cisco IOS Configuration Fundamentals Configuration Guide Release 12.4,* Part 7: Configuring Basic File Transfer Services, Transferring Files Using HTTP or HTTPS. San Jose, California, May 2005; May 2009*:* http://www.cisco.com/en/US/docs/ios/fundamentals/configuration/guide/cf_file-trnsfer_http_ps6350_TSD_Products_Configuration_Guide_Chapter.html

- Cisco Systems, Inc. *Cisco IOS Configuration Fundamentals Configuration Guide Release 12.4*, Part 8: Managing Configuration Files, Configuration Replace and Configuration Rollback. San Jose, California, March 2004; October 2008: http://www.cisco.com/en/US/docs/ios/fundamentals/configuration/guide/cf_config-rollback_ps6350_TSD_Products_Configuration_Guide_Chapter.html
- Cisco Systems, Inc. *Network Time Protocol: Best Practices White Paper*. San Jose, California, December 2008: http://www.cisco.com/en/US/tech/tk869/tk769/technologies_white_paper09186a008011701070.shtml
- Cisco Systems, Inc. *Cisco IOS Network Management Configuration Guide, Release 12.4*, Performing Basic System Management, Configuring NTP. San Jose, California, 2007: http://www.cisco.com/en/US/docs/ios/netmgmt/configuration/guide/nm_basic_sys_manage_ps6350_TSD_Products_Configuration_Guide_Chapter.html#wp1001170
- Cisco Systems, Inc. *Cisco IOS Network Management Configuration Guide, Release 12.4*, Troubleshooting and Fault Management, Logging System Messages. San Jose, California, 2009*:* http://www.cisco.com/en/US/docs/ios/netmgmt/configuration/guide/nm_troubleshooting_ps6350_TSD_Products_Configuration_Guide_Chapter.html#wp1054847
- Cisco Systems, Inc. Cisco Feature Navigator: http://tools.cisco.com/ITDIT/CFN/jsp/index.jsp
- Cisco Systems, Inc. Cisco Dynamic Configuration Tool: https://tools.cisco.com/qtc/config/html/configureHomeGuest.html
- Cisco Systems, Inc. SNMP Object Navigator: http://tools.cisco.com/Support/SNMP/public.jsp
- Cisco Systems, Inc. Cisco Power Calculator: http://tools.cisco.com/cpc/launch.jsp

Module Self-Check

Use the questions here to review what you learned in this module. The correct answers and solutions are found in the Module Self-Check Answer Key.

Q1) Which three benefits are benefits of a structured approach to network maintenance? (Choose three.) (Source: Applying Maintenance Methodologies)

A) Maintenance processes are better aligned to business needs.
B) Hardware discounts can be negotiated with the reseller.
C) The overall security of the network will be higher.
D) The total unplanned network downtime will be lower.
E) Users will never have to wait to get support.
F) Network maintenance can be outsourced to lower the cost.

Q2) Which three methodologies are methodologies that can be applied to network maintenance? (Choose three.) (Source: Applying Maintenance Methodologies)

A) Fault management, Configuration management, Accounting management, Performance management, and Security management
B) IT Infrastructure Library
C) Optimization and Maintenance
D) Telecommunications Management Network

Q3) Which two processes are common network maintenance processes? (Choose two.) (Source: Common Maintenance Processes and Procedures)

A) disaster recovery
B) network design
C) budget approval
D) documentation

Q4) Which two benefits are benefits of scheduled maintenance? (Choose two.) (Source: Common Maintenance Processes and Procedures)

A) Network engineers will not have to work outside regular work hours.
B) Lead times for change requests will be more predictable.
C) Disruptive maintenance tasks can be scheduled during assigned maintenance windows.

Q5) Which factors should be considered during the implementation of change procedures? (Source: Common Maintenance Processes and Procedures)

_____

_____

_____

_____

_____

Q6) Which three items do you need to have in order to replace a failed device? (Choose three.) (Source: Common Maintenance Processes and Procedures)

A) replacement hardware for the failed device
B) proof of purchase of the failed device
C) TAC support for the failed device
D) the current configuration of the failed device
E) the current software version of the failed device
F) the original box that the failed device was shipped in

Q7) Network monitoring is a fundamental aspect of a proactive network management strategy. (Source: Common Maintenance Processes and Procedures)

A) true
B) false

Q8) Which five protocols can be used to transfer a configuration file from a router to a server to create a configuration backup? (Choose five.) (Source: Network Maintenance Tools, Applications, and Resources)

A) HTTPS
B) HTTP
C) FTP
D) SNMP
E) TFTP
F) SCP

Q9) Which command is the command that is used to copy the running configuration of a router to a file named "test.cfg" residing on an FTP server with IP address 10.1.1.1, using the username "admin" and password "cisco"? (Source: Network Maintenance Tools, Applications, and Resources)

A) **copy running-config ftp://10.1.1.1/test.cfg user admin password cisco**
B) **copy running-config ftp://10.1.1.1/test.cfg /user:admin /password:cisco**
C) **copy running-config ftp://admin:cisco@10.1.1.1/test.cfg**
D) **archive running-config ftp://10.1.1.1/test.cfg user admin password cisco**
E) None of the above is correct; ftp does not require authentication.

Q10) The command that is used to create an archive copy of the running configuration manually is _____. (Source: Network Maintenance Tools, Applications, and Resources)

Q11) Which command is the command that is used to restore the current configuration to the archived configuration file "RO1-archive-config-5" residing in flash? (Source: Network Maintenance Tools, Applications, and Resources)

A) **archive rollback flash:/RO1-archive-config-5**
B) **configure replace flash:/RO1-archive-config-5**
C) **copy flash:/RO1-archive-config-5 running-config**
D) **archive restore flash:/RO1-archive-config-5**

Q12) The command that is used to configure a switch to log system messages to a syslog server at IP address 10.1.1.1 is _____. (Source: Network Maintenance Tools, Applications, and Resources)

Q13) What is the functionality delivered by the online Cisco Dynamic Configuration Tool? (Source: Selecting Network Maintenance Tools, Applications and Resources)

A) It interprets router configurations and recommends changes based on a set of best current practices.
B) It converts Cisco IOS firewall configurations to PIX or ASA configurations and vice versa.
C) It converts CatOS switch configurations to Cisco IOS configurations.
D) It validates hardware configurations and creates a bill of materials from it.

Q14) Which two processes are processes that benefit from the implementation of a network performance measurement system? (Choose two.) (Source: Network Maintenance Tools, Applications, and Resources)

A) disaster recovery
B) change management
C) capacity planning
D) SLA compliance

Module Self-Check Answer Key

Q1) A, C, D

Q2) A, B, D

Q3) A, D

Q4) B, C

Q5) Risk, impact, and resources should be balanced against urgency, necessity, and business needs.

Q6) A, D, E

Q7) A

Q8) A, B, C, E, F

Q9) C

Q10) **archive config**

Q11) B

Q12) **logging 10.1.1.1**

Q13) D

Q14) C, D

Module 2

Planning Troubleshooting Processes for Complex Enterprise Networks

Overview

As enterprises have become more and more dependent on their network infrastructure to support their business, the cost of network downtime has increased. When employees cannot perform their jobs due to network problems, the company loses productivity and therefore money. Network troubleshooting is an essential task that a network engineer needs to be capable of performing. The more efficiently and effectively problems can be diagnosed and resolved, the lower the impact will be to the business.

In complex environments, troubleshooting can be a daunting task, and the only way to diagnose and resolve problems quickly and effectively is by following a structured approach. This approach involves having well-defined and documented troubleshooting procedures and aligning troubleshooting procedures to general network maintenance procedures.

In this module, the benefits of a structured approach to troubleshooting are evaluated. The leading principles that are at the core of all troubleshooting methodologies are identified. Common approaches to network troubleshooting are evaluated, and generic troubleshooting processes and their relation to network maintenance processes are analyzed.

Module Objectives

Upon completing this module, you will be able to develop a troubleshooting process to identify and fix problems in complex enterprise networks. This ability includes being able to meet these objectives:

- Identify the challenges posed by troubleshooting in an unknown environment without access to tools and documentation and without established processes and procedures
- Evaluate and rate commonly practiced models and methodologies for network troubleshooting

- Plan and implement troubleshooting procedures as part of a structured troubleshooting methodology
- Plan and implement troubleshooting and network maintenance procedures that effectively support each other

Lesson 1

Lab 2-1 Debrief

Overview

Troubleshooting is as much about process, methodology, and communication as it is about technology. By experiencing the challenges of troubleshooting without access to documentation, an agreed troubleshooting method, supporting tools, or established communication procedures, you have the opportunity to observe the effect of unstructured troubleshooting and how the troubleshooting process could be improved.

In this lab, you practiced basic IP troubleshooting methods to resolve a routing protocol problem, and you discovered and documented the essential parts of the network topology. You diagnosed the problem and proposed a solution.

During the lab debrief, the instructor will lead a group discussion during which you can present your solutions. You will have an opportunity to verify your solution against a number of checkpoints provided by the instructor and compare your solution to the solutions of other students. The instructor will discuss alternate solutions and their benefits and drawbacks.

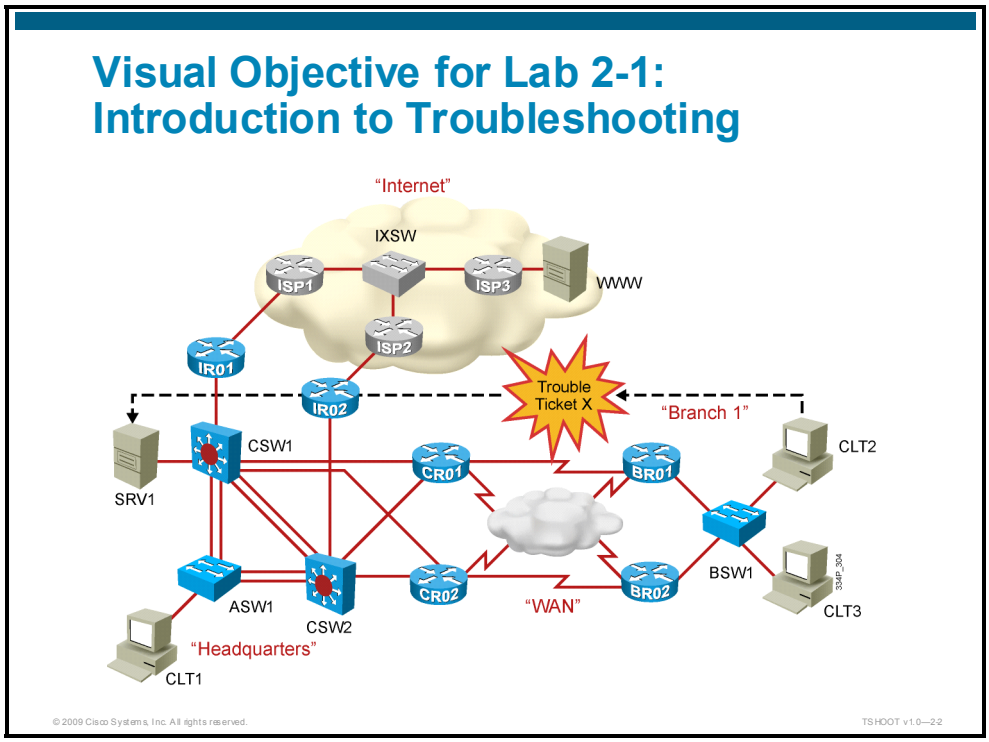
Objectives

Upon completing this lesson, you will be able to identify the primary procedural aspects of a troubleshooting process. This ability includes being able to meet these objectives:

- Compare your solution, findings, and action log against a set of checkpoints provided by the instructor and identify common and alternate solutions
- Consolidate the lessons learned during the review discussions into a set of best-practice methods and commands to aid you in future troubleshooting procedures

Review and Verification

This topic describes the problem that was introduced in Lab 2-1, asks how you can verify that you have solved the problems, and gives an example of a troubleshooting process that will allow you to find and resolve the issues.



This lab consists of a single identical problem on the routers CRO1 and CRO2, causing routing across the WAN to fail. The main objective of this lesson is not to solve the problem, but to experience the challenges of troubleshooting without access to tools, documentation, assigned responsibilities, or established procedures. Fifteen minutes is an extremely short time to find the problem, so the lessons learned from the process are more important than the solution in this introductory lab.

Trouble Ticket: No Connectivity to the Server

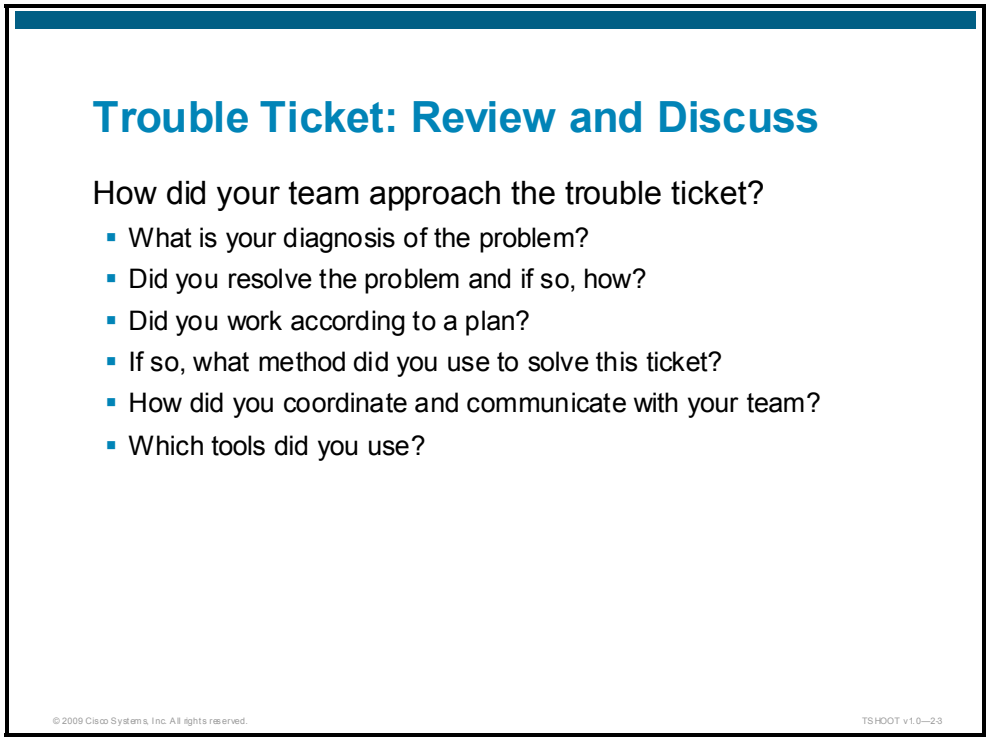
The text introducing this trouble ticket is the following:

You have just started your new job as a network engineer together with a few other engineers who are also newly hired. It is your first day at work, and your new team lead has just shown everybody to their desks and is busy arranging cell phones and all the other things that you need to get started. He takes a quick look at his PC and then tells you that a trouble ticket has just come in and that he would appreciate it if you and your other new teammates could do the initial troubleshooting while he is getting your things together. You are given the passwords to the routers and switches. He tells you to be careful in making changes, but to fix the problem if you can. He would like you to give him a diagnosis as soon as he returns, which will be 15 minutes from now.

The trouble ticket reads:

A user in Branch1 (PC CLT2) reports problems accessing the shared folder “\\SRV1\Public” on server SRV1. The user needed to leave for a meeting that will take all morning, but expects it to work when he returns after lunch.

Your task is to diagnose the issue, fix it if possible, and report back to your team lead in 15 minutes.



Note	This lesson provides sample answers and solutions to the questions posed during the debrief discussions. Keep in mind that alternate solutions and methods might be just as valid, and that the solutions provided here only highlight a sample approach and solution.

The biggest challenge in this lab is not the diagnosis of the problem itself, but finding your way in the network, establishing an effective process for working within the team without getting in each other's way, and using the results of different team members to get to a diagnosis efficiently.

The lack of network documentation—such as network diagrams, connection lists, IP address plans, and baseline configurations—complicates the troubleshooting procedure and increases the chance of making errors and the time it takes to diagnose and resolve the problem. As a result, you need to be even more careful to communicate and verify your findings with the other team members.

A sample approach would have been to first review the problem and the topology with the team, and then assign responsibilities to each team member. This should include primary responsibility for each of the devices and a division of the work. For a team consisting of three members, the work assignments could be as follows:

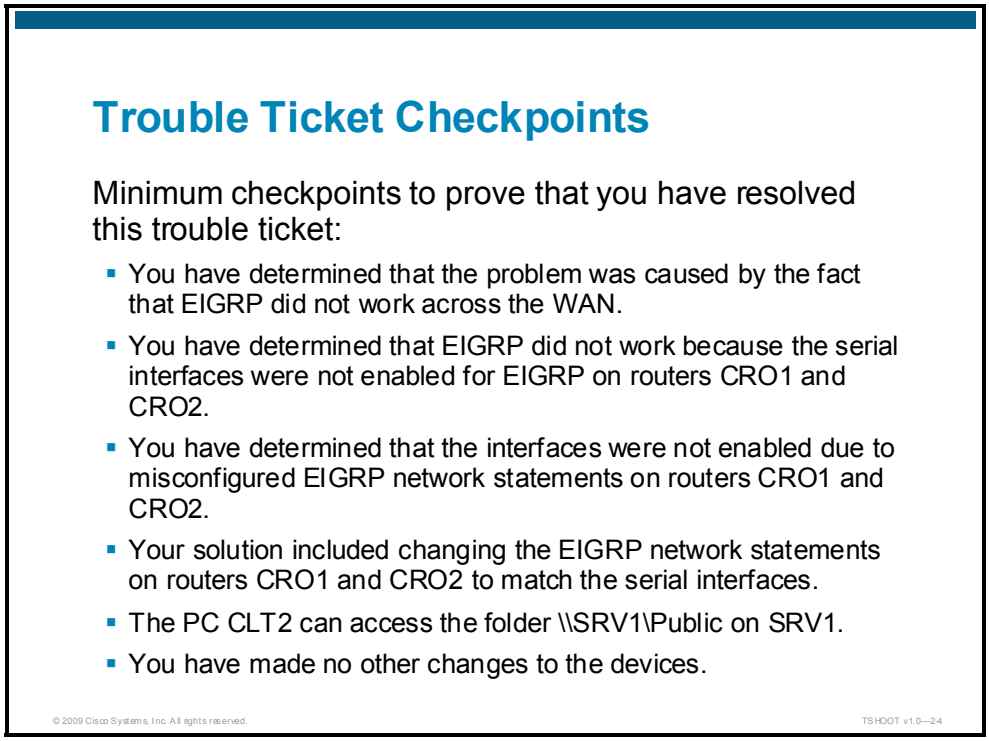
- Team member 1 has the primary responsibility for all the headquarters LAN switches. Team member 2 controls the headquarters routers. Team member 3 controls the branch routers and switch.
- Team member 1 will verify connectivity between the server and the LAN.
- Team member 2 will verify connectivity to the server from the routers and from the routers to the branch networks.
- Team member 3 will verify the problem and troubleshoot connectivity from the branch to the headquarters networks.

Dividing the work in a manner that allows the different team members to work simultaneously on different potential failure domains prevents work duplication. This allows problem causes to be eliminated more quickly than if each team member were working solo.

Although there are many possible methods that you could use to solve this problem, the following list is an example of a troubleshooting log to show you a methodical approach to finding the problem. This log is the log of team member 3, working to the plan outlined previously.

Trouble Ticket	Actions and results
	CLT2: Tried to ping SRV1. Could not resolve the name to IP address.
	Asked team member 1 for IP address of server SRV1. It is 10.1.152.1.
	CLT2: Tried to ping 10.1.152.1. Received destination unreachable from 10.1.160.124.
	Used Telnet to log into 10.1.160.124. This turns out to be router BRO1.
	BRO1: Tried to ping to 10.1.152.1: No response.
	BRO1: Checked the routing table for availability of a route to 10.1.152.1. No route is available.
	BRO1: Checked the routing protocol: EIGRP is running and enabled on all interfaces, including the WAN interfaces.
	BRO1: Checked the EIGRP neighbors. No neighbors are seen across the WAN.
	BRO1: Pinged the IP addresses on the other side of the WAN: All pings work.
	Checked with team member 2: He confirms that he is not receiving any routes across the WAN either.
	Team member 2 reports that EIGRP is not correctly enabled on the WAN interfaces.
	Team member 2 changes the network statements on CRO1.
	BRO1: EIGRP neighbors are now established and routes are learned.
	BRO1: Ping to 10.1.152.1 succeeds.
	CLT2: Ping to 10.1.152.1 succeeds.
	CLT2: Opened the share to \\SRV1\Public: Succeeds.
	Report back to team members: Problem solved.
	Discussed issue with team members. Conclusion: This cannot have been the only problem, because the WAN is redundant.
	BRO2: Similar problem exists.
	Team member 2 finds a similar problem on CRO2 and fixes it.

Because the objective of this lab was to learn about processes, methods, and communication, the technical aspects of this lab will not be reviewed in detail. More technical detail about troubleshooting EIGRP and routing in general will follow in later modules.



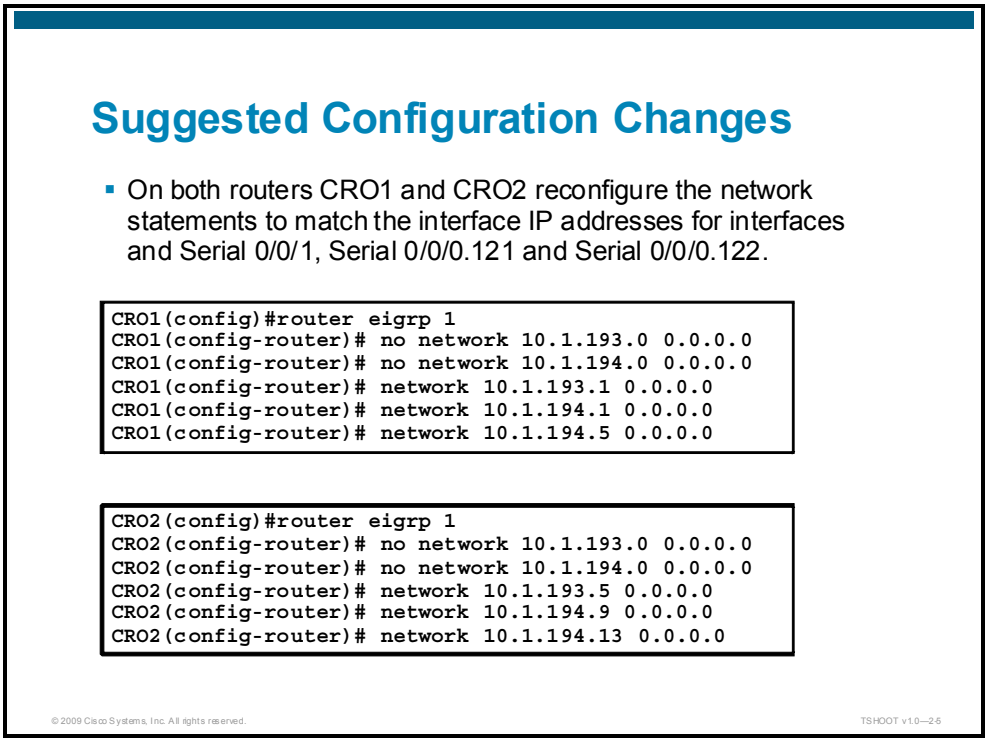
The figure shows the minimum requirements that you should meet to demonstrate that you have successfully resolved the problems in the trouble ticket.

Because the network has redundant connections and devices, it is not sufficient to verify that you have restored connectivity. You should also verify that the network would still operate correctly if any of the redundant components fail.

In this lab, there was an identical problem on routers CRO1 and CRO2. If the network statements are reconfigured on routers CRO1, the connectivity from PC CLT2 to server SRV1 is restored. However, when router CRO1 fails, the connectivity across the WAN is lost, even though there is physical redundancy. You will need to fix the network statements on both routers to have the level of availability that was intended in the network design.

Suggested Solutions

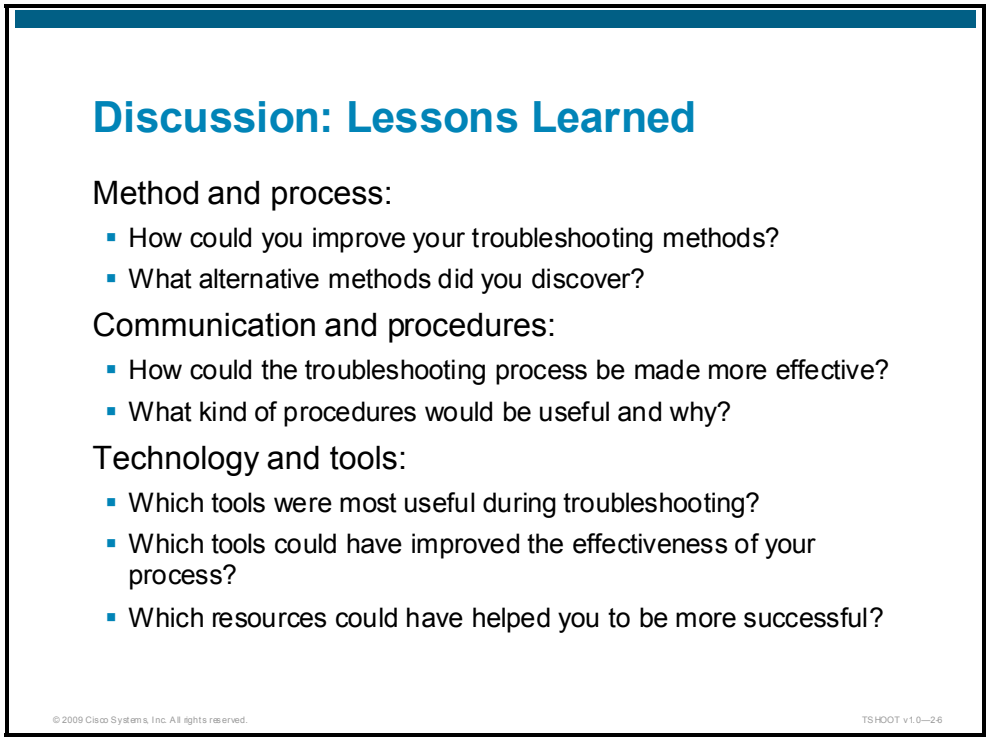
The problems introduced in the lab can be solved by correcting the switch configurations as follows.



The commands shown in the figure fix the problems that were introduced in this lab and return the configuration to its original baseline. Other solutions can be valid. If you have a different solution, bring it to the attention of the group and discuss the possible benefits or drawbacks of your solution. Most of the value of this exercise is in the process, not in the solution itself. Think about the process you followed and try to find ways to make improvements.

Consolidation

This topic describes the primary lessons that could be learned from the lab exercise.

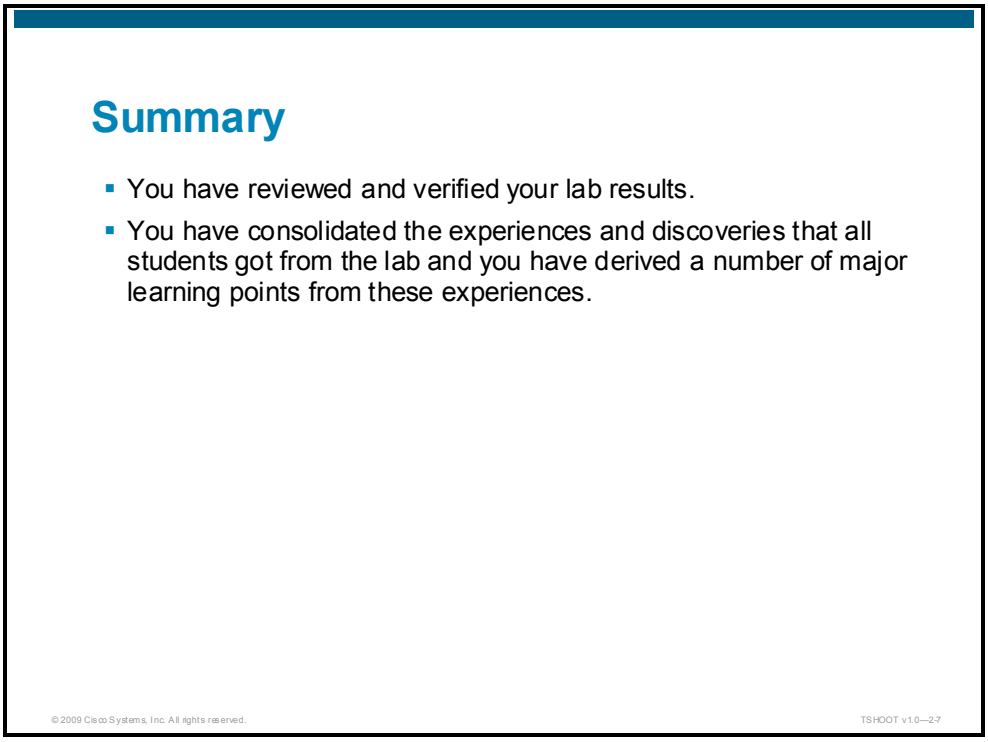


Think about all the things that you learned during the lab itself and during the debrief discussions. There is room to write down primary learning points in the Lab Debrief Notes section of the lab guide.

In addition to thinking of the methods, processes, and tools as they were used in the lab, reflect on how these would apply to your own organization.

Summary

This topic summarizes the key points that were discussed in this lesson.



Lesson 2

Applying Troubleshooting Methodologies

Overview

In an ideal world, things always work and problems never happen. However, in reality, people make mistakes and devices break. Diagnosing and resolving problems is an essential skill that network engineers use as a part of their many different job tasks.

There are no recipes for troubleshooting, and a particular problem can be diagnosed and sometimes even solved in many different ways. However, by employing a structured approach to the troubleshooting process, you can greatly reduce the average time it takes to diagnose and solve a problem.

There are many different structured troubleshooting approaches, and for some problems, one method might work better, while for others a different method may be more suitable. Therefore, the toolbox of a troubleshooter should contain a variety of structured approaches that a network engineer can choose from to select the best method or combination of methods for a particular problem.

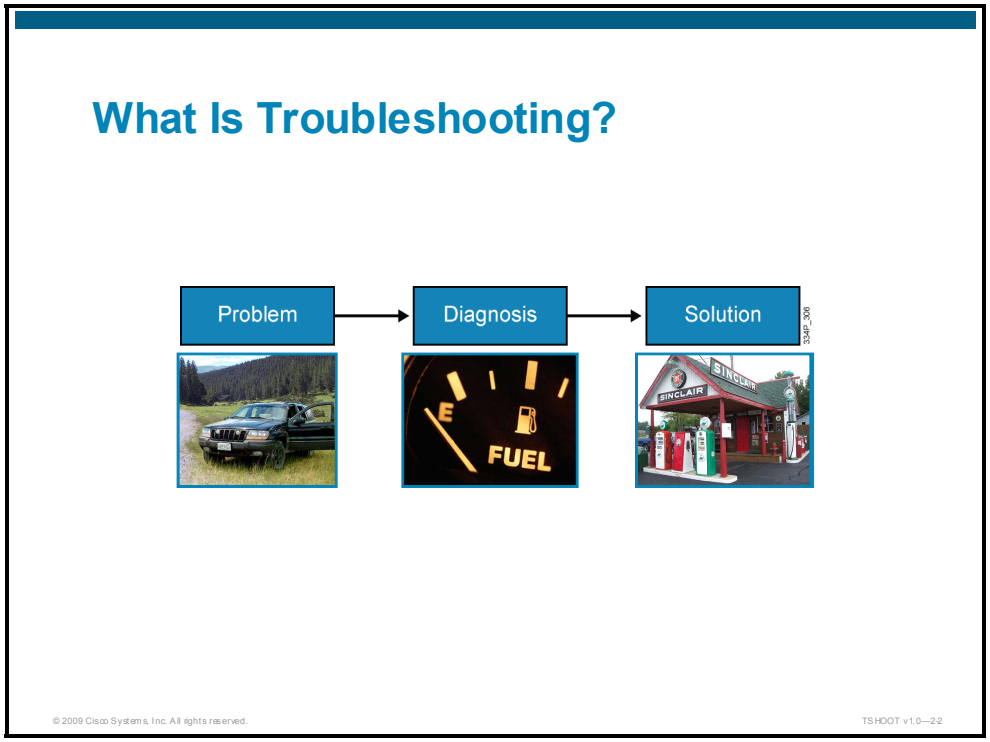
Objectives

Upon completing this lesson, you will be able to identify troubleshooting principles and evaluate and rate commonly practiced troubleshooting methodologies. This ability includes being able to meet these objectives:

- Identify the fundamental elements of a troubleshooting process
- Describe the advantages of a structured network troubleshooting method
- Evaluate and assess common troubleshooting approaches
- Select a combination of troubleshooting methods that are appropriate to a specific troubleshooting scenario

Troubleshooting Principles

This topic describes the main principles that are at the core of any process that aims at diagnosing and solving problems.



What is troubleshooting? Troubleshooting is the process that leads to the diagnosis and, if possible, resolution of a problem.

In general, a troubleshooting process starts when someone reports a problem. So in a way, you could say that a problem does not exist until it is noticed, considered a problem, and reported. What this shows is that you need to differentiate between a problem, as experienced by the user, and the cause of that problem.

Consequently, you also need to be aware that the time that a problem was reported is not necessarily the same as the time at which the event that caused that problem happened. Another consequence of this is that the reporting user generally equates the problem to the symptoms, whereas the troubleshooter often equates the problem to the root cause.

If the Internet connection fails on Saturday in a small company, is that a problem? Probably not, but you can be sure that it will turn into a problem on Monday morning if it is not fixed before then.

Although this distinction between symptoms and cause may seem philosophical, it is good to be aware of the potential communication issues that can arise from this.

A troubleshooting process starts with the reporting and definition of a problem. Next, the process of diagnosing the problem starts. During this process information is gathered, the problem definition is refined, and possible causes for the problem are proposed. Eventually this process should lead to a diagnosis of the root cause of the problem.

When the root cause has been found, possible solutions need to be proposed and evaluated. After the best solution is selected, that solution should be implemented. In some cases, the solution cannot immediately be implemented, and you will need to propose a workaround until the actual solution can be implemented. The difference between a solution and a workaround is that a solution resolves the root cause of the problem, whereas a workaround only remedies or alleviates the symptoms of the problem.

You and the user might have a different perception of what a problem actually is. When you implement a workaround, does that mean that the problem was solved? The reporting user will probably answer in the affirmative, because the user is not continuing to experience the symptoms, but the troubleshooter will probably not agree that the problem was solved, because the root cause was not addressed.

Diagnostic Principles

Diagnosis is the process of identifying the nature and cause of a problem.

Fundamental elements of this process are:

- Information gathering
- Analysis
- Elimination
- Proposing hypotheses
- Testing

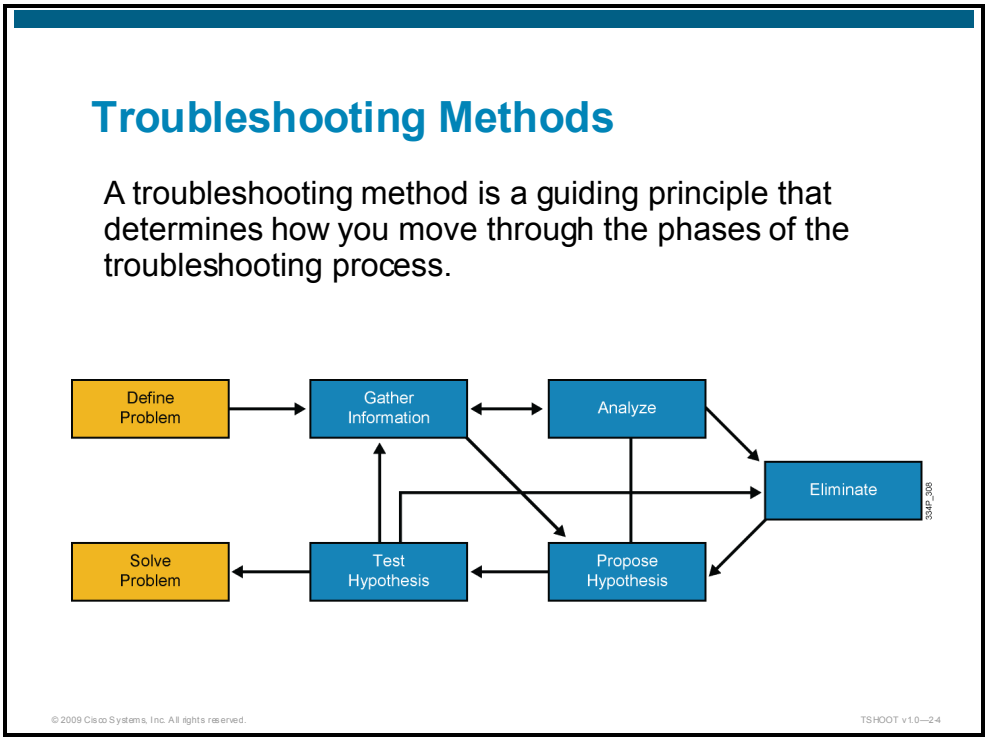
Although problem reporting and resolution are definitely essential elements of the process, most of the time is spent in the diagnostic phase, and you might even argue that this is really what troubleshooting consists of. However, if you look at troubleshooting in the context of network maintenance, problem reporting and resolution should definitely be taken into account as part of the troubleshooting process.

Diagnosis is the process of identifying the nature and cause of a problem. Essential elements of this process are:

- **Gathering of information and symptoms:** Gathering information about what is happening is essential to the troubleshooting process. Usually the problem report does not contain enough information to formulate a good hypothesis without first gathering more information. Information and symptoms can be gathered directly by observing processes, or indirectly by executing tests.
- **Analyzing information:** The gathered information is analyzed. You compare the symptoms against your knowledge of the system, processes, and baselines to separate normal behavior from abnormal behavior.
- **Eliminating possible causes:** By comparing the observed behavior against expected behavior, you can eliminate possible problem causes.
- **Formulating a hypothesis:** After gathering and analyzing information and eliminating the possible causes, you will be left with one or more potential problem causes. You need to assess the probability of each of these causes, and propose the most likely cause as the hypothetical cause of the problem.
- **Testing the hypothesis:** You will test the hypothetical cause to confirm or deny that it is the actual cause of the problem. The simplest way to do this is by proposing a solution based on this hypothesis, implementing that solution, and verifying if it solved the problem. If this method is impossible or disruptive, the hypothesis can be strengthened or invalidated by gathering and analyzing more information.

Structured Network Troubleshooting

This topic describes the benefits of a structured approach to network troubleshooting.



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

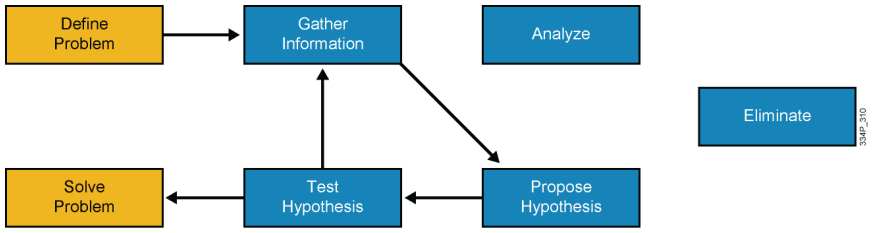
In a typical troubleshooting process for a complex problem, you would continually move between the different processes: Gather some information, analyze, eliminate some possibilities, gather more information, analyze again, formulate a hypothesis, test it, reject it, eliminate some more possibilities,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

If you do not use a structured approach to this process, but move between the phases in a more or less random way, you might eventually find the solution, but the process in general will be very inefficient. In addition, if your approach has no structure, it is practically impossible to hand it over to someone else without losing all the progress that was made up to that point. This even applies to resuming your own troubleshooting process.

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

The "Shoot from the Hip" Method

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



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

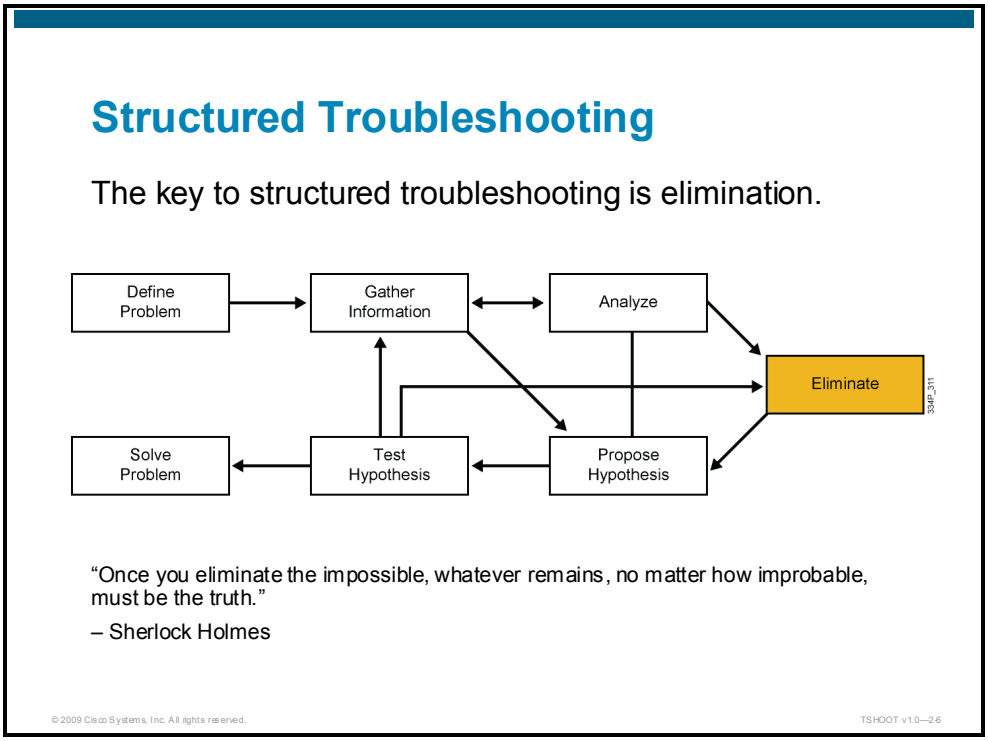
Look at the following example: A user reports a LAN performance problem to you. In 90 percent of similar problems in the past in this environment, the problem was caused by a duplex mismatch, and the solution was to configure the switch port for 100 Mb/s full duplex. An obvious thing to do now is to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

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

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

Common Troubleshooting Approaches

This topic describes a number of well-known troubleshooting methods.

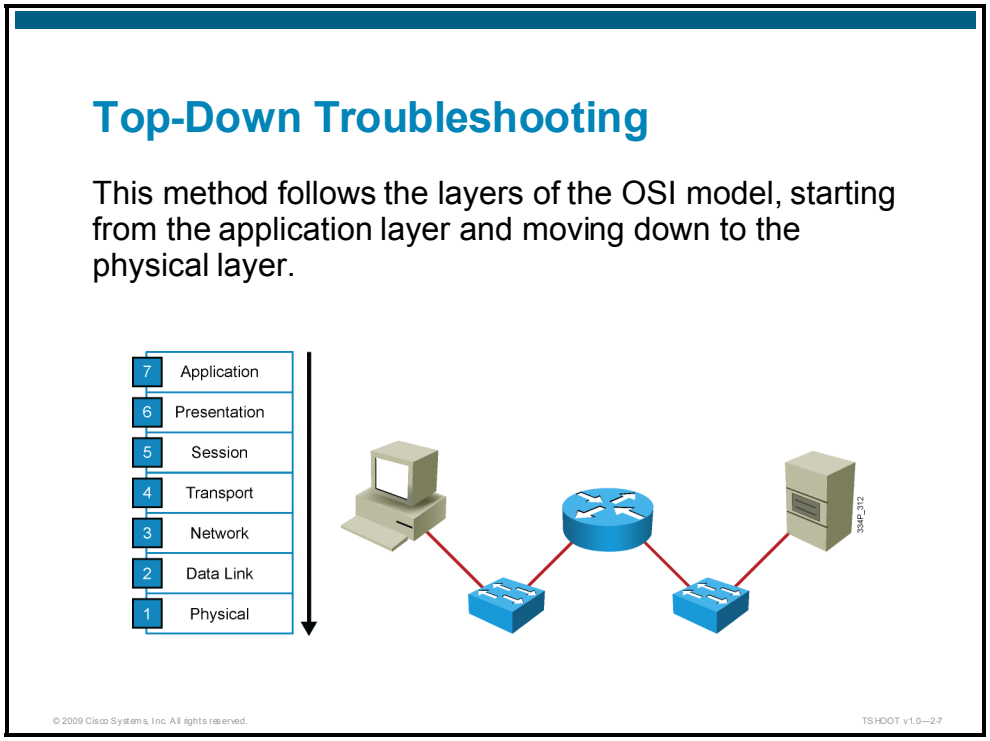


A structured troubleshooting method is a guideline that helps you move through the different phases of the troubleshooting process. The key to all structured troubleshooting methods is elimination of causes.

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

Some commonly used troubleshooting approaches are:

- **Top down:** Work from the application layer in the Open Systems Interconnection (OSI) model down to the physical layer.
- **Bottom up:** Work from the physical layer in the OSI model up to the application layer.
- **Divide and conquer:** Start in the middle of the OSI layers (usually the network layer) and then go up or down, depending on the results.
- **Follow the path:** Determine the path that packets follow through the network from source to destination and track the packets along the path.
- **Spot the differences:** Compare devices or processes of the network that are operating correctly to devices or processes that are not operating as expected. Gather clues by spotting significant differences.
- **Move the problem:** Physically move components and observe if the problem moves with the components or not.



The top-down troubleshooting method uses the OSI model as a guiding principle. One of the most important characteristics of the OSI model is that each layer depends on the underlying layers for its operation. This implies that if you find a layer to be operational, you can safely assume that all underlying layers are fully operational as well.

For example, if you are researching a problem of a user who cannot browse a particular website and you find that you can establish a TCP connection on port 80 from this host to the server and get a response from the server, you can typically draw the conclusion that the transport layer and all layers below must be fully functional between the client and the server, and that this is most likely a client or server problem and not a network problem.

Be aware that, in the example above, it is reasonable to conclude that Layers 1 through 4 must be fully operational, but this is not definitively proved. For example, unfragmented packets might be routed correctly, while fragmented packets are dropped. The TCP connection to port 80 might not uncover such a problem.

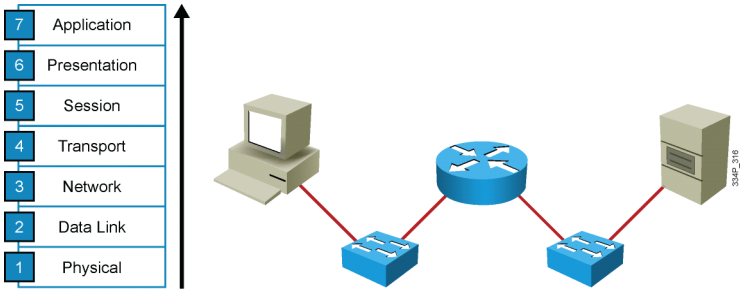
Therefore, the goal of this method is to find the highest OSI layer that is still working. All devices and processes that work on that layer or on the layers below are then eliminated from the scope of your problem. It might be clear that this method is most effective if the problem is on one of the higher OSI layers.

This is one of the most straightforward troubleshooting methods, because problems reported by users are typically defined as application layer problems, so starting the troubleshooting process at that layer is an obvious thing to do.

A drawback or impediment to this method is that you need to have access to the client's application layer software to initiate the troubleshooting process, and if the software is only installed on a small number of machines, it might be hard to test properly.

Bottom-Up Troubleshooting

This method follows the layers of the OSI model, starting from the physical layer and moving up to the application layer.



The bottom-up troubleshooting approach also uses the OSI model as the guiding principle, but this time you start on the physical layer and work your way up to the application layer. By verifying layer-by-layer that the network is operating correctly, you steadily eliminate more and more potential problem causes and narrow the scope of the potential problems.

A benefit of this method is that all the initial troubleshooting takes place on the network, so access to clients, servers, or applications is not necessary until a very late stage in the troubleshooting process.

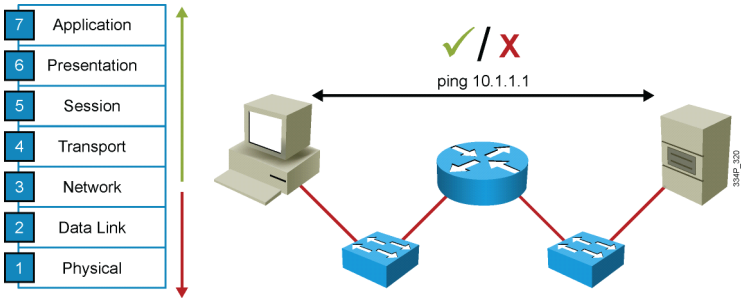
Also, the thoroughness and steady progress of this method will give you a relatively high probability of eventual success, or at the very least a decent reduction of the problem scope.

A disadvantage of this method is that, in large networks, it can be a very time-consuming process, because a lot of effort will be spent on gathering and analyzing data.

Therefore, the best use of this method is to first reduce the problem scope using a different strategy and then switch to this method for clearly bounded parts of the network topology.

Divide and Conquer

This method starts in the middle of the OSI model and moves up or down depending on results.

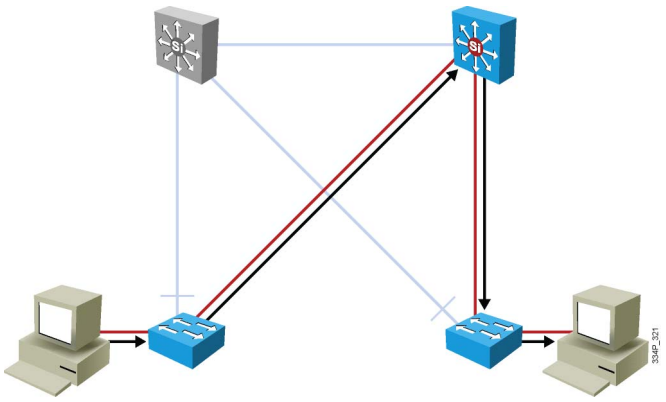


The divide-and-conquer troubleshooting method strikes a balance between the top-down and bottom-up troubleshooting approaches. If it is not clear if a top-down or bottom-up approach would be most effective, it can be helpful to start in the middle (typically the network layer) and run an end-to-end test such as ping. If this succeeds, you can assume that all lower layers are good, and you can start bottom-up troubleshooting from this point. Alternatively, if the test fails, you can start a top-down troubleshooting process from this point.

Whether the result of the initial test is positive or negative, this method usually results in a faster elimination of potential problems than what you would achieve by implementing a full top-down or bottom-up approach, which makes it a very effective troubleshooting strategy.

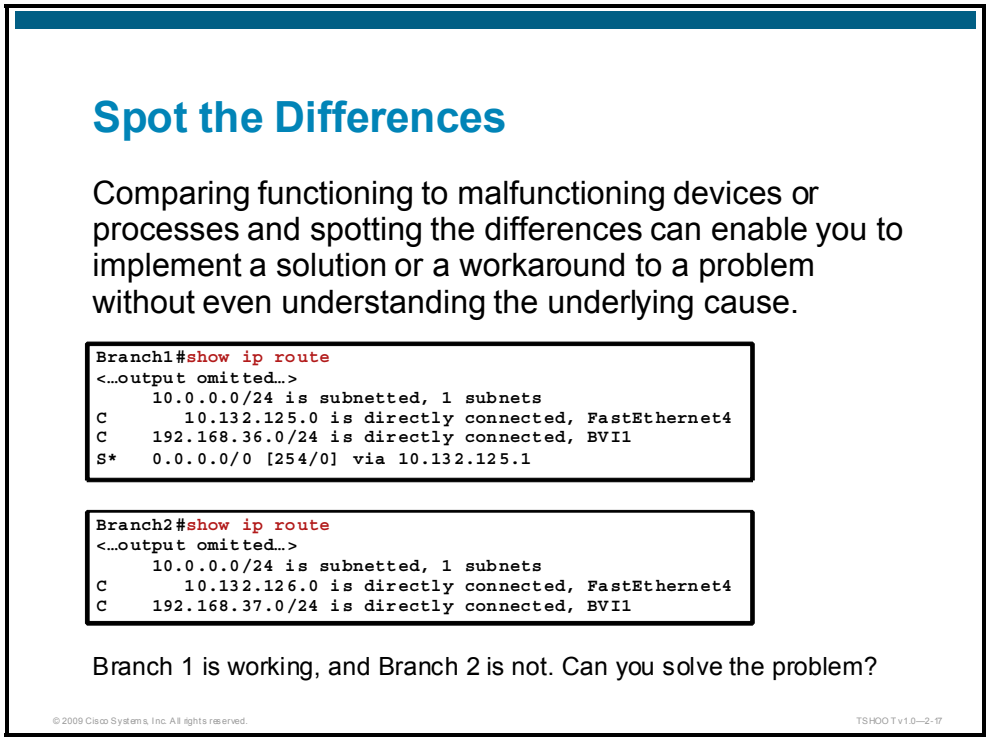
Follow the Path

Tracing the path of packets through the network eliminates irrelevant links and devices from the troubleshooting process.



Following the path is one of the most basic troubleshooting methods, and it usually complements one of the other troubleshooting methods, like the top-down or bottom-up approach.

Again, the objective of a troubleshooting method is to isolate the problem by eliminating potential problem areas from the scope of the troubleshooting process. By analyzing and verifying the path that packets and frames take through the network as they travel from the source to the destination, you can reduce the scope of your troubleshooting to just those links and devices that are actually in the forwarding path.



Another common troubleshooting method is spotting the differences. By comparing configurations, software versions, hardware or other device properties, links, or processes between working and nonworking situations and spotting significant differences between them, you might be able to resolve the problem by changing the nonoperational situation to be consistent with the working situation.

The biggest disadvantage of this method is that it can lead to a working situation, but not to an understanding of the root cause of the problem, so in some cases you cannot even be sure if you have implemented a real solution or only a workaround.

Here is an example. You are troubleshooting a connectivity problem with a branch office router. You have managed to narrow down the problem to some issue with the DSL link, but you cannot seem to find the cause. You notice that this router is an older type that was phased out in most of the other branch offices. You have one of the newer types of routers in the trunk of your car, because you plan to install that in another branch office next week. You decide to copy the configuration of the existing branch router to the newer router and replace it. Now everything starts to work as expected.

So what do you do? Do you consider the problem fixed? What was the root cause? What should you do with the old and new router now?

As you can see, this method has a number of drawbacks, but it is still a useful technique because you can use it even when you lack the background to troubleshoot based on knowledge of the technology.

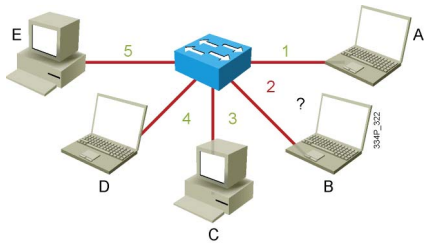
The effectiveness of this method depends on how easy it is to compare working and nonworking devices, situations, or processes. Having a good baseline of what constitutes normal behavior on the network makes it easier to spot abnormal behavior. Also, the use of consistent configuration templates makes it easier to spot the significant differences between functioning and malfunctioning devices. Consequently, the effectiveness of this method depends on the quality of the overall network maintenance process.

Like the follow-the-path method, this approach is best used as a supporting method in combination with other methods such as top-down or bottom-up troubleshooting.

Move the Problem

An elementary troubleshooting technique is to swap components and observe whether the problem stays, moves, or disappears.

- You install a couple of PCs, laptops, and a switch. Laptop B cannot establish a link. You suspect a hardware failure.
- How do you find out if the problem is the switch, the cable, or the laptop?



Another very elementary troubleshooting technique that can be used to isolate a problem is to physically swap components and see if the problem stays in place, moves with the component, or disappears entirely.

Look at the situation in the figure. One approach would be to start gathering data: Check settings on the laptop, examine settings and statistics on the switch, compare the settings on other laptops and switch ports, and so on.

However, what if you do not have the passwords for the PC and switch? The only data that you can gather is the status of the link LEDs on the switch and the laptops and PCs. What can you do? A common way to at least isolate the problem (if it is not solved outright) is to start swapping cables:

- Swap the cable connected to switch port 2 to one of the other ports, such as port 3. If the link LED on port 2 stays off and port 3 is still on, that means that the problem was not related to the PC or cable because they were both changed, and therefore the problem must be on the switch. If you have unused ports on the switch, you could try to change to one of those ports and see if that fixes the problem.
- If, after swapping the cable from port 2 to port 3, the situation changed and the LED on port 3 is now off, you can conclude that the problem must be with the cable or the PC, but not with the switch. So now you can swap the cables between the network interface card (NIC) of laptop B and the NIC of PC C. Again, see if the problem moves; if the LED on port 3 stays off and the LED on port 2 stays on, you must conclude that the problem is with the cable, because the PCs were swapped and the cable stayed in place. To verify this, you should swap the cables and confirm that the problem moves with the cable. Conversely, if the link status changed after swapping the cable from the NIC of B to the NIC of C, the conclusion must be that the problem is with the PC, not with the cable.

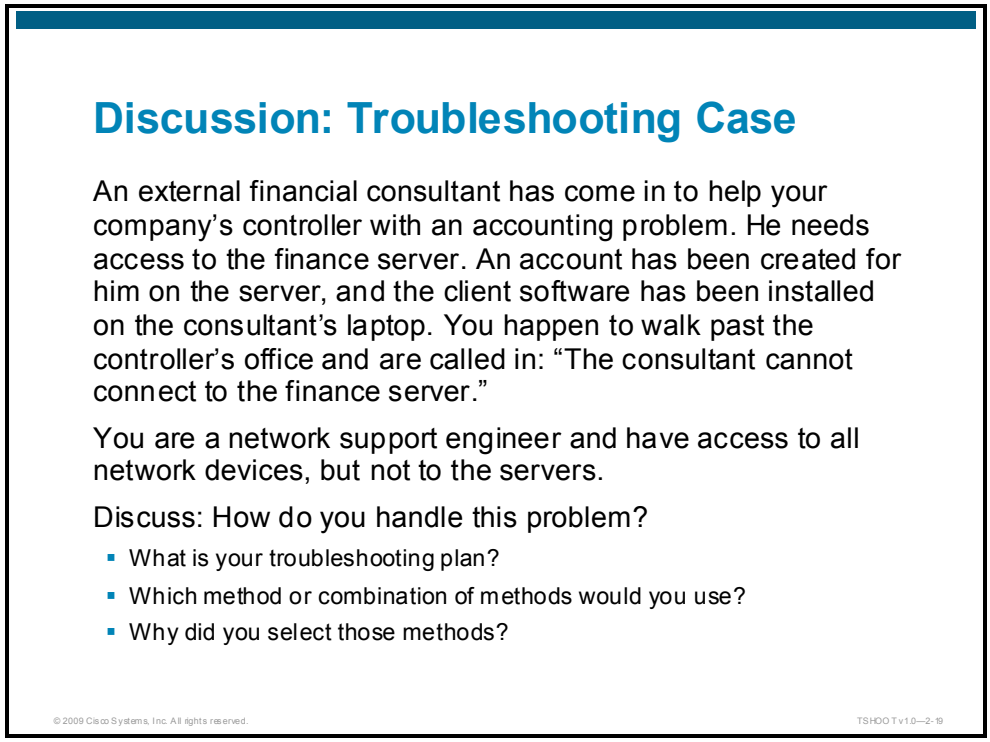
As you can see, this method allows you to isolate the problem, even if the information that you can gather is minimal, just by executing simple tests in a methodical way. So even if you do not solve the problem, you have scoped it to a single element, and further troubleshooting can now be focused on that element. (If you determined that the problem was with the cable, not the switch or PC, it is unnecessary to obtain the switch and PC password for further examination.)

What are the drawbacks of this method?

- You are isolating the problem to only a limited set of physical elements and not gaining any real insight into what is happening, because you are gathering only very limited, indirect information.
- This method assumes that the problem is with a single component. If the problem is with a particular combination of elements, you might not be able to isolate the problem correctly.

Troubleshooting Case Study

This topic describes a troubleshooting case and discusses which troubleshooting method would be most appropriate.



What are possible approaches to the case study described in the figure?

This case lends itself to many different approaches, but certain characteristics can help in deciding an appropriate method:

- You have access to the network devices, but not to the server. This implies that you will likely be able to handle layer 1–4 problems by yourself, but for layer 5–7 problems you will probably have to escalate to a different person.
- You have access to the client device, so it is possible to start your troubleshooting from it.
- The controller has the same software and access rights on his machine, so it is possible to compare between the two devices.

What are benefits and drawbacks of each of the possible methods?

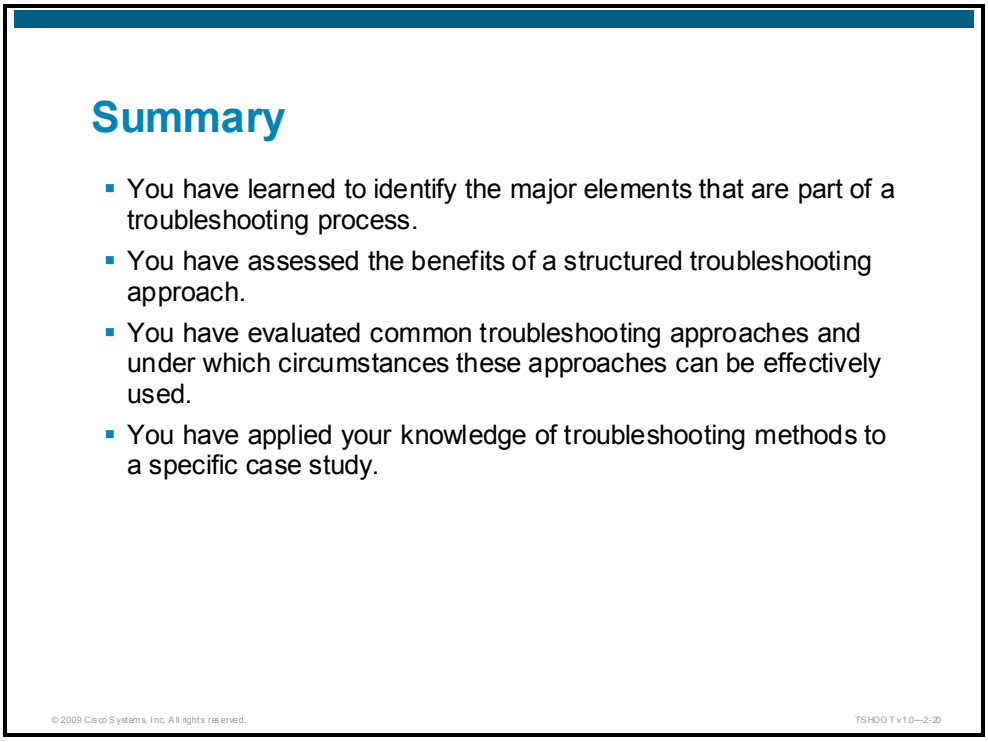
- **Top down:** You have the opportunity to start testing at the application layer. It is good troubleshooting practice to confirm the reported problem, so starting from the application layer is an obvious choice. The only possible drawback is that you will not discover simple problems, such as the cable being plugged into a wrong outlet, until later in the process.
- **Bottom up:** A full bottom-up check of the whole network is not a very useful approach because it takes too much time, and at this point there is no reason to assume that the network beyond the first access switch is causing the issue. You could consider starting with a bottom-up approach for the first stretch of the network, from the consultant's laptop to the access switch, to uncover potential cabling problems.

- **Divide and conquer:** A very viable approach. Ping from the consultant's laptop to the finance server. If that succeeds, you know that the problem is more likely to be with the application (although you have to consider potential firewall problems as well). If the ping fails, you are definitely dealing with a network issue and you are responsible for fixing it. The advantage of this method is that you can quickly decide on the scope of the problem and whether escalation would be necessary or not.
- **Follow the path:** Similar to the bottom-up approach, a full follow-the-path approach is not efficient under the circumstances, but tracing the cabling to the first switch can be a good start if it turns out that the link LED is off on the consultant's PC. This method might come into play after other techniques have been used to scope the problem down to where it seems to be an issue somewhere further up the path in the network.
- **Spot the differences:** You have access to both the controller's PC and the consultant's laptop, so this is a possible strategy. However, because these machines are not under the control of a single IT department, there will be many differences, and therefore it might be hard to spot the significant differences. This can be a useful method later in the process, after it has been determined that the problem is likely to be on the client.
- **Move the problem:** Using this approach alone is unlikely to solve the problem, but if following any of the other methods indicates a potential hardware issue between the consultant's PC and the access switch, this method may come into play. Also, as a first step you could consider swapping the cable connected to the consultant's laptop and the controller's PC to establish whether the problem is related to the laptop or not.

Many combinations of these different methods could be considered here. The most likely methods are top-down and divide-and-conquer, possibly switching to follow-the-path or spot-the-differences after the scope of the problem has been properly reduced. As an initial step in any approach, the move-the-problem method could be used to quickly separate client-related issues from network-related issues, or the bottom-up method could be used to verify the first stretch of cabling.

Summary

This topic summarizes the key points that were discussed in this lesson.



Lesson 3

Planning and Implementing Troubleshooting Procedures

Overview

It is impossible to write out a set of troubleshooting procedures that will solve any problem. The troubleshooting process can be guided by structured methods, but the exact steps that are taken at each point along the way cannot be prescribed because they depend on many different factors. Each network is different, each problem is different, and the skill set and experience of each engineer involved in a troubleshooting process are different.

However, to guarantee a certain level of consistency in the way that problems are diagnosed and solved in an organization, it is still important to evaluate the common subprocesses that are part of the troubleshooting process and to define procedures that outline how certain elements of these processes should be handled.

This lesson reviews the generic troubleshooting process and its subprocesses: Defining a problem, gathering information, analyzing the information, eliminating possible problem causes, formulating a hypothesis about the likely cause of the problem, testing that hypothesis, and solving the problem. This lesson analyzes the typical actions and decisions that are taken during each of those subprocesses and describes how these could be planned and implemented as troubleshooting procedures.

Objectives

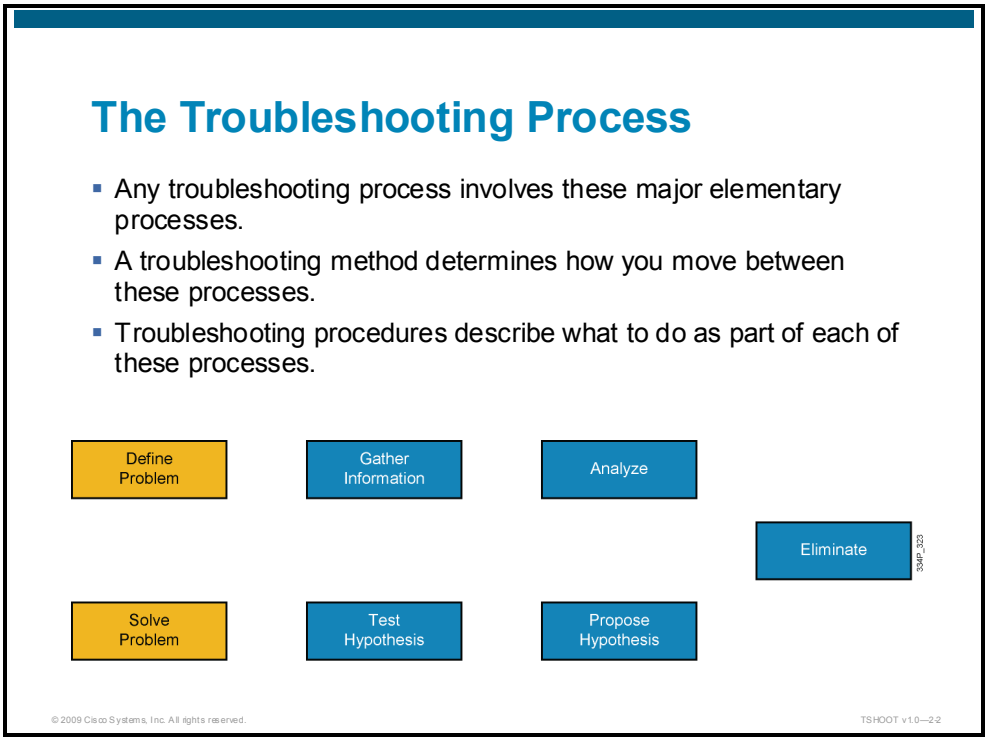
Upon completing this lesson, you will be able to plan and implement troubleshooting procedures as part of a structured troubleshooting methodology. This ability includes being able to meet these objectives:

- Identify the fundamental subprocesses of the generic troubleshooting process
- Formulate correct problem definitions and assign responsibilities
- Gather information in a structured manner
- Interpret and analyze the gathered information
- Isolate a problem through a process of elimination

- Formulate a hypothesis and evaluate the necessary actions to take after you have formulated a hypothesis
- Test a hypothesis and roll back if a hypothesis is not confirmed
- Integrate a solution into the existing network

Network Troubleshooting Procedures

This topic describes the components of the network troubleshooting process.



A network troubleshooting process can be reduced to a number of elementary subprocesses: Defining a problem, gathering information, analyzing the information, eliminating possible problem causes, formulating a hypothesis about the likely cause of the problem, testing that hypothesis, and eventually solving the problem.

These subprocesses are not strictly sequential in nature, and in many cases you will go through many of these subprocesses several times before eventually reaching the “solve the problem” phase.

A troubleshooting method provides a guiding principle that helps you move through these processes in a structured way.

There is no “recipe” for troubleshooting. Every problem is different, and it is impossible to create a script that will solve all possible problem scenarios. Troubleshooting is a skill that needs to be practiced to become effective at it. By practicing the different methods, you will become more effective at selecting the right methods for the problem at hand, gathering the most relevant information, and analyzing problems quickly and efficiently.

Although a complete script of troubleshooting actions and decisions cannot be created, you can have a closer look at each of the subprocesses and look at the best common practices that apply to each of those processes.

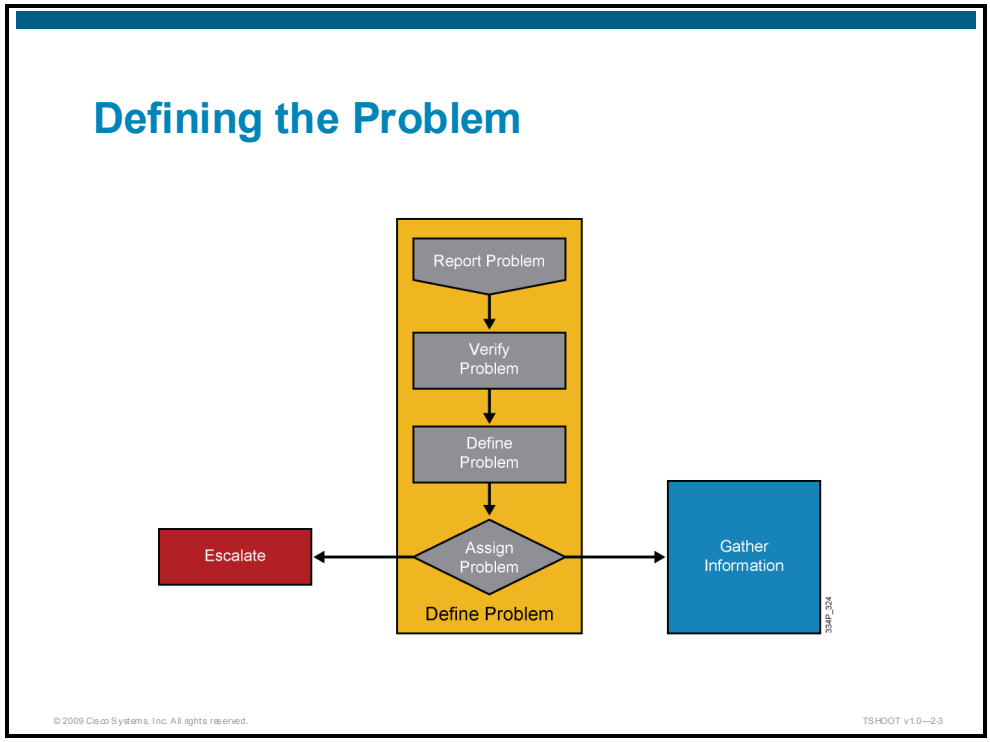
In this lesson, you will look at troubleshooting procedures and ask the following questions:

- What is the action plan for each of the elementary subprocesses or phases?
- What is it that you actually *do* during each of those subprocesses?
- What decisions do you need to make?
- What kind of support or resources do you need?
- What kind of communication needs to take place?
- How do you assign proper responsibilities?

Although the answers to these questions will be different for each organization, you can improve the consistency and effectiveness of the troubleshooting processes in your organization by planning, documenting, and implementing troubleshooting procedures.

Reporting, Defining, and Assigning Problems

This topic describes the typical actions and decisions that need to be taken after a problem is reported and before the diagnostic processes begin.



How Are Problems Reported?

Usually, you get a written or verbal description of a mix of different aspects of the problem. This description can include symptoms: “When I try to go to this location on the intranet, I get a page that says I don’t have permission”; a partial diagnosis: “The mail server isn’t working”; or consequences for the user: “I can’t file my expense report.”

To prevent wasting a lot of time based on false assumptions during the troubleshooting process, you must define the problem clearly. So the first step is to verify that the problem report is an accurate description of the problem and that the reported condition still exists. Sometimes a problem is intermittent in nature, and troubleshooting a problem when it is not occurring is nearly impossible.

A good problem description consists of accurate descriptions of the symptoms, not interpretations or conclusions. (Strictly speaking, consequences for the user are not part of the problem description itself but can help you assess the urgency of the issue.)

When a problem is reported as “The mail server isn’t working,” this is not a very useful description with which to start a troubleshooting process. What does it mean when someone says that the mail server is not working? Does it mean that the person just walked into the server room and noticed that all LEDs were off on the server? Does it mean that the server cannot be pinged from anywhere? Or does it mean that the mail client reports that the server cannot be reached?

In a situation like that, you want to get back to the person that reported the problem and ask what it means when the report says that the mail server is not working: What is not working? Can you tell me or show me what you are doing and how it is not working?

A better problem definition for the mail server problem could be: “When user X starts his email client he gets an error message saying that the client can not connect to the server. He can still access his network drives and browse the Internet.”

After you have clearly defined the problem, you have one more step to take before starting the actual troubleshooting process: you need to determine whether this problem is your responsibility or if it needs to be escalated to another department or person.

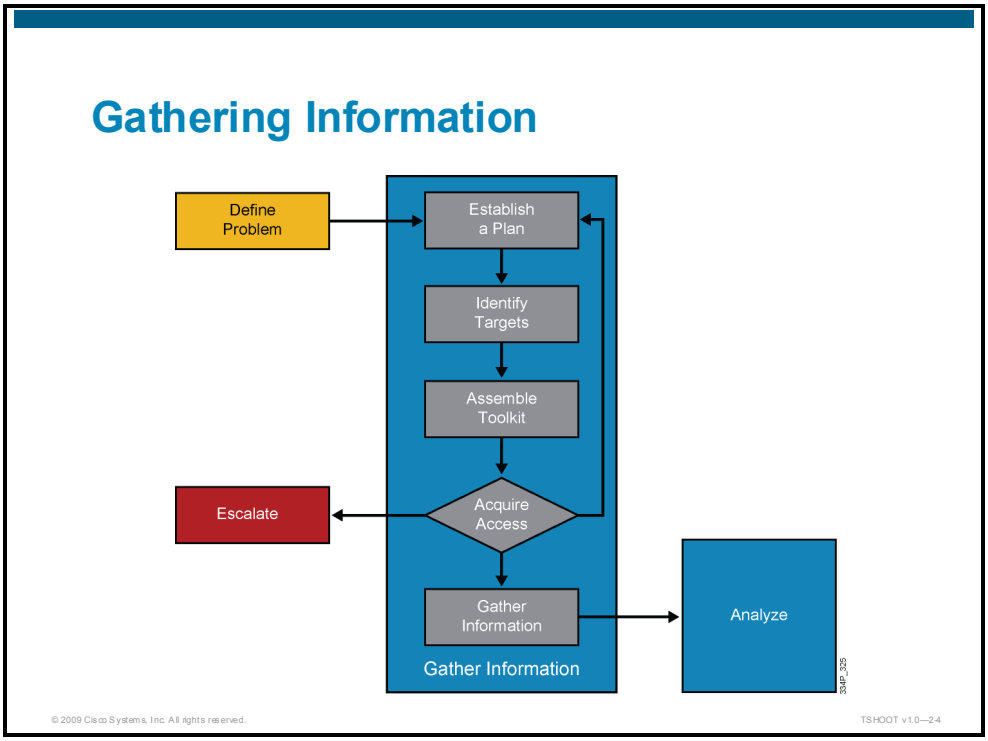
What if the problem definition is the following: “When user Y tries to access the corporate directory on the company intranet, she gets a message saying that permission is denied. She can access all other intranet pages.”

You are a network engineer and you do not have access to the servers. A separate department in your company manages the intranet servers. What do you do when this problem is reported to you as a network problem? Do you start troubleshooting or do you escalate it to the server department?

As part of the definition of your troubleshooting procedures, you need to answer these questions: Which types of problems are your responsibility to act on? What minimal actions do you need to take before you escalate a problem? How do you escalate a problem?

Gathering Relevant Information

This topic describes the decisions and actions that are part of the information-gathering process.



The first thing to do, after you have defined the problem and determined that it is your responsibility, is to start gathering more information about the problem. But what information do you need? Just randomly gathering information and hoping that at some point a hypothesis might materialize is not a very effective approach. before gathering information, you should select your initial troubleshooting method and develop an information-gathering plan. The information gathered will be used as the input for the troubleshooting approach that you have selected.

As part of this plan, you need to identify what the targets are for the information-gathering process. From which network devices, clients, or servers do you want to get information? What are the corresponding tools that you intend to use to gather that information?

When you have identified your targets and the tools that you intend to use to gather this information, you must acquire access to these targets. In many cases you might have access to these systems as a normal part of your job role, but in some cases, you might need to get information from systems that you cannot normally access. In such cases you might need to escalate the issue to a different department or person, either to obtain access or to get someone else to gather the information for you.

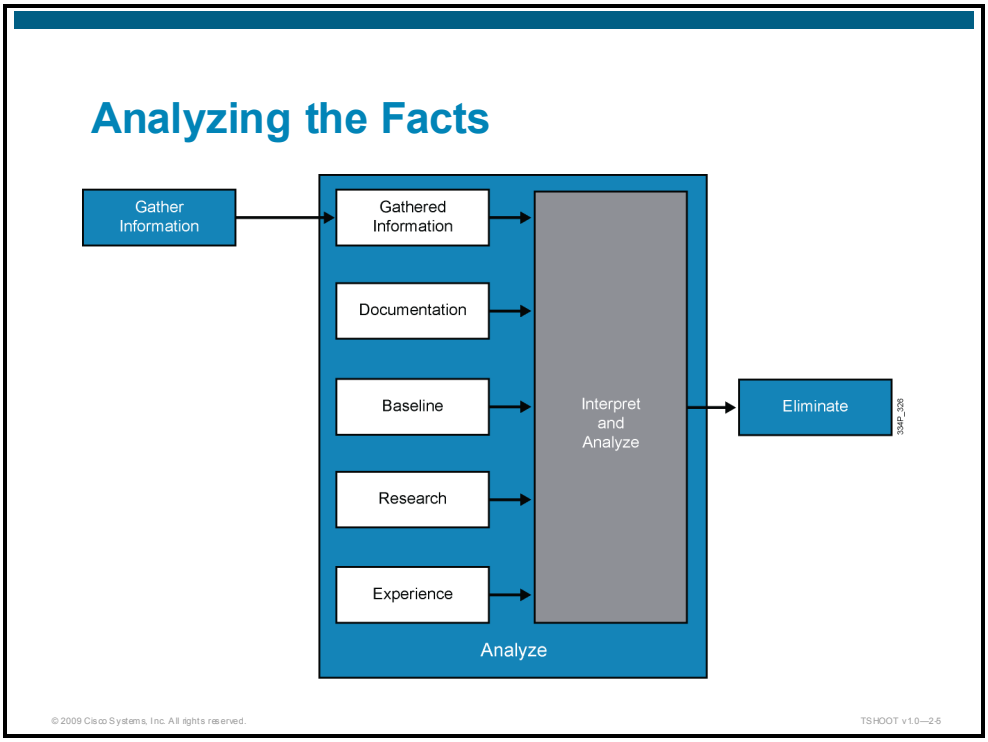
If the escalation process would slow the procedure down and the problem is urgent, you might want to reconsider the troubleshooting method that you selected and first try a method that uses different targets and would not require you to escalate.

For example, consider the following issue: Your company's sales manager reports that he cannot send or receive email from one of the branch offices, from which he is working this morning. This is urgent, because he needs to send out an important RFP response later this afternoon. Your initial thought is to start a top-down troubleshooting method by calling him and running through a couple of tests. However, he does not answer his phone, and when you check his calendar it turns out that he is in a meeting until 4:30 p.m. (1630). One of your colleagues in that branch office confirms that the sales manager is in a meeting, but left his laptop on his desk. The RFP response needs to be received by the customer before 5:00 p.m. (1700). It is 1:00 p.m. (1300) now. What do you do?

Even though a top-down troubleshooting approach might be the obvious choice, this means that you will need to wait until 4:30 p.m. (1630) before you can start troubleshooting, because you will not have access to the sales manager's laptop before then. This will put you under a lot of pressure to solve the problem in a half-hour. In this case you could consider a combination of the bottom-up and follow-the-path approaches to verify that there are no Layer 1–3 problems between the sales manager's laptop and the company's mail server. Even if you do not find an issue, you have eliminated many potential problem causes, and when you start a top-down approach at 4:30 p.m. (1630) you will be able to work more efficiently.

Analysis of the Gathered Information

This topic describes the processes and resources that are used during the interpretation and analysis of the gathered information.



After gathering information from various devices and tools, the next step is to interpret and analyze the information. In a way, this process is similar to detective work. On one hand, you are trying to discover clues that would point toward a particular root cause for the problem. On the other hand, you are trying to find proof that certain issues cannot be the cause of the problem, so that you can eventually eliminate those from the scope of your troubleshooting.

To interpret the raw information that you have gathered (for example, output from **show** commands and debugs, packet captures, and device logs) you might need to research commands, protocols, and technologies. You might also need to consult network documentation to be able to interpret the information in the context of the actual network implementation.

During the analysis of the gathered information, you are typically trying to determine two things: What is happening on the network, and what should be happening. If you discover differences between these two, they will usually give you clues about what is going wrong or at least a direction for further information gathering.

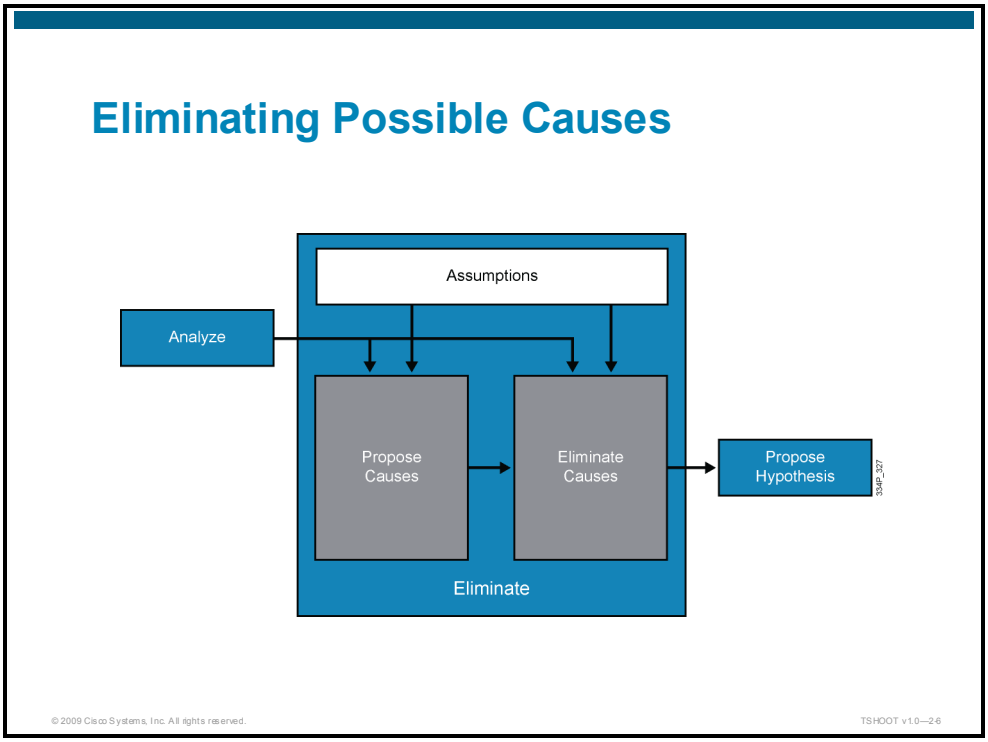
As mentioned, your image of what is actually happening will mostly be formed based on interpretation of the raw data, supported by research and documentation. But how do you know what should have been happening? For this, you need to have a good understanding of the operational processes of protocols and technologies. So if you are troubleshooting protocols and technologies that you are not very familiar with, you will need to invest some time in researching how they operate.

A second way to know what should have been happening under normal circumstances is by having a good baseline of the behavior of your network. If you know how your network performs and how things work under normal conditions, this will allow you to spot abnormalities in the behavior of the network and derive clues from those abnormalities. So as part of a proactive network maintenance approach, it is important to compile a baseline of your network's behavior that you can refer back to while troubleshooting.

Finally, this is the phase of the troubleshooting process where experience is important. An experienced network engineer needs to spend less time than an inexperienced engineer on researching processes, interpreting raw data, and distilling the relevant information.

Proposing and Eliminating Problem Causes

This topic describes the fundamental elements of the elimination process, which lies at the heart of all structured troubleshooting methods.



After you have interpreted and analyzed the information that you have gathered, you can start drawing conclusions from the results.

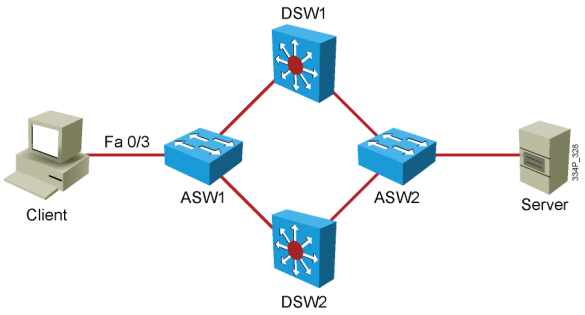
On one hand, you might have discovered clues that point toward certain issues that could be causing the problem, which add items to your list of potential problem causes (for instance, you might have observed high CPU loads on your multilayer switches, which might point to a bridging loop).

On the other hand, you might have observed behavior that rules out potential problem causes (for example, you successfully pinged the default gateway from the client, ruling out Layer 2 problems between the client and the default gateway).

Although the elimination process seems to be a rational, scientific procedure, you need to be aware that assumptions also play a role in this process, and you need to be willing to go back and reexamine and verify your assumptions. If you do not reexamine and verify, you might end up eliminating the actual root cause of the problem as a potential cause and therefore be unable to solve the problem.

Example: Elimination and Assumptions

- You are troubleshooting a connectivity problem between a client and a server.
- You verify the Layer 2 connection between the client and access switch ASW1 by use of the **show interface** and **show mac-address-table** commands.



Consider the following example: You are examining a connectivity problem between a client and a server. As part of a follow-the-path troubleshooting approach, you decide to verify the Layer 2 connectivity between the client and the access switch to which it connects.

Example: Elimination and Assumptions

The result of the **show interface** command is:

```
ASW1#show interface FastEthernet 0/3
FastEthernet0/3 is up, line protocol is up (connected)
  Hardware is Fast Ethernet, address is 001b.0c81.2703 (bia 001b.0c81.2703)
  MTU 1500 bytes, BW 100000 Kbit, DLY 100 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set
  Keepalive set (10 sec)
  Full-duplex, 100Mb/s, media type is 10/100BaseTX
  input flow-control is off, output flow-control is unsupported
  ARP type: ARPA, ARP Timeout 04:00:00
  Last input never, output 00:00:01, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  5 minute input rate 1000 bits/sec, 2 packets/sec
  5 minute output rate 3000 bits/sec, 5 packets/sec
     1672157 packets input, 130749802 bytes, 0 no buffer
     Received 6713 broadcasts (0 multicast)
     0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
     0 watchdog, 0 multicast, 0 pause input
     0 input packets with dribble condition detected
     3062213 packets output, 228795608 bytes, 0 underruns
     0 output errors, 0 collisions, 1 interface resets
     0 babbles, 0 late collision, 0 deferred
     0 lost carrier, 0 no carrier, 0 PAUSE output
     0 output buffer failures, 0 output buffers swapped out
```


You log into the access switch ASW1 and verify that the port connecting the client is up, that input and output packets are recorded on the port, and that there are no errors displayed in the packet statistics.

Example: Elimination and Assumptions

The result of the **show mac-address-table** command is:

```
ASW1#show mac-address-table interface FastEthernet 0/3
          Mac Address Table
-------------------------------------------

Vlan    Mac Address       Type        Ports
----    -----------       --------    -----
  10    000c.2905.19bd    DYNAMIC     Fa0/3
Total Mac Addresses for this criterion: 1
```

- The **show interface** command shows packets sent and received and no recorded input or output errors.
- The **show mac-address-table** command shows that the correct address was learned on the port.
- Conclusion: Layer 2 between switch ASW1 and the client is operational.

Next, you verify that the MAC address of the client was correctly learned on the port according to the MAC address table of the switch. Therefore, you conclude that Layer 2 is operational between the client and the switch, and you continue your troubleshooting approach, examining links further up the path.

Discussion: Elimination and Assumptions

- Is it correct to conclude that Layer 2 is operational between the client and switch ASW1?
- What are the assumptions that are made as part of the reasoning that leads to this conclusion?
- Are those assumptions reasonable?

Which assumptions are you making that you might need to reexamine later?

The first assumption is that the MAC address table entry and port statistics are current. How do you know that they are not old information? You might need to first clear the counters and MAC address and then verify that the counters are still increasing and that the MAC address is learned again.

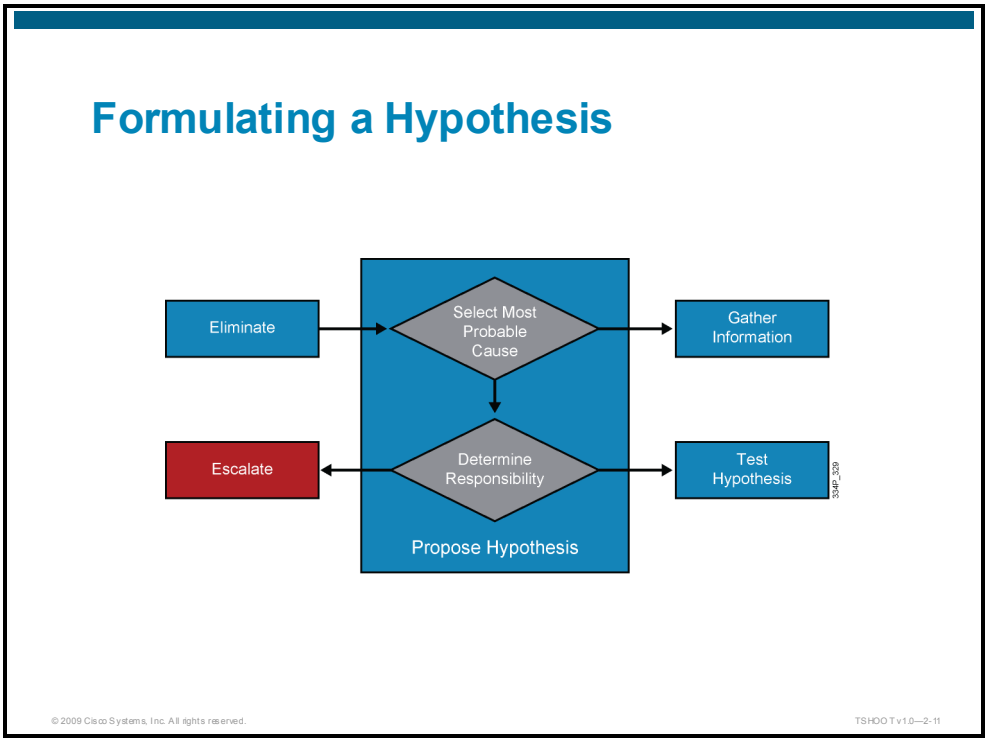
The second assumption is hidden in the conclusion that is being drawn: The only thing that you can really prove is that Layer 2 is operational from the client to the switch, because the switch has received frames from the client.

The fact that the interface is up and that frames were recorded as being sent by the switch does not give you definitive proof that the client has correctly received those frames. So even though it is reasonable to assume that if a link is operational on Layer 2 in one direction, it is also operational in the other direction, you still might need to reexamine this assumption later when it turns out that return packets from the server are not reaching the client.

Spotting faulty assumptions is one of the trickier aspects of troubleshooting, because in many cases, you are not consciously making those assumptions, but they are part of your thought processes. One of the things that can be very helpful in uncovering these hidden assumptions is to explain your reasoning to one of your colleagues or peers. Because different people think differently, they might be able to spot the hidden assumptions that you are making and help you uncover them.

Proposing Hypotheses

This topic describes how to propose a hypothesis about the possible problem cause and how to act on that hypothesis.



Having followed the process of proposing and eliminating potential problem causes, you should now have a list of those causes. Based on experience, you might even be able to assign a certain measure of probability to each of these potential causes.

If this list still has many different possible problem causes and none of them clearly stands out as the most likely cause, you might need to go back and gather more information and eliminate more problem causes before you can propose a good problem hypothesis.

After you have reduced the list of potential causes to a just a few (ideally just one) that emerge as the most likely causes at this point, select one of these as your problem hypothesis. You will assume that this is what is causing the problem and attempt to solve the problem based on this assumption.

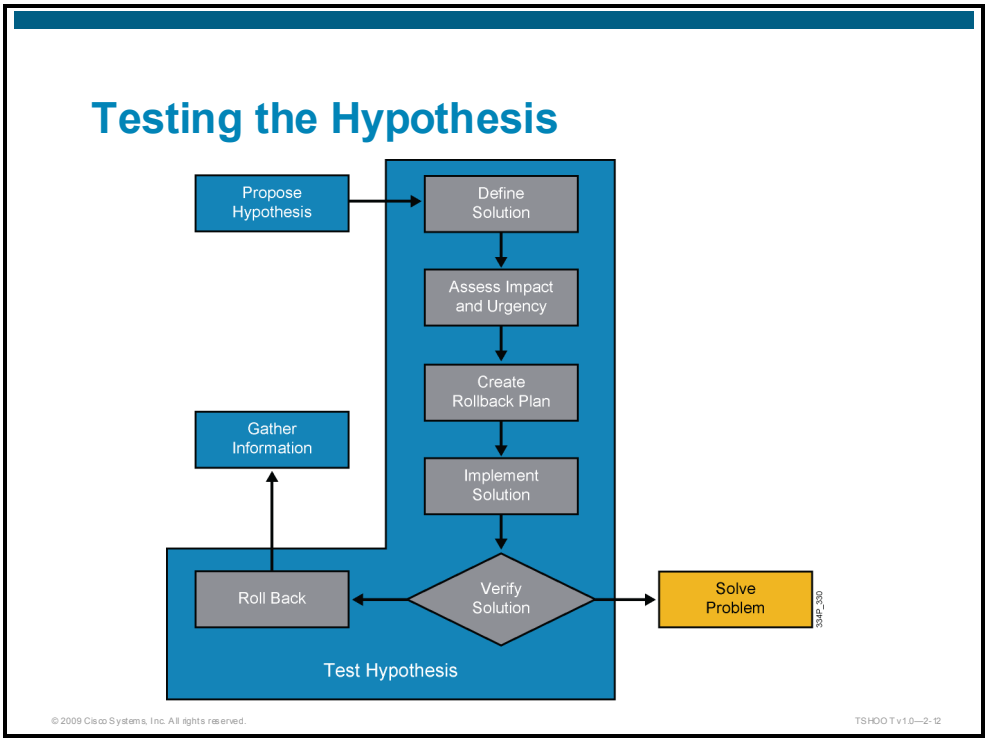
This means that you will now need to reassess whether the proposed problem cause is your responsibility. If the issue that you are proposing as your hypothesis causes the problem, is it then your responsibility to solve it, or will you need to escalate it to some other person or department?

If you decide to escalate the problem, you should ask yourself if this ends your involvement in the process, because the escalation by itself does not immediately solve the problem. How long do you think the other person will take to solve the problem? And how urgent is this problem? Can you afford to wait for someone else to fix it?

If you cannot solve the problem, but it is too urgent to wait for the problem to be solved through an escalation, you might need to come up with a workaround—a temporary fix that alleviates or remedies the symptoms experienced by the user, even if it does not address the root cause of the problem.

Testing and Verifying a Proposed Hypothesis

This topic describes the actions and decisions that should be considered while implementing a possible solution based on a proposed hypothesis.



After you have proposed a hypothesis about the cause of a problem, the next step is to come up with a possible solution (or workaround) to that problem and start planning to implement the solution.

Usually, implementing a possible solution involves making changes to the network. Therefore, if your organization has defined procedures for regular network maintenance, you can follow those regular change procedures.

The next step is to assess the impact of the change on the network and balance that against the urgency of the problem. If the urgency outweighs the impact and you decide to go ahead with the change, it is important to ensure that you have a way to revert to the original situation after you make the change. Even though you have determined that your hypothesis is the most likely cause of the problem and your solution is intended to fix it, you can never be entirely sure that your proposed solution will actually solve the problem. If it does not solve it, you need to have a way to undo your changes and revert to the original situation.

After you have created a rollback plan, you implement your proposed solution (or workaround) according to your organization's change procedures and you verify that it has solved the problem. You should verify that the change that you made did what you expected it to do, that it solved the root cause and problem symptoms (or in case of a workaround, just the symptoms), and that it has not introduced any new problems.

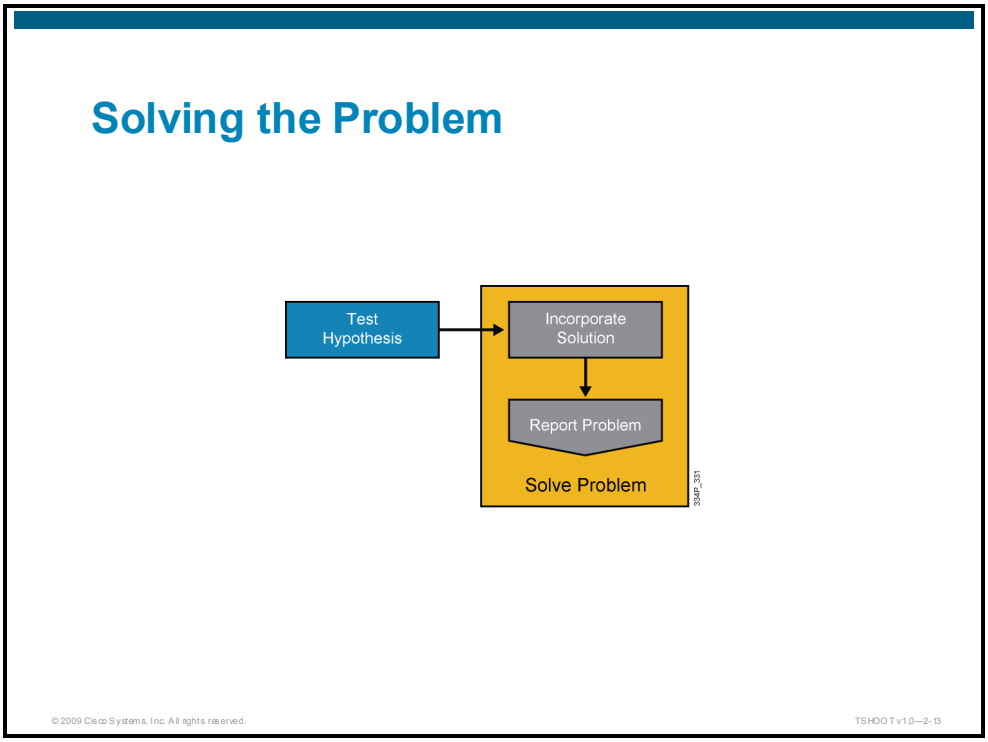
After you verify that your hypothesis was correct and the problem symptoms have disappeared, you can move to the final stage of the troubleshooting process.

But what do you do if it turns out that the problem was not fixed, the symptoms did not disappear, or new problems have been introduced by the change that you made?

In those cases, you should execute your rollback plan, revert to the original situation, and resume the troubleshooting process. If the problem was not fixed, a good first step is to determine whether the root cause hypothesis was invalid or if it was simply the proposed solution that did not work.

Wrapping Up the Process

This topic describes the final steps that you need to take after you have confirmed your hypothesis and solved the problem.



When you have confirmed your hypothesis and verified that the symptoms have disappeared, you have essentially solved the problem. All you need to do now is make sure that the changes that you made are integrated into the regular implementation of the network and that maintenance procedures associated with the changes that you made are executed.

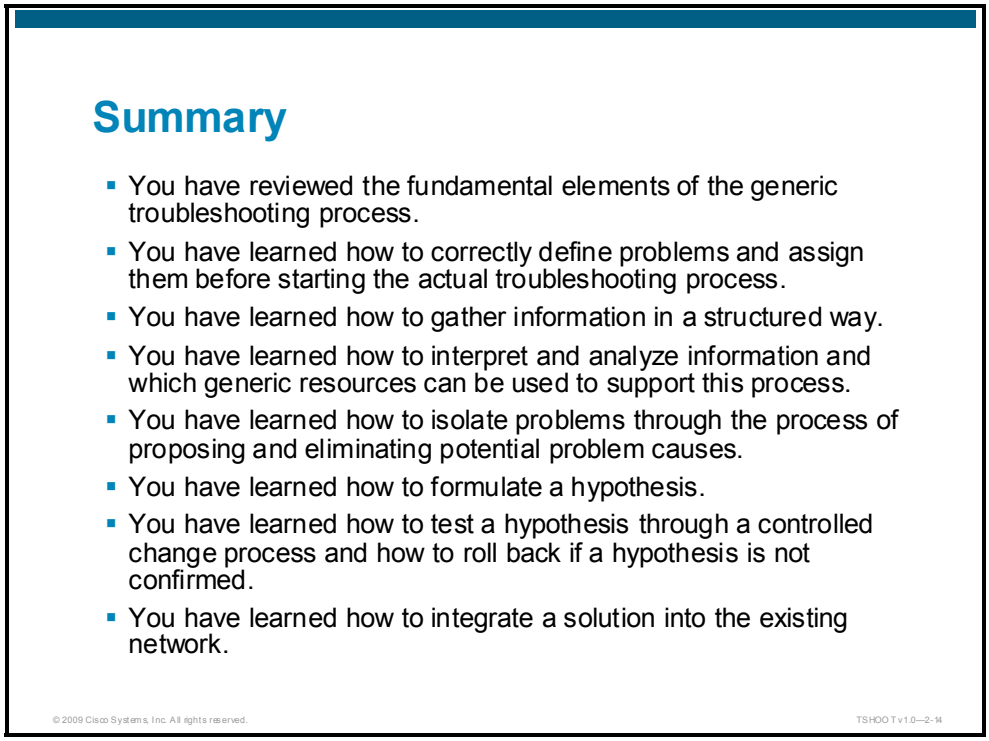
You will need to create backups of any elements that were changed, such as device configurations and software. You will need to document all changes to make sure that the network documentation still accurately describes the current state of the network. In addition, you will need to perform any other actions that are prescribed by the regular change control procedures that are in use on the network.

Finally, you will need to communicate that the problem has been resolved. At a minimum, you will need to communicate back to the original user that reported the problem, but if you have involved others as part of an escalation process, you will need to update them as well.

For any of the processes and procedures described in this lesson, each organization will need to make its own choices in how much of these procedures should be described, formalized, and followed. However, for anyone involved in troubleshooting it is beneficial to review these processes, compare them to their own troubleshooting habits, and see what they could benefit from.

Summary

This topic summarizes the key points that were discussed in this lesson.



Lesson 4

Integrating Troubleshooting into the Network Maintenance Process

Overview

Troubleshooting is a process that takes place as part of many different network maintenance tasks. During the verification phase, it could be necessary to troubleshoot as part of the implementation of new devices or new technology, in response to problems reported by users, or while executing regular network maintenance tasks like software upgrades. Therefore, troubleshooting processes should be considered and integrated into network maintenance procedures, and vice versa. When troubleshooting procedures and maintenance procedures are properly aligned, the effectiveness of the overall network maintenance process increases. In this lesson, the interactions between troubleshooting and maintenance processes are analyzed to learn how they can be most efficiently integrated.

Objectives

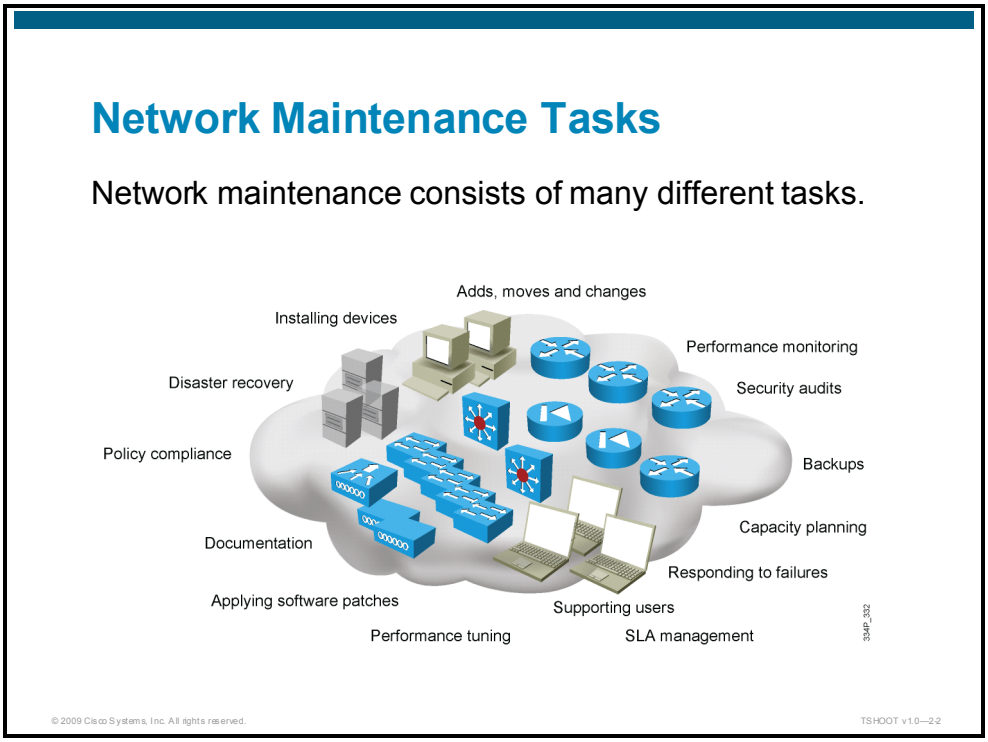
Upon completing this lesson, you will be able to plan and implement troubleshooting and network maintenance procedures that effectively support each other. This ability includes being able to meet these objectives:

- Evaluate the benefits gained by aligning troubleshooting procedures to network maintenance procedures
- Create and update documentation as part of routine maintenance to support the troubleshooting process and routinely update documentation as part of the troubleshooting process to keep the documentation accurate and up-to-date
- Identify abnormal network behavior through the comparison of actual behavior to a baseline created as part of the network maintenance process

- Implement communication processes that increase the effectiveness of the troubleshooting process
- Implement change procedures that are flexible enough to support the changes that need to be made during troubleshooting, but also controlled enough so that changes are integrated into the standard maintenance and documentation procedures

Troubleshooting and Network Maintenance

This topic describes the interaction between network troubleshooting and maintenance processes.

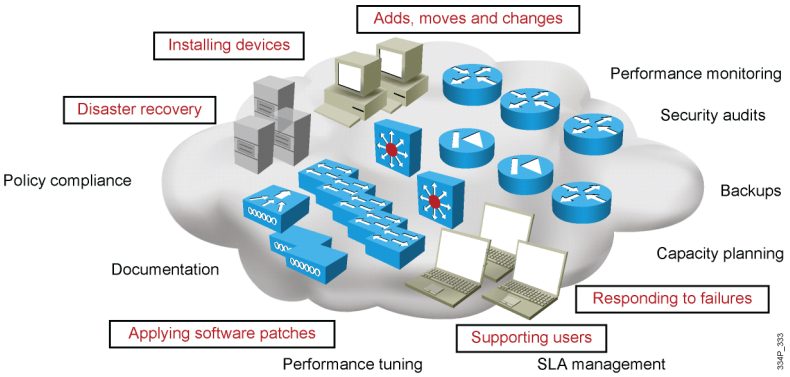


Network maintenance involves many different tasks. For some of these tasks, such as supporting users, responding to network failures, and disaster recovery, troubleshooting is a major component of the tasks. Tasks that do not revolve around fault management, such as adding or replacing equipment, moving servers and users, and performing software upgrades, also regularly include troubleshooting processes.

In that sense, troubleshooting should not be seen as a standalone process, but as an essential skill that plays an important role in many different types of network maintenance tasks.

Troubleshooting as Part of Maintenance

Troubleshooting is an essential skill that plays an important role in many different types of network maintenance tasks.



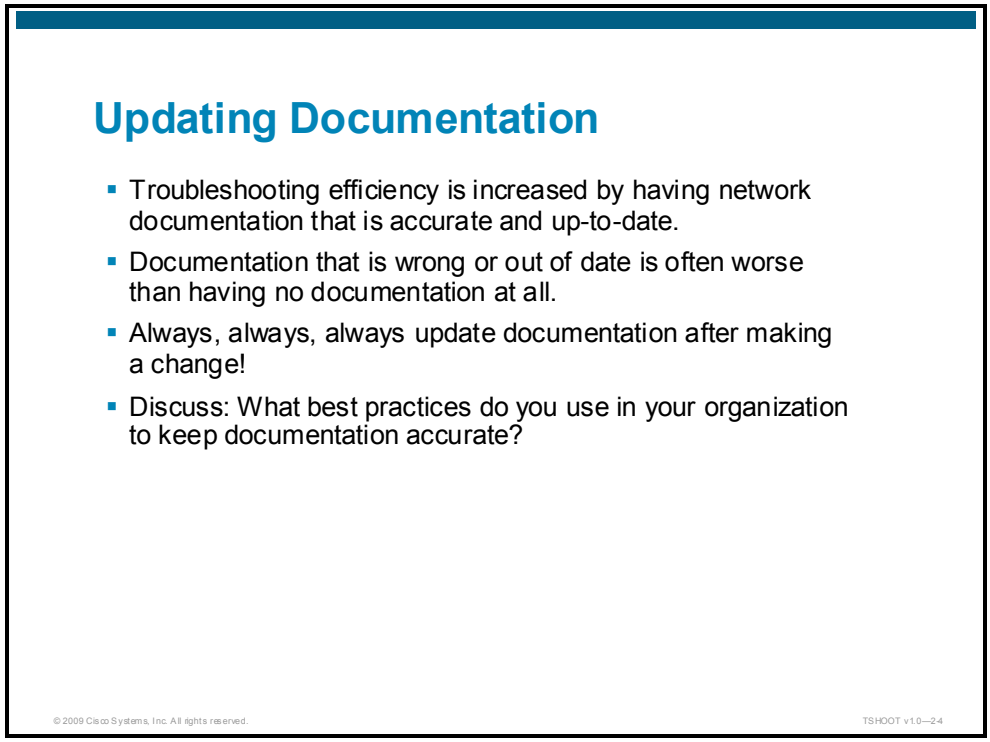
To troubleshoot effectively, you are dependent on many processes and resources that are part of the network maintenance process: You need to have access to documentation that is up-to-date and accurate. You are dependent on good backup and restore procedures to be able to roll back changes if they do not resolve the problem that you are troubleshooting. You need to have a good baseline of the behavior of the network so you know which conditions are supposed to be normal on your network and what kind of behavior is considered abnormal. You need to have access to logs that are properly time stamped to find out when particular events happened.

So in many ways, the quality of your troubleshooting processes is very dependent on the quality of your network maintenance processes. Therefore, it makes sense to plan and implement troubleshooting activities as part of the overall network maintenance process and to make sure that troubleshooting processes and maintenance processes are aligned and support each other, increasing the effectiveness of both sets of processes.

This lesson highlights a number of the major aspects of these processes that should be synchronized between the maintenance and troubleshooting processes. This lesson also shows how each set of processes can support the other.

Documentation

This topic describes the interaction between the maintenance and troubleshooting processes related to network documentation.



Having accurate and current network documentation can tremendously increase the speed and effectiveness of troubleshooting processes. Having good network diagrams can especially help in quickly isolating problems to a particular part of the network, tracing the flow of traffic, and verifying connections between devices. Having a good IP address and patching administration is also valuable and can save a lot of time in simply trying to locate devices and IP addresses.

However, documentation that is wrong or outdated is often worse than no documentation at all. At least when you have no documentation, you know that you need to gather information yourself and that you will still be working from correct information. It might take more time than simply referring to available documentation, but the conclusions that you draw are still correct. However, if the documentation that you have is inaccurate or out of date, you could start working with incorrect information and you could draw the wrong conclusions. In such a situation you can lose a lot of time before you discover that you are working from the wrong assumptions.

Interestingly enough, the troubleshooting process is often its own worst enemy when it comes to keeping the documentation up-to-date and accurate. Although everyone who is involved in network maintenance agrees that updating documentation is an essential network maintenance task, they all recognize that in the heat of the moment, when you are troubleshooting a problem that is affecting network connectivity for many users, documenting the process and the changes that you are making is one of the last things on your mind.

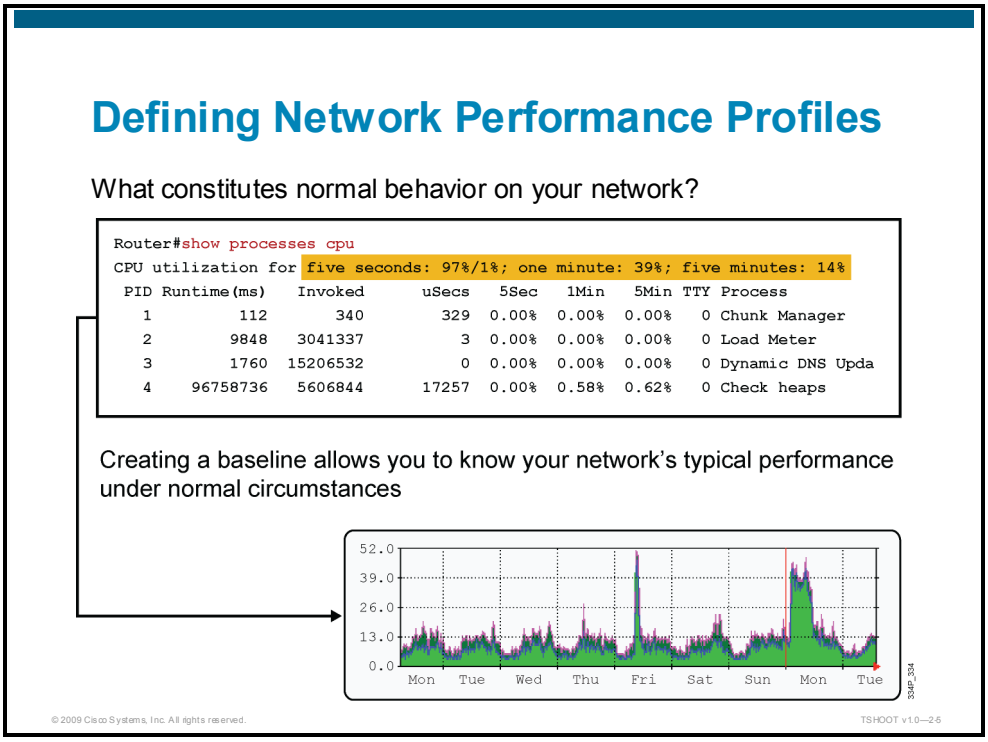
There are several things you can do to alleviate this problem. First, you should make sure that any changes you make during troubleshooting are handled in accordance with normal change procedures, if not during the troubleshooting process itself, then at least after the fact. You might loosen the requirements concerning authorization and scheduling of changes during major failures, but you need to ensure that after the problem has been solved, or a workaround has been implemented to restore connectivity, you always go through the standard administrative processes such as updating the documentation. In fact, knowing that you will need to update the documentation afterwards serves as an incentive to keep at least a minimal log of the changes that you make during troubleshooting.

Another way to keep your documentation accurate is to assume that people will forget to update the documentation and schedule regular checks of the documentation. However, verifying documentation by hand is tedious work, so you probably will want to implement an automated system for that. For example, for configuration changes, you could implement a system that downloads all device configurations on a regular basis and compares the configuration to the last version to spot any differences.

There are also various Cisco IOS features (in particular the Configuration Archive and Rollback feature and the Embedded Event Manager) that can be leveraged to create automatic configuration backups, to log configuration commands to a syslog server, or even to send configuration differences via email.

Creating a Baseline

This topic describes the use of a network baseline in troubleshooting processes.



An essential troubleshooting technique is to compare what is happening on the network to what is expected or to what is normal on the network. Whenever you spot abnormal behavior in an area of the network that is experiencing problems, there is a good chance that it is related to the problems. It could be the cause of the problem or it could be another symptom that could help point toward the underlying cause (of course, it could also be a coincidence and not be related to the problem at all). Either way, it is always worth investigating abnormal behavior to find out if it is related to the problem.

But how do you know what is normal on your network? Consider the following example: You are troubleshooting an application problem, and while following the path between the client and the server you notice that one of the routers is also a bit slow in its responses to your commands. You use a **show processes cpu** command and notice that the average CPU load over the past five seconds was 97 percent, and over the last one minute was around 39 percent. You wonder if this might be causing the problem.

This could be an important clue that is worth investigating, or it could merely be that your router regularly runs at 40 to 50 percent CPU and it is not related to this problem at all. In this case you could potentially waste a lot of time trying to find the cause for the high CPU load, which is entirely unrelated to the problem at hand.

The only way to know what is normal for your network is to measure the network's behavior continuously. Knowing what to measure is different for each network. In general, the more you know, the better; but obviously you need to balance this against the effort and cost involved in implementing and maintaining a performance management system.

Useful statistics to gather and create a baseline for are the following:

- **Basic performance statistics such as the interface load for critical network links and the CPU load and memory usage of routers and switches:** These values can be polled on a regular basis through the use of Simple Network Management Protocol (SNMP) and then collected and graphed.
- **Accounting of network traffic:** Remote Monitoring (RMON), Network-Based Application Recognition (NBAR), and NetFlow statistics can be used to profile different types of traffic on the network.
- **Measurements of network performance characteristics:** The IP service level agreement (SLA) feature in Cisco IOS Software can be used to measure critical performance indicators such as delay and jitter across the network infrastructure.

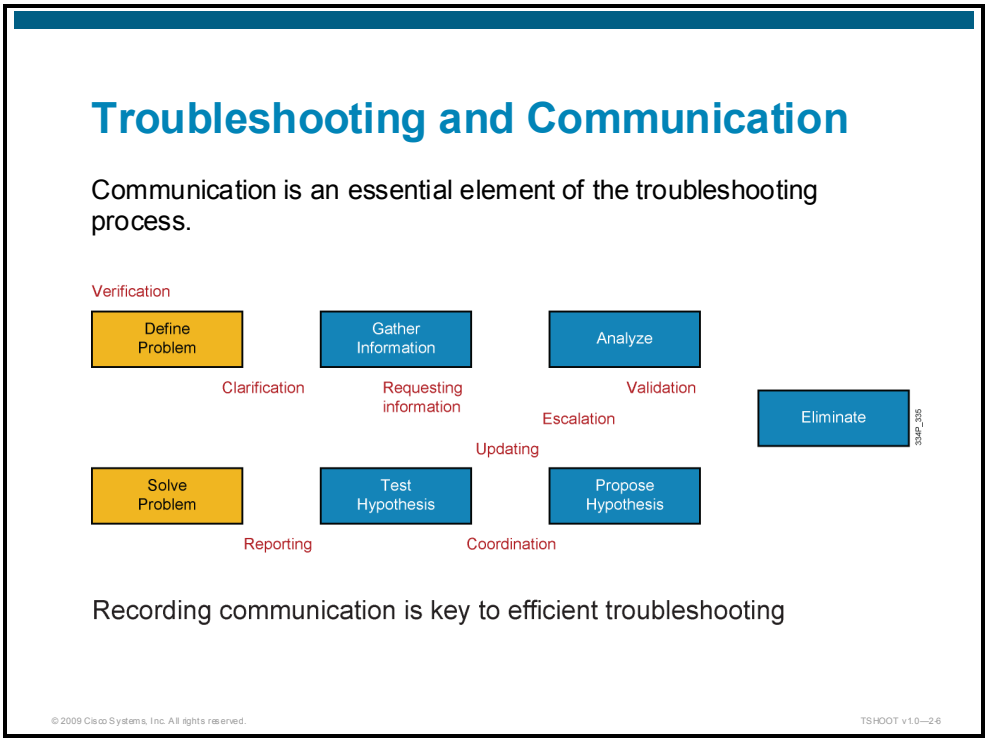
These baseline measurements are useful for troubleshooting, and they can also be used as input for capacity planning, network usage accounting, and SLA monitoring.

There is clearly a synergy between gathering traffic and performance statistics as part of regular network maintenance, and using those statistics as a baseline during troubleshooting. Also, after you have the infrastructure in place to collect, analyze, and graph network statistics, you can leverage this infrastructure to troubleshoot specific performance problems.

For example, if you notice that a router crashes once every two weeks and you suspect a memory leak as the cause of this issue, you could decide to graph the memory usage of the router for a certain period of time to see if you can find a correlation between the crashes and the memory usage.

Communication

This topic describes the importance of communication during troubleshooting processes and how to integrate communication in the normal network maintenance processes.



Communication is an essential part of the troubleshooting process. The following is a review of the elementary phases of the troubleshooting process:

- **Report the problem:** Clearly, this phase is all about communication. When someone reports a problem, the problem definition is often too vague to act on it immediately. You need to verify the problem and gather as much information as you can about the symptoms from the person that reported the problem. Asking good questions and carefully listening to the answers is essential in this phase. Some of the questions that you can ask during this phase are What do you mean exactly when you say that something is failing? Did you make any changes before the problem started? Did you notice anything special before this problem started? When did it last work? Has it ever worked?
- **Gather information:** During this phase of the process, you will often depend on other engineers or users to gather information for you. You might need to obtain information that is contained in server or application logs, configurations of devices that you do not manage, information about outages from a service provider, or information from users in different locations to compare against the location that is experiencing the problem. Clearly communicating what information you need and how that information can be obtained is essential to obtain the information that you require.
- **Analyze and eliminate:** In itself, interpretation and analysis is mostly a solitary process, but there are still some communication aspects to this phase. First, you cannot be experienced in every aspect of networking, so if you find that you are having trouble interpreting certain results or if you lack knowledge about certain processes, you can ask specialists on your team to help you out. Also, there is always a chance that you are misinterpreting results, misreading information, making wrong assumptions, or having other flaws in your interpretation and analysis. A different viewpoint can often help in

these situations, so discussing your reasoning and results with teammates to validate your assumptions and conclusions can be very helpful when you are stuck.

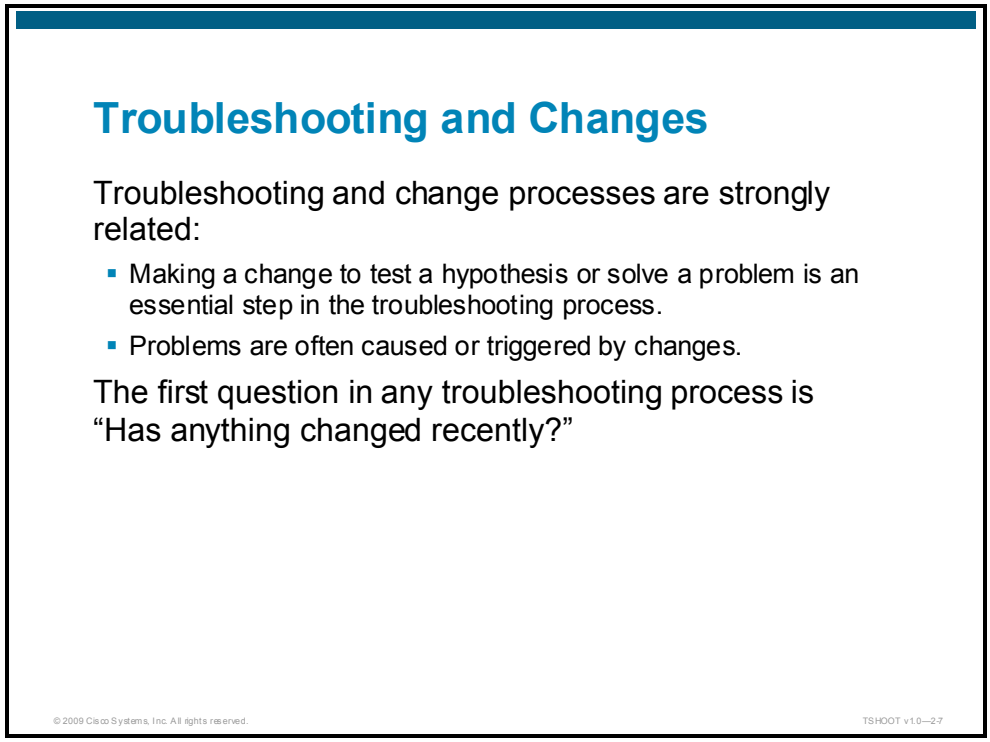
- **Propose and test a hypothesis:** Most of the time, testing a hypothesis involves making changes to the network. These changes might be disruptive, and users might be impacted. Even if you have decided that the urgency of the problem outweighs the impact and that the change needs to be made, you should still communicate clearly what you are doing and why you are doing it. Even if your changes will not have a major impact on the users or the business, you should still always coordinate and communicate any changes that you are making. Especially when other team members are working on the same problem, you need to ensure that you are not both making changes. Any results from the elimination process might be rendered invalid if a change was made during the information-gathering phase and you were not aware of it. Also, if two changes are made in quick succession and it turns out that the problem was resolved, you do not know which of the two changes actually fixed it. So does that mean that you cannot be working on the same problem as a team? No, but you need to adhere to certain rules. Having multiple people working on different parts of the network, gathering information in parallel or pursuing different strategies, can help in finding the cause faster. Especially during a major disaster when every minute counts, the extra speed that you can gain by working in parallel is very valuable. However, any changes or other disruptive actions should be carefully coordinated and communicated.
- **Solve the problem:** Clearly, this phase involves some communication as well. You will need to report back to the person who originally reported the problem that it has been resolved. Also, you will need to communicate this to the other people that were involved during the process. And finally, you will need to go through any communication that is involved in the normal change processes, to ensure that the changes that you made are properly integrated in the standard network maintenance processes.

Especially when you are troubleshooting as a team, it should be possible to hand over the results of the process up to a certain point and escalate it to someone else. Common reasons for this could be that you know that you do not have sufficient knowledge and skills and you want to escalate it to a specialist or more senior engineer, or that you are working in shifts and you need to hand over the problem when your shift ends. Handing over the results of the process requires not only clear communication of the results of your process like gathered information and conclusions that you have drawn, but it also includes any communication that has been going on up to this point. This is where an issue tracking or trouble ticketing system can be of tremendous value, especially if it integrates well with other means of communication such as email.

Finally, another communication process that requires some attention is how to communicate the progress of your troubleshooting process to the business. When you are experiencing a major outage, there will usually be a barrage of questions from business managers and users: What are you doing to repair this issue? How long will it take before it is solved? Can you implement any workarounds? What do you need to fix this? All are reasonable questions, but many of those questions cannot be answered until the cause of the problem is found. At the same time, the time spent communicating about the process is taken away from the actual troubleshooting effort itself. Therefore, it is worthwhile to streamline this process; for example, by having one of the senior team members act as a conduit for all communication. Any questions from the business are routed to this person and any updates and changes are communicated to this person, who then updates the main stakeholders. This way, the engineers that are actually working on the problem can work with a minimal amount of distraction.

Change Control

This topic describes the interaction between troubleshooting processes and change control procedures.



Change control is one of the most fundamental processes in network maintenance. By strictly controlling when changes are made, and defining what type of authorization is required and what actions need to be taken as part of that process, you can reduce the frequency and duration of unplanned outages and thereby increase the overall uptime of your network.

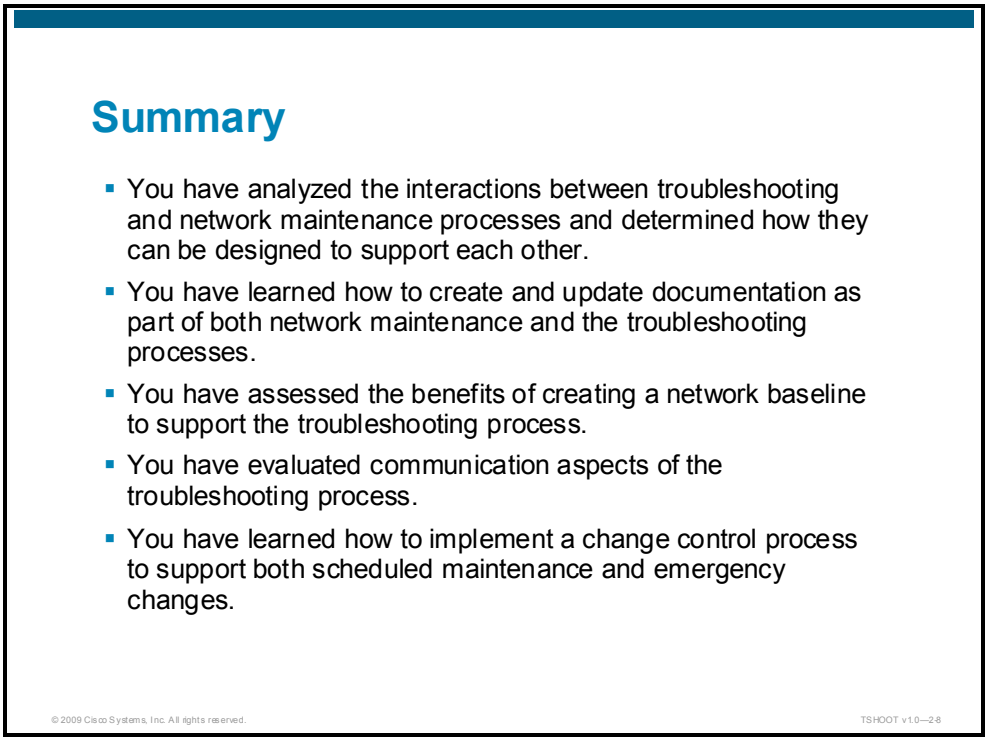
How do changes made as part of troubleshooting fit into the overall change processes? Essentially, there is little difference between making a change as part of the maintenance process and making a change as part of troubleshooting. Most of the actions that you take are the same: you implement the change, verify that it achieved the desired results, roll back if it did not, back up the changed configurations or software, document your changes, and so on. The main differences between regular changes and emergency changes regard the authorization that is required to make a change, and the scheduling of the change. When you are creating change control procedures, there is always an aspect of balancing urgency, necessity, impact, and risk. The outcome of this assessment determines whether a change can be executed immediately or needs to be scheduled at a later time and also the level of authorization that will be required to make the change. This means that changes that are made to resolve problems or outages should be designed into the overall change procedures.

The troubleshooting process can benefit greatly from having well-defined and well-documented change processes. It is uncommon for devices or links to simply fail from one moment to the next. In most cases, problems are triggered or caused by some sort of change. This can be a simple direct change, such as changing a cable or reconfiguring a setting, but it could also be more subtle, like a change in traffic patterns due to the outbreak of a new worm or virus. Also, a problem could be caused by a combination of changes, where the first change is the underlying cause of the problem, but the problem is not triggered until you make another change. For example, consider a situation where someone accidentally erases the router software from its flash. This does not cause the router to fail immediately, because it is running the software from its RAM. However, if that router reboots due to a short power failure a month later, it will not boot, because of the missing software in flash. In this example the underlying cause of the failure is the erased software, but the trigger is the power failure. This type of problem is very hard to catch, and only in the most tightly controlled environments will you be able to find the underlying cause or prevent this type of problem. (For example, in this case a log of all privileged EXEC commands executed on the router might have revealed that the software had been erased.)

What does this mean for troubleshooting processes? It means that, when a problem is reported, the first question you should always ask yourself is “Has anything been changed?” The more thoroughly changes are reported and documented on your network, the higher the chance that you will be able to get an answer to that question and find the cause or trigger for the problem.

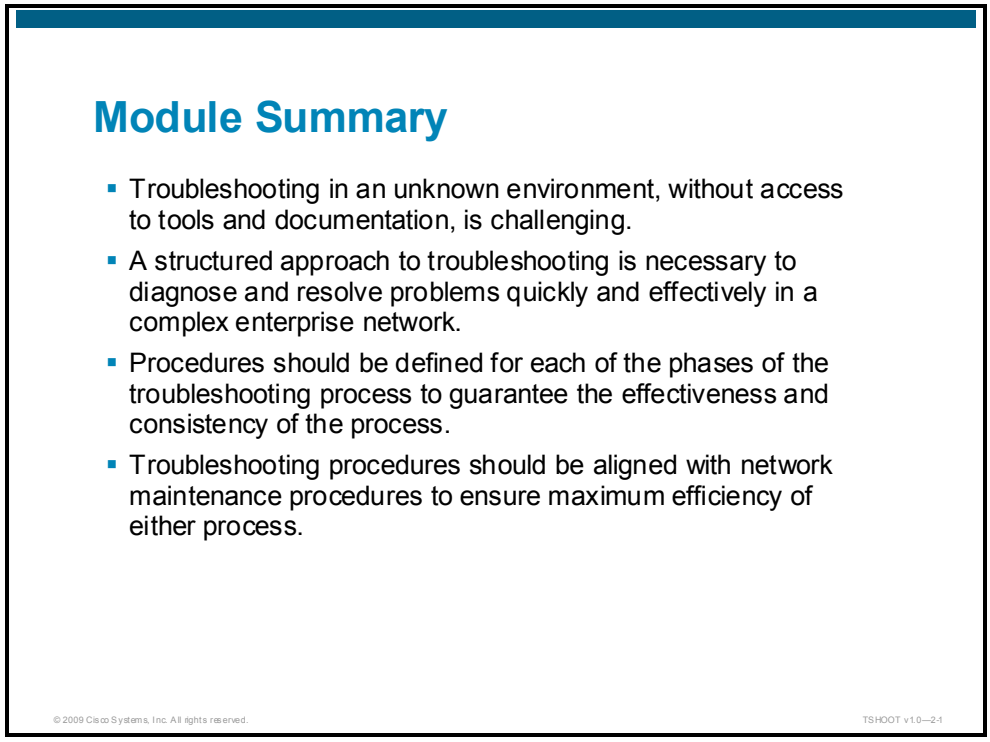
Summary

This topic summarizes the key points that were discussed in this lesson.



Module Summary

This topic summarizes the key points that were discussed in this module.



Troubleshooting in an unknown environment without established procedures and access to documentation and tools is challenging. Random troubleshooting without good coordination can lead to more problems and a lengthy recovery time.

Applying a structured methodology to network troubleshooting optimizes the process and reduces network downtime. A troubleshooting plan can be defined by combining common troubleshooting methods, such as top-down, bottom-up, divide-and-conquer, follow-the-path, spot-the-differences, and move-the-problem, and selecting the methods that are most suited to the particular problem.

The troubleshooting process consists of defining the problem, gathering information, analyzing information, eliminating causes, proposing hypotheses, testing hypotheses, and solving the problem. In this module, each of those phases has been analyzed to identify the major elements that should be considered when defining troubleshooting procedures.

Troubleshooting and maintenance procedures should be aligned with regards to aspects such as documentation, creating network performance baselines, communication, and change control.

Module Self-Check

Use the questions here to review what you learned in this module. The correct answers and solutions are found in the Module Self-Check Answer Key.

Q1) Which three processes are subprocesses or phases of a troubleshooting process? (Choose three.) (Source: Applying Troubleshooting Methodologies)

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

Q2) Which four approaches are valid troubleshooting methods? (Choose four.) (Source: Applying Troubleshooting Methodologies)

A) top down
B) bottom up
C) follow the path
D) seek and destroy
E) divide and conquer

Q3) Which three troubleshooting approaches use the OSI reference model as a guiding principle? (Choose three.) (Source: Applying Troubleshooting Methodologies)

A) top down
B) bottom up
C) follow the path
D) spot the differences
E) move the problem
F) divide and conquer

Q4) Which troubleshooting method is most appropriate to find a bad cable? (Source: Applying Troubleshooting Methodologies)

A) top down
B) bottom up
C) follow the path
D) spot the differences
E) move the-problem
F) divide and conquer

Q5) Which conditions make troubleshooting by spotting the differences more effective? (Source: Applying Troubleshooting Methodologies)

_____
_____
_____
_____
_____

Q6) Which problem has a clear problem definition? (Source: Planning and Implementing Troubleshooting Procedures)

A) I cannot order printer cartridges because the Internet is down.
B) My email does not work.
C) I cannot log in to the network because the server is down.
D) When I try to access http://www.cisco.com, my Internet Explorer says that it cannot display the web page.

Q7) Which two resources will help in interpreting and analyzing information gathered during troubleshooting? (Choose two.) (Source: Planning and Implementing Troubleshooting Procedures)

A) documentation
B) network baseline
C) packet sniffers
D) assumptions

Q8) Which four steps are part of testing a hypothesis? (Choose four.) (Source: Planning and Implementing Troubleshooting Procedures)

A) Define a solution.
B) Create a rollback plan.
C) Implement the solution.
D) Define the problem.
E) Assess impact and urgency.

Q9) During which three of the troubleshooting phases could it be necessary to escalate a problem to a different department? (Choose three.) (Source: Planning and Implementing Troubleshooting Procedures)

A) defining the problem
B) gathering information
C) analyzing the facts
D) eliminating possible causes
E) formulating a hypothesis
F) solving the problem

Q10) Which technology can be deployed to measure critical network performance indicators such as delay and jitter? (Source: Integrating Troubleshooting into the Network Maintenance Process)

A) NetFlow
B) RMON
C) IP SLA
D) NBAR

Q11) Which phase of the troubleshooting process does not have communication as a major component? (Source: Integrating Troubleshooting into the Network Maintenance Process)

A) defining the problem
B) solving the problem
C) eliminating possible causes
D) gathering information

Module Self-Check Answer Key

Q1) A, B, D

Q2) A, B, C, E

Q3) A, B, F

Q4) E

Q5) Troubleshooting by spotting differences is made easier by having consistent templates for configurations and a baseline of network behavior under normal circumstances.

Q6) D

Q7) A, B

Q8) A, B, C, E

Q9) A, B, E

Q10) C

Q11) C

Module 3

Maintenance and Troubleshooting Tools and Applications

Overview

Troubleshooting can be a time-consuming process. During outages, lost time comes at a price in terms of lost productivity or lost revenue.

Any tool that can help shorten the time to diagnose and resolve problems quickly starts to yield return on the initial investment of acquiring and implementing the tool. Optimizing the use of the tools built into Cisco IOS Software should therefore be a first priority for any engineer that is involved in troubleshooting processes. Cisco IOS Software supports many technologies and protocols that can be leveraged in combination with specialized tools and applications to support troubleshooting and maintenance processes, such as fault notification and baseline creation.

This module reviews the built-in Cisco IOS tools and commands. It also assesses and evaluates their usage, features, and technologies that can be used in combination with specialized tools and applications.

Module Objectives

Upon completing this module, you will be able to select tools that best support specific troubleshooting and maintenance processes in large, complex enterprise networks. This ability includes being able to meet these objectives:

- Use the Cisco IOS commands to selectively gather information in support of basic diagnostic processes.
- Identify the tools that are commonly used for specific maintenance and troubleshooting processes and prepare the infrastructure for their use.
- Survey an unknown network environment and assemble a maintenance and troubleshooting toolkit.

Lesson 1

Assembling a Basic Diagnostic Toolkit Using Cisco IOS Software

Overview

A large amount of the total time spent on troubleshooting processes is spent on gathering information. One of the challenges during this process is how to gather only the relevant information and not at the same time collect and process a lot of information that is irrelevant and distracts from the information that you are really interested in.

For the elementary diagnostic processes that will be repeated time and again during the troubleshooting process, it is worthwhile to invest some time in learning how to apply the basic tools that support these processes in an efficient and effective manner.

In this lesson, the basic Cisco IOS commands used to troubleshoot connectivity problems will be reviewed and the inherent capabilities of the Cisco IOS Software to filter and select information are explained in the context of a number of practical examples.

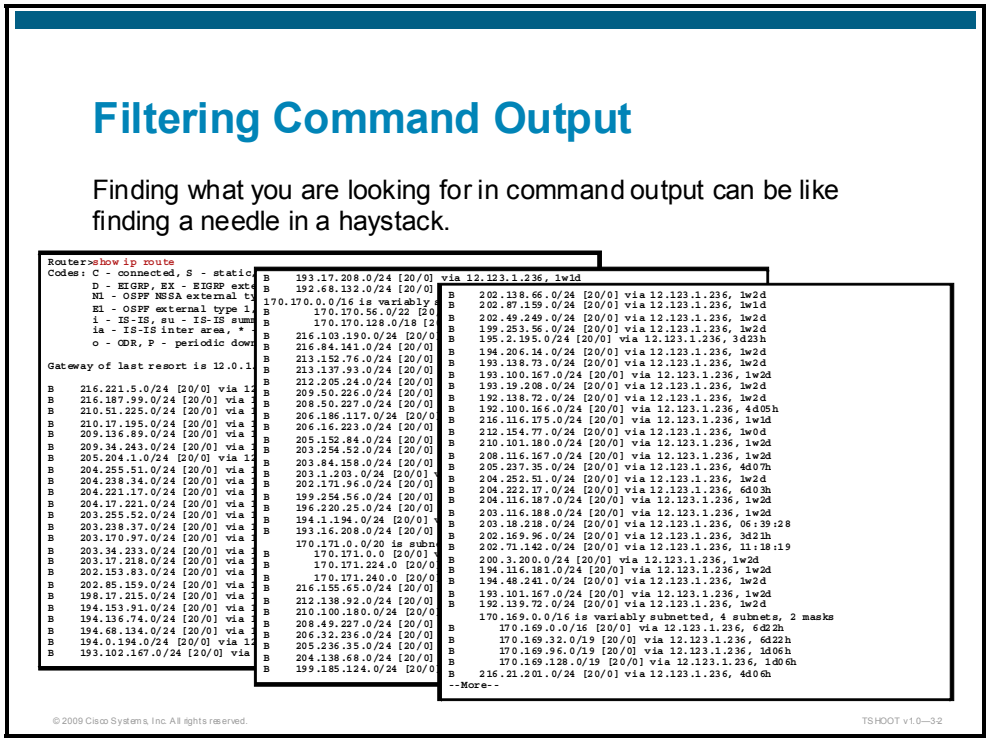
Objectives

Upon completing this lesson, you will be able to use Cisco IOS commands to selectively gather information in support of basic diagnostic processes. This ability includes being able to meet these objectives:

- Apply filtering to Cisco IOS commands to select relevant output
- Test network connectivity using Cisco IOS commands
- Diagnose basic hardware-related problems

Selecting and Filtering Information

This topic describes how to apply filtering to Cisco IOS commands to optimize the gathering of information using Cisco IOS Software.

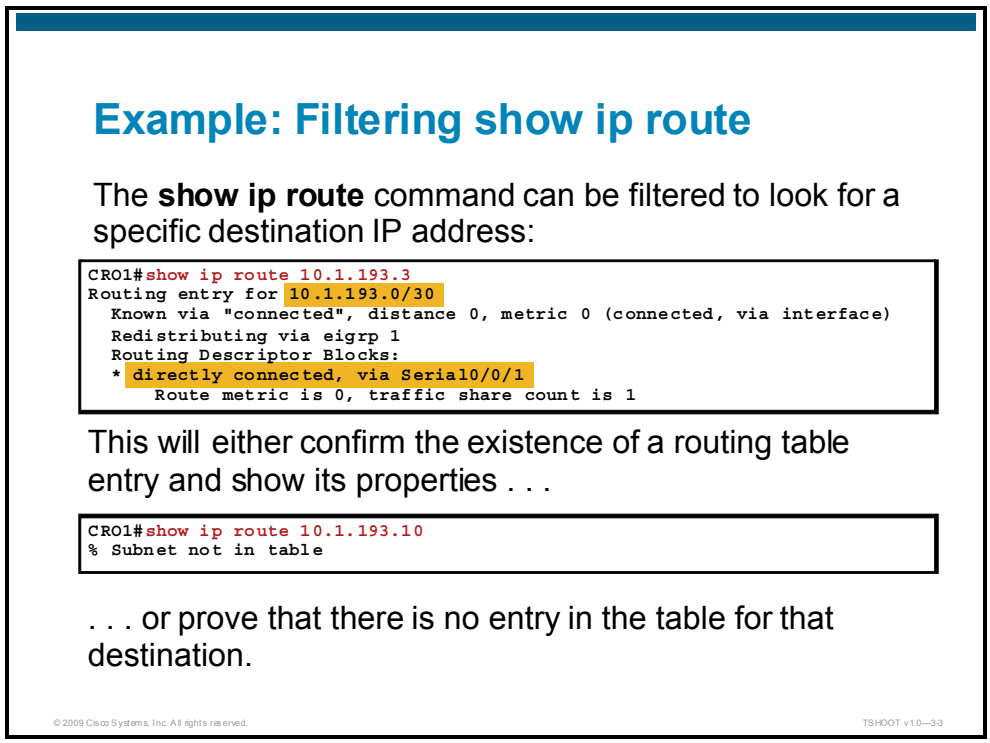


When you are troubleshooting, you are often looking for very specific information: You want to know if there is a route in the routing table that matches destination IP address 10.1.193.3. Or you want to verify if MAC address 0019.55df.ad66 is being learned on interface Fast Ethernet 0/15. Or maybe you want to check the percentage of CPU time being used by the “IP Input” process.

In the previous three examples you could use the **show ip route** command to display the routing table, the **show mac address-table** command to examine the MAC address table, and the **show processes cpu** command to check the CPU utilization for all processes on the router or switch. However, the routing table and MAC address table can contain thousands to tens of thousands of routes or addresses. Scanning through these tables to find a particular entry only really works in very small environments; and even then, there is a high probability of overlooking things. Also, what if you cannot find the entry that you are looking for? Is it really not in the table? Or did you simply not see it? Should you repeat the command? Furthermore, if you executed the command three times in a row and you still did not see the entry, does that prove that the entry is not in the table?

The list of processes on a router or switch is not hundreds or thousands of entries long; so in this case, you could just look through the full list and find the “IP Input” process. But what if you want to repeat the command every minute to see how the CPU usage of the “IP Input” process changes over time?

In all these cases, you are interested in only a small subset of the information that the commands can provide. The Cisco IOS Software provides options to limit the output that is being displayed and to select only the information that you are interested in.



For example, to limit the output of the **show ip route** command, you can enter a specific IP address on the command line as an option. This causes the router to execute a routing table lookup for that specific IP address and see if it finds a match. If it finds a match in the routing table, it displays the corresponding entry with all its details. If it does not find a match in the routing table, it displays a message saying "% Subnet not in table."

Caution	Be aware that the default route is excluded from this process. If a default route is present, but no other route matches the IP address, the router will display "% Subnet not in table" even if packets for that destination IP address would be forwarded according to the default route entry.

Example: Filtering show ip route (Cont.)

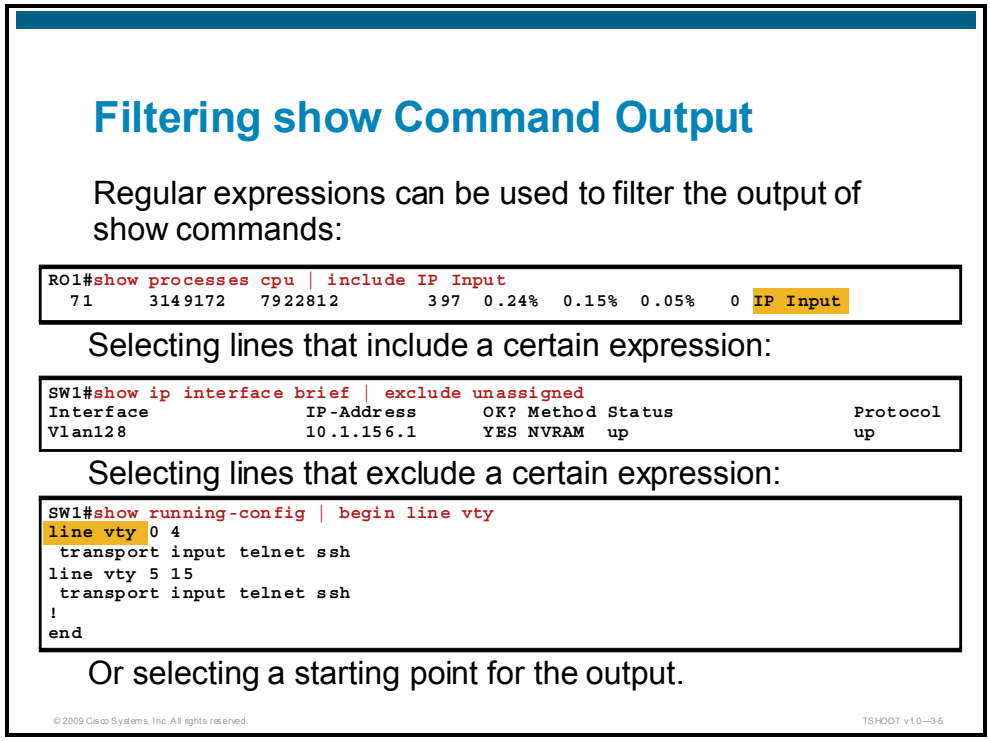
Alternatively, you can limit the output of the command to the subset of prefixes that fall within a particular block.

```
CR01#show ip route 10.1.193.0 255.255.255.0 longer-prefixes
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

     10.0.0.0/8 is variably subnetted, 46 subnets, 6 masks
C       10.1.193.2/32 is directly connected, Serial0/0/1
C       10.1.193.0/30 is directly connected, Serial0/0/1
D       10.1.193.6/32 [90/20517120] via 10.1.192.9, 2d01h, FastEthernet0/1
                      [90/20517120] via 10.1.192.1, 2d01h, FastEthernet0/0
D       10.1.193.4/30 [90/20517120] via 10.1.192.9, 2d01h, FastEthernet0/1
                      [90/20517120] via 10.1.192.1, 2d01h, FastEthernet0/0
D       10.1.193.5/32 [90/41024000] via 10.1.194.6, 2d01h, Serial0/0/0.122
```


Another way that you can limit the output from the **show ip route** command to a particular subset of routing information is by specifying a prefix followed by the **longer-prefixes** keyword. The router will then list all subnets that fall within the prefix that you have specified (including that prefix itself, if it is listed in the routing table).

If the network that you are troubleshooting has a good hierarchical IP numbering plan, this command option can be very useful to select all routes from a particular part of the network. For example, all subnets from a particular branch office, all VLANs in a particular building, or all subnets in the data center can be selected if there is an appropriate summary for these blocks.

For many commands, if you find yourself skimming through multiple pages of output, looking for that one line, it is worth researching the command syntax to see if you can limit the command output to find the specific items that you are interested in.



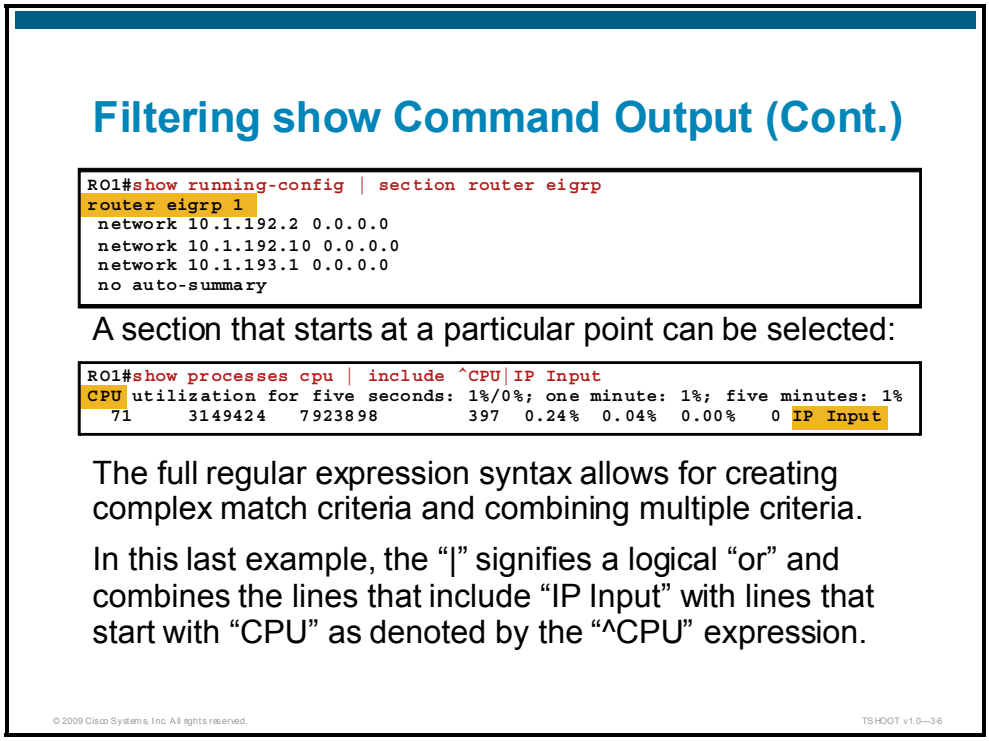
Unfortunately, **show** commands do not always have the specific options that allow you to select exactly what you need. In that case, you can still use a more generic way of filtering. The output of Cisco IOS **show** commands can be filtered by appending a pipe character "|" to the show command followed by one of the keywords **include, exclude**, or **begin** and then a regular expression.

Regular expressions are patterns that can be used to match strings in a piece of text. In its simplest form, you can use regular expressions to match words or text fragments in a line of text, but full use of the regular expression syntax allows you to build complex expressions that match very specific text patterns.

In the examples in the figure, we simply use a word or line fragment to select the lines in the text that we are interested in. So for instance, if you are interested only in the "IP Input" process in the output of the **show processes cpu** command, you can select only the lines that contain the string "IP Input" by using the command **show processes cpu | include IP Input**.

Similarly, you can exclude lines from the output through use of the **| exclude** option. This can be useful on a switch if you are trying to obtain all the IP addresses on the interfaces by use of the **show ip interface brief** command. On a switch that has many ports, the output of this command will also list all the ports that have no IP address assigned. If you are looking for the interfaces that have an IP address only, those lines obscure the output. If you know that all interfaces without an IP address have the string "unassigned" in place of the IP address, then you can exclude those lines from the output by issuing the command **show ip interface brief | exclude unassigned**.

Another useful option is **| begin**, which allows you to skip all output of the command up to the first occurrence of the regular expression pattern. For example, if you are interested only in checking the configuration for the vty lines and you know that these are at the very end of the configuration, you can jump straight to that point by issuing the command **show running-config | begin line vty**.



For selecting pieces of the configuration file, there is a newer option (introduced in Cisco IOS version 12.3(2)T) that allows you to select lines from the configuration that match a particular regular expression and any following associated lines.

For instance, the command **show running-config | section router** will select all lines that include the expression "router" and the configuration section that follows that line. Therefore, this command will select all routing protocol configuration sections, but leave out the rest of the configuration. This makes it more restrictive than the **| begin** option, but more useful than the **| include** option when you want to select sections of a command instead of only lines that contain a specific expression.

Although the **show running-config** command is the most obvious candidate for the use of the **| section** option, it can be applied to any show command that separates its output in sections. For example, if you want to display only the standard access-lists in the output of the **show access-lists** command you can achieve that by issuing the command **show access-lists | section Standard**.

Although in many cases the **include**, **exclude**, **begin**, and **section** options will simply be followed by a word or text fragment, it is actually possible to use the full power of the regular expression string matching syntax.

For example, in the second command in the figure you can see how the "^" sign can be used to denote that a particular string will be matched only if it occurs at the beginning of a line. The expression "^CPU" will therefore match only lines that start with the characters "CPU" and not any lines that contain the string "CPU." In addition, you can see how the pipe character "|" can be used as part of a regular expression to signify a logical OR. Therefore, the output being matched by the **show processes cpu | include ^CPU|IP Input** command are all lines that start with the string "CPU" or contain the string "IP Input."

For more information about regular expressions, see the Understanding Regular Expressions section in the Cisco IOS Configuration Fundamentals Configuration Guide at:

http://www.cisco.com/en/US/docs/ios/fundamentals/configuration/guide/cf_cli-basics_ps6350_TSD_Products_Configuration_Guide_Chapter.html#wp1002051

Redirecting show Command Output

The output of a **show** command can be redirected, copied, or appended to files:

```
RO1#show tech-support | redirect tftp://192.168.37.2/show-tech.txt
!
```

The output can be redirected to a file using **redirect:**

```
RO1#show ip interface brief | tee flash:show-int-brief.txt
Interface                  IP-Address      OK? Method Status
Protocol
FastEthernet0/0            10.1.192.2      YES manual up                    up
FastEthernet0/1            10.1.192.10     YES manual up                    up
Loopback0                  10.1.220.1      YES manual up                    up

CRO1#dir flash:
Directory of flash:/

    1  -rw-    23361156   Mar 2 2009 16:25:54 -08:00  c1841-advipservicesk9-
mz.124-23.bin
    2  -rw-         680   Mar 7 2009 02:16:56 -08:00  show-int-brief.txt
```

Or the output can be copied to a file using **tee.**

Instead of filtering the output of a **show** command, the output can also be redirected, copied, or appended to a file by using the pipe character (|), followed by the options **redirect**, **tee**, or **append** and a URL that denotes the file.

When you use the | **redirect** option on a **show** command, the output is not displayed on the screen but is redirected to a text file instead. This file can be stored locally on the device's flash memory or it can be stored on a network server such as a TFTP or FTP server.

The | **tee** option is similar to the | **redirect** option, but the difference is that this command both displays the output on your screen and copies it to a text file.

Redirecting show Command Output (Cont.)

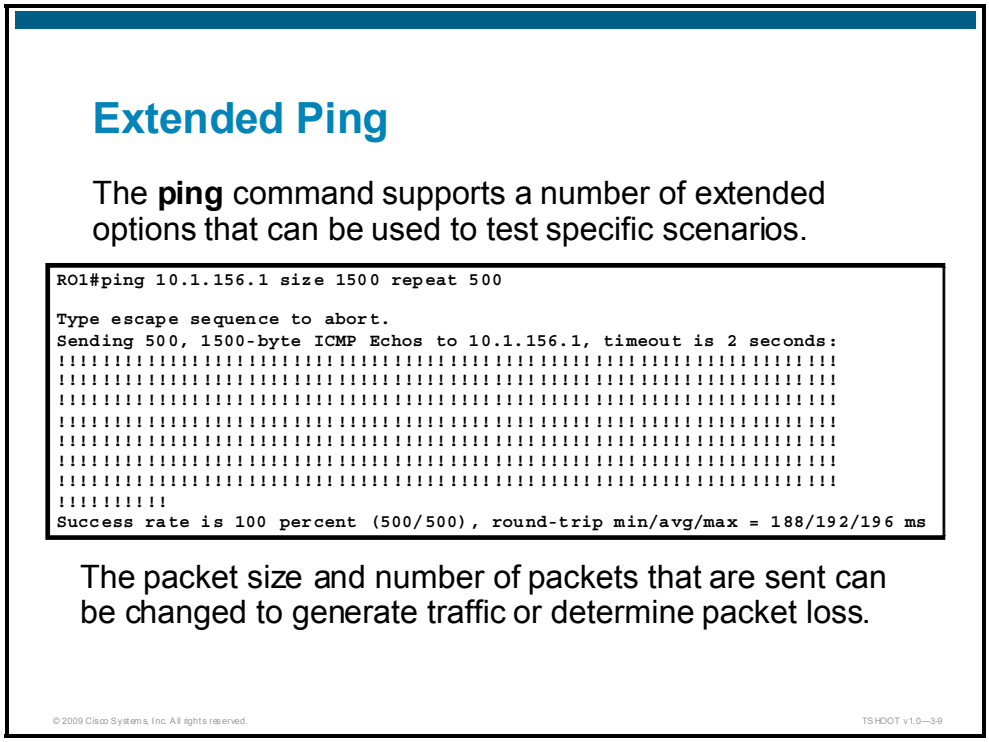
Or, the output of multiple commands can be appended to a file using **append** if the file system supports append operations:

```
RO1#show version | append flash:show-commands.txt
RO1#show ip interface brief | append flash:show-commands.txt
RO1#more flash:show-commands.txt
Cisco IOS Software, 1841 Software (C1841-ADVIPSERVICESK9-M), Version 12.4(23),
RELEASE SOFTWARE (fc1)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2008 by Cisco Systems, Inc.
Compiled Sat 08-Nov-08 20:07 by prod_rel_team

ROM: System Bootstrap, Version 12.3(8r)T9, RELEASE SOFTWARE (fc1)

CRO1 uptime is 3 days, 1 hour, 22 minutes

<…output omitted…>

Interface                  IP-Address      OK? Method Status
Protocol
FastEthernet0/0            10.1.192.2      YES manual up                    up
FastEthernet0/1            10.1.192.10     YES manual up                    up
Loopback0                  10.1.220.1      YES manual up                    up
```


Finally, the | **append** option is analogous to the | **redirect** option, but it allows you to append the output to a file instead of replacing that file. The use of this command makes it very simple to collect the output of several **show** commands in a text file, either directly on a server, or first on the device itself and then copied to a server. A prerequisite for this option is that the file system that you are writing to must support “append” operations, so for example a TFTP server cannot be used in this case.

Connection Testing

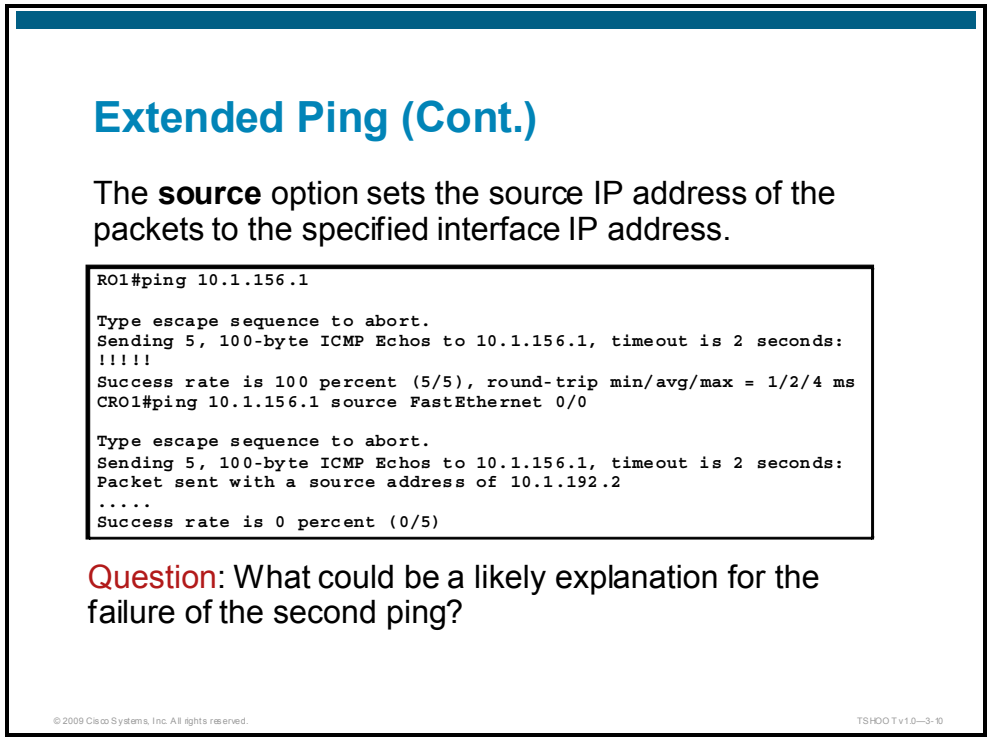
This topic describes how to use Cisco IOS commands to test network connectivity on the network and transport layers.



Ping is probably the most widely known connectivity-testing tool in networking and has been part of the Cisco IOS Software from the beginning. Still, it is worth pointing out some of the extended options and how these can be used to test specific scenarios.

The following options can be useful while troubleshooting:

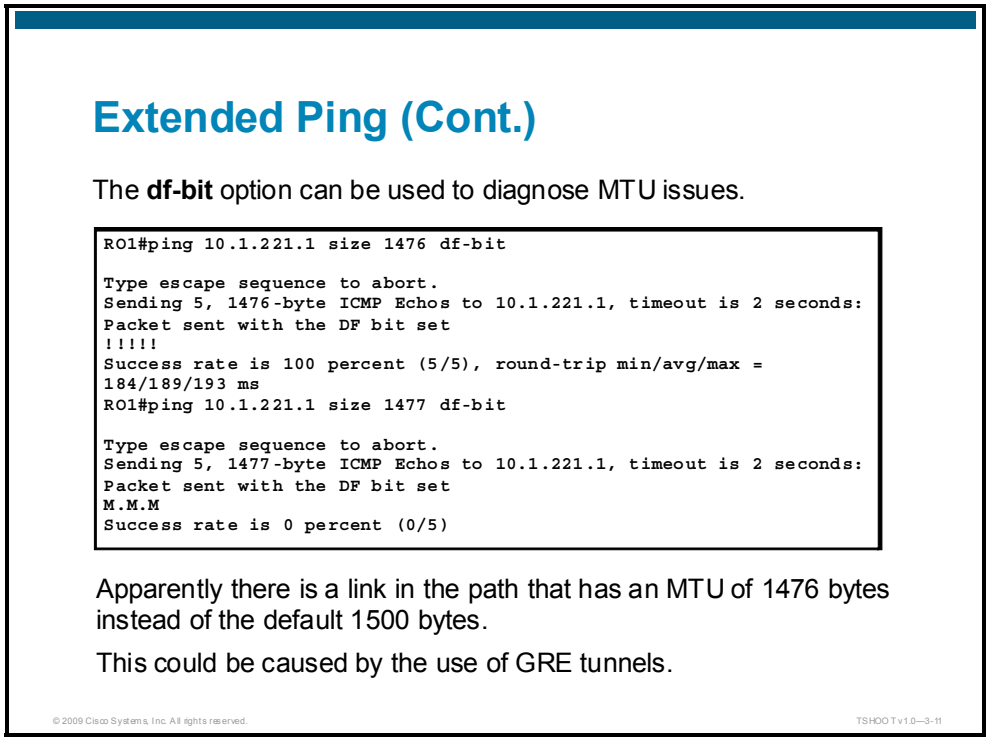
- **repeat** *repeat-count*: By default, the **ping** command sends out five Internet Control Message Protocol (ICMP) echo request packets. By using the **repeat** option, you can specify how many packets should be sent. This is particularly useful when you are troubleshooting a situation where you have detected some packet loss. Using this option, you can send out hundreds to thousands of packets and then look for a pattern in the occurrence of the packet loss. If you see a particular pattern, this can give you a clue as to what is happening. For instance, if you see a pattern where every other packet is lost, resulting in exactly 50 percent packet loss, then this usually indicates that traffic is being load-balanced across two paths and on one of the two paths, the packets are dropped.
- **size** *datagram-size*: This allows you to specify the total size of the packet (including headers) in bytes that will be sent. This can be useful for various reasons. In combination with the **repeat** option, it allows you to send a steady stream of large packets, and this way it can be used to generate some load. The quickest way to generate a lot of load by use of the **ping** command is to combine a very large **repeat** number, a **size** set to 1500 bytes, and the **timeout** option set to 0 seconds. A common second application is to use this option in combination with the **df-bit** option to determine the maximum transmission unit (MTU) across a particular path.



- **source [***address* | *interface***]**: This option allows you to set the source IP address or interface. The IP address must be one of the device's own IP addresses. This is very useful because by default, the router will select the IP address of the outbound interface as the source of the packets.

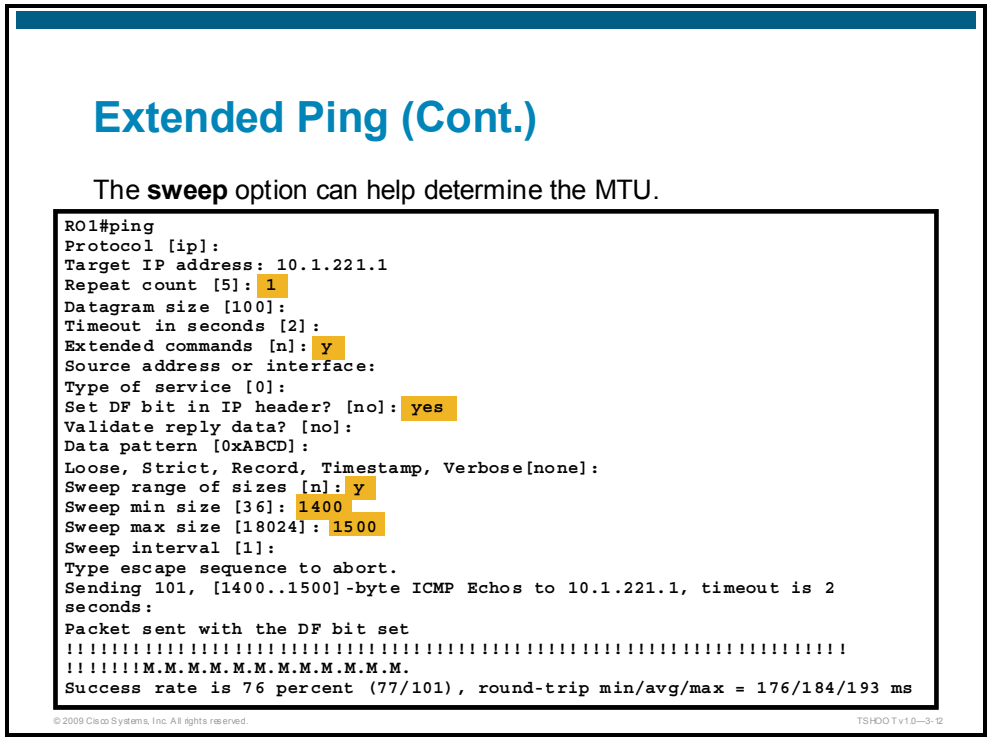
Look at the example in the figure. What can explain why the initial ping succeeds, but the ping with the source IP address set to the IP address of the FastEthernet 0/0 interface fails?

To begin with, you can conclude from the successful initial ping that apparently this router has a working path to IP address 10.1.156.1. So for the second ping, it is not likely that it would be the path toward 10.1.156.1 that fails. The difference must be in the return path. Because a different source address is used for the packets, the return packets will have a different destination IP address. The most likely explanation for the failure of the second ping in this example is that at least one of the routers on the return path does not have a route to the subnet of the FastEthernet 0/0 interface.



- **df-bit**: This option sets the Don't Fragment bit in the IP header to indicate that routers should not fragment this packet. If it is larger than the MTU of the outbound interface, the router should drop the packet and send an ICMP "Fragmentation needed and DF bit set" message back to the source. This option can be very useful when you are troubleshooting MTU-related problems. By setting the **df-bit** option and combining it with the **size** option, you can force routers along the path to drop the packets if they would need to fragment them. By varying the size and looking at which point the packets start being dropped, you can determine the MTU.

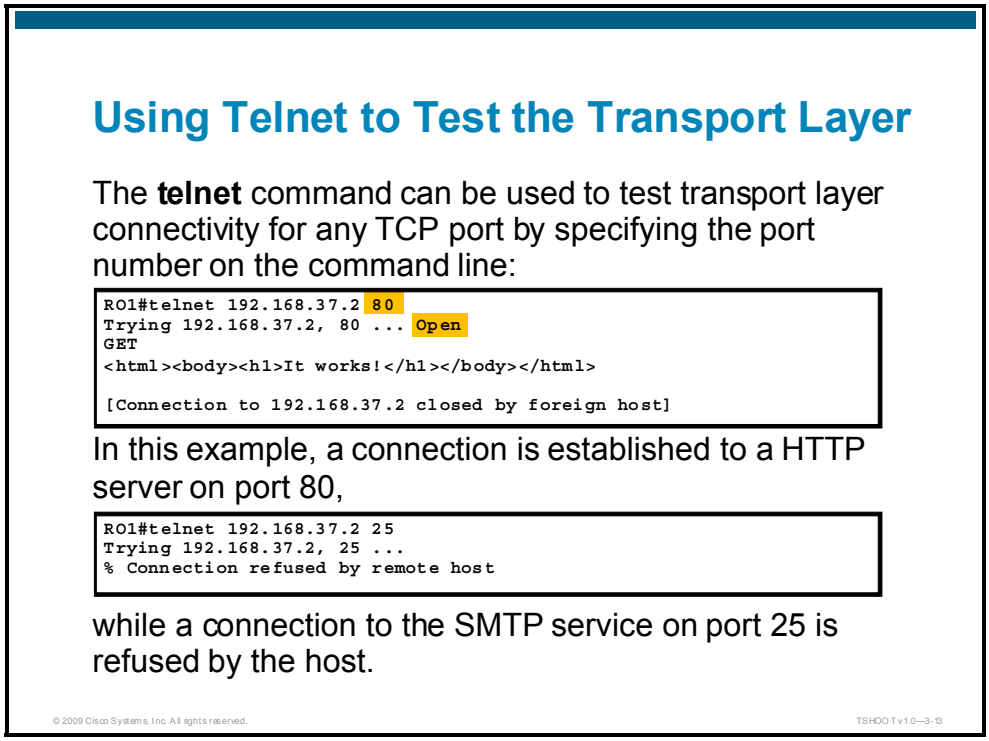
In the example in the figure, you can see that packets with a size of 1476 bytes are successfully sent, while packets with a size of 1477 bytes are dropped. The M in the output of the **ping** command signifies that an ICMP "Fragmentation needed and DF bit set" message was received. From this, you can conclude that somewhere along the path to the destination there must be a host that has an MTU of 1476 bytes. A possible explanation for this could be the use of a Generic Routing Encapsulation (GRE) tunnel, which typically has an MTU of 1476 bytes (1500 bytes default MTU minus 24 bytes for the GRE and IP headers).



Even more extended options can be set for the **ping** command through the use of an interactive dialog. By entering ping without any additional options, you get a series of questions, including the options that you can also set directly as on option. In addition, there are several other options. In the figure, the “sweep range of sizes” option is highlighted. This option allows you to send a series of packets that increase in size and can be useful to determine the MTU along a path, similar to the previous example.

The biggest issue in the previous example is that you needed to specify the size for the packets on the **ping** command, and if you do not really have a good initial guess of the MTU, it can take you many tries to find the exact MTU. In this example the router is told to send packets starting at a size of 1400 bytes, sending a single packet per size and increasing the size one byte at a time until a size of 1500 bytes is reached. Again, the DF bit is set on the packets.

The result of this is that the router sends out 101 consecutive packets, the first one being 1400 bytes in size, and the last one being 1500 bytes. As can be seen, 77 of the pings are successful and 24 of the pings fail. Again, this means that there must be a link along the path that has an MTU of 1476 bytes.



While ping can be used to test network layer connectivity, it can also be very useful to be able to test transport layer connections from the command line. For instance, consider that you are troubleshooting a problem where someone cannot send email through a particular SMTP server. You decide to use a divide-and-conquer approach. You ping the server and it responds. This means that the network layer between you and the server is operational. Now, how do you verify the transport layer? Of course, you could configure a client and start a top-down troubleshooting procedure, but it would be convenient if you could first establish that Layer 4 is operational. So how can you do this?

Even though the Telnet server application runs on its own well-known port number 23 and Telnet clients connect to that port by default, you can specify a specific port number on the client and connect to any TCP port that you want to test. At least this will show you if the connection is accepted (as indicated by the word “Open” in the figure), if the connection is refused, or if it times out. From any of those responses you can draw further conclusions concerning the connectivity. Certain applications, if they use an ASCII-based session protocol, might even display an application banner, or you might be able to trigger some responses from the server by typing in some keywords. Good examples of these types of protocols are SMTP, FTP, and HTTP.

Hardware Diagnostics

This topic describes the generic Cisco IOS commands that can be used to gather information about the status and performance of the hardware.

Basic Hardware Diagnostics

Cisco IOS Software includes many commands to diagnose hardware operation.

Due to their nature, many of those commands and features are product- and platform-specific.

Essential commands that are common to both routers and switches are:

- **show processes cpu**
- **show memory**
- **show interface**

TSHOOT v1.0—3-14

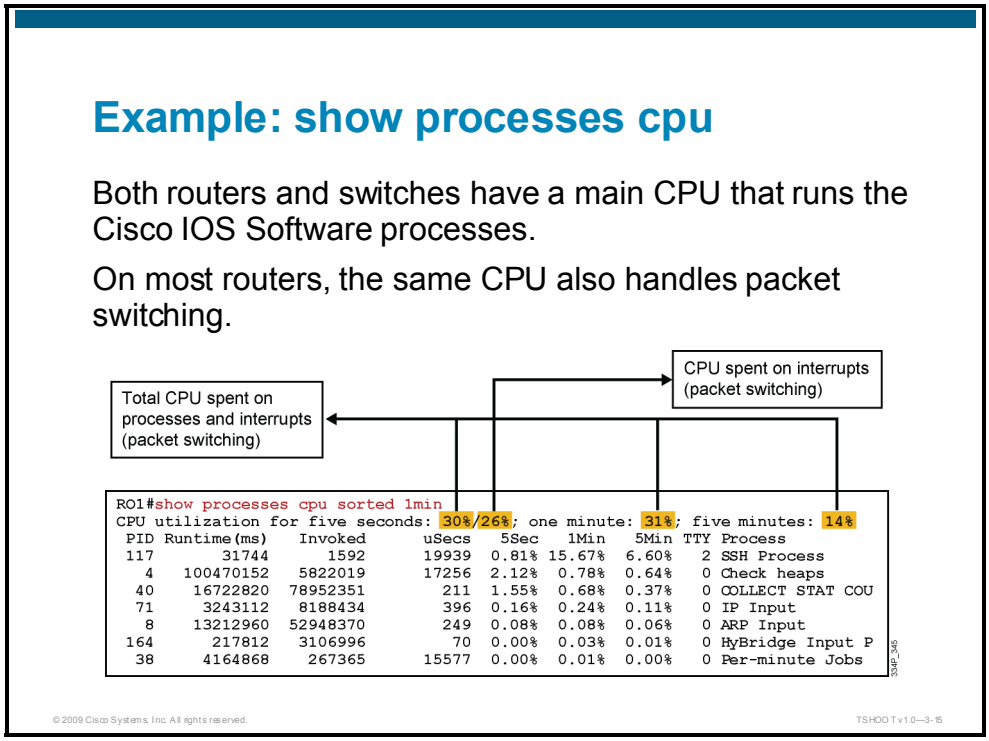
Inevitably, troubleshooting processes involve a component of hardware troubleshooting. In the end, there are really only three main categories of things that could be the cause of a failure on the network: hardware failures, software failures (bugs), and configuration errors. One could argue that performance problems form a fourth category, but performance problems are not really a problem cause, but a symptom. Having a performance problem means that there is a difference between the expected behavior and the observed behavior. This can mean one of two things. One option is that the expectation was not in line with reality: the system is functioning as it should, but the result is not what was expected or promised. In this case, the problem is not technical but organizational in nature and cannot be resolved through technical means. The other option is that the system is not functioning as could be reasonably expected. However, in this case there must be an underlying reason that causes the system to behave differently from what was expected, and again this underlying cause will be a hardware failure, a software failure, or a configuration error.

Most of the focus in this course is on diagnosing and resolving configuration errors. There are a number of reasons for this focus. First, out of those three categories, configuration is generally the only element that is created or changed by engineers during implementation and change processes. Hardware and software can only really be swapped out if they are suspected to be the cause of the problem, so the actions that can be taken to resolve the problem are limited. Second, the detailed information that would be needed to pinpoint a specific hardware or software problem is often not publicly available, and therefore hardware and software troubleshooting processes are generally executed as a joint effort with a vendor (or a reseller or partner for that vendor). Documentation of the configuration and operation of software features is generally publicly available, and therefore configuration problems can often be diagnosed without needing direct assistance from the vendor or reseller.

However, even if you decide to focus your troubleshooting on configuration errors initially, as your troubleshooting progresses and you eliminate common configuration problems from the equation, you might pick up clues that hardware components are the underlying cause of the problem, and you will need to be able to at least do a first analysis and possible diagnosis of the problem before it is escalated to the vendor. The move-the-problem method is an obvious candidate to approach suspected hardware problems, but this approach only really works well if the problem is a more or less black-and-white—works or does not work—type of problem. Performance problems that might be caused by hardware failures generally require a more subtle approach and require more-detailed information gathering.

Due to its nature, diagnosing hardware problems is highly product and platform dependent. However, there are a number of generic commands that can be used to diagnose performance-related hardware issues on all platforms based on Cisco IOS Software.

In essence, any network device is a specialized computer, which at a minimum consists of a CPU, RAM, and storage, allowing it to boot and run the operating system and interfaces, which allow for the reception and transmission of network traffic. Therefore, after you decide that a problem that you are observing on a given device might be hardware related, it is worthwhile to at least verify the operation of these generic components. The most commonly used Cisco IOS commands for this purpose are the **show processes cpu**, **show memory**, and **show interface** commands.



Both routers and switches have a main CPU that executes the processes that constitute the Cisco IOS Software. Processes are scheduled to share the available CPU cycles and take turns executing their code. The **show processes cpu** command gives you an overview of all processes that are currently running on the router, the total amount of CPU time that they have consumed over their lifetime, their CPU usage over the last 5 seconds, and a 1-minute and 5-minute weighted average of their CPU usage. At the top, you will also see the total percentages of available CPU cycles spent. From this, you can see if the total CPU usage is high or low and which processes might be causing the CPU load. By default, the processes are sorted by process ID, but they can be sorted based on the 5-second, 1-minute, and 5-minute averages.

In the example in the figure you can see that over the past minute, 31 percent of the available CPU have been used and the “SSH Process” was responsible for roughly half of these CPU cycles (15.67 percent). However, the next process in this sorted list is the “Check heaps” process, which has consumed only 0.78 percent of the total available CPU time over the last minute, and the list quickly drops off after that. So what were the remaining 15 percent CPU cycles that were recorded over the last minute spent on?

The command output in this example was taken from a low-end router, and on most of the low-range and midrange routers, the same CPU that is used to run the operating system processes is also responsible for packet switching. To maximize the speed at which packets can be switched, the packet switching task is not handled by a specific process, but the CPU can be interrupted to suspend the current process that it is executing, switch one or more packets, and then resume the execution of scheduled processes. The CPU time spent on interrupt-driven tasks is therefore not accounted for under any particular process. Therefore, the time spent in interrupt mode can be calculated by adding the CPU percentages for all processes and then subtracting that total from the total CPU percentage listed at the top. For the 5-second CPU usage, this figure is actually even listed separately behind the slash.

This means that in the example in the figure, 30 percent of the total available CPU cycles over the past 5 seconds were used, out of which 26 percent were spent in interrupt mode and 4 percent for the execution of scheduled processes.

Because of this, it is quite normal for routers to be running at high CPU loads during peaks in network traffic. In those cases, most of the CPU cycles will be consumed in interrupt mode. If, however, particular processes consistently use large chunks of the available CPU time, this could be a clue that there is a problem associated with those particular processes. However, to be able to draw any definitive conclusions, you would need to have a baseline of the CPU usage over time.

For switches, the essential elements of this **show** command are the same, but the interpretation of the numbers tends to be a bit different. Switches have specialized hardware that handles the packet switching task, and the main CPU should in general not be involved in this. When you see a high percentage of the CPU time being spent in interrupt mode this usually indicates that packets are being forwarded in software instead of by the ternary content addressable memory (TCAM).

After you determine that the CPU load is abnormally high, and you decide to investigate further, you will generally need to resort to platform-specific troubleshooting commands to gain more insight into what is happening.

Example: show memory

Both routers and switches have an amount of generic RAM memory, used by processes and for temporary packet buffering.

```
RO1#show memory
                Head    Total(b)     Used(b)     Free(b)   Lowest(b)  Largest(b)
Processor   820B1DB4    26534476    19686964     6847512     6288260     6712884
      I/O     3A00000     6291456     3702900     2588556     2511168     2577468
```

Not having sufficient free memory can cause memory allocation problems.

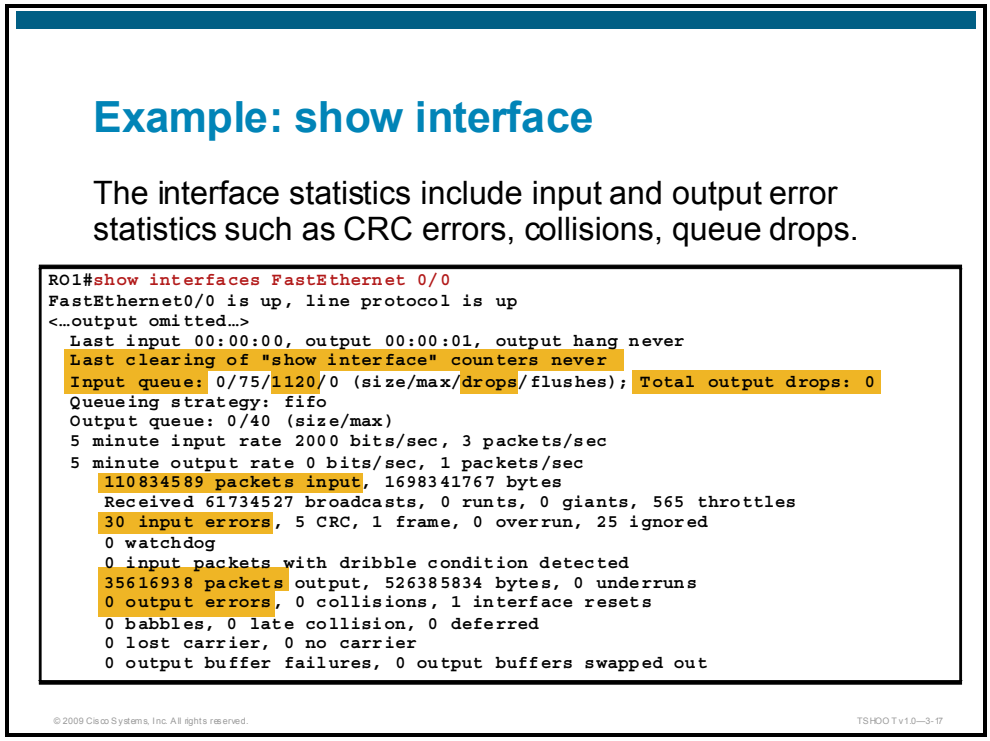
Establishing a baseline can help discover these issues before they cause disruption.

Like available CPU cycles, memory is a finite resource, shared by the various processes that together form the Cisco IOS technology. Similar to the division between CPU cycles spent for scheduled processes and CPU cycles spent on interrupt-driven packet switching, the memory is divided into different pools and used for different purposes.

At least there will be a processor pool that contains memory that can be used by the scheduled processes, and an I/O pool that is used to temporarily buffer packets during packet switching. Processes allocate and release memory as needed from the processor pool, and generally, there should be more than enough free memory for all the processes to share. Memory on routers and switches is dimensioned to be more than enough for what they would need in the typical environment that they were designed for. However, in unusual deployment scenarios, for example, when you are running Border Gateway Protocol (BGP) on your router and you decide that you need to carry the full Internet routing table, you might need more memory than the default memory included with the router. Also, when you decide to upgrade the Cisco IOS Software on your router you should be sure to verify the recommended memory for the new software version.

As with CPU usage, it can be useful to create a baseline of the memory usage on your routers and switches and graph the usage over time to ensure that you will notice when the devices start to run low on memory.

If a router or switch does not have enough free memory to satisfy the request of a process, it will log a memory allocation failure, signified by a %SYS-2-MALLOCFAIL message. The result of this is that the process cannot get the memory that it requires, which can result in unpredictable disruptions or failures. Apart from the processes filling the memory through normal use, you could also encounter a situation where, due to a software defect, processes do not properly release memory, causing memory to “leak” away, eventually leading to memory allocation failures and, most likely, a crash of the router. Creating a baseline and graphing memory usage over time can also help in pinpointing these types of failures.

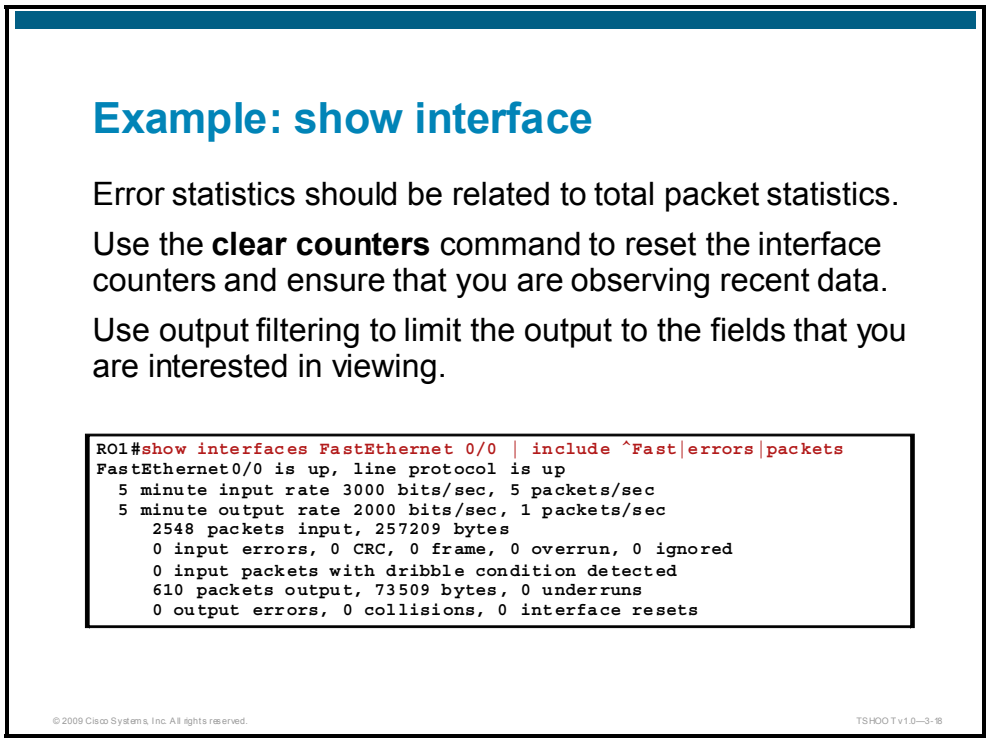


The interfaces that the traffic passes through are another component that is always worth verifying when you are troubleshooting performance-related issues and you suspect the hardware to be at fault. In most cases the interfaces are one of the first things that you will verify while tracing the path between devices.

The output of the **show interface** command lists a number of important statistics that should be checked:

- **Input queue drops:** Input queue drops (and the related ignored and throttle counters) signify that at some point more traffic was delivered to the router than it could process. This does not necessarily indicate a problem, because it could be normal during traffic peaks. However, it could be an indication that the CPU cannot process packets in time, so if this number is consistently high, it is worth trying to spot at which moments these counters are increasing and how this relates to CPU usage.
- **Output queue drops:** These indicate that packets were dropped due to congestion on the interface. Seeing output drops is normal for any point where the aggregate input traffic is higher than the output traffic. During traffic peaks, packets are dropped if traffic is delivered to the interface faster than it can be sent out. However, even if this is considered normal behavior, it leads to packet drops and queuing delays, so applications that are sensitive to those, such as voice over IP (VoIP), might suffer from performance issues. Consistently seeing output drops can therefore be a good indicator that you might need to implement an advanced queuing mechanism to provide good quality of service (QoS) to each application.
- **Input errors:** This counter indicates errors experienced during the reception of the frame, such as cyclic redundancy check (CRC) errors. High numbers of CRC errors could indicate cabling problems, interface hardware problems, or, in an Ethernet-based network, duplex mismatches.

- **Output errors:** This counter indicates errors, such as collisions, during the transmission of a frame. In most Ethernet-based networks today, full-duplex transmission is the norm and half-duplex is the exception. In full-duplex operation collisions cannot occur; therefore collisions, and especially late collisions, often indicate duplex mismatches.



For all these counters, the absolute number of drops or errors is not very significant, but the statistics should be evaluated against the total number of input and output packets. For example, a total of 25 CRC errors in relation to 123 input packets is reason for concern, while 25 CRC errors for 1.458.349 packets is not a problem at all.

Also, these statistics are collected from the point where the router is booted up, so the numbers that you are looking at might be statistics over several months. Based on those statistics, how can you diagnose a problem that has been happening for the past two days? After you decide that you need to investigate the interface counters in more detail, it is good practice to reset the interface counters by using the **clear counters** command, let it accumulate statistics for a while, and then reevaluate the outcome.

If you repeatedly want to display selected statistics to see how the counters are increasing, it is very useful to filter the output by use of a regular expression to include only the lines that you are interested in. For instance, in the example in the figure, the output is limited to only the lines that start with the word “Fast,” include the word “errors,” or include the word “packets.”

Additional Hardware Diagnostics

Other commands that can be useful in troubleshooting hardware-related problems:

- **show controllers**
- **show platform**
- **show inventory**
- **show diag**

Some features that can be used to diagnose interface hardware or cabling issues:

- Generic Online Diagnostics (GOLD)
- Time Domain Reflectometer (TDR)

Hardware troubleshooting is by definition platform dependent, so research the capabilities of the platform you are working on.

The commands that are covered in this topic form a very limited toolkit of hardware troubleshooting commands, but they are a good starting point to collect some initial clues to either confirm that the problem might be hardware related, or eliminate hardware problems from the list of potential problem causes. When you have decided that the cause of the problem might be hardware related, you should research the more-specific hardware troubleshooting tools that are available for the platform that you are working with.

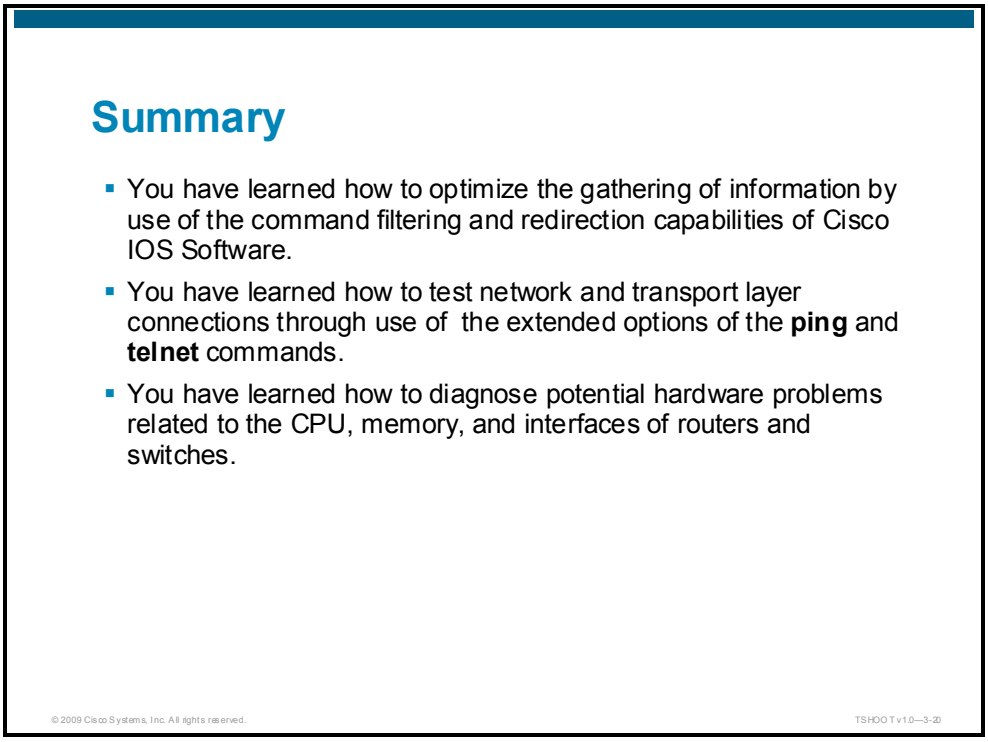
The following are some hardware troubleshooting features and commands that are supported in Cisco IOS Software:

- **show controllers:** The output of this command is dependent on the interface hardware, but in general, it gives you more-detailed packet and error statistics for that particular type of hardware, in addition to information about hardware elements such as buffers, queues, and other data structures associated with that interface.
- **show platform:** On many of the Cisco Catalyst LAN switches, the **show platform** family of commands can be used to examine the TCAMs and other specialized switch hardware components.
- **show inventory:** This command lists the hardware components of a router or switch. The output includes the product code and serial number for each component. This is useful to document your hardware and to order replacement or spare parts.
- **show diag:** On routers, this command can be used to gather even more-detailed information about the hardware than what is provided by the **show inventory** command. For example, the output of this command includes the hardware revision of the individual components. In case of known hardware issues, this command can be used to determine whether the component is susceptible to a particular hardware fault.
- **Cisco Generic Online Diagnostics (GOLD):** Cisco GOLD is a platform-independent framework for runtime diagnostics. It includes command-line interface (CLI)-based access to bootup, health monitoring, on-demand, and scheduled diagnostics. It is available on many of the midrange and high-end Catalyst LAN switches and high-end routers such as the 7600 series and CRS-1 routers.
- **Time Domain Reflectometer (TDR):** Some of the Catalyst LAN switches support the TDR feature, which allows you to detect cabling problems such as open or shorted UTP wire pairs.

For further hardware troubleshooting commands and tools, research the appropriate sections of the configuration guides and command references on http://www.cisco.com for the platform that you are working with.

Summary

This topic summarizes the key points that were discussed in this lesson.



Lesson 2

Using Specialized Maintenance and Troubleshooting Tools

Overview

Access to information is one of the most important aspects of troubleshooting and maintenance. Information is gathered in many ways: on demand, during troubleshooting processes, continuously, as part of baseline creation, and triggered by network events.

In addition to the tools that are available in the Cisco IOS command-line interface (CLI), there are many specialized network maintenance and troubleshooting tools that you can use to support these information-gathering processes. These tools and applications typically require communication with the network devices, and several different underlying technologies can be used to transfer the information between the devices and the tools.

This lesson evaluates a number of commonly used network management platforms and troubleshooting tools. This lesson also shows examples of the implementation of the underlying technologies that enable communication between the network and the tools and applications.

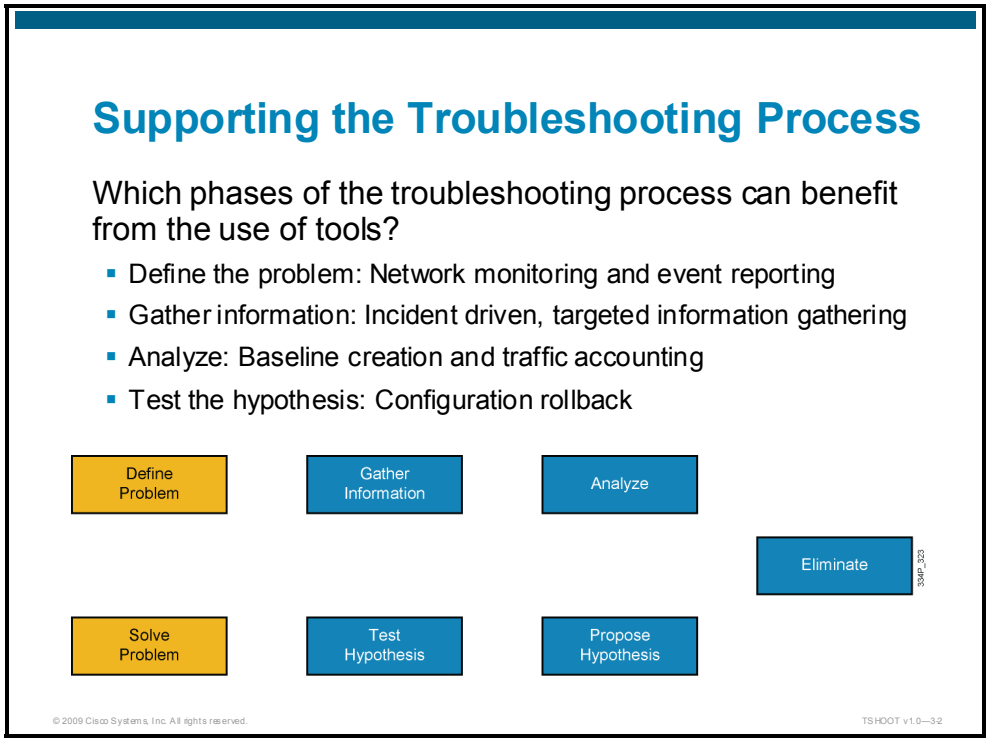
Objectives

Upon completing this lesson, you will be able to identify tools that are commonly used for specific maintenance and troubleshooting processes. This ability includes being able to meet these objectives:

- Identify tools and their underlying technologies to support the troubleshooting process
- Enable SPAN and RSPAN to facilitate the use of packet sniffers
- Configure routers and switches for communication with SNMP-based or NetFlow-based network management systems to facilitate the collection of device and traffic statistics that are part of a network baseline
- Configure routers and switches to send SNMP traps to provide fault notification to SNMP-based network management systems

Troubleshooting and Supporting Tools

This topic describes the underlying technologies and features that can be implemented to facilitate the use of troubleshooting tools.

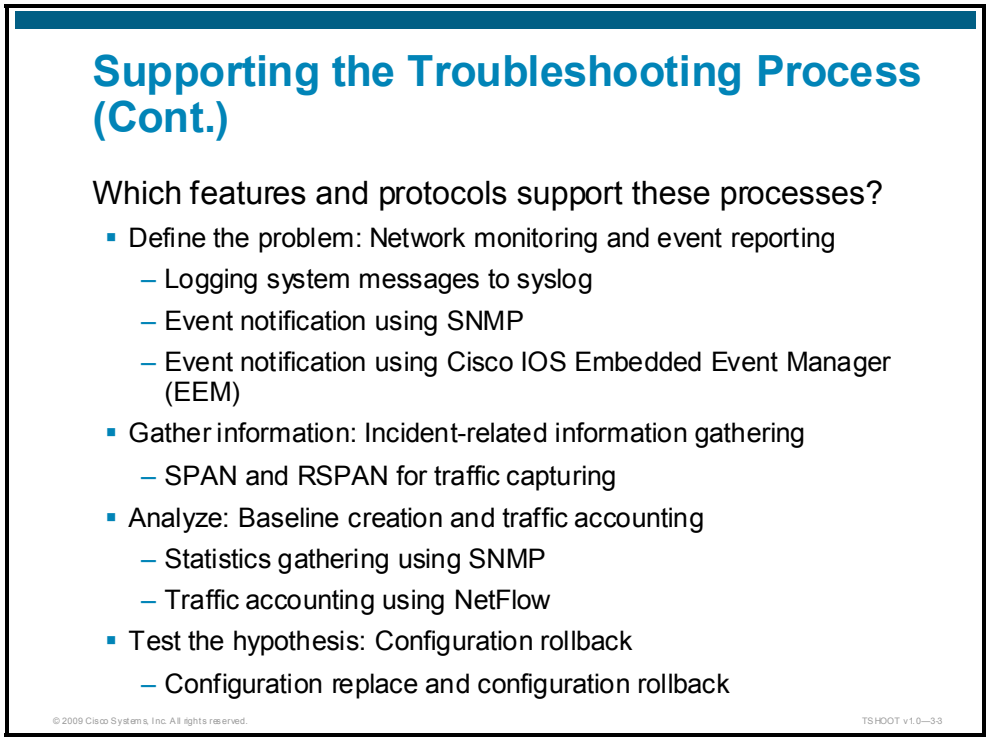


A generic troubleshooting process consists of several phases, or subprocesses. Some of these processes are primarily mental, such as the elimination process. Some of these processes are administrative in nature, such as documenting and reporting changes and solutions. And some of these processes are more technical in nature, such as the gathering and analysis of information.

The processes that benefit the most from the deployment of network maintenance and troubleshooting tools are the processes that are technical in nature, and therefore this lesson will focus primarily on the use of those tools and how to prepare the network to support those tools.

The following processes have elements that you can optimize by the use of tools:

- **Define the problem:** One of the main objectives of deploying a proactive network management strategy is to be aware of potential problems before users report that they are experiencing outages or performance degradation. Network monitoring and event reporting systems can notify the network support team of events as they happen, giving them time to respond to the problem before the users notice and report them.
- **Gather information:** This is one of the essential steps in the troubleshooting process, and you can leverage any tool to obtain detailed information about events in an effective way.
- **Analyze:** A major component of the interpretation and analysis of the gathered information is comparison against a baseline. The ability to differentiate between normal and abnormal behavior can yield important clues about the potential problem cause. Collecting statistics about network behavior and network traffic is therefore a key process to support troubleshooting data analysis.
- **Test your hypothesis:** Testing a hypothesis commonly involves making changes to the network, and you might need to roll back those changes if they did not resolve the problem. Tools that enable easy rollback of changes are therefore important to an efficient troubleshooting process.



With the exception of configuration rollback, which is more a generic change management tool than a specific troubleshooting tool, most of the mechanisms mentioned in the preceding section fall into one of the following three categories:

- **Collection of information on demand, driven by incidents:** This is the typical information gathering that you do during the troubleshooting process itself. You gather, interpret, and analyze information, and based on the outcome of this process you gather more information. Examples of this are the capturing of network traffic or debugging of device processes.
- **Continuous collection of information to establish a baseline:** A set of major network performance indicators are established, and based on those indicators, statistics about the behavior of the network are collected over a long period of time. These statistics together form a baseline that you can use to judge whether the behavior that you observe is normal or not. This process also provides historical data that you can correlate to events. Examples of this are the collection of statistics through use of Simple Network Management Protocol (SNMP) and traffic accounting by use of NetFlow technology.
- **Notification of network events:** Instead of collecting information from the network driven by incidents or through a continuous process, information is reported by network elements triggered by the occurrence of specific events. Examples of this are the reporting of events via syslog log messages or SNMP traps and the definition and the reporting of specific events through the use of the Embedded Event Manager that is part of the Cisco IOS Software.

What these categories have in common is that their functionality depends on interaction between a tool or application running on a host, and the network devices. In the first two categories, the information is pulled from the network elements to the application or tool; in the last category, the information is pushed to the application or tool by the network devices.

A very broad spectrum of tools and applications can perform the processes mentioned here, and it is virtually impossible to list them all, let alone compare and contrast them. However, many of these tools depend on the same underlying technologies and protocols for the communication between the application and the network. This includes protocols like syslog, SNMP, and NetFlow, and also the technology that allows you to forward packets that are received on a port to a specific system for analysis.

Beyond understanding the main benefits that a particular tool or application brings to the network troubleshooting process, it is therefore also very important for an engineer to know how to enable the necessary communication between the network devices, and the tools and applications for doing this.

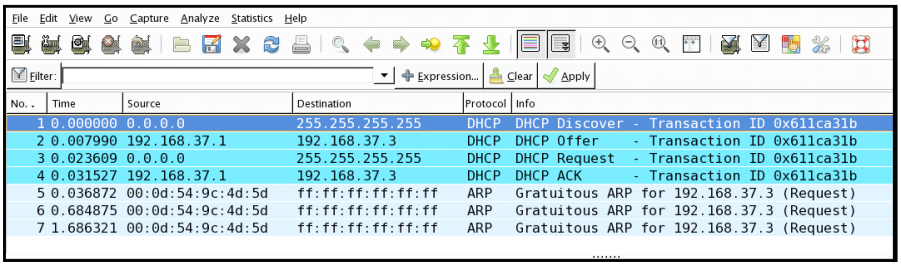
Traffic Capturing

This topic describes the use of packet sniffers in support of troubleshooting processes and how to use the SPAN feature to relay the packets to a system that has a packet sniffer installed.

Using a Packet Sniffer

Packet sniffers can be used to capture packets to allow detailed analysis of packet flows.

Taking packet captures at various points in the network allows you to spot potential differences.



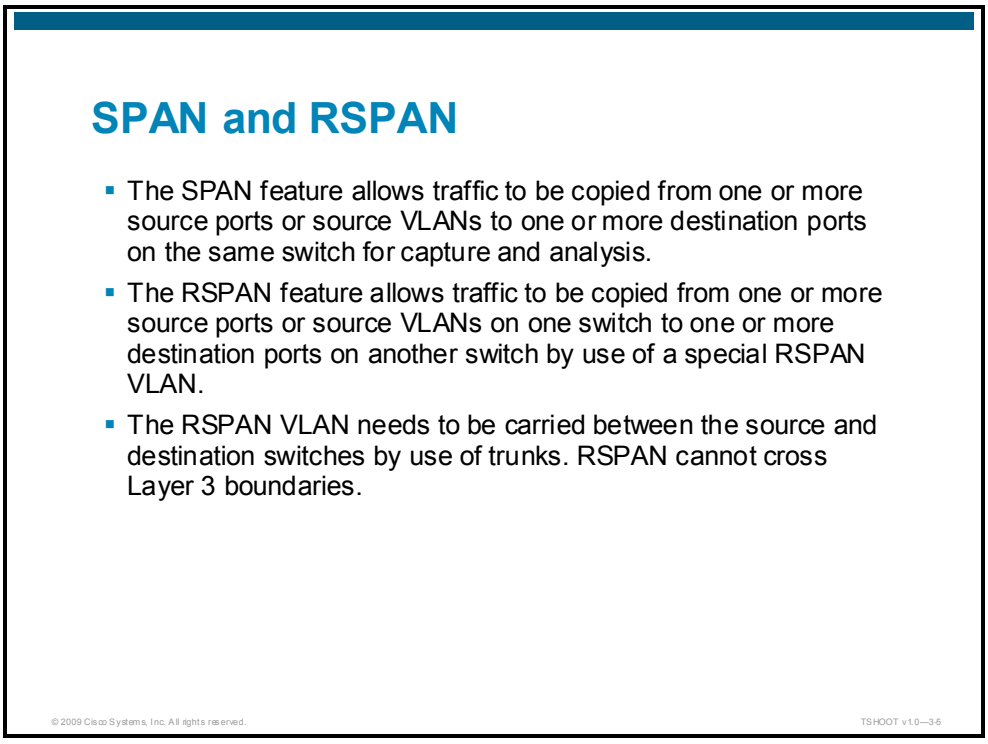
Packet sniffers or network analyzers are important tools of the trade for network engineers. They allow you to see protocol errors like retransmissions or session resets, and they also allow you to spot missing packets. When communication between two hosts is failing, it can be very useful to capture the traffic between those two hosts at various points in the network and look for differences. If you can spot where the packets start to go missing, this will help in pinpointing the problem.

Packet sniffing is a very powerful tool, because large amounts of very detailed data can be gathered, but at the same time that is also its drawback. Unless you know exactly what you are looking for and you know how to filter the information to select the traffic that you are interested in, it can be very difficult to analyze packet captures, because of the large amount of information that they contain.

Various free and commercial tools on the market enable packet capturing (or sniffing) and packet analysis. These tools can be either software based and installed on a regular PC, or specialized hardware devices that can capture huge amounts of data in real time. Whatever tool you select, it is always important to learn the filtering capabilities of the product, so that you are capable of selecting just the information that you are interested in from the captured data.

One of the issues that you generally run into is that it is not always practical, or even possible, to install the software on the machines on which you are troubleshooting a problem. Especially on servers, but also on clients, the installation of software is often tightly controlled, and in many cases it is not possible to capture traffic on the end-systems themselves.

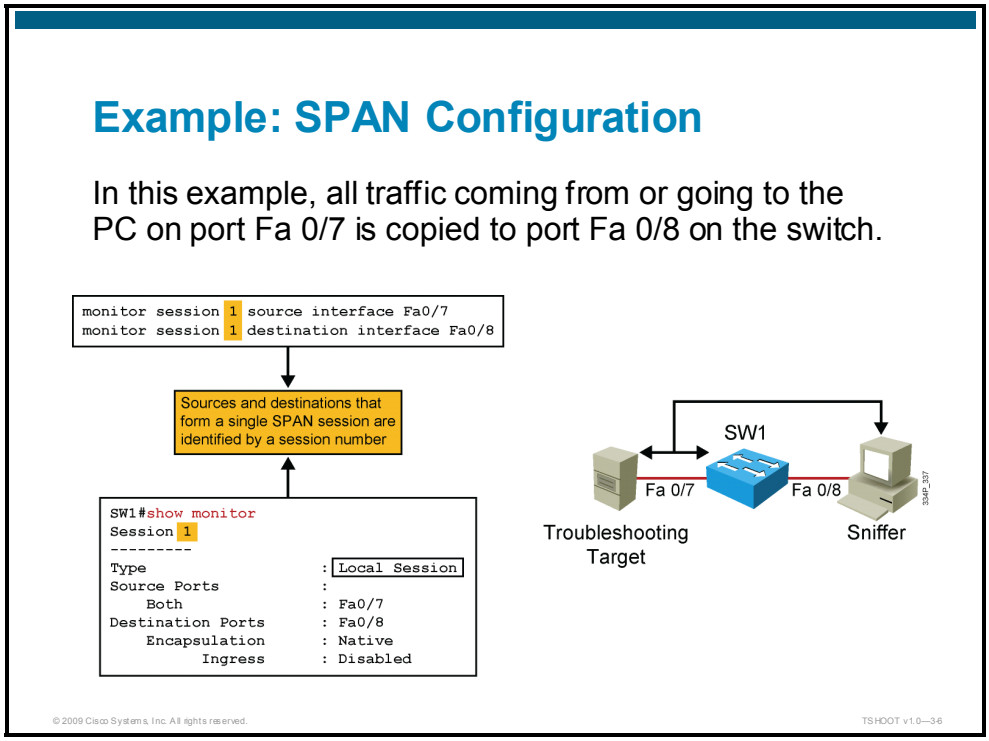
Luckily, there is a solution to this problem. If you cannot get the packet capturing software on a particular machine itself, the next best thing is to transport the traffic that you want to capture to a machine that already has the software installed.



The Switched Port Analyzer (SPAN) feature that is available on Cisco Catalyst switches allows you to copy the traffic from one or more switch ports or VLANs to another port on the same switch. This feature allows you to connect a system that has packet-sniffing software installed to a port on the switch. That port will then receive a copy of the traffic from the selected ports or VLANs, allowing it to be captured and analyzed.

The SPAN feature can only copy traffic between ports and VLANs on a single switch. The Remote Switched Port Analyzer (RSPAN) feature allows you to copy traffic from ports or VLANs on one switch to ports on a different switch, as long as these switches are in the same Layer 2 domain. A special RSPAN VLAN is used to carry the traffic between switches, and therefore this VLAN needs to be carried on the trunks between the switches.

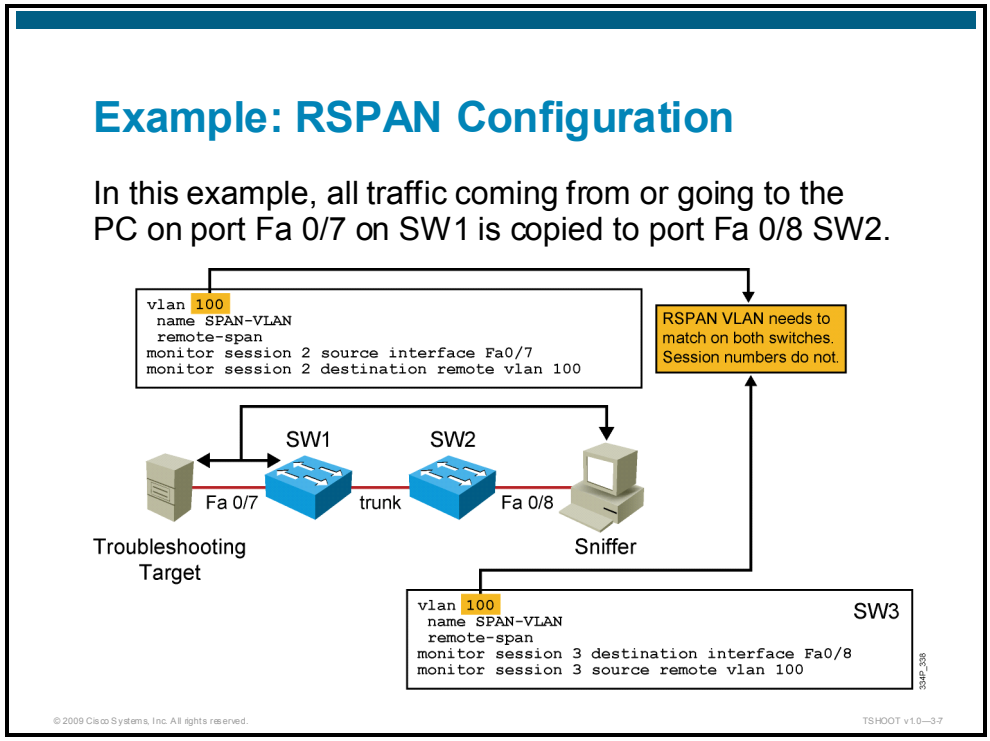
There are specific hardware-based limitations for the SPAN and RSPAN features on each of the different switching platforms, so you should check the documentation for the platform that you are working with to find out exactly which capabilities are supported and what the limitations are for that particular platform.



In the example shown in the figure, the objective is to capture all the traffic sent or received by the server connected to port FastEthernet 0/7 to troubleshoot a problem with that server. A packet sniffer is connected to port FastEthernet 0/8. The switch is now instructed to copy all the traffic that it sends and receives on port Fa 0/7 to port Fa 0/8 by configuring a SPAN session.

The SPAN session is identified by a session number, "1" in this example. The source ports or VLANs are identified by use of the **monitor session** *number* **source** command and the destination ports are identified by use of the **monitor session** *number* **destination** command. The session number is what binds the commands together to form a single session.

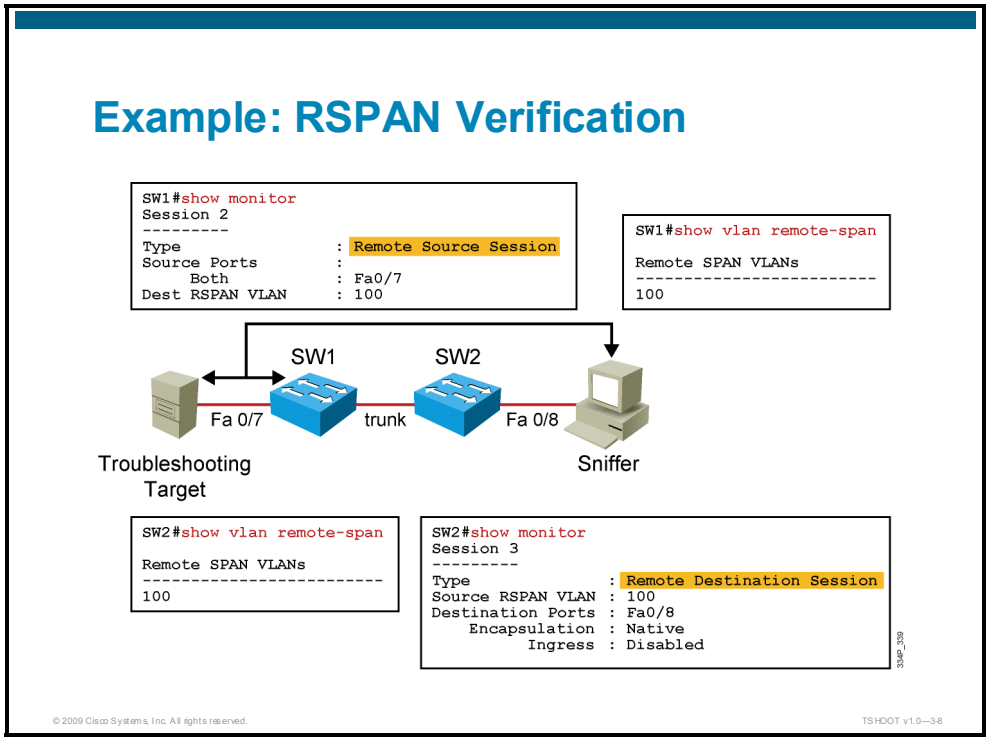
The configuration of the SPAN session can be verified by using the **show monitor** command.



Configuration of RSPAN is similar to configuration of local SPAN in the sense that it uses the **monitor session** *number* **source** and **monitor session** *number* **destination** commands to define the port that traffic is captured from and the port that traffic is copied to. However, because the ports are now on two different switches, a medium is needed to transport the traffic from one switch to the other. This is done by use of a special RSPAN VLAN. This VLAN is configured like any other VLAN, but as an addition, the keyword “remote-span” is entered in (config-vlan)# mode to signify that this is an RSPAN VLAN. This VLAN needs to be defined on all switches in the path and it needs to be allowed on the trunk or trunks between the source and destination switches.

On the source switch, the RSPAN VLAN is configured as the destination for the SPAN session through use of the **monitor session** *number* **destination remote vlan *vlan*** command, and in a similar way the destination switch is configured to use the RSPAN VLAN as the source of the SPAN session through use of the **monitor session** *number* **source remote vlan** *vlan* command.

The RSPAN VLAN needs to match on the source and destination switches, but the session numbers do not need to match. The session numbers are local identifiers that define the relationship between sources and destinations for a session on that switch. The session numbers are not communicated between switches.



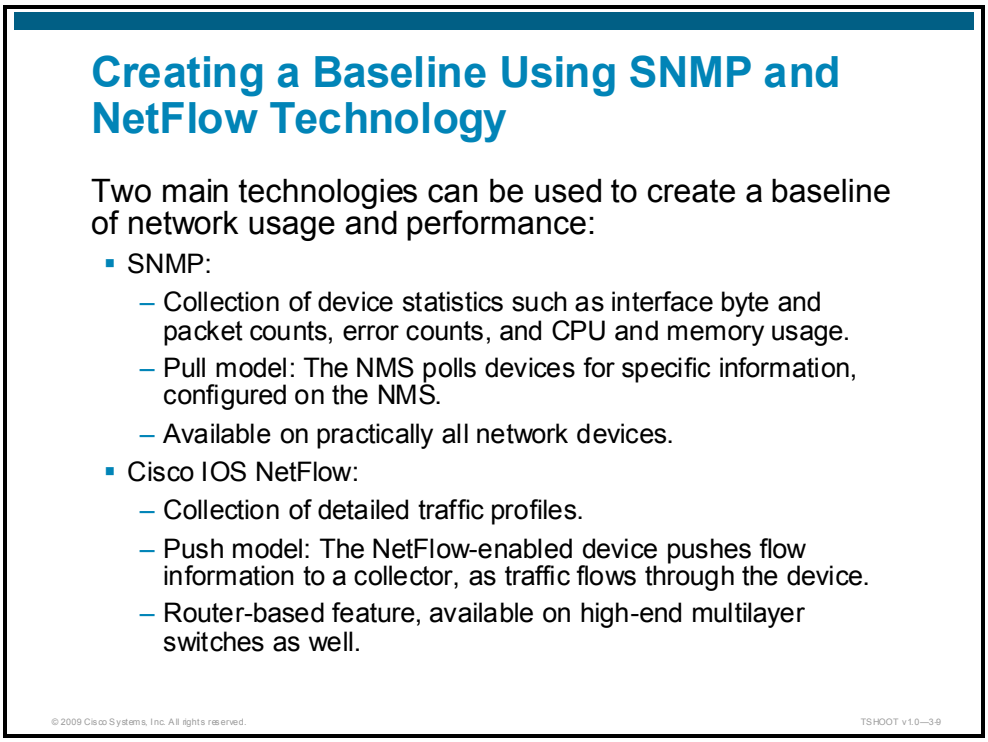
Again, you can use the **show monitor** command to verify the configuration of the SPAN session. Notice that on the source switch, the session is identified as "Remote Source Session," while on the destination switch it is marked as "Remote Destination Session."

In addition to verifying the correct configuration of the RSPAN session, it is also important that you verify that the VLAN is configured correctly as a RSPAN VLAN on both switches. The **show vlan remote-span** command is useful for this verification.

Finally, if VLANs are pruned on the trunks between the switches, you should verify that the RSPAN VLAN is allowed on those trunks.

Statistics Gathering and Traffic Accounting

This topic describes the use of SNMP-based or NetFlow-based systems to gather statistics in order to create a network baseline and explains how to configure the network devices to communicate with these systems.



Two main technologies are used to gather statistics from Cisco switches and routers: SNMP and NetFlow. Although there is a certain amount of overlap between the data that you can collect by the use of these protocols, they have a different focus.

SNMP primarily focuses on the collection of device statistics. Routers and switches (and other network devices) keep statistics about the operation of their processes and interfaces locally. These statistics can be viewed through the CLI or the GUI. This is fine if all you need is a snapshot view of the statistics or parameters at a particular moment in time, but if you want to collect and analyze these statistics over time, neither the CLI nor the GUI is very well suited to gather these statistics.

The SNMP protocol solves this issue by running a special process, called an agent, on a device. You can query this agent by using the SNMP protocol to obtain the values of statistics or parameters. By periodically querying or “polling” the SNMP agent running on a device, statistics can be gathered and collected over time by a network management station (NMS). This data can then be processed and analyzed in various ways. Averages, minimums, and maximums can be calculated, the data can be graphed, and thresholds can be set to trigger a notification process when they are exceeded.

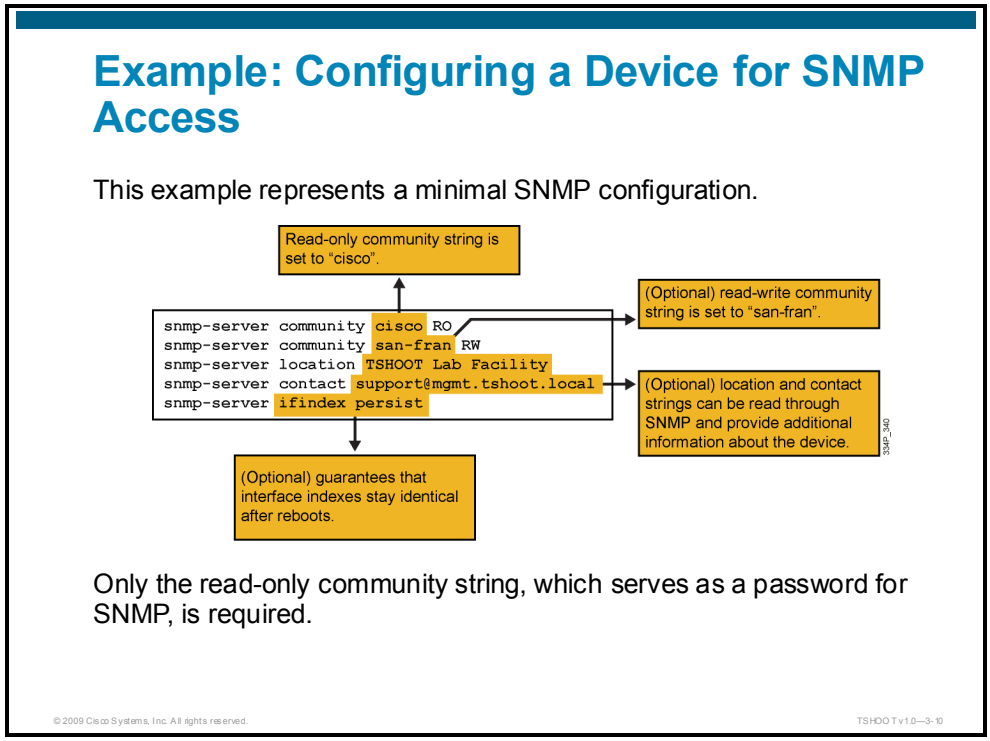
Statistics gathering with SNMP is essentially a “pull”-based system, where the NMS polls devices periodically to obtain the values of the MIB objects that it is set up to collect.

NetFlow has a different focus and uses different underlying mechanisms. A NetFlow-enabled device, such as a router or Layer 3 switch (currently only supported on the 4500 and 6500 series switches), collects information about the IP traffic that is flowing through the device. The NetFlow feature classifies traffic by flow. A flow is identified as a collection of packets that have the same essential header fields, such as ingress interface, source and destination IP address, protocol number, type of service (ToS) field, and, if applicable, port number. For each individual flow, the number of packets and bytes is tracked and accounted. This information is kept in a flow cache, and flows are expired from the cache when the flows are terminated or time out, or when the cache is full.

You can enable this feature as a standalone feature on a router, and it allows you to examine the NetFlow cache through the use of CLI commands. This can be a useful tool during troubleshooting, because it allows you to see the flow entries being created as packets enter the router. In that sense, you could utilize NetFlow as a diagnostic tool. However, the biggest strength of the NetFlow technology lies in the fact that instead of only keeping a local cache and temporarily accounting flows on the device itself, you can export the flow information to a NetFlow collector. Before entries are expired from the cache, the flow information, consisting of the main packet headers and additional information (such as packet and byte counts, egress interface, and flow start and duration), is sent to a collector that receives the flow information and records it in a database. Although the content of the packets themselves is not recorded, the flow information that is transferred to the collector by the router essentially contains a full view of all the traffic that has flowed through the router. Enabling NetFlow and exporting the flows from a number of key routers can yield a fairly complete view of all the traffic on the network. After collection, the NetFlow data can be processed and graphed.

In contrast with SNMP, NetFlow uses a “push”-based model. The collector simply listens to NetFlow traffic, and the routers are in charge of sending NetFlow data to the collector, based on changes in their flow cache.

Another difference between NetFlow and SNMP is that NetFlow gathers only traffic statistics, whereas SNMP can also collect many other performance indicators such as interface errors and CPU and memory usage. On the other hand, the traffic statistics collected using NetFlow have a higher degree of granularity than the traffic statistics collected using SNMP.



To configure a router for SNMP-based access is fairly simple. Although SNMP version 3 is the official current standard, version 2c is still the most widely used version. In SNMP version 2c, access to the SNMP agent is granted based on an SNMP community string. In general two different SNMP community strings are defined, one for read-only access and a different one for read-write access. For statistics gathering, only read access is required, and therefore a read-write community is optional and does not need to be defined.

Although it is not strictly necessary, it is good habit to define the SNMP contact and location. These parameters can be read via SNMP and define the support contact and physical location of the device.

Another command that is useful, especially when creating a baseline or graphing interface-related variables, is the **snmp-server ifindex persist** command. This command guarantees that the SNMP interface index for each interface will stay the same even if the device is rebooted. Without this command, you could encounter instances where the interface's ifindex changes after a reboot and counters for that interface are no longer correctly graphed.

For increased security, access lists can be defined to allow only SNMP access from certain locations. In addition, in scenarios where access needs to be granted for only a small collection of MIB objects, an SNMP view can be defined together with a corresponding community string. This limits access, by use of that community string, to only the MIB objects that are defined in the view and no others.

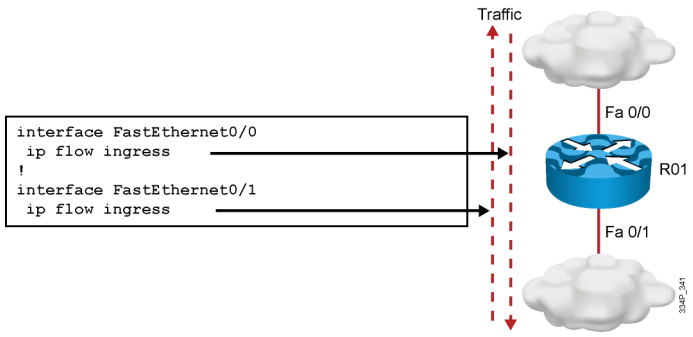
SNMP version 3 offers enhanced security, through authentication and encryption of SNMP access.

For more information about SNMP, including SNMPv3, see

http://www.cisco.com/en/US/docs/ios/netmgmt/configuration/guide/nm_cfg_snmp_sup_ps6350_TSD_Products_Configuration_Guide_Chapter.html.

Example: Configuring a Device for NetFlow Accounting

In this example, ingress NetFlow accounting is turned on for both interfaces on RO1 to account for both outbound and inbound traffic.



The **ip flow ingress** command replaces the older **ip route-cache flow** command

The figure shows how NetFlow accounting for ingress flows is enabled on both interfaces. The definition of a flow is unidirectional, so if you want to account for both inbound and outbound traffic, the feature needs to be turned on for both interfaces. In older Cisco IOS versions, the command to enable NetFlow on an interface was **ip route-cache flow**, but this has been replaced in newer versions with the **ip flow ingress** command.

Example: Examining the NetFlow Cache

The **show ip cache flow** command can be used to display the content of the cache.

```
RO1#show ip cache flow
<…output omitted…>

SrcIf         SrcIPaddress    DstIf         DstIPaddress    Pr SrcP DstP  Pkts
Se0/0/0.121   10.1.194.10     Null          224.0.0.10      58 0000 0000    27
Se0/0/0.122   10.1.194.14     Null          224.0.0.10      58 0000 0000    28
Fa0/0         10.1.192.5      Null          224.0.0.10      58 0000 0000    28
Fa0/1         10.1.192.13     Null          224.0.0.10      58 0000 0000    27
Fa0/1         10.1.152.1      Local         10.1.220.2      01 0000 0303     1
Se0/0/1       10.1.193.6      Null          224.0.0.10      58 0000 0000    28
Fa0/1         10.1.152.1      Se0/0/1       10.1.163.193    11 0666 E75E  1906
Se0/0/1       10.1.163.193    Fa0/0         10.1.152.1      11 E75E 0666  1905
```

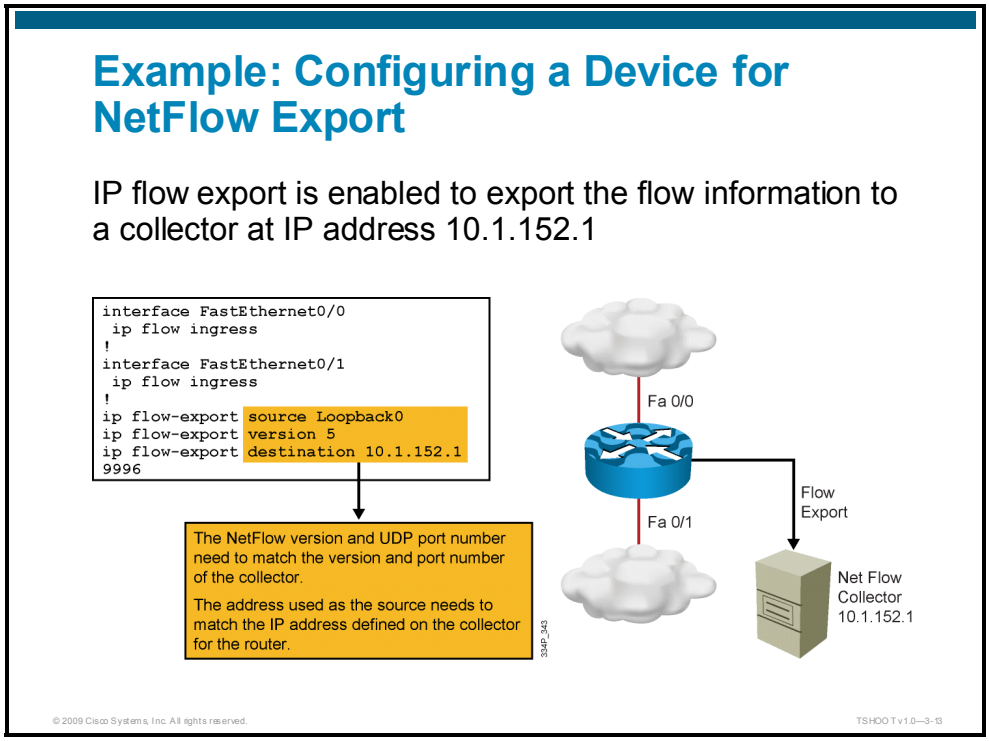
334P_342

This command can be a powerful tool for troubleshooting connection issues.

In this example, it confirms that traffic is flowing between host 10.1.152.1 and host 10.1.163.193.

After configuring these initial commands, the router starts caching and accounting flow information locally in its memory. You can display the cache by using the **show ip cache flow** command. This command can be very useful when you are troubleshooting connection problems, because it shows the active flows that are sending packets through the router.

The output filtering options for **show** commands can be used to select only those IP addresses that you are interested in. For example, for the sample output in the figure the command **show ip cache flow | include 10.1.163.193** could have been used to limit the output to only those flows that have 10.1.163.193 as the source or destination IP address.



To export the flows to a NetFlow collector, three extra items need to be configured.

First, the version of the NetFlow protocol should be configured. The most commonly used and supported version is NetFlow version 5. The most current and flexible version is NetFlow version 9, which is recommended if your collector supports it. Consult the documentation of your collector to find out which versions of NetFlow are supported.

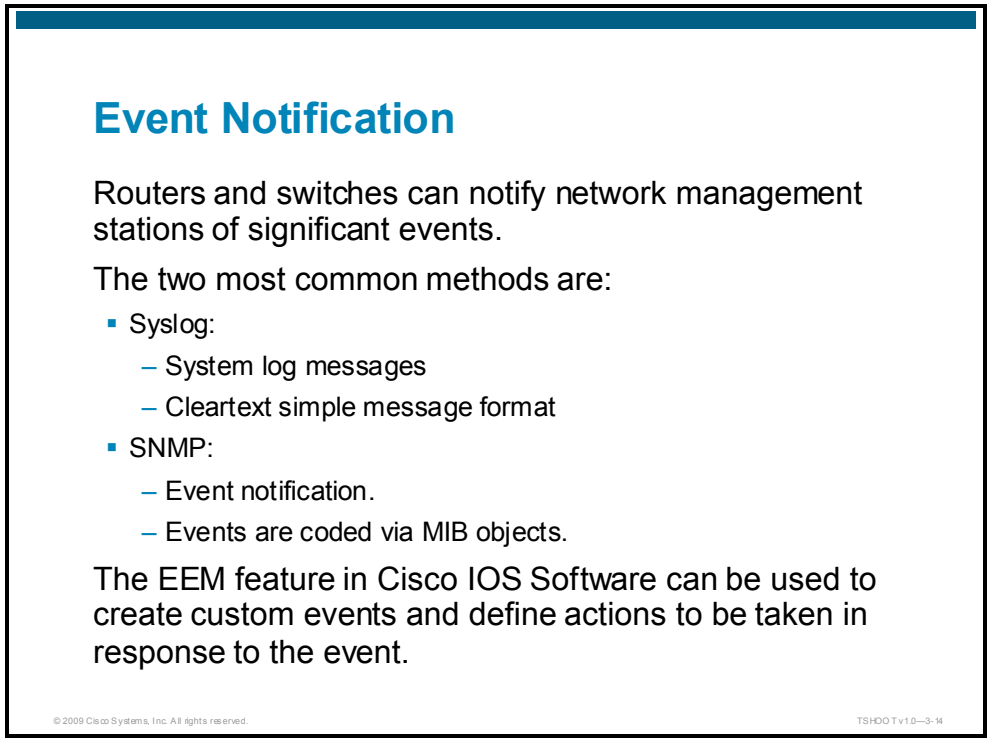
Second, the IP address and User Datagram Protocol (UDP) port number of the collector need to be configured. There is no default port number for NetFlow, so you should check the documentation of your collector to ensure that the port number on your collector and the exporting router match.

Third, a collector is usually configured with the IP addresses of all routers that can send flow information to the collector. The collector verifies that the source IP address of the incoming packets matches the configured IP address. Therefore, it is important that the NetFlow packets are always sourced from the same interface. Using a loopback interface as the source interface for NetFlow ensures that the packets will always be sourced from the same address, regardless of the interface that is used to transmit the NetFlow packets.

For more information about NetFlow configuration, see the Cisco IOS NetFlow configuration guide:
http://www.cisco.com/en/US/docs/ios/netflow/configuration/guide/12_4/nf_12_4_book.html.

Notification

This topic describes how syslog and SNMP can be used to notify network management stations of the occurrence of significant events.



A key element of a proactive network management strategy is fault notification. When a significant event happens on your network, you do not want to wait for users to start reporting problems caused by the event; you want the network devices to report that event to a central system, so that you will be aware of the issue before problems associated to the event are being reported. In addition to being aware of the problem earlier, you also have the advantage of getting a report of the underlying event instead of only a description of symptoms.

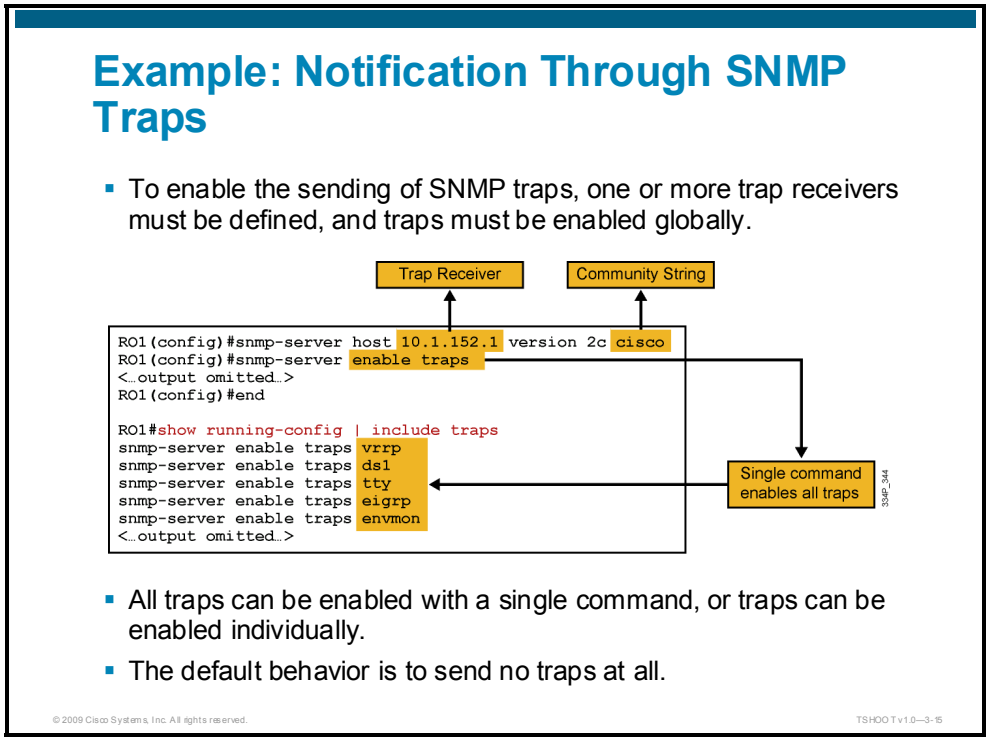
The two protocols that are most commonly used for this purpose are syslog and SNMP.

Syslog is a simple protocol that sends text-based log messages. These messages are the same messages that are displayed on the console of the device. The syslog protocol allows these messages to be forwarded across the network to a central log server that collects and stores the messages from all the devices. By itself, this constitutes only a very basic form of event notification. The network device notifies the log server and the log message is stored, but who notifies the network support team that a significant event has happened? Syslog capabilities are included as a component of many network management systems, and these systems often include advanced mechanisms to notify network support engineers of significant events.

As seen in the previous topic, SNMP allows an agent running on a network device to be queried by an SNMP manager for statistics. In addition to responding to polling, the agent can also be enabled to send messages to the SNMP manager based on the occurrence of events such as an interface going down, device configuration, and so on. These messages, called traps, do not contain user-readable text, but SNMP MIB objects and associated variables. Therefore, these messages will always need to be processed by an SNMP-based network management

system that is capable of interpreting and processing the MIB object information contained in the trap.

Both syslog messages and SNMP traps use predefined messages that are embedded in Cisco IOS Software. They will trigger on predefined conditions, and the content of the message is also fixed. The number of defined syslog messages and SNMP traps is very extensive, and as a result, they will fulfill the fault notification needs of most organizations. However, there could be special cases where you want to be notified of a particular condition or event that is not part of the standard collection of log messages and events included in Cisco IOS Software. For these special cases, Cisco IOS Software contains a feature called the Cisco IOS Embedded Event Manager (EEM), which allows you to define custom events and corresponding actions.



In the example shown in the figure, SNMP trap notification is enabled. This involves two major steps: defining one or more trap receivers, and enabling traps.

Trap receivers are configured using the **snmp-server host** *host* **traps [version {1 | 2c | 3}]** *community-string* command. By default, SNMP version 1 traps are sent. Higher versions need to be configured explicitly. If a particular trap receiver needs to receive only specific traps, this can be configured by specifying the intended trap categories as an additional option.

To enable the sending of SNMP traps globally, each category of traps can be individually enabled by using the **snmp-server enable traps** *notification-type* command. To enable all categories by use of a single command, the command **snmp-server enable traps** can be used. This command does not appear in the configuration as a command, but executes a macro that enables all available categories of traps. You can see this effect in the output of the **show running-config | include traps** command in the example.

Example: Cisco IOS EEM Configuration

In this example, an EEM applet is created that does the following:

- When the configure terminal command is entered:
  - Log a critical message to the console and any other log destinations that states that configuration mode was entered.
  - Log a notification message to the console and any other log destinations that reminds the engineer of the change control policies that apply.

```
event manager applet CONFIG-STARTED
event cli pattern "configure terminal" sync no skip no occurs 1
action 1.0 syslog priority critical msg "Configuration mode was entered"
action 2.0 syslog priority informational msg "Change control policies
apply. Authorized access only."
```


SNMP and syslog both act on predefined triggers and send predefined messages. Both protocols allow for a limited amount of filtering, but it is not possible to define entirely new event triggers or messages.

The EEM framework enables the creation of custom policies that trigger actions based on events. Events can be triggered based on various Cisco IOS subsystems such as syslog messages, Cisco IOS counter changes, SNMP MIB object changes or traps, CLI command execution, timers, and many others. Actions can consist of sending SNMP traps and syslog messages, executing CLI commands, sending email, or even running Tool Command Language (TCL) scripts. This allows for the creation of very powerful and complex policies.

The figure shows one of the many possible applications of EEM.

Imagine that you have the following policy in your organization: All network engineers get privileged access to the routers and switches and can make changes if necessary. However, only level 3 support engineers are allowed to make emergency changes if required. Level 1 and 2 engineers always need to obtain authorization before making any change to the system. Whenever an engineer configures a router or switch, a %SYS-5-CONFIG_I message is logged to the syslog server. However, this is message is logged as a syslog level 5 “notification” message and does not show up in the logs as a high-priority item. You want to change the behavior of the router to the following: A message should be logged as soon as anybody enters configuration mode, in addition to the %SYS-5-CONFIG_I message that is logged after leaving configuration mode. This message should be logged as a critical message. Secondly, an informational message should be logged reminding the engineer of the existing change control policies.

The EEM applet shown in the figure achieves this as follows:

- The applet is created and named “CONFIG-STARTED” using the **event manager applet CONFIG-STARTED** command.
- The event that should trigger this applet is defined by using the command **event cli pattern "configure terminal" sync no skip no occurs 1**. This line effectively says that the policy should be triggered if a command that includes “configure terminal” is entered in the CLI. The **occurs 1** option forces the event to be triggered on a single occurrence of the CLI pattern.
- An action named “1.0” is defined (actions are sorted in alphabetical order) by using the **action 1.0 syslog priority critical msg "Configuration mode was entered"** command. This command tells the router to log a critical message containing the text “Configuration mode was entered.”
- An action named “1.0” is defined (actions are sorted in alphabetical order) by using the **action 2.0 syslog priority informational msg “Change control policies apply. Authorized access only."** command. This command tells the router to log an informational message containing the text “Change control policies apply. Authorized access only.”

Example: EEM Configuration (Cont.)

When the configuration mode is entered on the router, this messages appears on the console:

```
RO1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
CRO1(config)#
Mar 13 03:24:41.473 PDT: %HA_EM-2-LOG: CONFIG-STARTED: Configuration
mode was entered

Mar 13 03:24:41.473 PDT: %HA_EM-6-LOG: CONFIG-STARTED: Change control
policies apply. Authorized access only.
```

EEM supports many other event triggers and actions. Refer to the EEM documentation for more information.

This figure shows the effect of the configured EEM policy. As soon as a user enters configuration mode, two messages appear, one critical message (syslog level 2) stating “%HA_EM-2-LOG: CONFIG-STARTED: Configuration mode was entered” and one informational message (syslog level 6) “%HA_EM-6-LOG: CONFIG-STARTED: Change control policies apply. Authorized access only.”

This is a very simple example of the operation of EEM. EEM is a very powerful tool, and by incorporating the use of the TCL scripting language, the system can be built out into a complete distributed notification system.

For more information about the Cisco IOS Embedded Event Manager, see:

http://www.cisco.com/go/eem

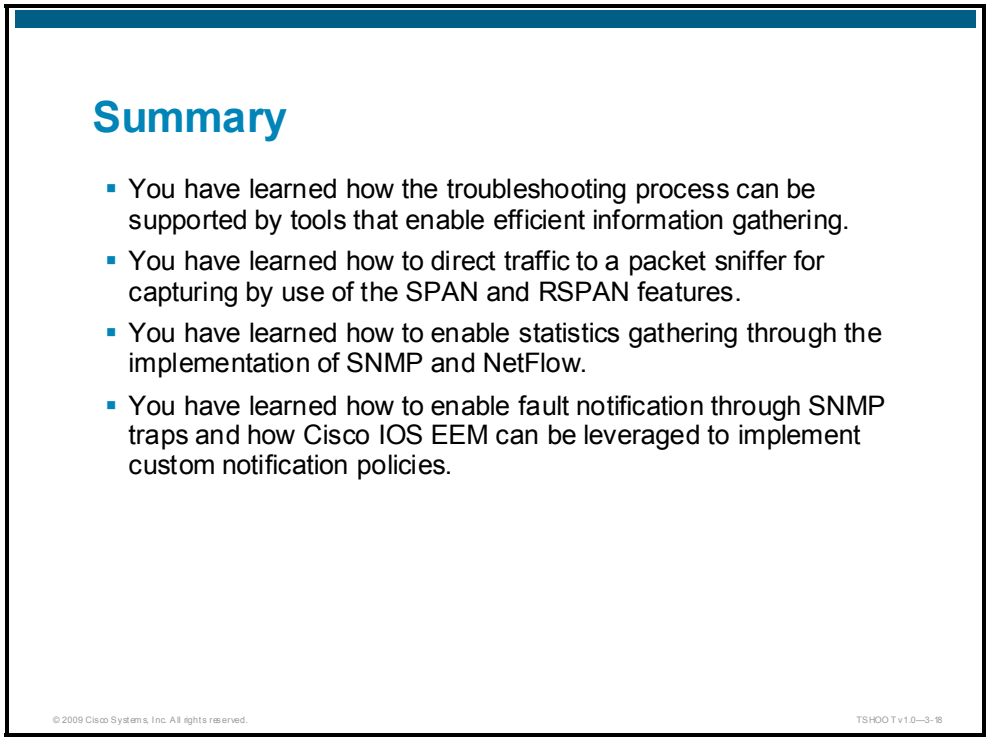
Good examples of EEM TCL scripts can be found at the EEM scripting community pages on Cisco Beyond:

http://forums.cisco.com/eforum/servlet/EEM?page=main

For more information about the Cisco IOS Embedded Event Manager, see:
http://www.cisco.com/go/eem

Summary

This topic summarizes the key points that were discussed in this lesson.



Lesson 3

Lab 3-1 Debrief

Overview

Knowing the environment in which you are troubleshooting is fundamental to efficient fault resolution. Gathering and creating up-to-date and correct documentation and assembling a troubleshooting and maintenance toolkit are well worth spending time on, because the effort invested in this will pay off during outages, when every minute counts.

In this lab, you have surveyed the network, reviewed and supplemented the documentation, and explored the available troubleshooting and maintenance toolkit.

During the lab debrief, the instructor will lead a group discussion during which you can validate your findings and clear up any remaining questions about the lab topology and tools.

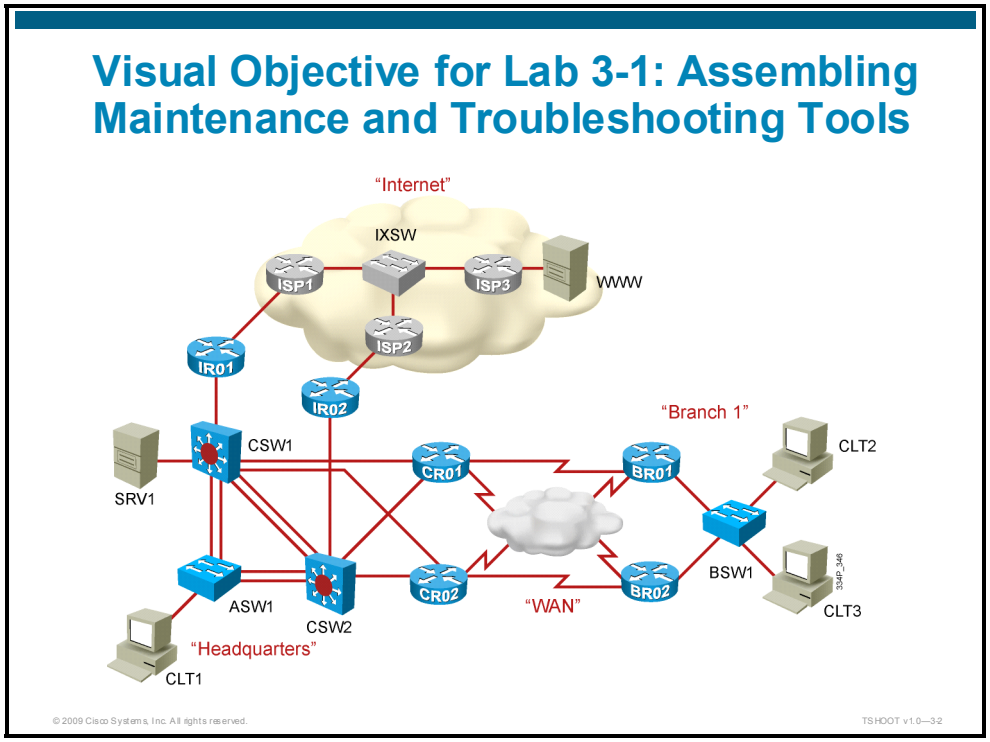
Objectives

Upon completing this lesson, you will be able to describe the primary aspects of the baseline design and implementation of the lab network. This ability includes being able to meet these objectives:

- Validate your understanding of the network and available tools and fill in any potential gaps in your documentation
- Consolidate the lessons learned during the review discussions into a set of best-practice methods and commands to aid in future troubleshooting procedures

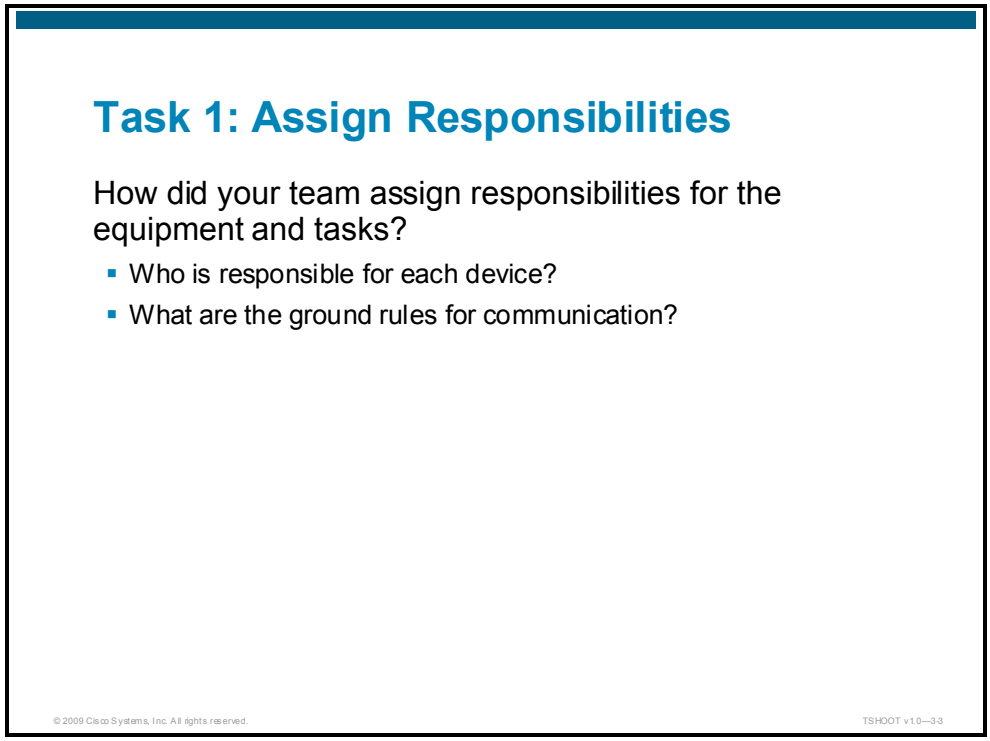
Review and Verification

This topic reviews the topology of the lab network and the implemented technologies, features and tools.



The lab pod consists of three separate areas: A headquarters site, consisting of a switched campus, WAN routers to connect to the branch offices, and Internet routers to connect to two Internet service providers (ISPs). The routers and switches that represent the Internet are managed by the instructor and are not accessible to you as a student. If you ever suspect problems with these ISP connections, you should escalate the problem to your instructor, similarly to how you would escalate a problem to an ISP in your job role.

The equipment in a single pod consists of six routers and four switches, and you will be responsible for maintaining and troubleshooting this network as a team.



To prevent situations where you and your teammates are getting in each other's way and to facilitate smooth troubleshooting, you should establish responsibilities and communication procedures within your team.

Sample responsibilities and ground rules for a team of four members could look like this table:

Device	Responsible Team Member
ASW1	Team member 1
CSW1	Team member 2
CSW2	Team member 3
IRO1	Team member 1
IRO2	Team member 1
CRO1	Team member 2
CRO2	Team member 3
BRO1	Team member 4
BRO2	Team member 4
BSW1	Team member 4

Ground rules for working together and communicating are as follows:

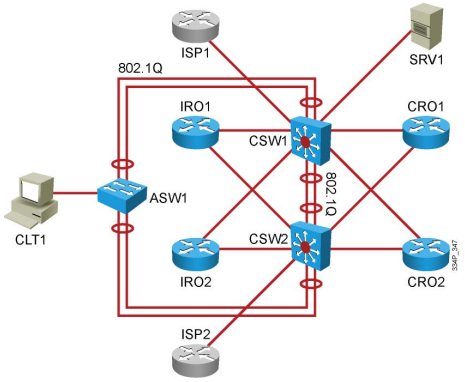
- The person that is in control of a particular device is the only person allowed to connect to that device's console port. Team members may use Telnet or Secure Shell (SSH) to access all devices and use **show** commands to diagnose, but any disruptive actions such as making changes to the configuration, reloading the device, or starting debugging can be done only with the permission of the controlling team member.
- You should communicate problems diagnosed and changes made to the configurations to all team members.

You are free to work according to different ground rules or to reassign responsibilities for each lab, as long as your work plan ensures that the actions of one team member will not spoil the lab experience for the other team members.

Task 2: Review the Physical Lab Topology

Do you have any questions about the physical lab topology of the headquarters LAN?

Did you find anything remarkable about the implementation?



The physical design of the headquarters LAN is a relatively standard campus LAN design using Layer 2 switching in the access layer and multilayer switching in a collapsed core and distribution layer.

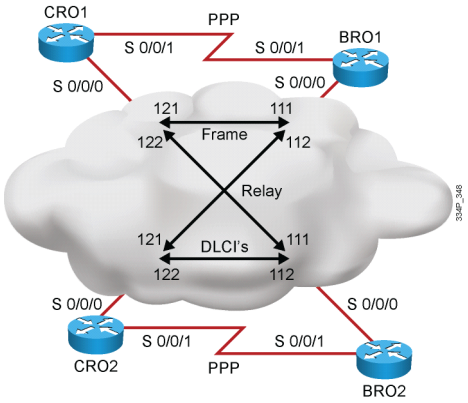
Some points that are worth some attention for later troubleshooting exercises are the following:

- EtherChannel technology has been used to bundle the two redundant links between the switches.
- VLANs have been pruned on the trunks to allow only those VLANs that are currently implemented and that are necessary.
- The ISP routers are not directly connected to the IRO1 and IRO2 routers, but the links to the ISP routers are connected to the switches CSW1 and CSW2. The connection to the routers IRO1 and IRO2 is made using two VLANs (VLAN 11 and 12) that are not routed on the switches but use Layer 2 switching to pass the traffic between router IRO1 and ISP1, and IRO2 and ISP2, respectively.
- The connections between the core switches and the WAN routers use a routed port, and the connections between the core switches and the Internet routers use a transit VLAN (VLAN 129).
- Rapid spanning tree is implemented, and the switches CSW1 and CSW2 are the root bridges for different sets of VLANs.

Task 2: Review the Physical Lab Topology (Cont.)

Do you have any questions about the physical lab topology of the WAN?

Did you find anything remarkable about the implementation?



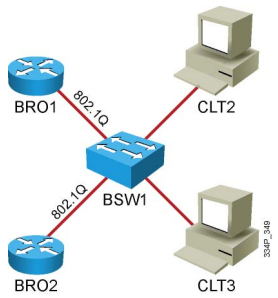
The physical design of the headquarters WAN is very redundant and includes both Frame Relay and PPP links. In some labs, only one of the two types of WAN might be used and the interfaces of the other type might be shut down.

It is worth noticing that because the lab is physically implemented using serial crossover cables, and CSU/DSUs are not used, the router that is the physical DCE is configured to provide the clock signal for the serial link running PPP. For the Frame Relay links, the Frame Relay switch provides clocking.

Task 2: Review the Physical Lab Topology (Cont.)

Do you have any questions about the physical lab topology of the branch LAN?

Did you find anything remarkable about the implementation?

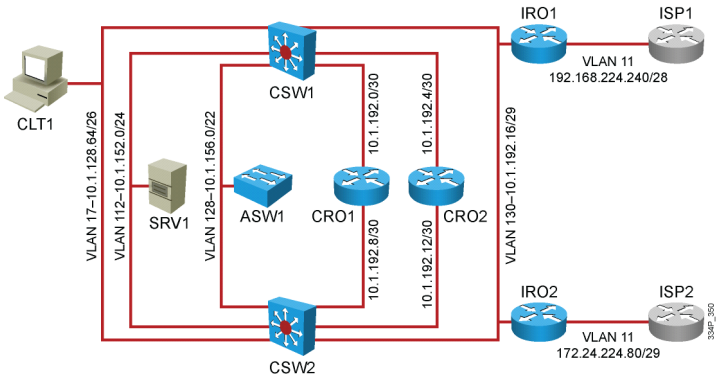


The branch LAN design is fairly straightforward. Both routers BRO1 and BRO2 are connected to switch BSW1 via trunks, and both routers perform inter-VLAN routing. The switch performs only Layer 2 switching.

Task 3: Review the Logical Lab Topology

Do you have any questions about the logical lab topology of the headquarters LAN?

Did you find anything remarkable about the implementation?



The backbone of the headquarters LAN consists of the two switches CSW1 and CSW2. The Enhanced Interior Gateway Routing Protocol (EIGRP) is used to route to the Internet routers and WAN routers.

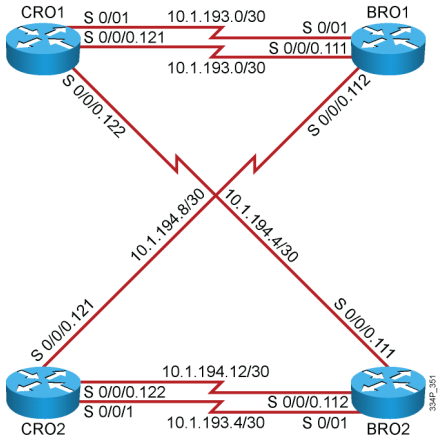
Some points that are worth attention for later troubleshooting exercises are the following:

- The routers IRO1 and IRO2 perform network address translation (NAT) toward the ISP routers. They both inject a default route into EIGRP to enable routing to the Internet from the other routers.
- The switches CSW1 and CSW2 are configured for the Hot Standby Router Protocol (HSRP) to provide redundant default gateways to all devices in the office, guest, voice, server, and management VLANs.
- The switches CSW1 and CSW2 act as Dynamic Host Configuration Protocol (DHCP) servers for the office, guest, and voice VLANs.

Task 3: Review the Logical Lab Topology (Cont.)

Do you have any questions about the logical lab topology of the WAN?

Did you find anything remarkable about the implementation?

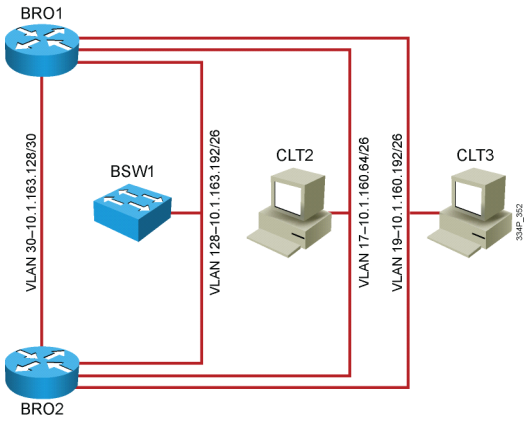


The logical topology of the WAN is straightforward, using separate point-to-point subinterfaces for each Frame Relay permanent virtual circuit (PVC).

Task 3: Review the Logical Lab Topology (Cont.)

Do you have any questions about the logical lab topology of the branch LAN?

Did you find anything remarkable about the implementation?

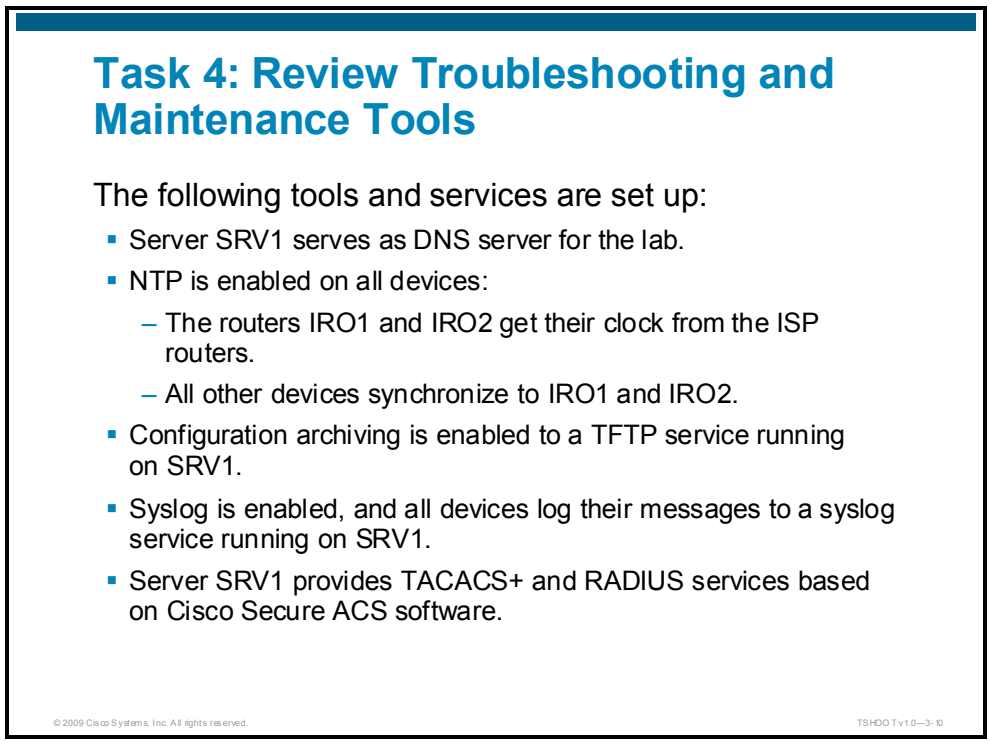


The logical topology of the branch LAN is a simplified version of the headquarters LAN, using the routers BRO1 and BRO2 to route between the VLANs. A noteworthy difference is that in this case the Gateway Load Balancing Protocol (GLBP) is implemented instead of HSRP to load-balance outbound traffic to the WAN and provide redundant default gateways to the hosts.

As with the situation at headquarters, the routers BRO1 and BRO2 act as DHCP servers for the office, voice, and guest VLANs.

It is also worth noting that for the guest VLAN an access list has been implemented on the routers, which implements the following policy:

- Permit all Internet Control Message Protocol (ICMP) traffic for troubleshooting purposes
- Permit GLBP traffic to ensure proper GLBP operation for the guest VLAN
- Permit Domain Name System (DNS) to the DNS server SRV1 to provide name resolution to guest PCs
- Permit DHCP client side traffic to enable automatic IP address assignment
- Deny all traffic to the corporate networks included in the prefix 10.1.128.0/17
- Permit any other traffic to allow Internet access



The following maintenance tools and services are present in the lab: DNS, syslog, and TFTP services are provided on SRV1. Network Time Protocol (NTP) service is provided through the ISP routers in the backbone. NetFlow and Simple Network Management Protocol (SNMP) traps are enabled to communicate with SRV1. It can depend on the lab provider whether actual network management software has been installed to process the NetFlow and SNMP information.

To support these tools and services the following configuration sections can be found in the configurations of the devices:

To send DNS requests to SRV1 and to define the default domain:

```
ip name-server 10.1.152.1
ip domain-name mgmt.tshoot.local
```

To send syslog messages to SRV1 and to buffers in RAM on the router and to limit the output on the console to messages of level 4 (warning) or lower:

```
logging buffered 16384
logging console warnings
logging 10.1.152.1
```

To archive configurations to SRV1 via TFTP any time the configuration is saved, and to log all configuration changes to syslog, the following is configured:

```
archive
 log config
  logging enable
  logging size 50
  notify syslog
  hidekeys
 path tftp://srv1.mgmt.tshoot.local/$h-archive-config
 write-memory
```

To synchronize the clock to IRO1 or IRO2 (which themselves synchronize to the ISP routers), the following is configured:

```
ntp server 10.1.220.4
ntp server 10.1.220.3
```

The time zone and summertime are configured for the US Pacific time zone and the time stamps reflect these time stamps as configured by the following commands:

```
clock timezone PST -8
clock summer-time PDT recurring
service timestamps debug datetime msec localtime show-timezone
service timestamps log datetime localtime show-timezone
```

Cisco Secure Access Control Server (ACS) software has been installed to provide TACACS+ and RADIUS services to the network devices to be used in future labs. Currently, authentication, authorization, and accounting (AAA) have been configured to use the local username and password database for Telnet and SSH access. Console access has been configured so that authentication for user mode access is not required. This is configured using the following commands:

```
aaa new-model
aaa authentication login default local
aaa authentication login CONSOLE none
aaa authorization exec default local
line con 0
 login authentication CONSOLE
```


Task 4: Review Troubleshooting and Maintenance Tools (Cont.)

The following tools and services are set up:

- SNMP traps are enabled:
  - All devices send their traps to server SRV1.
  - Optionally, a trap receiver can be installed on server SRV1, but this is not essential to the labs.
- NetFlow is enabled on routers IRO1, IRO2, CRO1, and CRO2:
  - Flows are exported to server SRV1.
  - Optionally, flow collector software can be installed on server SRV1, but this is not essential to the labs.

Packet-capturing software and other tools are available from the Tools folder on the server.

An SNMP read-only community string of "cisco" and a read-write community string of "san-fran" is configured; SRV1 is defined as a trap host and all traps are enabled. Additionally, an SNMP location and contact are defined:

```
snmp-server community cisco RO
snmp-server community san-fran RW
snmp-server trap-source Vlan128
snmp-server location TSHOOT Lab Facility
snmp-server contact support@mgmt.tshoot.local
snmp-server host 10.1.152.1 version 2c cisco
snmp ifmib ifindex persist
snmp-server enable traps
```

On CRO1 and CRO2 NetFlow accounting is enabled and flows are exported to SRV1:

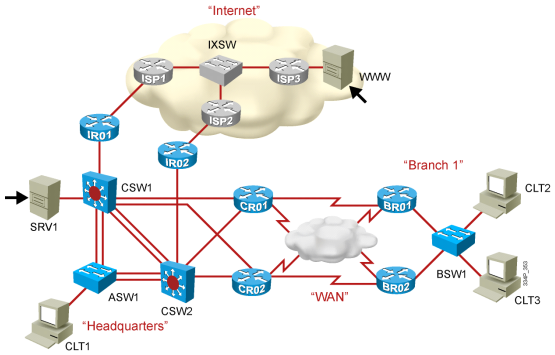
```
interface FastEthernet0/0
 ip flow ingress
interface FastEthernet0/1
 ip flow ingress
interface Serial0/0/0
 ip flow ingress
interface Serial0/0/1
 ip flow ingress
ip flow-export source Loopback0
ip flow-export version 5
ip flow-export destination 10.1.152.1 9996
```

In a similar way, NetFlow is enabled on IRO1 and IRO2.

Connectivity Testing

During troubleshooting procedures, connectivity can be tested as follows:

- By accessing the shared folder \\SRV1\Public on server SRV1 (10.1.152.1) from client PCs CLT1 and CLT2
- By accessing the web page http://www.isp3.local on server WWW (172.34.224.1) from client PCs CLT1, CLT2, and CLT3



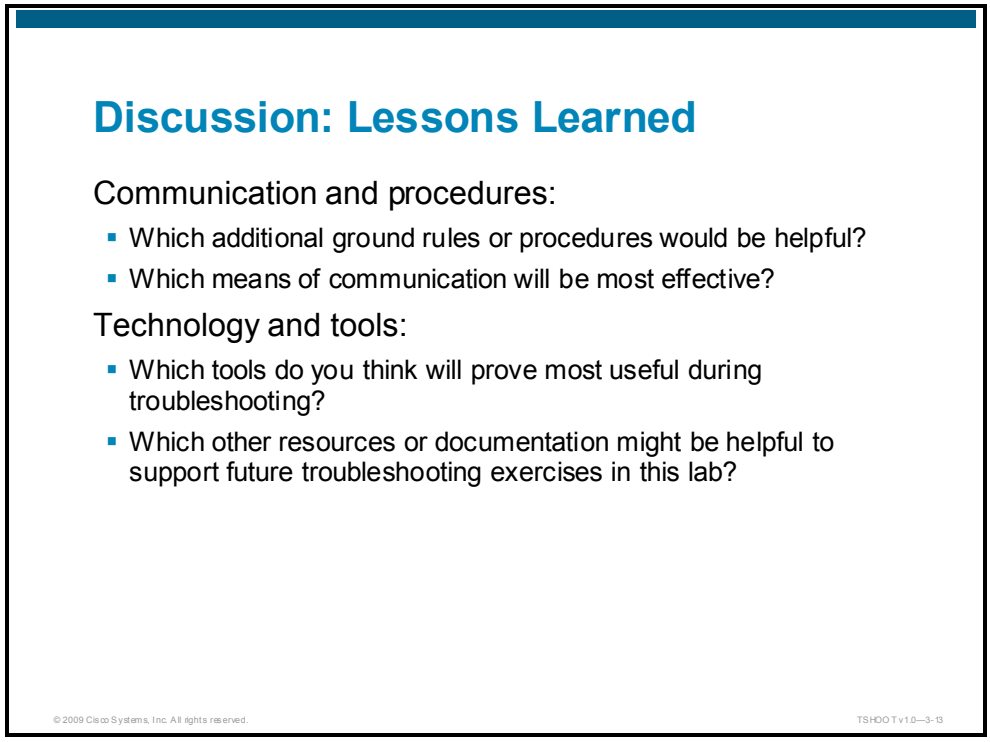
To test connectivity during future troubleshooting labs it is important to understand which services are being provided. The most common tests that you can execute to verify proper operation of the network are the following:

- Opening the shared folder \\SRV1\Public on server SRV1 from client PCs CLT1 and CLT2. This can be done manually by connecting a network drive on the client PCs or by clicking an icon provided on the desktop of the client PC.
- Opening the web page http://www.isp3.local using a browser on client PCs CLT1, CLT2, and CLT3.

It is also important to note the IP addresses of these two servers so that you can test based on IP address instead of hostname when DNS is not functioning correctly.

Consolidation

This topic describes the key lessons that can be learned from the lab exercise.

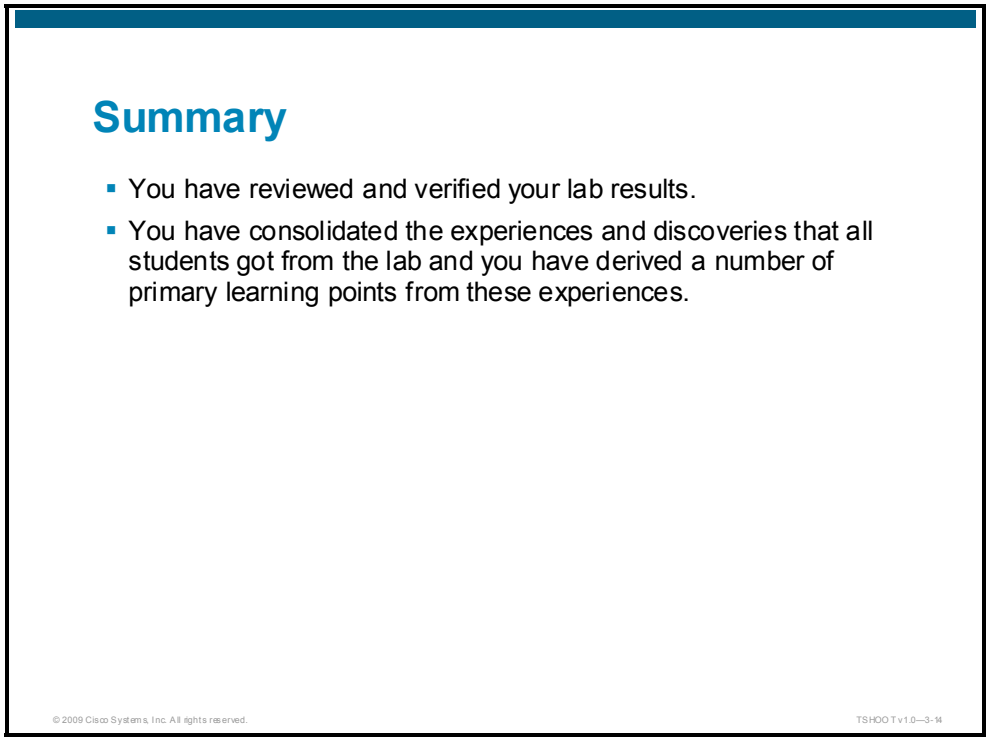


Think about all the things that you learned during the lab itself and during the debrief discussions. There is room to write down primary learning points in the Lab Debrief Notes section of the lab guide.

In addition to thinking of the methods, processes, and tools as they were used in the lab, reflect on how these would apply to your own organization.

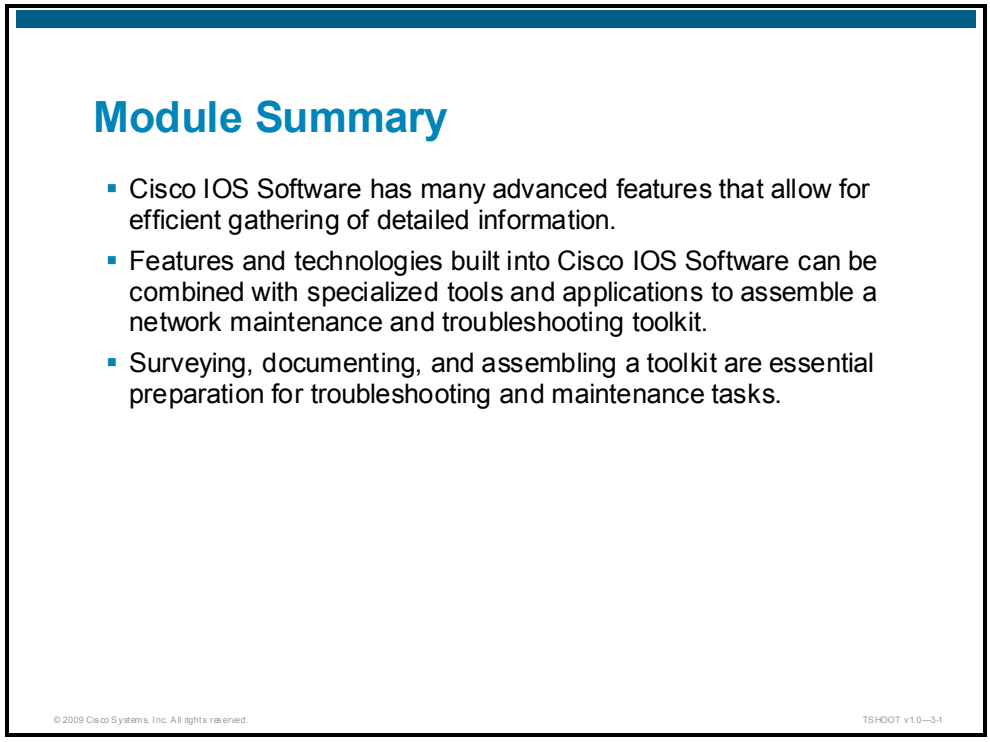
Summary

This topic summarizes the key points that were discussed in this lesson.



Module Summary

This topic summarizes the key points that were discussed in this module.



Gathering information is one of the most time-consuming processes that are executed in support of troubleshooting and maintenance tasks. Gathering information efficiently is therefore of the greatest importance. Cisco IOS Software contains many features that help select and gather information, test connections, and assess the basic health of the hardware of a device. Using these tools effectively and efficiently is essential to any network engineer.

Protocols and technologies, such as Switched Port Analyzer (SPAN), Simple Network Management Protocol (SNMP), syslog, and NetFlow can be used with specialized tools and applications in support of troubleshooting processes and maintenance processes. Preparing the infrastructure for the use of these tools and applications is an important task that is part of any network engineer's job.

Surveying a network, acquiring and creating documentation, and assembling a toolkit are fundamental steps to prepare for troubleshooting and network maintenance processes.

References

For additional information, refer to these resources:

- Cisco Systems, Inc., *Configuration Replace and Configuration Rollback:* http://www.cisco.com/en/US/docs/ios/12_3t/12_3t7/feature/guide/gtrollbk.html
- Cisco Systems, Inc., *Searching and Filtering CLI Output:* http://www.cisco.com/en/US/docs/ios/fundamentals/configuration/guide/cf_cli-basics_ps6350_TSD_Products_Configuration_Guide_Chapter.html#wp1002025
- Cisco Systems, Inc., *Show Command Output Redirection:* http://www.cisco.com/en/US/docs/ios/fundamentals/configuration/guide/feat_show_cmd_redrct_ps6350_TSD_Products_Configuration_Guide_Chapter.html
- Cisco Systems, Inc., *Show Command Section Filter:* http://www.cisco.com/en/US/docs/ios/12_3t/12_3t2/feature/guide/gtshfltr.html
- Cisco Systems, Inc., *Understanding the Ping and Traceroute Commands:* http://www.cisco.com/en/US/products/sw/iosswrel/ps1831/products_tech_note09186a00800a6057.shtml
- Cisco Systems, Inc., *Resolve IP Fragmentation, MTU, MSS, and PMTUD Issues with GRE and IPSEC:* http://www.cisco.com/en/US/tech/tk827/tk369/technologies_white_paper09186a00800d6979.shtml
- Cisco Systems, Inc., *Troubleshooting Ethernet:* http://www.cisco.com/en/US/docs/internetworking/troubleshooting/guide/tr1904.html
- Cisco Systems, Inc., *The show processes Command:* http://www.cisco.com/en/US/products/sw/iosswrel/ps1828/products_tech_note09186a00800a65d0.shtml
- Cisco Systems, Inc., *Troubleshooting Memory Problems:* http://www.cisco.com/en/US/products/sw/iosswrel/ps1831/products_tech_note09186a00800a6f3a.shtml
- Cisco Systems, Inc., *Cisco Generic Online Diagnostics (GOLD):* http://www.cisco.com/en/US/products/ps7081/products_ios_protocol_group_home.html
- Cisco Systems, Inc., *Catalyst Switched Port Analyzer (SPAN) Configuration Example:* http://www.cisco.com/en/US/products/hw/switches/ps708/products_tech_note09186a008015c612.shtml
- Cisco Systems, Inc., *Configuring SNMP Support:* http://www.cisco.com/en/US/docs/ios/netmgmt/configuration/guide/nm_cfg_snmp_sup_ps6350_TSD_Products_Configuration_Guide_Chapter.html

Module Self-Check

Use the questions here to review what you learned in this module. The correct answers and solutions are found in the Module Self-Check Answer Key.

Q1) Which command will display all subnets that are contained in the prefix 10.1.32.0/19? (Source: Assembling a Basic Diagnostic Toolkit Using Cisco IOS Software)

A) **show ip route 10.1.32.0 /19 longer-prefixes**
B) **show ip route 10.1.32.0 255.255.224.0 subnets**
C) **show ip route 10.1.32.0 /19 subnets**
D) **show ip route 10.1.32.0 255.255.224.0 longer-prefixes**

Q2) You execute the command **show ip route 10.1.1.1** and the response of the router is "% Subnet not in table." Which conclusion can be drawn from this response? (Source: Assembling a Basic Diagnostic Toolkit Using Cisco IOS Software)

A) The host entry 10.1.1.1/32 is not in the routing table.
B) There is no route in the routing table that matches IP address 10.1.1.1. All packets to that destination will be dropped.
C) There is no specific route in the routing table that matches IP address 10.1.1.1. Packets to that destination might be forwarded by the default route, if it is present.
D) The classful network 10.0.0.0 is not present in the routing table.

Q3) Which command will display the part of the running configuration that contains all statements for the EIGRP routing protocol? (Source: Assembling a Basic Diagnostic Toolkit Using Cisco IOS Software)

A) **show running-config | section router eigrp**
B) **show running-config | include router eigrp**
C) **show running-config | exclude router eigrp**
D) **show running-config | start router eigrp**
E) none of the above; requires using a regular expression

Q4) Which command will display the output of the command **show ip interface brief** on screen and copy the output of the command to the file show-output.txt on TFTP server 10.1.1.1? (Source: Assembling a Basic Diagnostic Toolkit Using Cisco IOS Software)

A) **show ip interface brief | tee tftp://10.1.1.1/show-output.txt**
B) **show ip interface brief | append tftp://10.1.1.1/show-output.txt**
C) **show ip interface brief | redirect tftp://10.1.1.1/show-output.txt**
D) **show ip interface brief | copy tftp://10.1.1.1/show-output.txt**
E) none of the above; can only be copied to a file in the flash memory of the device

Q5) Which Cisco IOS command will send 154 ICMP request packets of 1400 bytes each with the Do Not Fragment bit set to IP address 10.1.1.1? (Source: Assembling a Basic Diagnostic Toolkit Using Cisco IOS Software)

A) **ping 10.1.1.1 –l 1400 –r 154 -f**
B) **ping 10.1.1.1 size 1400 repeat 154 df-bit**
C) **ping 10.1.1.1 repeat 154 size 1400 df 1**
D) none of the above; can be accomplished only by use of the extended **ping** interactive dialog

Q6) You execute the command **telnet 192.168.37.2 80** and the response of the router is "Trying 192.168.37.2, 80 ... Open." Which conclusion can be drawn from this response? (Source: Assembling a Basic Diagnostic Toolkit Using Cisco IOS Software)

A) The web server on host 192.168.37.2 is running and serving files.
B) There is a service running on TCP port 80 on host 192.168.37.2 and it accepts connections.
C) The server on 192.168.37.2 is accepting Telnet connections.
D) No conclusions can be drawn about the server. The word "Open" means only that the IP address could be found in the routing table.

Q7) You execute the command **show processes cpu** and the output includes "CPU utilization for five seconds: 30%/26%." Which two statements are correct? (Choose two.) (Source: Assembling a Basic Diagnostic Toolkit Using Cisco IOS Software)

A) The total CPU load over the past 5 seconds was 56 percent.
B) The total CPU load over the past 5 seconds was 30 percent.
C) The percentage of CPU time spent on scheduled processes was 26 percent.
D) The percentage of CPU time spent on scheduled processes was 30 percent.
E) The percentage of CPU time spent on scheduled processes was 4 percent.

Q8) Which two technologies can be implemented to create a baseline of network usage? (Choose two.) (Source: Using Specialized Maintenance and Troubleshooting Tools)

A) SPAN
B) SNMP
C) EEM
D) syslog
E) NetFlow

Q9) SPAN traffic is carried between switches using _____. (Source: Using Specialized Maintenance and Troubleshooting Tools)

Q10) Which two commands are necessary to configure a switch to copy all traffic from interface FastEthernet 0/1 to a packet sniffer on port FastEthernet 0/5? (Choose two.) (Source: Using Specialized Maintenance and Troubleshooting Tools)

A) **monitor session 1 source interface Fa0/1**
B) **span session 1 destination interface Fa0/5**
C) **span session 1 destination remote interface Fa0/5**
D) **monitor session 1 destination interface Fa0/5**
E) **span session 1 source interface Fa0/1**
F) **span session 1 destination remote interface Fa0/5**

Q11) Cisco IOS NetFlow technology allows you to gather detailed traffic profiles and performance statistics such as CPU and memory usage. (Source: Using Specialized Maintenance and Troubleshooting Tools)

A) true
B) false

Q12) Which two commands are necessary to configure a router to send SNMP traps to an SNMP manager at IP address 10.2.2.2 using community string "cisco"? (Choose two.) (Source: Using Specialized Maintenance and Troubleshooting Tools)

A) **snmp-server community cisco RW**
B) **snmp-server ifindex persist**
C) **snmp-server contact 10.2.2.2 community cisco**
D) **snmp-server enable traps**
E) **snmp-server host 10.2.2.2 cisco**
F) **snmp-server trap-host 10.2.2.2**

Q13) Which technology is best described by each sentence below? (Source: Using Specialized Maintenance and Troubleshooting Tools)

A) EEM
B) SNMP
C) NetFlow
D) SPAN

_____ 1. framework that allows for the creation of custom events and corresponding actions
_____ 2. technology that allows traffic to be copied from one switch port to another to allow it to be captured
_____ 3. protocol that allows statistics to be gathered from a device and notifications to be sent by that same device
_____ 4. technology that enables the collection of detailed traffic profiles

Module Self-Check Answer Key

Q1) D

Q2) C

Q3) A

Q4) A

Q5) B

Q6) B

Q7) B, E

Q8) B, E

Q9) trunks

Q10) A, D

Q11) false

Q12) D, E

Q13) 1-A
2-D
3-B
4-C

Module 4

Maintaining and Troubleshooting Campus Switching-Based Solutions

Overview

Ethernet switching-based solutions are found in many different areas of current enterprise networks. Layer 2 and 3 switching solutions are at the heart of all campus networks, and can also be found in the data center and WAN environments.

A good understanding of campus switching technologies, such as VLANs, trunks, spanning tree, multilayer switching, and First Hop Redundancy Protocols (FHRPs), and the ability to diagnose and resolve problems associated with those technologies is therefore essential to any network engineer.

This module reviews prominent campus multilayer switching technologies and focuses on the resolution of problems in environments that utilize these technologies.

Module Objectives

Upon completing this module, you will be able to diagnose and correct problems found in a network infrastructure consisting of campus LAN switches. This ability includes being able to meet these objectives:

- Diagnose and solve VLAN and trunking problems using the Cisco IOS CLI
- Diagnose and solve spanning-tree problems using the Cisco IOS CLI
- Diagnose and solve problems with SVIs and inter-VLAN routing
- Diagnose and solve problems with FHRPs such as HSRP, VRRP, and GLBP
- Diagnose and solve performance problems on Cisco Catalyst switches

Lesson 1

Troubleshooting VLANs

Overview

Switched Ethernet has been the dominant LAN technology for more than a decade. This is why a good understanding of the processes involved in Layer 2 switching is essential to any engineer who is involved in network troubleshooting. VLAN-based switched infrastructures are at the core of every campus network, and being able to diagnose and resolve Layer 2 switching problems in those environments is a fundamental skill that any network engineer should have.

In this lesson, you will review the Layer 2 switching process and associated switch data structures. You will learn how to gather information from these data structures using Cisco IOS commands and how to interpret and analyze the gathered information to verify the proper operation of the Layer 2 switching process, or to pinpoint and resolve problems if the Layer 2 switching process is not functioning properly.

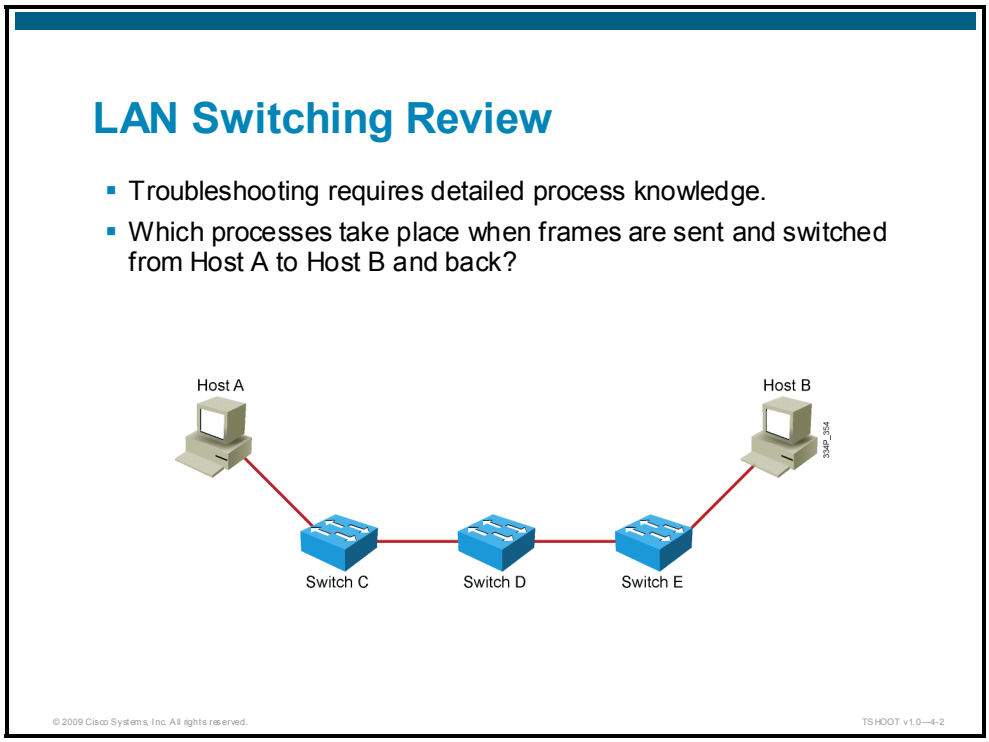
Objectives

Upon completing this lesson, you will be able to diagnose VLAN and trunking problems using the Cisco IOS command-line interface (CLI). This ability includes being able to meet these objectives:

- Understand the process involved in switching a frame from a host in a VLAN to another host in the same VLAN across multiple switched hops
- Analyze information gathered from switch data structures to verify proper operation of Layer 2 forwarding within a VLAN

LAN Switching Operation

This topic describes the processes involved in switching a frame from one host to another in a network infrastructure based on LAN switches. In addition to straightforward Layer 2 switching functionality, this topic also reviews VLAN and trunking concepts.



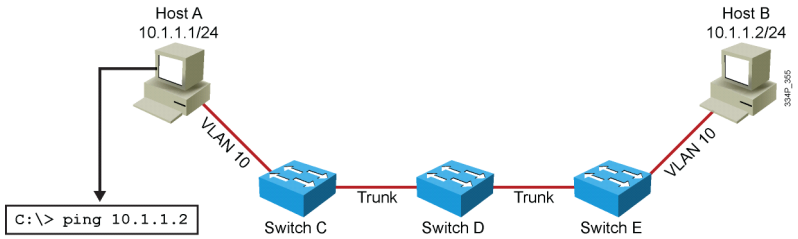
Thorough knowledge of the core processes performed by hosts and network devices is some of the most important knowledge that a network engineer can have. You could argue that knowing how things work does not necessarily help you in building better networks, and that as long as you know what the design principles and the implementation guidelines and commands are, you should be able to develop and build your network—that when things simply work, it is not always necessary to understand exactly how they work.

However, when things break down and devices are not functioning as they should, a good understanding of process helps you to determine where exactly a process breaks down and, consequently, it helps you determine which parts of the network are functioning correctly and which parts are not functioning correctly. So this lesson starts by asking a simple question: "What are the processes that take place when two hosts communicate using IP over a switched LAN?"

First, determine which parts of this process you are interested in when specifically looking at switched network infrastructures. This lesson makes the following assumptions: You are examining generic IP connectivity at this point, which means that you are not interested in application, presentation, session, or transport layer processes. Therefore, you can assume that computer names have been correctly resolved to IP addresses and that the application is ready to send out its first packet on the LAN to the destination IP address of the other host. Further, assume that you are dealing with two hosts that are on the same VLAN and subnet. This limits the scope to the processes involved in Layer 2 switching and excludes Layer 3 routing processes.

Layer 2 Switching Process

- User initiates a connection.



Because the application being used is irrelevant in this context, this lesson uses ping as the sample application, as shown in the figure.

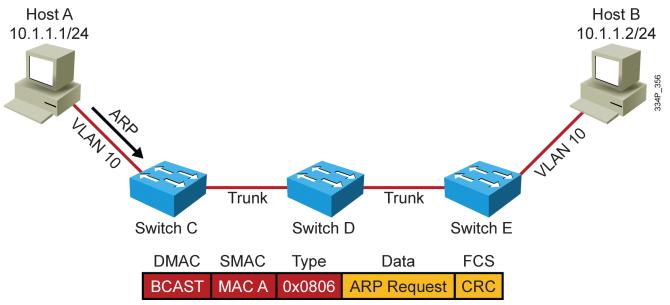
This process can be broken down into the following steps:

1. Host A checks the destination (Host B) IP address and compares it against its own IP address and mask. From this, it concludes that Host B is on the same subnet.

2. Because Host B is on the same subnet, Host A consults its Address Resolution Protocol (ARP) cache to find the MAC address of Host B. If the cache contains an entry for Host B, Host A skips the ARP process, encapsulates the IP packet in an Ethernet frame destined for Host A, and transmits the frame.

Layer 2 Switching Process (Cont.)

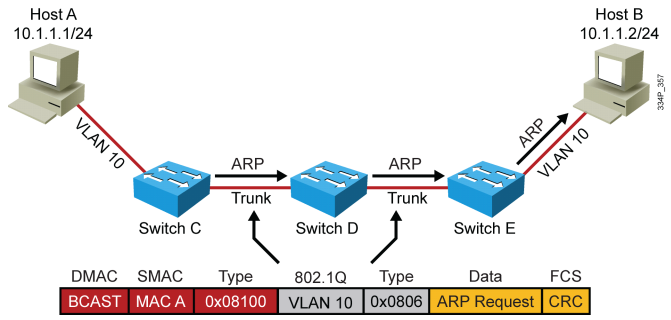
- Host A sends an ARP request as a broadcast.



3. If the ARP cache on Host A does not contain an entry for the IP address of Host B, Host A sends out an ARP request as a broadcast to obtain the MAC address of Host B.

Layer 2 Switching Process (Cont.)

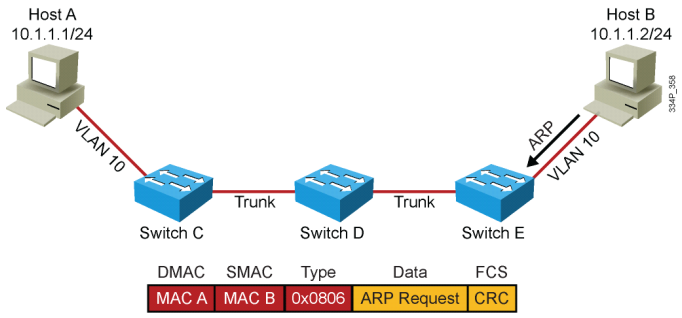
- The switches flood the ARP broadcast on all ports in VLAN 10 including trunks, learning the MAC address of Host A in the process.



4. Switch C checks the VLAN of the port it receives the frame on, records the source MAC address in its MAC address table, and associates it to that port and VLAN. It performs a lookup in its MAC address table to try to find the port that is associated to the MAC address of Host B. Then—assuming that this table does not contain an entry for Host B—it floods the frame on all ports in that VLAN, including all trunks that this VLAN is allowed on. Alternately, if the switch did find an entry for Host B's MAC address in the MAC address table, it forwards it only on the port associated to that MAC address. Switches D and E repeat this process as they receive the frame.

Layer 2 Switching Process (Cont.)

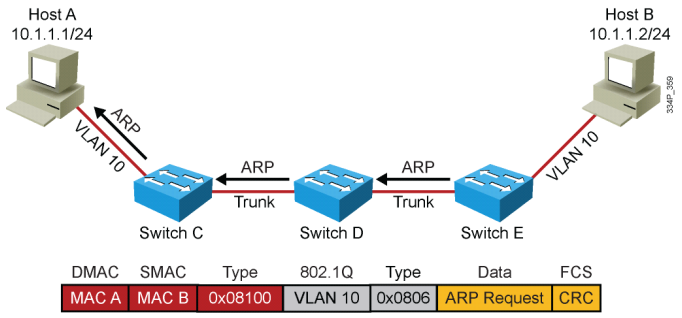
- Host B responds to the ARP using unicast to Host A.



5. Host B receives the ARP request, records the IP address and MAC address of Host A in its own ARP cache, and then proceeds to send an ARP reply as a unicast back to Host A.

Layer 2 Switching Process (Cont.)

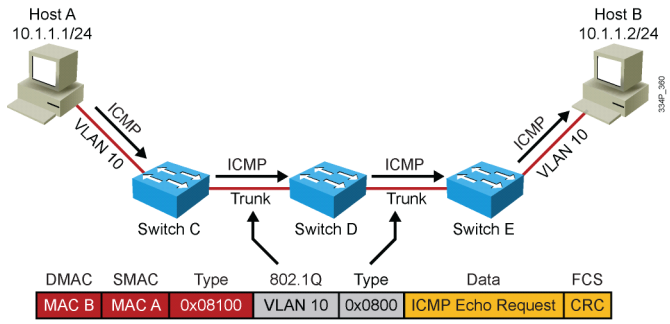
- The switches forward the ARP reply on the port on which they learned the MAC address of Host A, at the same time recording the MAC address of Host B.



6. The switches check the VLAN of the port they received the frame on, and because all switches now have an entry in their MAC address table for the MAC address of Host A, they forward the frame containing the ARP reply on the path to Host A only, not flooding it out on any other port. At the same time, they record Host B's MAC address and corresponding interface and VLAN in their MAC address table if they do not already have that entry.

Layer 2 Switching Process (Cont.)

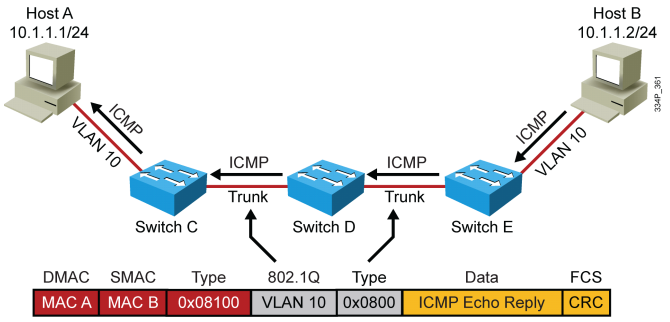
- Host A now sends the original packet as a unicast to Host B, and the switches forward it on their port toward Host B.



7. Host A receives the ARP reply and records the IP and MAC address of Host B in its ARP cache. Now Host A is ready to send the original IP packet.
8. Host A encapsulates the IP packet (Internet Control Message Protocol [ICMP] echo request) in a unicast frame destined for Host B and sends it out.
9. The switches again consult their MAC address tables, find an entry for the Host B MAC address, and forward it on the path toward Host B.

Layer 2 Switching Process (Cont.)

- Host B responds using unicast to Host A, and the switches forward this frame on their port toward Host A.

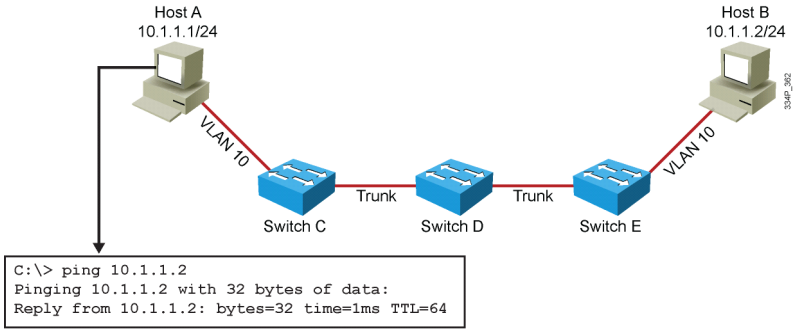


10. Host B receives the packet and responds to Host A (by sending an ICMP echo reply packet).

11. The switches again consult their MAC address tables and forward the frame straight to Host A, without any flooding.

Layer 2 Switching Process (Cont.)

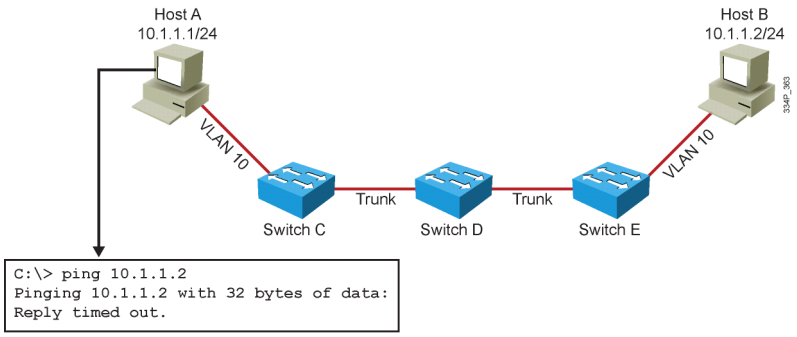
- This completes the exchange.



12. Host A receives the packet. This step concludes this simple packet exchange.

Discussion: Typical LAN Switching Problems

- What can cause this process to fail?
- How can you determine where it fails?
- What commands can you use to diagnose these problems?



Although this process is most likely very familiar to you and might seem elementary, listing the steps like this clearly shows that even for the simplest communication, an elaborate chain of events takes place. If at any point this chain is broken due to faulty cabling, failing devices, or misconfiguration, the communication fails. Therefore, you look at how you can leverage your knowledge of these processes to diagnose problems in a switched environment.

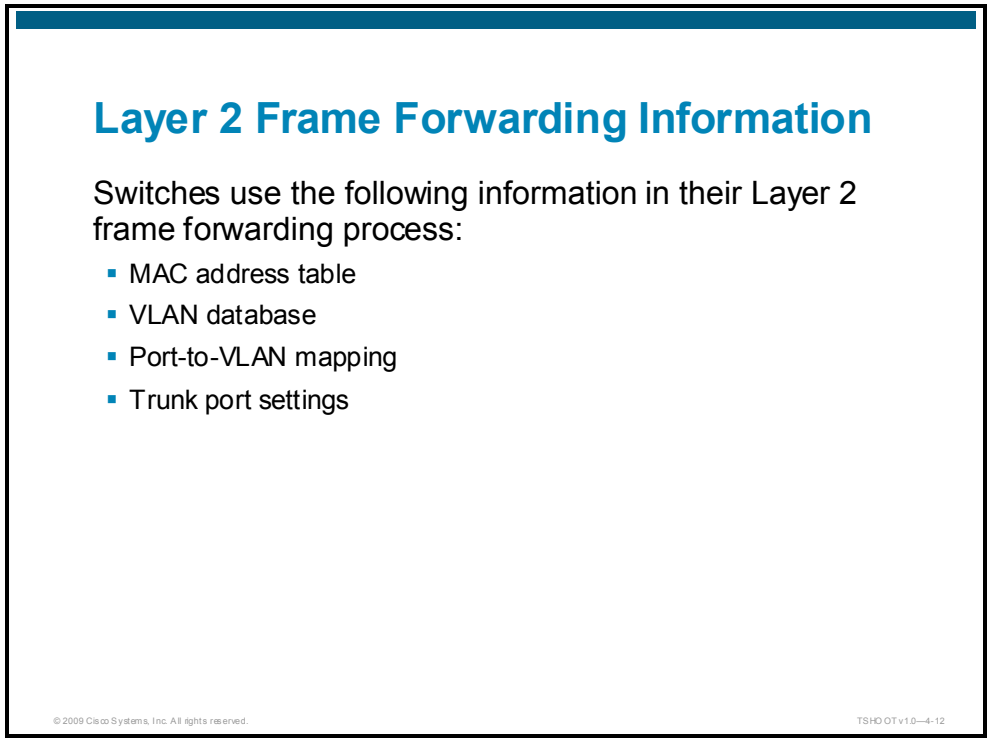
First, look at possible issues that could cause the communication to fail:

- Physical problems:
  - Bad, missing, or miswired cables
  - Bad ports
  - Power failure
- Device problems:
  - Software bugs
  - Performance problems
- Misconfiguration:
  - Missing or wrong VLANs
  - Misconfigured Virtual Terminal Protocol (VTP) settings
  - Wrong VLAN setting on access ports
  - Missing or misconfigured trunks:
    - Native VLAN mismatch
    - VLANs not allowed on trunk

This lesson focuses specifically on LAN switching, and therefore does not discuss generic physical problems. This lesson investigates how specific commands available on the Cisco Catalyst LAN switches can supplement your troubleshooting toolkit to help troubleshoot data link and physical layer problems.

Switch Data Structures

This topic describes which data structures LAN switches are using to switch frames and how to obtain information from those structures that can be used to diagnose problems.



A common method to troubleshoot Layer 2 switching problems is to follow the path of the frames through the switches. If you actually try to follow the frames themselves in real time, you would spend a lot of time and effort to make packet captures at various points in the network, which is not always practically feasible.

So instead of trying to follow the frame in real time, you can try to follow the trail of the frame. The objectives are to confirm that frames have passed through the switches and to verify the information that the switch must have used to make its forwarding decisions. If you find a point where the trail suddenly stops, or if you find that the information that the switch uses to forward its frames does not match your expectations, you gain important clues that help you reduce the scope of the possible problem areas, formulate a hypothesis on the cause of the problem, or even outright point at the cause of the problem.

How can you follow the trail of a frame? Which data structure would prove that frames have passed through a switch?

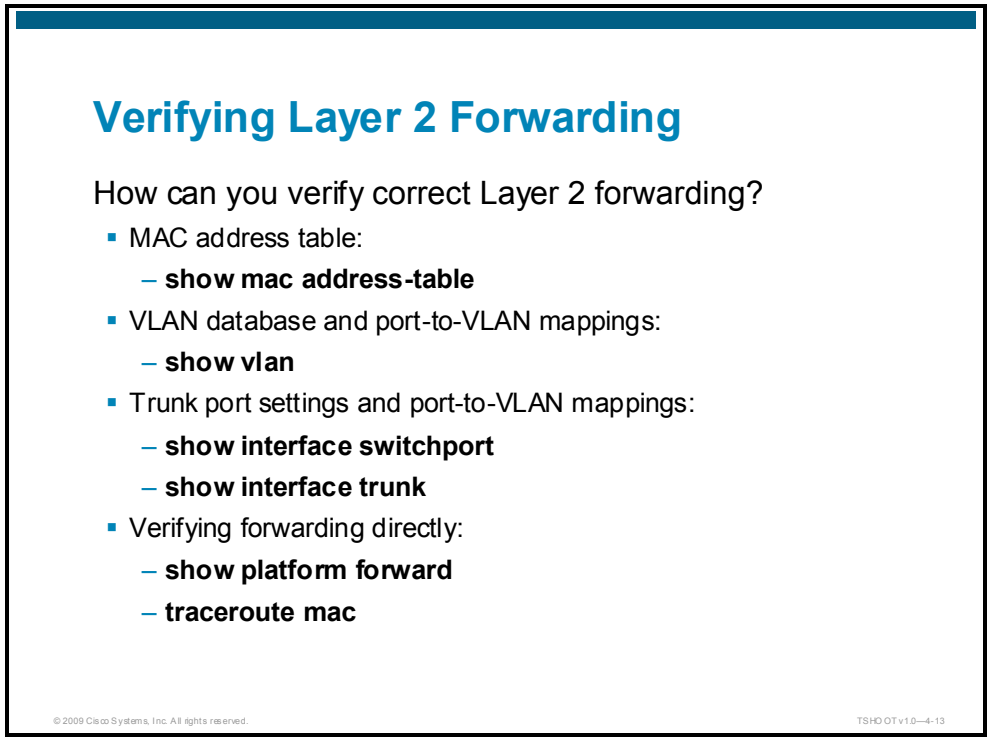
One of the key data structures that you can consult is the switch's MAC address table. In this table, the switch registers the source MAC address of each frame that it receives in combination with the port and VLAN on which it was received. So when you see an entry for a particular MAC address in this table, it proves that at some point this switch received frames from that source. It does not necessarily tell you anything about a particular frame or how long ago the last frame was received. Therefore, it can be good practice to clear the MAC entry from the table by using the **clear mac-address-table** *address* command and verify that the MAC address is learned again when you reinitiate the connection. Also, the MAC address table allows you to verify that frames are received on the port and VLAN where they are expected. If somehow the output of this command does not match your expectations and assumptions, this provides you with a clue about the cause of the problem.

For example:

- You discover that frames are not received on the correct VLAN. This could point to VLAN or trunk misconfiguration as the cause of the problem.
- You find that frames are received on a different port from what you expected. This could point to a physical problem, spanning tree issues, or duplicate MAC addresses.
- You find that the MAC address is not registered in the MAC address table at all. This tells you that the problem is most likely upstream from this switch. You should retrace your steps and investigate between the last point where you know that frames were received and this switch.

On the other hand, if the output matches your expectations and assumptions, this command confirms that everything is working as expected up to this point. (Keep in mind that this does not necessarily prove that everything is working properly, because it is still possible that your assumptions and expectations were wrong to begin with.) The next step is to use your knowledge of the forwarding process combined with information that can be gathered from the switch's command output to determine what the next step in the process would be. Again, you should validate the facts that you gather about the switch's forwarding behavior against your expectations and assumptions.

When you have confirmed that the behavior of the switch matches your expectations, you have successfully reduced the possible scope of the problem: You have confirmed that everything works as expected up to this point.



After you have looked at a common troubleshooting method to diagnose Layer 2 switching problems, you look at how to gather the information from the switch that is needed to validate your assumptions. Some commonly used commands that help you obtain information about the Layer 2 switching process, VLANs, and trunks are as follows:

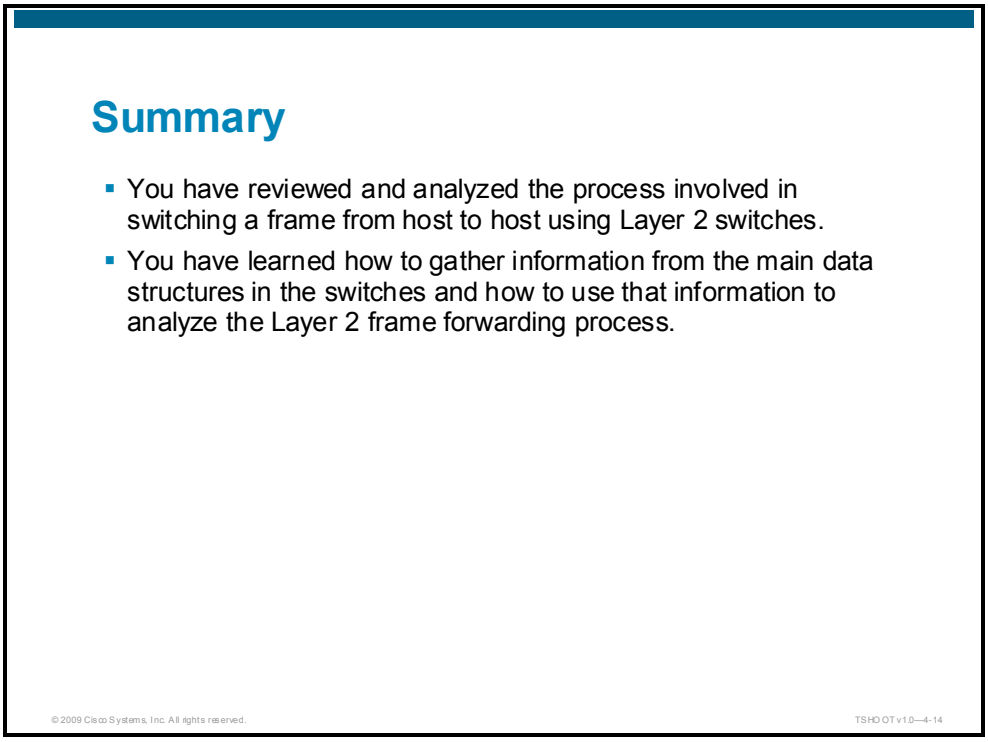
- **show mac address-table:** The most important command to verify Layer 2 forwarding. It shows you the MAC addresses learned by the switch and their corresponding port and VLAN associations. This command indicates whether frames sourced by a particular host have succeeded in reaching this switch. It also helps you verify if these frames were received on the correct inbound interface.
- **show vlan:** Verifies VLAN existence and port-to-VLAN associations. This command lists all VLANS that were created on the switch (either manually or through the VTP). It also lists the ports that are associated to each of the VLANs. Note that trunks are not listed, because they do not belong to any VLAN in particular.
- **show interface trunk:** Checks which interfaces are configured as trunks. This also shows which VLANs are allowed on the trunk and what the native VLAN is for that trunk.
- **show interface switchport:** Combines some of the information found in **show vlan** and **show interface trunk**. This command is most useful if you are not looking for a switch-wide overview of trunk- or VLAN-related information, but would prefer to have a quick summary of all VLAN-related information for a single interface.
- **show platform forward:** Directly gathers frame forwarding information from the ternary content addressable memory (TCAM) hardware. This command exists on many of the Cisco Catalyst Switches.
- **traceroute mac:** Displays a list of switch hops that a frame from a particular source MAC address to a particular destination MAC address passes through, if Cisco Discovery Protocol is enabled.

For sample output of these commands and guidelines regarding their use, see the “Sample Troubleshooting Flows” section in the course lab guide.

To review the complete syntax and a description of all command options, see the Command Reference sections of the Cisco Product Support pages on http://www.cisco.com.

Summary

This topic summarizes the key points that were discussed in this lesson.



Lesson 2

Troubleshooting Spanning Tree

Overview

High availability is an important requirement for campus LANs today. The more dependent enterprises have become on their networks to support their business, the more important it is that those networks are highly available and that network downtime is minimized. One of the primary tools in building highly available networks is the use of redundant devices and links.

However, with the introduction of redundancy in a Layer 2 switched network, bridging loops can be introduced, resulting in broadcast storms that can potentially cripple the network. The IEEE 802.1D Spanning Tree Protocol (STP) acts as a safety net by breaking those loops and thereby preventing broadcast storms.

This is why a good understanding of the operation of STP is essential to any network engineer. It is important to know how to predict the spanning-tree topology or, in the absence of documentation of the spanning-tree parameters, determine the spanning-tree topology based on the interpretation of the output of Cisco IOS commands. Spanning tree failures can be catastrophic when they happen, and therefore recognizing the symptoms and having an action plan for these types of failures is a skill that is essential in reducing network downtime.

In this lesson, you will learn how to verify the proper operation of STP, how to recognize the typical symptoms of a spanning tree failure, and how to recover from this type of failure.

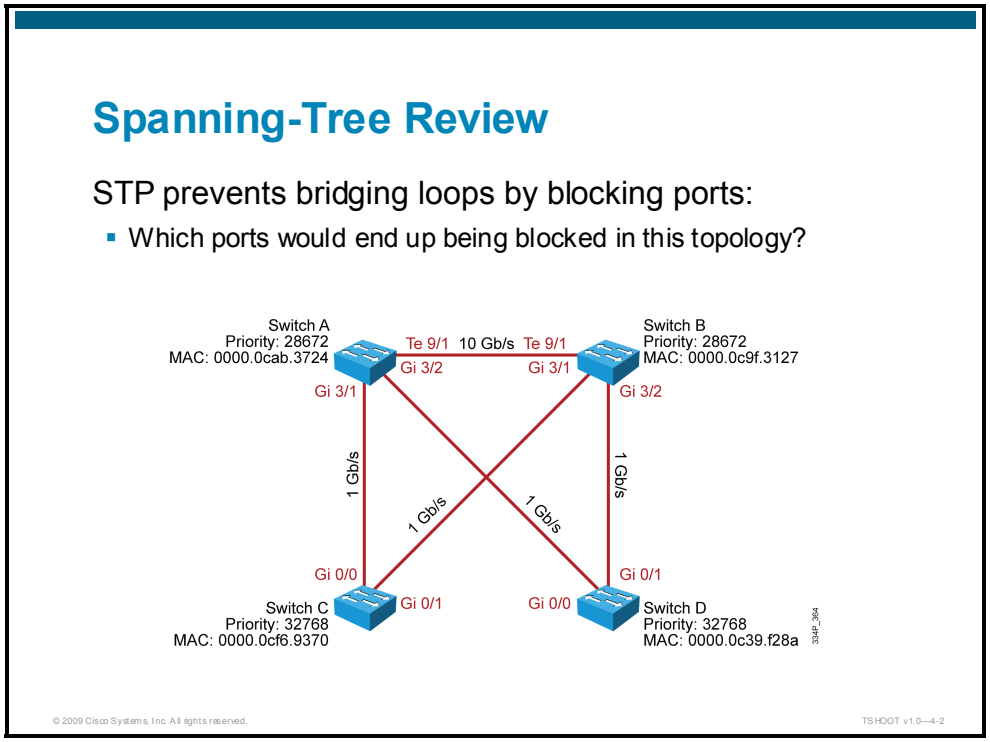
Objectives

Upon completing this lesson, you will be able to diagnose spanning-tree problems using the Cisco IOS command-line interface (CLI). This ability includes being able to meet these objectives:

- Understand the steps that spanning tree goes through to attain a loop-free topology
- Determine the spanning-tree topology using Cisco IOS commands
- Recognize the symptoms of spanning-tree failures and remediate the failures
- Understand the mechanisms involved in EtherChannel load balancing

Spanning Tree and Rapid Spanning Tree

This topic describes the process that switches running STP go through to achieve a loop-free topology.



The IEEE 802.1D STP is one of the most pervasive and important protocols in the campus switching environment. You will hardly find a LAN that does not run this protocol in some form or another. You might sometimes hear people say that they do not use spanning tree. However, in most cases what they mean is that in their network spanning tree is not actively blocking ports or involved in the reconvergence process when a failure occurs. In those instances, spanning tree is usually running in the background as a safety net.

So from what is spanning tree saving your network? The main purpose of spanning tree is the prevention of bridging loops and the packet storms that might ensue from loop conditions. If you have ever been in the situation where loops in the switching topology were introduced and spanning tree was not running or was not functioning correctly, you know how badly this type of failure can affect your network. The complete LAN can become fully saturated, and switches can become entirely unresponsive until the loops are broken. This is why understanding the spanning-tree process, recognizing the symptoms of a spanning-tree failure, and knowing how to act to resolve those issues should be core knowledge for any engineer that implements or supports switched LANs.

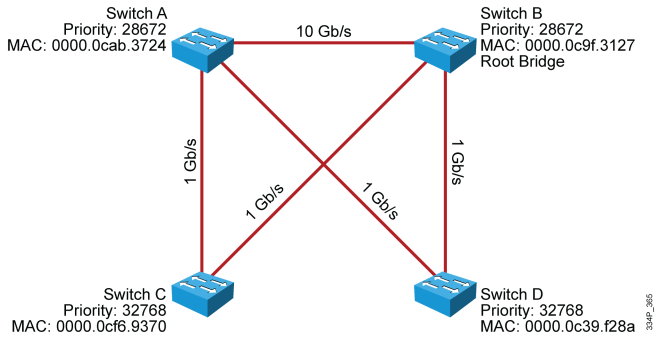
You can review spanning-tree operation by running through a simple case study. Look at the figure and take a little time to review what you know of spanning-tree operation. See if you can find an answer to the following questions:

- Which switch will be the root switch?
- Which ports are considered root ports?
- Which ports are considered designated ports?
- Which ports will be blocking and which ports will be forwarding?

Spanning Tree: Root Bridge

Step 1: Elect a root bridge:

- Decision based on lowest bridge ID



The first step in the spanning-tree algorithm is the election of a root bridge. Initially, all switches assume that they are the root. They start transmitting Bridge Protocol Data Units (BPDU) with the Root ID field containing the same value as the Bridge ID field. This implies that each switch essentially claims that it is the root bridge on the network.

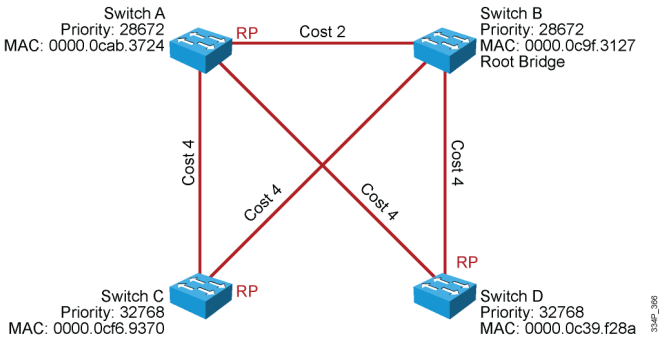
As soon as the switches start receiving BPDUs from the other switches, each switch compares the root ID in the received BPDUs against the value that it currently has recorded as the root ID. If the received value is lower than the recorded value (which was originally the switch's own bridge ID), the switch replaces the recorded value with the received value and starts transmitting this in the Root ID field in its own BPDUs.

As a result of this, after a while, all switches will have learned and recorded the bridge ID of the switch that has the lowest bridge ID of all switches, and the switches will all be transmitting this ID in the Root ID field of their BPDUs.

Spanning Tree: Root Ports

Step 2: Elect a root port for each nonroot switch:

- Decision based on lowest root path cost.
- Possible ties are broken by upstream bridge ID and port ID.



As soon as a switch recognizes that it is not the root (because it is receiving BPDUs that have a root ID value that is lower than its own bridge ID), it marks the port on which it is receiving those BPDUs as its root port.

Naturally, it could happen that BPDUs are received on multiple ports. In this case, the switch elects the port that has the lowest-cost path to the root as its root port. If two ports have an equal path cost to the root, the switch looks at the bridge ID values in the received BPDUs to make a decision (where the lowest bridge ID is considered better, similar to root bridge election). If the root path cost and the bridge ID in both BPDUs are the same, because both ports are connected to the same upstream switch, the switch looks at the Port ID field in the BPDUs and selects its root port based on the lowest value in that field.

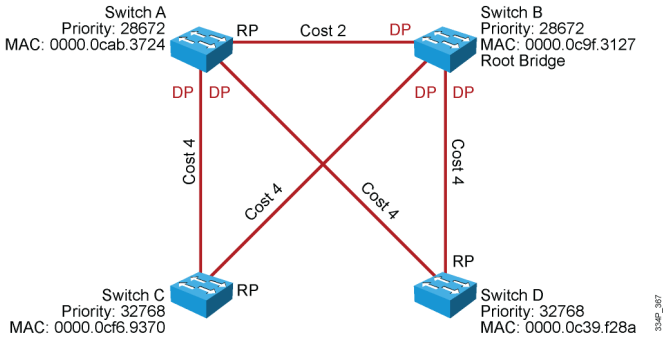
The way a switch determines the path cost is by adding the cost associated to the port on which it receives the BPDU to the value in the Root Path Cost field in the received BPDU. The lowest value determines the switch's root port, and this value is in turn transmitted in the switch's own BPDUs. In short, this means that the root bridge starts sending BPDUs with the root path cost set to 0, and then each switch adds the cost of its root port to the received cost when it sends BPDUs to neighboring switches.

The cost associated to each port is, by default, related to its speed, but can be manually changed.

Spanning Tree: Designated Ports

Step 3: Elect a designated port for each segment:

- Decision based on lowest root path cost.
- Possible ties are broken by upstream bridge ID and port ID.

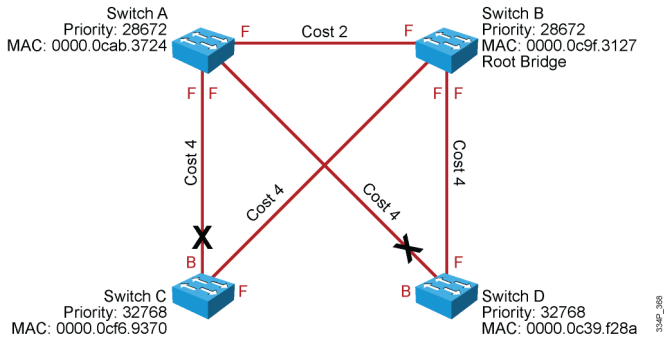


After electing the root bridge and root ports, the switches determine which of the switches will become the designated bridge for each Ethernet segment. This process has similarities to both root bridge and root port elections. Each switch connected to a segment sends BPDUs out of its port connected to that segment, essentially claiming to be the designated bridge for that segment. At this point, it considers its port to be a designated port.

As soon as a switch starts receiving BPDUs from other switches on that segment, it compares the received values of the root path cost, bridge ID, and port ID fields (in that order) against the values in its own BPDUs that it is sending out that port. If it turns out that the other switch has lower values than this switch, it stops transmitting BPDUs on the port and marks it as a nondesignated port (which can be either an alternative or a backup port).

Spanning Tree: Forwarding and Blocking

Step 4: Root ports and designated ports transition to forwarding state; other ports stay in blocking state.



TSHOOT v1.0—4-6

To prevent bridging loops during the time it takes the STP to execute its algorithm, all ports initially start out in the blocking state. As soon as a switch marks a port as either a root port or a designated port, it starts to transition this port to the forwarding state.

Notice that up to this point no distinction has been made between the classical (802.1D-1998) and rapid (802.1w / 802.1D-2004) versions of STP. They both execute the same algorithm when it comes to the decision-making process. On the other hand, when it comes to the process of transitioning a port from the blocking (or discarding, in rapid spanning-tree terms) to the forwarding state, there is a big difference between those two spanning-tree versions. Classical 802.1D would simply take 30 seconds to transition the port to forwarding, whereas rapid spanning tree can leverage additional mechanisms to transition the port to forwarding in less than a second.

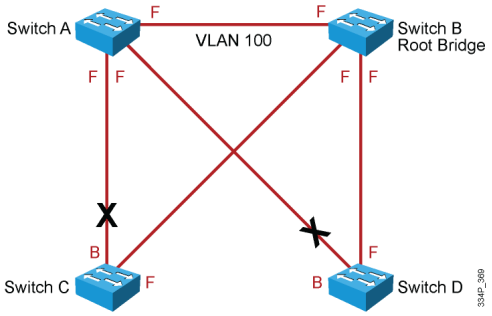
Although the order of the steps listed in the diagrams suggests that STP goes through these steps in a coordinated sequential manner, this is not actually the case. If you look back at the descriptions of each of the steps in the process, you will see that each switch is going through these steps in parallel and that it might adapt its selection of root bridge, root ports, and designated ports as new BPDUs are received. As the BPDUs are propagated through the network, it will eventually settle into a situation where all switches have a consistent view of the topology of the network. Also, note that when this stable state is reached, BPDUs are only transmitted by designated ports.

Analyzing the Spanning-Tree Topology

This topic describes how you can use Cisco IOS commands to analyze the spanning-tree topology and verify proper operation of STP.

Spanning-Tree Verification

- What tools can be used to verify that the actual spanning-tree topology matches the expected topology?
- How can you discover the spanning-tree topology if there is not enough information to predict it?



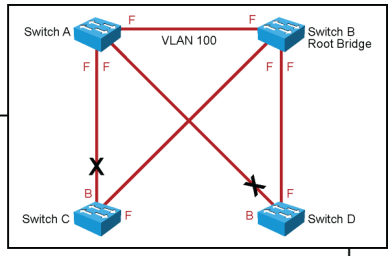
In many networks, the optimal spanning-tree topology is determined as part of the network design and then implemented through manipulation of spanning-tree priority and cost values. Sometimes you might run into situations where spanning tree was not considered in the design and implementation, or where it was considered initially, but the network has undergone significant growth and changes. In such situations, it is important for an engineer to know how to analyze the actual spanning-tree topology in the operational network.

In addition, a big part of troubleshooting consists of comparing the actual state of the network against the expected state of the network and spotting the differences to gather clues about the problem that you are troubleshooting. To do that, you should be able to examine the switches and determine the actual topology, in addition to knowing what the spanning-tree topology is supposed to be.

Spanning Tree Cisco IOS Commands

To get an overview of spanning-tree status and topology:

- **show spanning-tree [vlan *vlan-id*]**




```
SwitchA#show spanning-tree vlan 100

VLAN0100
  Spanning tree enabled protocol rstp
  Root ID     Priority    28772
              Address     0000.0c9f.3127
              Cost        2
              Port        88 (TenGigabit9/1)
              Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID   Priority    28772  (priority 28672 sys-id-ext 100)
              Address     0000.0cab.3724
              Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
              Aging Time 300

Interface        Role Sts Cost      Prio.Nbr Type
---------------- ---- --- --------- -------- --------------------------------
Gi3/1            Desg FWD 4         128.72   P2p
Gi3/2            Desg FWD 4         128.80   P2p
Te9/1            Root FWD 2         128.88   P2p
```


The following are some of the most common commands to gather information about the status of the STP and the corresponding topology:

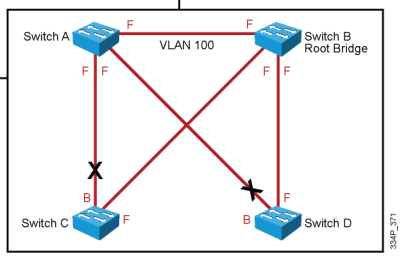
- **show spanning-tree:** Using this command without specifying any additional options is a good way to get a quick overview of the status of the STP for all VLANs that are defined on a switch. If you are interested only in a particular VLAN, you can limit the scope of this command by specifying the VLAN as an option.

Spanning Tree Cisco IOS Commands (Cont.)

To see detailed BPDU information:

- **show spanning-tree interface** *interface* **detail**

```
SwitchA#show spanning-tree interface Ten 9/1 detail
 Port 88 (TenGigabitEthernet9/1) of VLAN0100 is root forwarding
   Port path cost 2, Port priority 128, Port Identifier 128.88.
   Designated root has priority 28772, address 0000.0c9f.3127
   Designated bridge has priority 28772, address 0000.0c9f.3127
   Designated port id is 128.88, designated path cost 0
   Timers: message age 15, forward delay 0, hold 0
   Number of transitions to forwarding state: 1
   Link type is point-to-point by default
   BPDU: sent 10, received 670
```



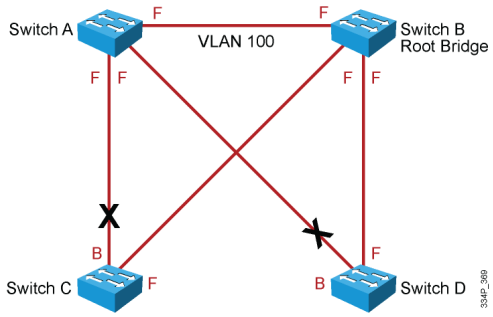
- **show spanning-tree interface** *interface-id* **detail:** This command is useful if you need to see the exact content of the BPDUs. It will either give you the BPDU content of the BPDUs received from the upstream switch (if that switch is the designated bridge) or the content of the BPDUs that are being sent out by this switch (if this switch is the designated bridge for the segment connected to the interface). For example, in the figure you can see that the port is a root port and the upstream switch is the designated bridge. This is also reflected by the fact that this switch is receiving BPDUs (it received 670 BPDUs), but not transmitting them (it sent 10 BPDUs during initial spanning-tree convergence and stopped after that). At the same time, you can see that the upstream switch is also the root bridge. This can be concluded from the fact that the designated bridge ID and the root bridge ID are the same and is further confirmed by the fact that the designated path cost is reported as a cost of 0.

Spanning-Tree Failures

This topic describes the typical symptoms that you would experience during a major spanning-tree failure and how you can recover from such a failure.

Bridging Loops and Broadcast Storms

- What happens when Spanning Tree Protocol fails to do its job?
- What will happen to this network if Switch D erroneously transitions both its ports to the forwarding state?



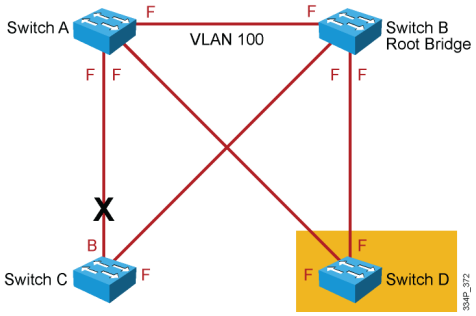
The biggest problem with STP is not the fact that it could fail, because any protocol can fail and, in fact, STP is one of the most well known and reliable protocols available. However, the biggest problem with spanning tree is the fact that when it fails, it can fail in very catastrophic manners.

With many protocols, when they malfunction, all that happens is that you lose the functionality that the protocol was providing. For example, if Open Shortest Path First (OSPF) is malfunctioning on one of your routers, you might lose connectivity to networks that are reachable via that router, but it would generally not affect the rest of your OSPF network; if you still have some way to connect to that router, you can still perform your troubleshooting routines to diagnose and fix the problem.

With spanning tree, there are two different types of failures. The first one is similar to the type of problem described in the previous paragraph. STP might erroneously decide to block ports that should have gone to the forwarding state. This causes problems that are similar to the OSPF problem: You might lose connectivity for traffic that would normally pass through this switch, but the rest of the network is unaffected and you can still troubleshoot on the switch, if you still have a way to access it. The second type of failure is much more disruptive; it happens when spanning tree erroneously decides to move one or more ports to the forwarding state.

Bridging Loops and Broadcast Storms (Cont.)

- What happens when Spanning Tree Protocol fails to do its job?
- What will happen to this network if Switch D erroneously transitions both its ports to the forwarding state?



Remember that an Ethernet frame header does not include a Time To Live (TTL) field, which means that any frame that enters a bridging loop will continue to be forwarded by the switches indefinitely. The only exceptions are the frames that have their destination address recorded in the MAC address table of the switches. These frames are simply forwarded to the port that the MAC address is associated with and do not enter a loop. However, any frame that is flooded by a switch—broadcasts, multicasts, and unicasts with an unknown destination MAC address—enter the loop and start circling.

What are the consequences and corresponding symptoms of this behavior?

- The load on all links in the switched LAN will quickly start increasing as more and more frames enter the loop. Note that this is not limited to just the links that form the loop, but also any other links in the switched domain, because the frames are flooded on all links. Naturally, when the spanning-tree failure is limited to a single VLAN, only links in that VLAN will be affected and switches and trunks that do not carry that VLAN will operate normally.
- If the spanning-tree failure has caused more than one bridging loop to exist, traffic will increase exponentially; because frames will not only start circling, but will also start getting duplicated. This happens because, in the case of multiple loops, there will be switches that receive a frame on a port and then flood it out on multiple ports, essentially creating a copy of the frame every time they forward it.
- When control plane traffic starts entering the loop (for example, Hot Standby Router Protocol [HSRP] hellos, OSPF hellos, or Enhanced Interior Gateway Routing Protocol [EIGRP] hellos), the devices that are running these protocols quickly start getting overloaded. Their CPU will approach 100 percent utilization while they are trying to process an ever-increasing load of control plane traffic. In many cases, the earliest indication that you get of a broadcast storm in progress is that routers or Layer 3 switches are reporting control plane failures such as continual HSRP state changes and that they are running at a very high CPU load.

- Switches will experience very frequent MAC address table changes. This happens because frames will usually start looping in both directions, causing a switch to see a frame with a certain source MAC address coming in on a port and then see a frame with the same source MAC coming in on a different port just a fraction of a second later.
- Due to the combination of very high load on all links and the CPU running at maximum load on Layer 3 switches or routers, these devices typically become unreachable, making it nearly impossible to diagnose the problem while it is happening.

As you can see, the consequences of this type of spanning-tree failure, when it results in a broadcast storm, are severe, and the execution of proper troubleshooting methods is hindered by the fact that the links and devices are overloaded. What can you do if you have determined—based on the symptoms described above—that you might be dealing with a broadcast storm?

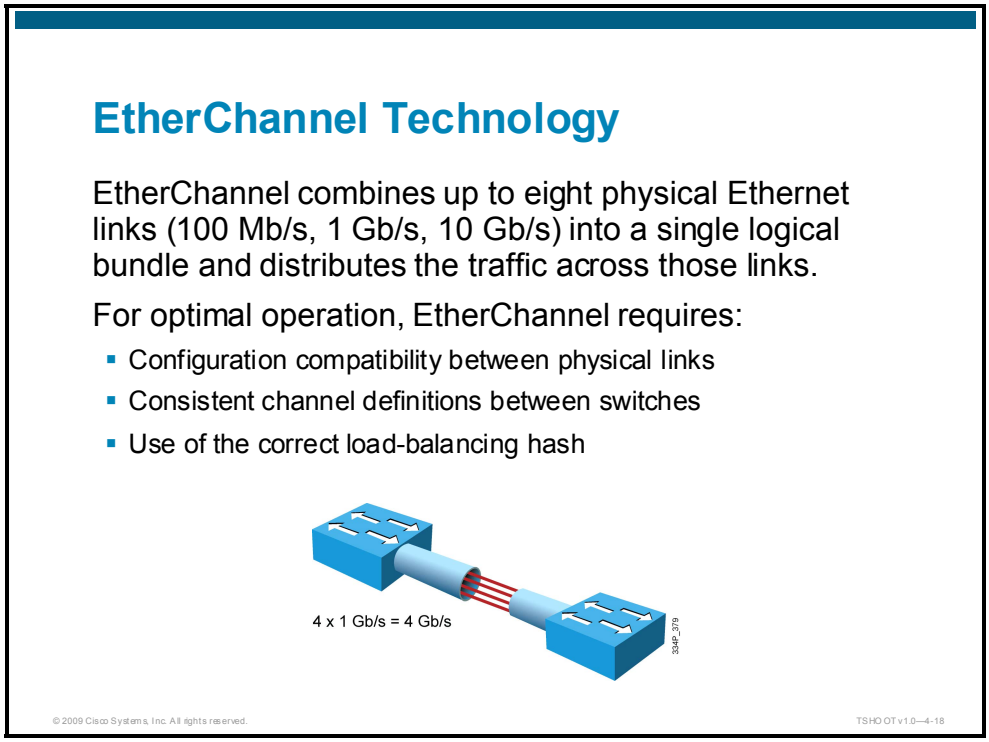
A viable approach is to take over the role of the failing STP by manually, either physically or through configuration (if that is still possible), removing redundant links in the switched network until all loops are eliminated from the topology. When the loops are broken, the traffic and CPU loads should quickly drop to normal levels, and you should regain connectivity to your devices.

Although this restores connectivity to the network, you cannot consider this the end of your troubleshooting process. You have removed all redundancy from your switched network and you need to restore the redundant links.

Of course, if the underlying cause for the spanning-tree failure has not been fixed, chances are that restoring the redundant links will trigger a new broadcast storm. To find the underlying cause of the failure, before you restore the redundant links, you should spend sufficient time investigating what happened at the moment that the broadcast storm started. When you eventually start restoring the redundant links, you should carefully monitor the network and have an emergency plan to fall back on if you see a new broadcast storm developing.

EtherChannel Operation

This topic describes how EtherChannel splits traffic over multiple links and what conditions could cause this mechanism to fail.



EtherChannel is a technology that bundles multiple physical Ethernet links (100 Mb/s, 1 Gb/s, 10 Gb/s) into a single logical link and distributes the traffic across these links. This logical link is represented in Cisco IOS syntax as a port-channel interface.

Control protocols such as spanning tree or routing protocols interact only with this single port-channel interface and not with the associated physical interfaces. Packets and frames are routed or switched to the port-channel interface, and then a hashing mechanism determines which physical link will be used to transmit them.

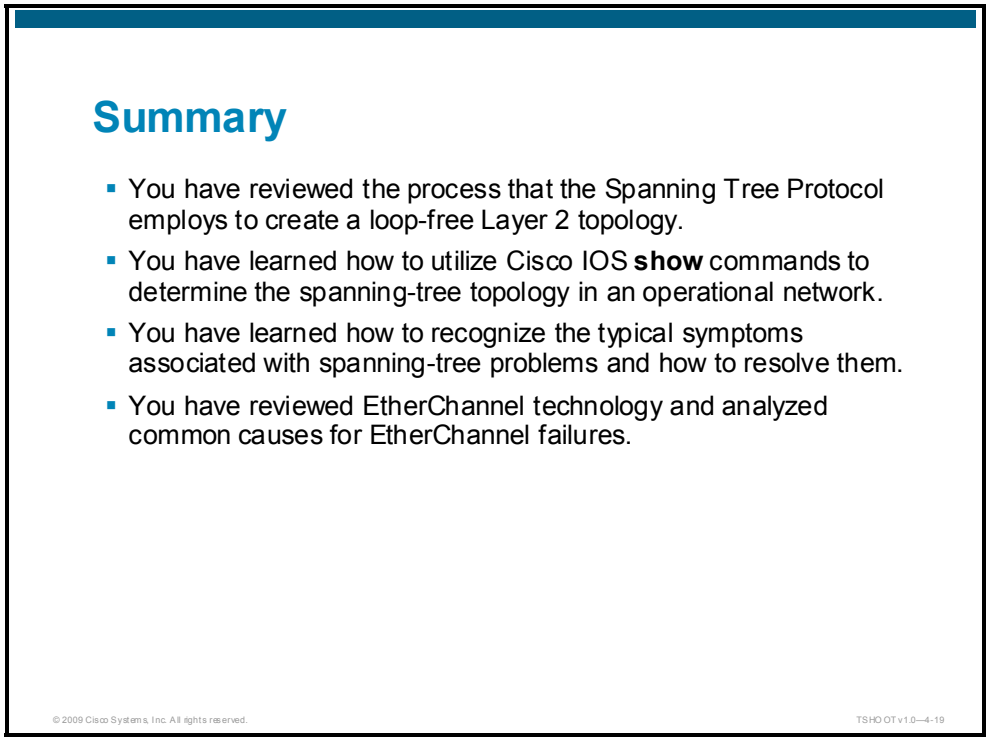
There are three main categories of problems that you might run into when deploying EtherChannel technology:

- **Inconsistencies between the physical ports that are members of the channel:** The physical links in an EtherChannel must have the same operational characteristics (same speed, same duplex, same trunk or access port status, same native VLAN when they are trunking, and same access VLAN when they are access ports, for example). It is therefore recommended that the configuration of all physical links in the channel be identical. If at a certain point in time (usually due to misconfiguration) one of the physical links changes its operational status in such a way that a mismatch with the other physical links is created, this port will be suspended and removed from the EtherChannel bundle until consistency is restored. When the switch suspends a physical link in the channel due to incompatibilities, it generates a %EC-5-CANNOT_BUNDLE2 log message.

- **Inconsistencies between the devices on both sides of the EtherChannel:** If the switch on one side is configured to bundle these links into an EtherChannel and the switch on the other side is not, the switch that is configured for EtherChannel will detect this (by detecting inconsistencies in the spanning-tree behavior) and move the port to an error-disabled state. The switch will generate a %SPANTREE-2-CHNL_MISCFG message when it error-disables the port. The use of an EtherChannel negotiation protocol like the 802.3ad Link Aggregation Control Protocol (LACP) or the Port Aggregation Protocol (PAgP), will prevent this situation from happening because both sides must agree before they will start using EtherChannel.
- **Underutilized links in the channel:** Most people expect that when EtherChannel is used, the traffic will be equally balanced across all physical links in the bundle. However, it is important to realize that the method used to distribute traffic over the physical links is to calculate a hash of a combination of fields in the Ethernet and IP headers of a frame and then send the frame to a physical interface based on the hash result. Therefore, the distribution of traffic depends on two things: the distribution of hash values over the physical links, and the header fields that are used as a key into the hash calculation. The Cisco EtherChannel hash algorithm results in a value between 0 and 7. This means that in case of an eight-port EtherChannel, one hash value is assigned to each of the links and (assuming a random traffic mix) traffic is equally balanced across all eight links. However, if the channel consists of six links, the distribution will be 2:2:1:1:1:1 instead, meaning that the first two links in the channel each handle twice as much traffic as the other links. The second factor in EtherChannel load balancing is which header fields are used as the base of the hash value. If you can assume those fields in the traffic to be entirely random, it would not matter what hashing mechanism is used, but because header fields are typically not random, the choice of header fields to be hashed affects the distribution. The simplest example of this is where only the destination MAC address is used as the input for the hash calculation. If 90 percent of all frames would be destined for a single MAC address (for example, the MAC address of the default gateway), all of that traffic would end up on the same physical link. So if you see a very uneven distribution of traffic over the links in the channel, you should examine the hashing method and the traffic mix to determine the cause.

Summary

This topic summarizes the key points that were discussed in this lesson.



Lesson 3

Lab 4-1 Debrief

Overview

In this lab, you have practiced troubleshooting Layer 2 connectivity problems related to VLANs, trunking, spanning tree, and EtherChannel. You have created a troubleshooting plan to coordinate the work between the team members. You have methodically researched the problems, kept a log of your actions and findings, and resolved the problems and documented your solutions.

During the lab debrief, the instructor will lead a group discussion during which you can present your solutions. You will get an opportunity to verify your solution against a number of checkpoints provided by the instructor and compare your solution to other students' solutions. The instructor will discuss alternate solutions and their advantages and disadvantages.

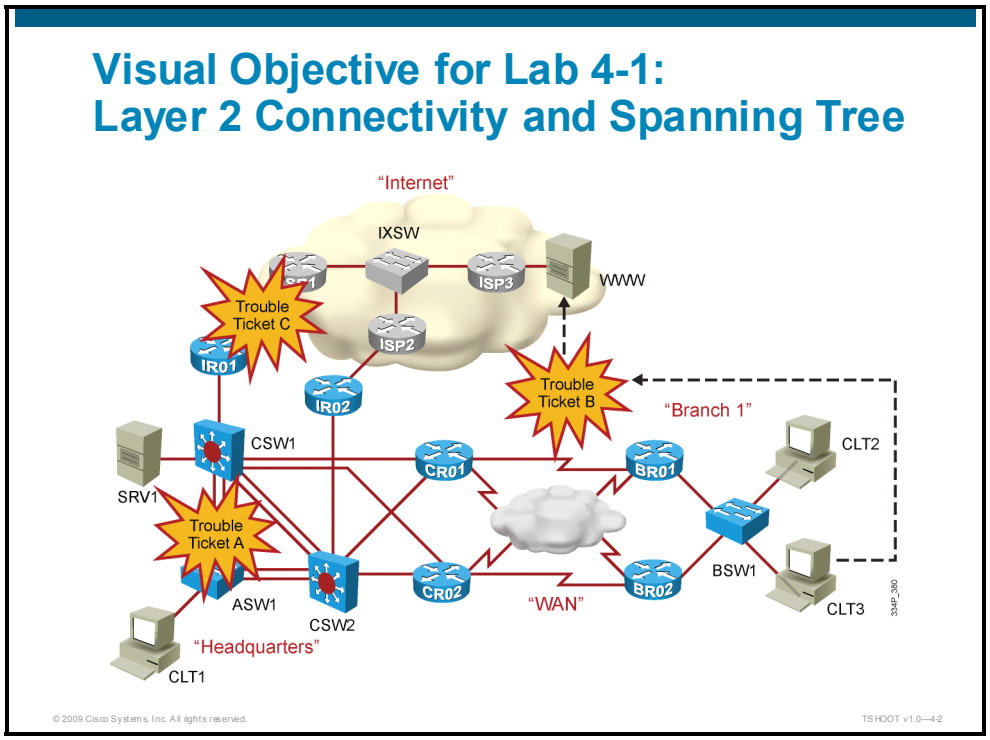
Objectives

Upon completing this lesson, you will be able to diagnose and resolve Layer 2 connectivity problems using a structured, methodical approach. This ability includes being able to meet these objectives:

- Compare your solution, findings, and action log against a set of checkpoints provided by the instructor and identify common and alternate solutions
- Consolidate the lessons learned during the review discussions into a set of best-practice methods and commands to aid you in future troubleshooting procedures

Review and Verification

This topic describes the problems that were introduced in Lab 4-1, asks how you can verify that you have solved the problems, and gives an example of a troubleshooting process that will allow you to find and resolve the issues.



This lab consists of three different trouble tickets: Trouble Ticket A, which revolves around the connectivity from access switch ASW1, Trouble Ticket B, which is concerned with connectivity from the guest VLAN in the branch office to the Internet, and Trouble Ticket C, which dealt with the connectivity between headquarters and service provider ISP 1. The problems introduced in each ticket do not affect the other tickets, so the troubleshooting processes for each of these three tickets can take place in parallel.

Trouble Ticket A: Switch Replacement Gone Bad

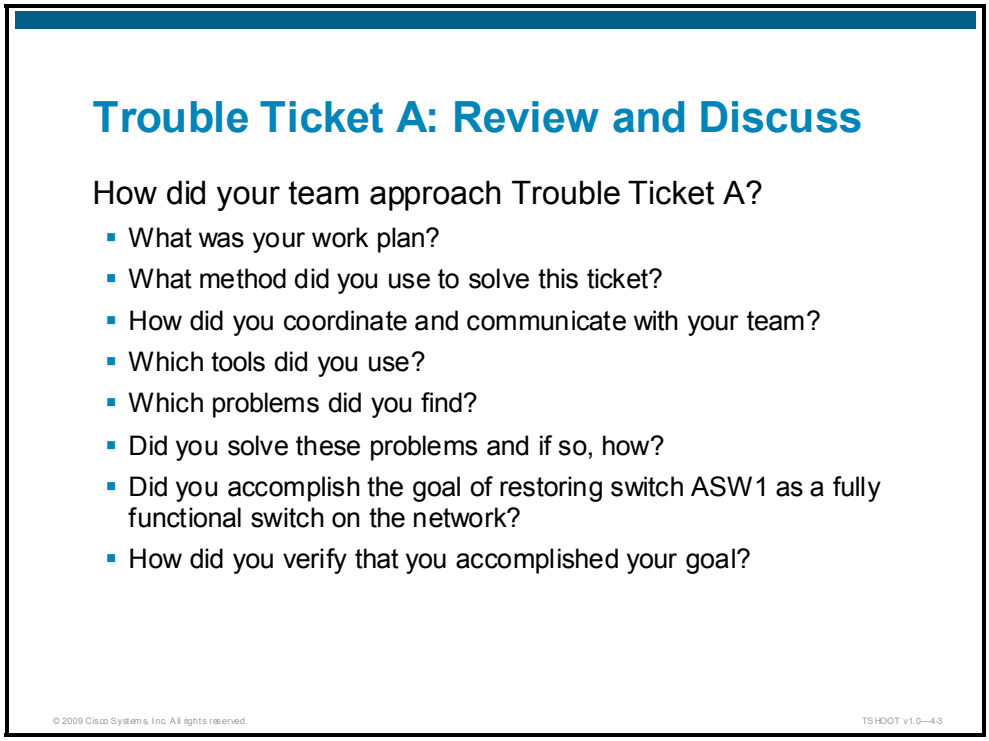
The text introducing this trouble ticket was the following:

Late yesterday afternoon, access switch ASW1 failed and you quickly had to come to the conclusion that the power supply had gone bad and that the switch needed to be replaced. Luckily, you still had a comparable switch on the shelf and you tasked a couple of your junior colleagues (who have only been with the company for two weeks) with the replacement of this switch in order to get an idea of their skill level.

This morning, when you come in and ask them how things went, they tell you that they stayed late trying to restore switch ASW1, but in the end, they could not, so they ask you to have a look because they are out of ideas. When you ask them what the exact problem is, they tell you that they do not know and that it “simply does not work”.

Users on the first floor have already started to complain that they cannot get access to the network and they had expected this problem to be fixed today.

Your task is to diagnose the issues and restore switch ASW1 as a fully functional access switch on the network.



Note	This lesson provides sample answers and solutions to the questions posed during the debrief discussions. Keep in mind that alternate solutions and methods might be just as valid and that the solutions provided here only highlight a sample approach and solution.

One of the biggest potential problems when troubleshooting is multiple people trying to diagnose the same problem at the same time and making changes without coordination or communication. A sample work plan for a team consisting of four members would be:

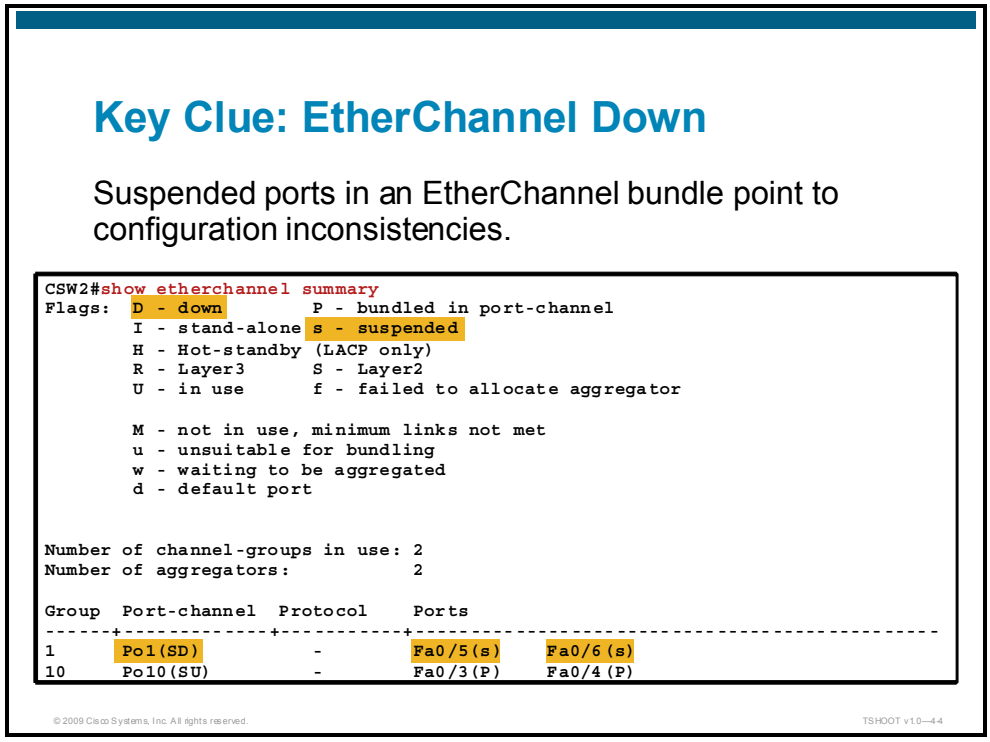
- Team member 1 will be responsible for Trouble Ticket B. Team member 1 will control the switch BSW1 and the routers BRO1 and BRO2.
- Team member 2 will take responsibility for Trouble Ticket C. Team member 1 will control switch CSW1 and routers IRO1 and IRO2.
- Team members 3 and 4 will work on Trouble Ticket A together. Team member 3 will control switch ASW1 and router CRO1, while team member 4 will control switch CSW2 and router CRO2.
- The person that is in control of a particular device is the only person allowed to connect to that device's console port. Team members may use Telnet or SSH to access all devices and use **show** commands to diagnose, but any disruptive actions such as making changes to the configuration, reloading the device, or starting debugging can be done only with the permission of the controlling team member.
- Problems diagnosed and changes made to the configurations should be communicated to all team members.

Although there are many possible methods that you could use to solve this problem, we will list an example of a troubleshooting log for team member 3 that follows the method described in the Sample Troubleshooting Flows section of the lab guide.

Trouble Ticket	Actions and results
Ticket A	CLT1: Tried to release and renew the IP address: No address is assigned.
	ASW1: Verified that PC1's MAC address is learned on port fa 0/7 in VLAN 17
	CSW1: The MAC address of CLT1 is not learned on CSW1.
	CSW2: The MAC address of CLT1 is not learned on CSW2 either.
	ASW1: Verified Cisco Discovery Protocol: Neither CSW1 nor CSW2 are listed as neighbors.
	Conclusion: Clearly, there are two problems: one between ASW1 and CSW1 and one between ASW1 and CSW2. Decision: Try to restore connectivity via CSW2 first, because we control both ASW1 and CSW2. Will research the problem between ASW1 and CSW1 later. Notified team members.
	ASW1: Verified link status to CSW1. Port channel 2 and constituting interfaces are up and **show etherchannel summary** shows no issues.
	CSW2: Port channel 1 is down. **show etherchannel summary** shows that both physical interfaces are suspended.
	CSW2: Log shows two %EC-5-CANNOT_BUNDLE2 messages. Reason listed is that Fa 0/5 and Fa0/6 are not compatible with Po1 because the VLAN mask is different
	CSW2: Compared port channel interface to physical interfaces: VLAN allowed list is missing on physical interfaces.
	Team member 4 on CSW2: Configuration changed: Added allowed VLAN list to physical interfaces. Both prechange and postchange configurations were saved in flash on the switch. Communicated to team members.
	CSW2: Port-channel 1 is now up and the Cisco Discovery Protocol shows ASW1.
	ASW1: Received a log message on the console: %SPANTREE-2-PVSTSIM_FAIL: Blocking designated port Po2: Inconsistent superior PVST BPDU received on VLAN 17, claiming root 24593:001f.2721.8400
	ASW1: Verified spanning tree for VLAN 17: Port-channel 2 is in spanning tree state "BKN*" and Type "Bound(PVST) *PVST_Inc" for VLAN 17. Need to research!
	CSW2: Verified spanning tree for VLAN 17: Port-channel 1 is listed with port type "P2P Peer (STP)." Research spanning tree command output for meaning of "Peer (STP)."
	ASW1 & CSW2: Verify STP protocol used: CSW2 uses Rapid PVST+, ASW1 uses MST.
	Checked baseline: All switches should run Rapid PVST+.
	ASW1: Configuration changed: **spanning-tree mode rapid-pvst**. Both prechange and postchange configurations were saved in flash on the switch. Communicated to team members.
	CSW2: MAC address of PC1 is now learned in VLAN 17.
	CLT1: DHCP works. Ping to default gateway works. Ping to SRV1 works. Browsing to http://www.isp3.local works.
	SRV1: Telnet to ASW1 does not work. Ping does not work either.
	ASW1: Verified IP interfaces: Interface VLAN 128 is down.
	ASW1: Verified VLANs: VLAN 128 does not exist.
	ASW1: Configuration changed: Added VLAN 128 with name "MGMT." Both prechange and postchange configurations were saved in flash on the switch. Communicated to team members.
	SRV1: Telnet to ASW1 works.
	Problem fixed. Documented changes. Backed up configurations to server. Closed ticket.

Trouble Ticket	Actions and results
	Follow up on problem between ASW1 and CSW1.
	ASW1 & CSW1: Links in EtherChannel between ASW1 and CSW1 are all up. No errors on the ports. Statistics show input and output packets after clearing the counters.
	ASW1 & CSW1: Verified that CDP is enabled on the interfaces that belong to the EtherChannel between ASW1 and CSW1.
	CSW1: Verified trunk configuration. It turns out that port channel 1 is configured for ISL.
	Checked with team member in charge of CSW1.
	CSW1: Configuration changed: Encapsulation set to 802.1q for port channel 1. Both prechange and postchange configurations were saved in flash on the switch. Communicated to team members.
	ASW1 & CSW1: The switches now list each other in the output of **show cdp neighbors**.
	ASW1: Spanning tree shows port channel 1 being blocked for VLAN 17 in accordance with the baseline.
	Problem fixed. Documented changes. Backed up configurations to server. Closed ticket.

Next, you will review a number of important **show** commands from this troubleshooting flow that highlight the problems.



A primary command used to troubleshoot problems with EtherChannel bundles is the **show etherchannel summary** command. It presents a concise overview of all links that are configured for EtherChannel, the status of the individual physical interfaces, and the logical port-channel interfaces. In the output in the figure, you can clearly see that there is a problem with Port-channel 1, because it is marked with a "D" to indicate that it is down, while the physical interfaces are marked with an "s" to indicate that they have been suspended.

When you see physical interfaces in an EtherChannel that are marked as suspended, this usually indicates that there is a configuration mismatch, either between interfaces in the channel itself or between the configuration on this end of the EtherChannel and the configuration at the other end. To find more detail about what exactly caused the physical interfaces you can use the **show etherchannel** *number* **detail** command, or search the log for a message that tells you why the links were suspended.

In this example, the output of **show etherchannel 1 detail** would indicate the following:

```
CSW2#show etherchannel 1 detail
Group state = L2
Ports: 2   Maxports = 8
Port-channels: 1 Max Port-channels = 1
Protocol:    -
Minimum Links: 0
                            Ports in the group:
                            -------------------
Port: Fa0/5
------------

Port state    = Up Cnt-bndl Suspend Not-in-Bndl
Channel group = 1           Mode = On              Gcchange = -
Port-channel  = null        GC   =   -             Pseudo port-channel = Po1
Port index    = 0           Load = 0x00            Protocol =    -

Age of the port in the current state: 0d:00h:25m:13s

Probable reason: vlan mask is different
Port: Fa0/6
```

```
------------

Port state    = Up Cnt-bndl Suspend Not-in-Bndl
Channel group = 1           Mode = On              Gcchange = -
Port-channel  = null        GC   =   -             Pseudo port-channel = Po1
Port index    = 0           Load = 0x00            Protocol =    -

Age of the port in the current state: 0d:00h:25m:14s

Probable reason: vlan mask is different
                             Port-channels in the group:
                             ---------------------------


Port-channel: Po1
------------

Age of the Port-channel   = 0d:00h:24m:48s
Logical slot/port   = 2/1          Number of ports = 0
GC                  = 0x00000000      HotStandBy port = null
Port state          = Port-channel Ag-Not-Inuse
Protocol            =    -
Port security       = Disabled
```

In this case, the output indicates that the cause of the problem is the VLAN mask, which means that there must be a mismatch between the VLANs allowed on the port channel versus the VLANs allowed on the physical interfaces.

You could also find the problem indication from the log, which contains the following messages:

```
Mar 20 08:12:39 PDT: %EC-5-CANNOT_BUNDLE2: Fa0/5 is not compatible with Po1
and will be suspended (vlan mask is different)
Mar 20 08:12:39 PDT: %EC-5-CANNOT_BUNDLE2: Fa0/6 is not compatible with Po1
and will be suspended (vlan mask is different)
```


Key Clue: Spanning Tree Incompatibility

This switch is running 802.1s MST and has detected an incompatible configuration between itself and the neighboring switch running (Rapid) PVST+.

```
ASW1#show spanning-tree vlan 17

MST0
  Spanning tree enabled protocol mstp
  Root ID     Priority    32768
              Address     001e.79a9.b580
              This bridge is the root
              Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID   Priority    32768  (priority 32768 sys-id-ext 0)
              Address     001e.79a9.b580
              Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

Interface           Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Fa0/7               Desg FWD 200000    128.9    P2p Edge
Po1                 Desg BLK 100000    128.56   P2p
Po2                 Desg BKN*100000    128.64   P2p Bound(PVST) *PVST_Inc
```

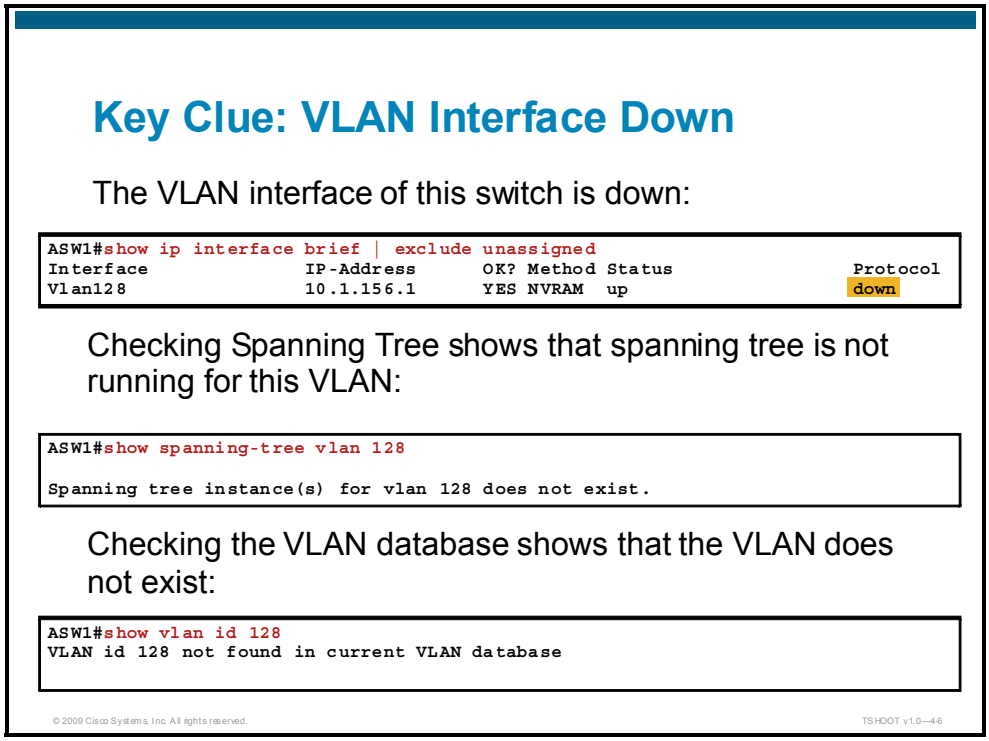

The output of this command has a couple of elements that clearly point to a spanning-tree configuration issue. The “BKN*” and “*PVST_Inc” elements in the output point toward a spanning-tree inconsistency, while the “Bound (PVST)” element points toward a boundary between two different spanning-tree varieties. Because all other switches run Rapid PVST+, it is reasonably safe to assume that switch ASW1 should not be running MST, but should be running Rapid PVST+.

The output of the command is dependent on the order in which you solve the problems. In this example, the problem with the EtherChannel between switches ASW1 and CSW2 has been solved, while the problem between switches ASW1 and CSW1 has not been solved yet. This is why an inconsistency is reported for Port-channel 2, but not for Port-channel 1.

As soon as you solve either of the two problems with the EtherChannels you will also see a log message similar to this:

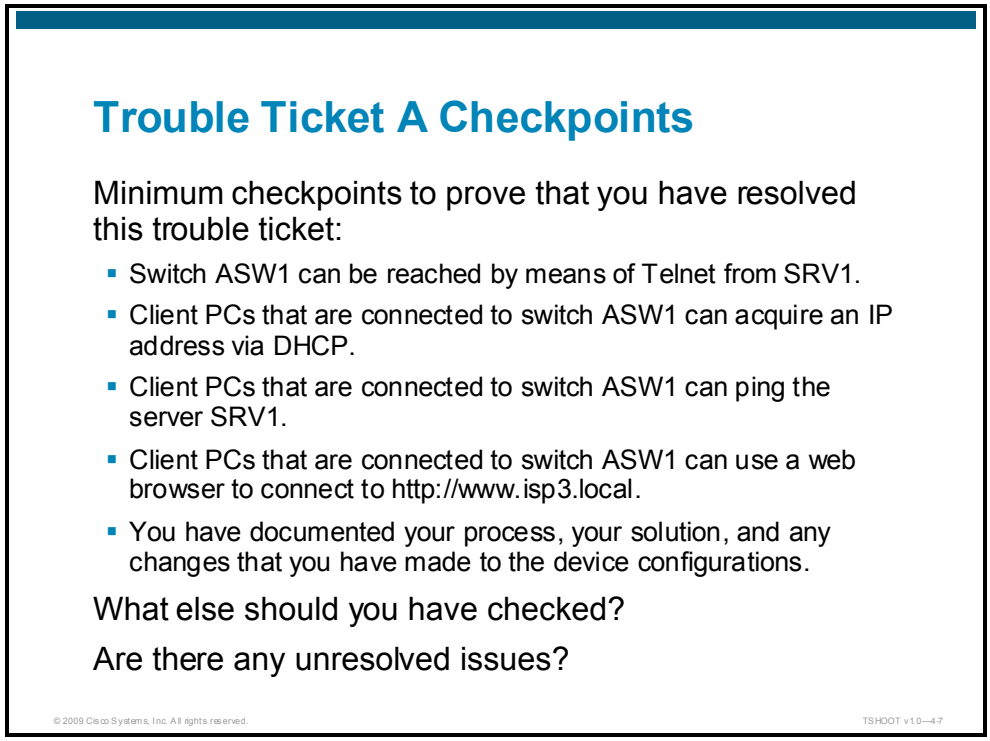
```
Mar 23 02:05:34 PDT: %SPANTREE-2-PVSTSIM_FAIL: Blocking designated
port Po2: Inconsistent superior PVST BPDU received on VLAN 17,
claiming root 24593:001f.2721.8400
```

This message also points toward an inconsistency between spanning-tree protocol varieties.



For the ISL/802.1Q mismatch, there are no commands that clearly indicate that a trunk encapsulation is the root cause of the problem. You will need to discover this by carefully checking baselines and checking matching trunk parameters on both sides of the switches.

The missing VLAN 128 is easier to detect. A VLAN interface is up as long as the VLAN exists and there is an active port in that VLAN that is in spanning-tree forwarding state. Therefore, when you discover that a VLAN interface is down, it is a good idea to first check the spanning-tree status for that VLAN. By doing so, you discover that spanning tree is not running for that VLAN, which should quickly lead to the hypothesis that the VLAN does not exist, which is then confirmed.



These are the minimal requirements that you should meet to be able to say that you have successfully resolved the problems in Trouble Ticket A.

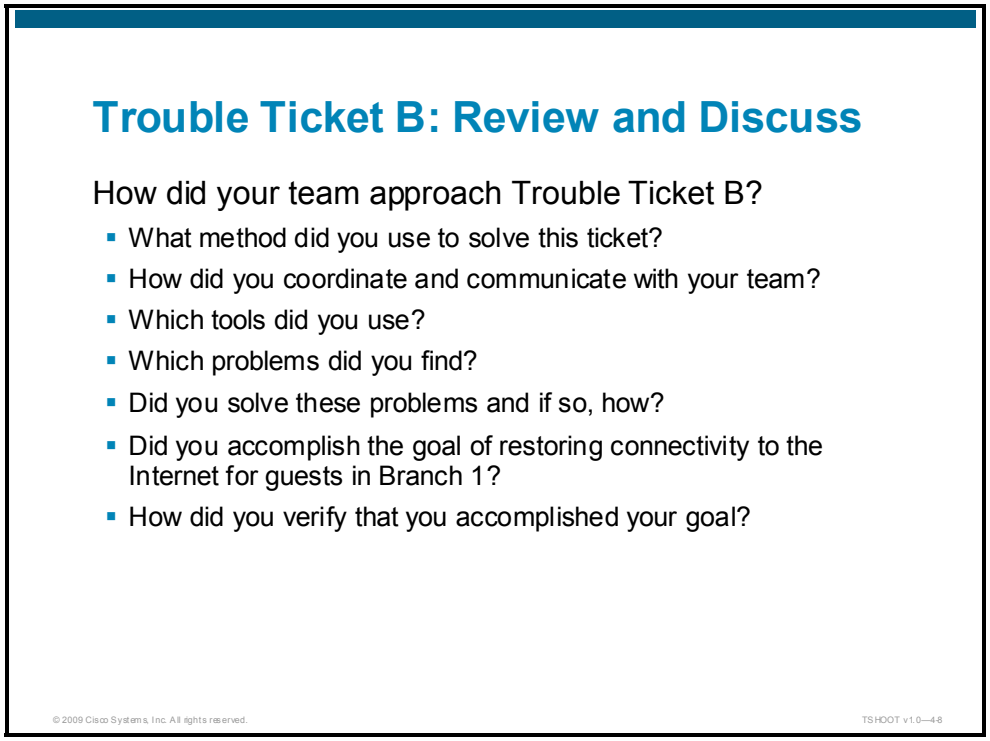
What else should you have verified? Because the network has redundant connections and devices, it is not sufficient to verify that you have connectivity when all links are connected. You should also verify that the network would still operate correctly if any of the redundant components fail. In this lab, there was a problem with the links between switches CSW1 and ASW1 and another problem with the links between switches CSW2 and ASW1. Fixing one of these problems will restore connectivity, but you will need to fix both issues to have the level of availability that was intended in the design of the network.

Trouble Ticket B: Guest Access Problem in Branch

The text introducing this trouble ticket was the following:

This morning, there was a call from one of the branch offices: An external consultant came in today and needs access to the Internet and email. The consultant's PC, CLT3, was plugged into one of the outlets that are patched to the guest VLAN on switch BSW1. However, the consultant has not been able to get an IP address and cannot get onto the network.

Your task is to diagnose and solve this problem, making sure that the consultant gets Internet access.



Most likely, the work plan and method that you used to tackle Trouble Ticket B were very similar to the approach used to fix the problems in Trouble Ticket A. The primary issue in this ticket is the fact that the guest VLAN was excluded from the list of allowed VLANs on the trunks of BSW1 that lead to routers BRO1 and BRO2.

The following troubleshooting log provides an example of how this problem could have been resolved.

Trouble Ticket	Actions and results
Ticket B	CLT3: Tried to release and renew the IP address. No address is assigned
	CLT2: Verified connectivity on CLT2, which is in the office VLAN. No problems here. Conclusion: The problem is restricted to the guest VLAN, or even CLT3 alone.
	Check baseline: BRO1 is the DHCP server for VLAN 19 B1S1-GUEST.
	BSW1: MAC address of CLT3 is learned on port Fa0/8 in VLAN 19.
	BRO1: Tried to ping BRO2 in VLAN 19: No response. Conclusion: Problem seems to be with VLAN 19 in general, not just CLT3.
	BSW1: Investigated VLAN 19. Interfaces Fa0/3 and Fa 0/4 are not listed as members of VLAN 19.
	BSW1: Investigated trunking on Fa 0/3 and Fa 0/4. VLAN 19 is not included in the list of allowed VLANs.
	BSW1: Configuration changed: Added VLAN 19 to the allowed VLAN list on both trunks. Both prechange and postchange configurations were saved in flash on the switch. Communicated to team members.
	CLT3: DHCP works. Ping to default gateway works. Browsing to http://www.isp3.local works.
	CLT3: Verified that CLT3 cannot access file-services on SRV1.
	BRO1: Ping to BRO2 in VLAN 19 succeeds.
	Problem fixed. Documented changes. Backed up configurations to server. Closed ticket.

The following section highlights the primary **show** commands used in this procedure.

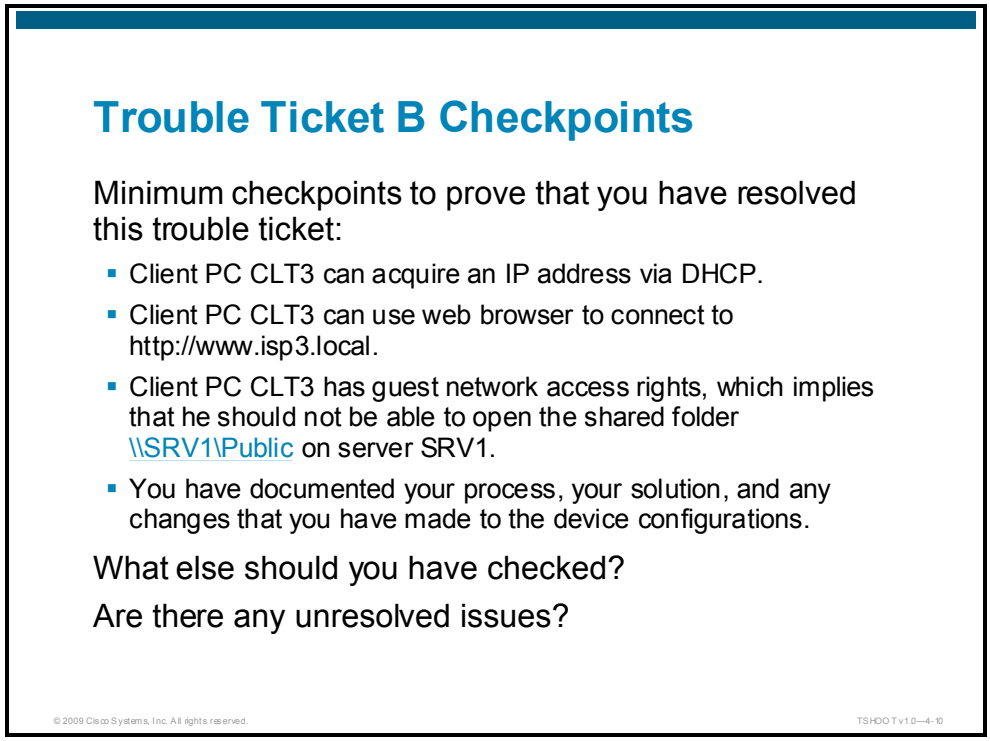
Key Clue: Trunk Ports Missing in VLAN

The list of ports that are members of VLAN 19 does not include the uplink ports FastEthernet 0/3 and FastEthernet 0/4.

```
BSW1#show vlan id 19

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
19   B1S1-GUEST                       active    Fa0/8

VLAN Type  SAID       MTU   Parent RingNo BridgeNo Stp  BrdgMode Trans1 Trans2
---- ----- ---------- ----- ------ ------ -------- ---- -------- ------ ------
19   enet  100019     1500  -      -      -        -    -        0      0

Remote SPAN VLAN
----------------
Disabled

Primary Secondary Type              Ports
------- --------- ----------------- ------------------------------------------
```


When you have an issue that seems to be limited to a particular VLAN, it is good practice to verify that all ports that you expect to be a member of that VLAN are actually listed as a member of that VLAN in the output of the **show vlan id** *vlan-id* command. In this example, the trunk ports that lead from switch BSW1 to routers BRO1 and BRO2 are not listed, which points to a problem with these trunk ports.

In this example, verification of the trunking status of these ports by use of the **show interfaces trunk** command will reveal that VLAN 19 is not included in the list of allowed VLANs for the trunk ports.



The figure shows a list of minimum requirements that you should meet to consider this trouble ticket to be resolved.

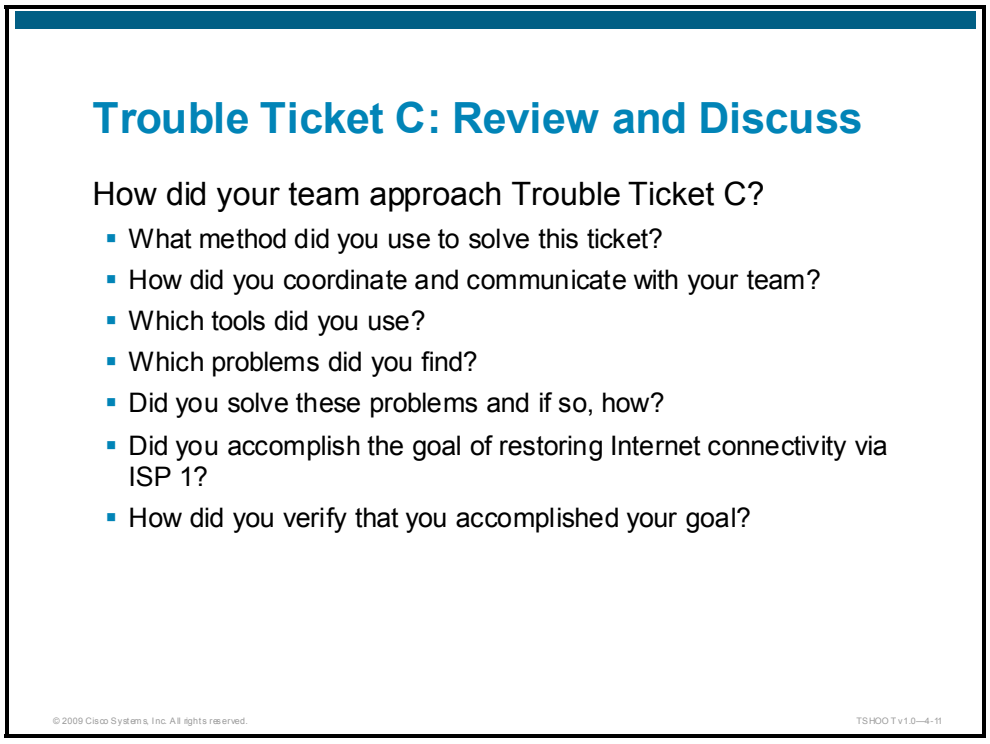
If at any point during troubleshooting you have removed the access list that limits guest access on routers BRO1 and BRO2, you need to ensure that it is reapplied and that CLT3 does not have more rights than necessary. Check your baseline to see what the access list should have permitted and denied.

Trouble Ticket C: Internet Service Provider 1 Seems to be Down

The text introducing this trouble ticket was the following:

The network management system has reported that the connection to Internet Service Provider 1, which is tracked by pinging the IP address of their router, is down. This does not cause any immediate problems because all traffic is routed via Internet Service Provider 2, but the issue needs to be researched and either solved or escalated to Internet Service Provider 1.

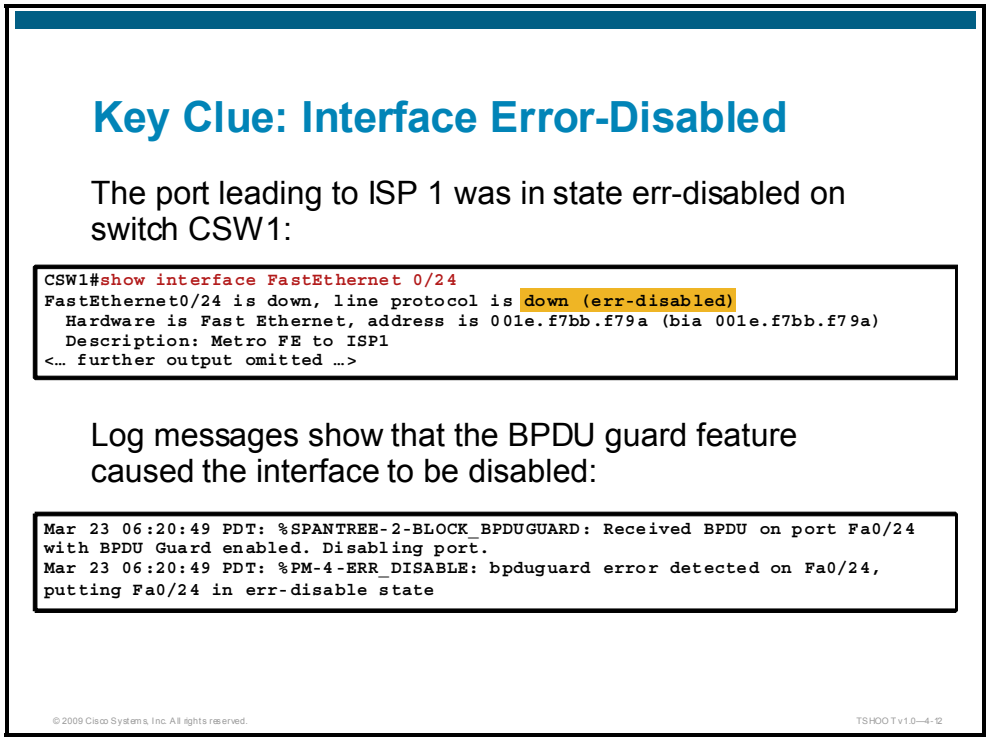
Your task is to research this issue and then to either resolve the problem, or if it cannot be resolved on your side, to escalate it to Internet Service Provider 1 with a clear report of why you think that the problem is on their end.



Again, the work plan and method you used to tackle Trouble Ticket C are likely to be similar to the approach used to fix the problems in Trouble Tickets A and B. The biggest difference between this ticket and the other tickets is that the problem is not caused by a change in the configurations of the devices in your pod, but by a change in the service provider network. Therefore, the troubleshooting procedure for this ticket will not result in a change made on a device in your pod, but in an escalation procedure to report the problem to the service provider.

A troubleshooting log is listed here to provide you with an example of how this problem could have been resolved.

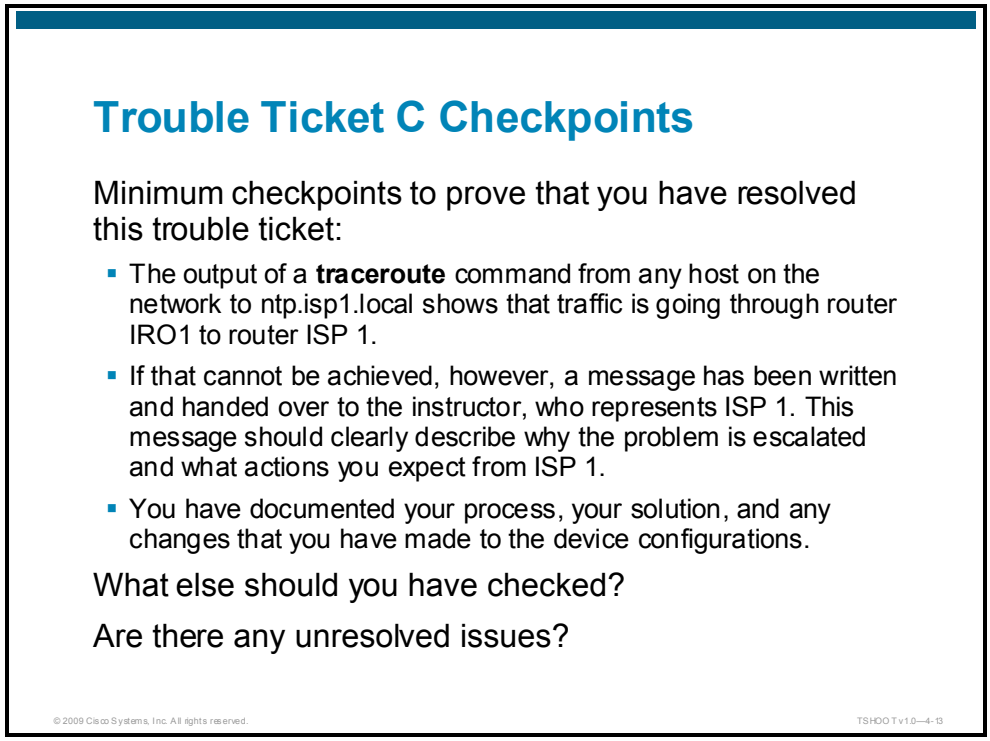
Trouble Ticket	Actions and results
Ticket C	SRV1: Confirmed problem: Traceroute to ntp.isp1.local goes through IRO2 and ISP2.
	IRO1: Ping to 192.168.224.254 (router ISP1) fails.
	CSW1: Verified that the MAC address of IRO1 is learned on Fa 0/15 in VLAN 11.
	CSW1: Interface Fa 0/24 is down (err-disabled).
	CSW1: Log message shows that the BPDU guard feature has disabled Fa 0/24, because BPDUs were received from ISP 1 on port Fa0/24.
	Check baseline: BPDU guard should be enabled toward ISP 1 and ISP 2.
	Escalated problem to ISP. Check again in 30 minutes if no feedback is received.
	After 15 minutes, the connection to ISP 1 came back.
	SRV1: Verified that traceroute to ntp.isp1.local goes through IRO1 and ISP1.
	Documented solution. Closed ticket. Communicated to team members.



A quick verification of the physical interfaces that are involved in the connection between routers IRO1 and ISP1 shows that interface FastEthernet 0/24 on switch CSW1, which leads to ISP1, is in the error-disabled state. The error-disabled state is usually caused by features that are configured to disable a port when a certain condition occurs. In this case, the bridge protocol data unit (BPDU) guard feature has disabled the interface, triggered by the reception of BPDUs from ISP 1.

On switch CSW1, the BPDU guard feature is configured to try to re-enable an error-disabled port after 5 minutes (300 seconds). As long as ISP 1 keeps sending BPDUs, switch CSW1 will display a log message on the console every 5 minutes, showing that it has disabled the port again due to reception of a BPDU.

After the service provider fixes the problem, the port will be automatically re-enabled within 5 minutes and traffic will start flowing via ISP 1 again.

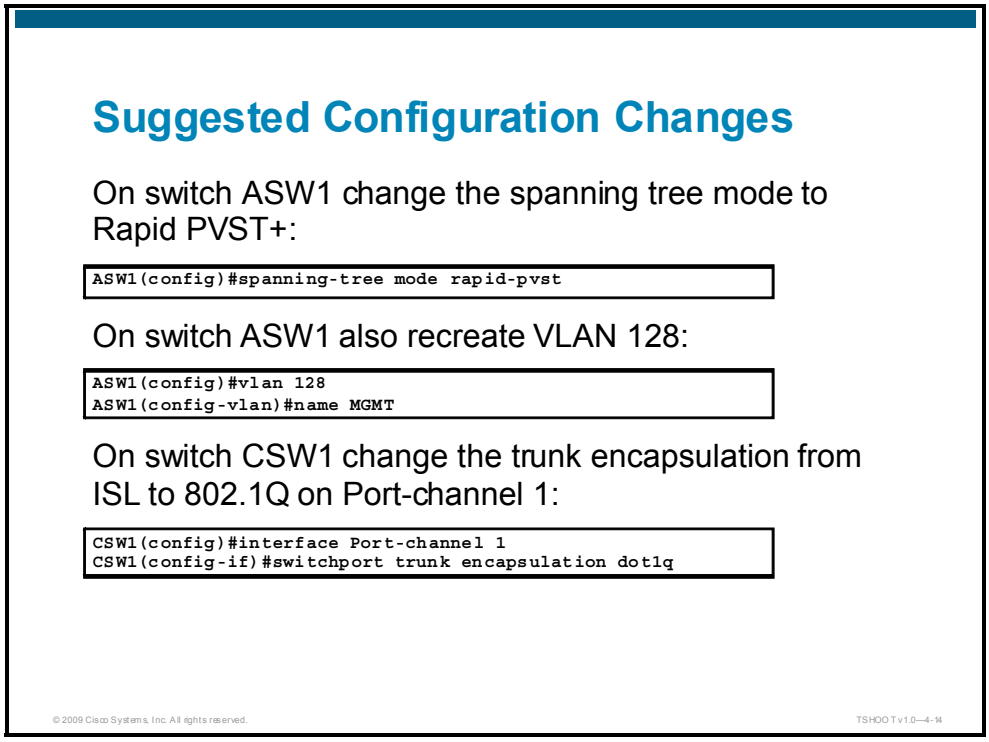


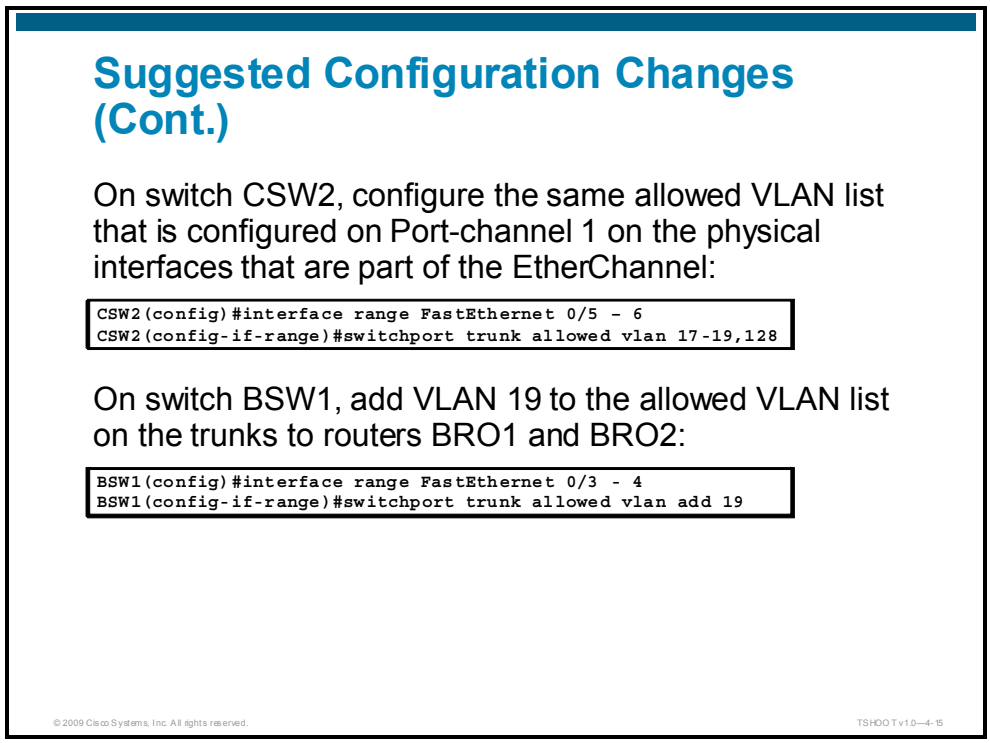
The figure shows a list of minimum requirements that you should meet to consider this trouble ticket to be resolved.

Whether the ticket is fully resolved by re-enabling the interface to ISP 1 depends on the timing of your lab. You can consider your troubleshooting process finished if you have escalated the problem in a clear manner to your instructor who takes the role of ISP 1. Depending on the timing of the lab, the instructor might act on this by stopping the transmission of BPDUs from the service provider switch, thereby allowing the link to ISP1 to be re-enabled.

Suggested Solutions

Correcting the switch configurations as follows can solve the problems introduced in the lab.





The commands shown in the figures will fix the problems that were introduced in this lab and will return the configuration to its original baseline. Other solutions might also be valid. If you have a different solution, bring it to the attention of the group and discuss the possible advantages or disadvantages of your solution.

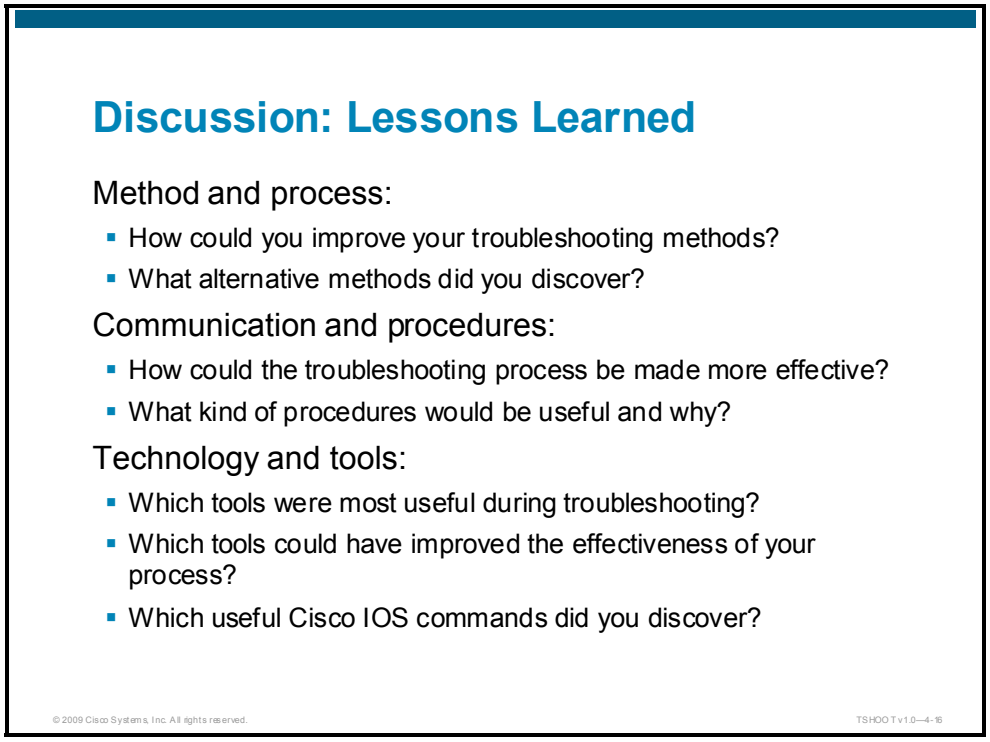
Some things that you might have configured but that are not considered a valid solution for these scenarios are:

- **Removing BPDU guard on switch CSW1:** It might be tempting to remove the BPDU guard feature from interface FastEthernet 0/24 on switch CSW1 to solve the problem between switches CSW1 and ISP 1. However, BPDU guard is a safety feature, and it was enabled in the baseline configurations, so removing it would alter the functionality that was intended by the network design. The situation did not warrant making an emergency change and overruling the baseline configuration, because there was no real outage. The Internet could still be reached via ISP2.
- **Changing the guest access list on routers BRO1 or BRO2:** As part of the troubleshooting process, you might have removed or changed the access list that limits guest access from routers BRO1 or BRO2. If you did so, you should have restored the access list, because it was not the cause of the problem, and by removing or changing it, you alter the functionality of the guest access to the network.

Most of the value of these exercises is in the process, not in the solution itself. Reflect on the process that you have followed and try to find ways to improve that process.

Consolidation

This topic describes the primary lessons that you could learn from the lab exercise.

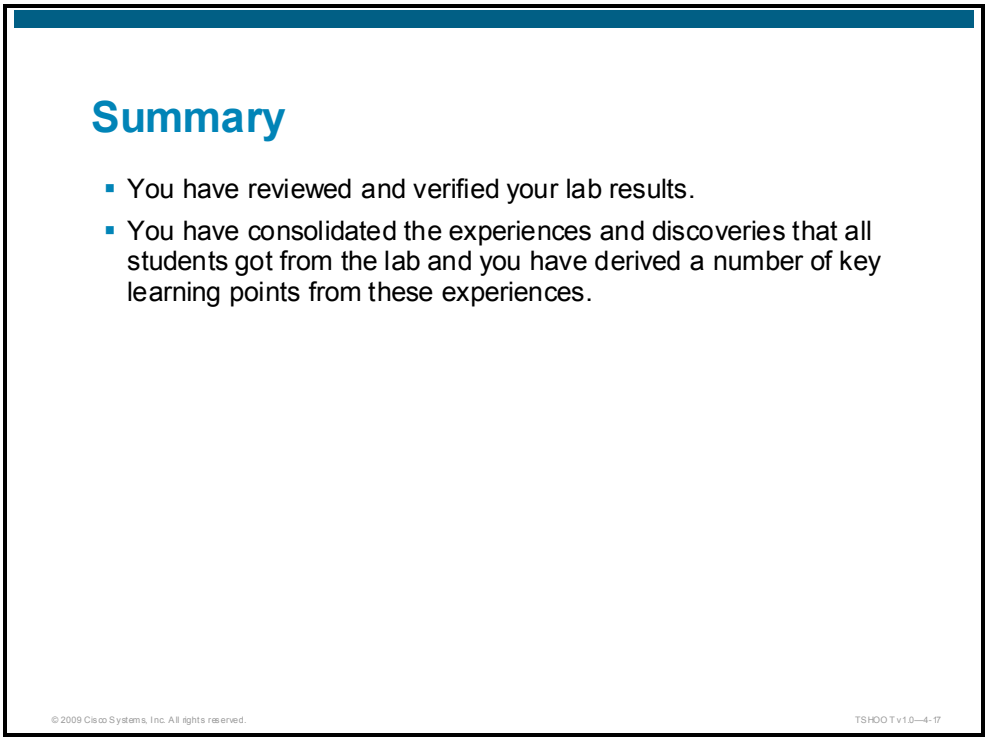


Think about all the things that you learned during the lab itself and during the debrief discussions. There is room to write down primary learning points in the Lab Debrief Notes section of the lab guide.

In addition to thinking of the methods, processes, and tools as they were used in the lab, reflect on how these would apply to your own organization.

Summary

This topic summarizes the key points that were discussed in this lesson.



Lesson 4

Troubleshooting Switched Virtual Interfaces and Inter-VLAN Routing

Overview

The traditional distinction between routers and switches has become blurred over the past decade. Multilayer switches have taken over the role of the router in the campus LAN environment and are even being used in other Ethernet-based environments as a replacement for the traditional router.

It is important for network engineers to understand the differences between hardware-accelerated Layer 3 switching and software-based routing architectures, and how these differences translate to the troubleshooting process that you would employ to troubleshoot Layer 3 problems on a multilayer switch versus troubleshooting Layer 3 problems on a router.

In this lesson, you will review multilayer switching concepts and learn how to diagnose specific problems related to multilayer switching and switched virtual interfaces (SVIs).

Objectives

Upon completing this lesson, you will be able to diagnose problems with SVIs and inter-VLAN routing. This ability includes being able to meet these objectives:

- Utilize the information contained in the data structures used in the operation of multilayer switching to diagnose issues related to multilayer switching
- Diagnose problems related to SVIs and routed ports based on an understanding of the essential differences between these two types of Layer 3 interfaces on a multilayer switch

Inter-VLAN Routing and Multilayer Switching

This topic describes the differences and similarities between routing and multilayer switching.

Routing and Multilayer Switching

Discussion: compare and contrast:

- What are the differences between routing and multilayer switching?
- What are the similarities between routing and multilayer switching?
- How can you recognize a router?
- How can you recognize a multilayer switch?

Cisco Catalyst 6504 Switch

Cisco 7206 Router

Over the past decade, the distinction between a router and a switch has become more and more blurred. In essence, a multilayer switch is a "switch that can route," while on the other hand a "router that can switch" can be created by inserting an Ethernet switching module in a modular router. Therefore, when it comes to troubleshooting, there is not much difference between troubleshooting IP routing on a multilayer switch and troubleshooting IP routing on a router.

So ask yourself this question: "What are the similarities and what are the differences between multilayer switches and routers?"

The following are the similarities between multilayer switching and routing:

- Both routers and multilayer switches use routing protocols or static routes to maintain information about the reachability and direction of network prefixes and record this information in a routing table.
- Both routers and multilayer switches perform the same functional packet switching actions: They receive a frame, strip off the Layer 2 header, perform a Layer 3 lookup to determine the outbound interface and next hop, encapsulate the packet in a new Layer 2 frame, and transmit the frame.

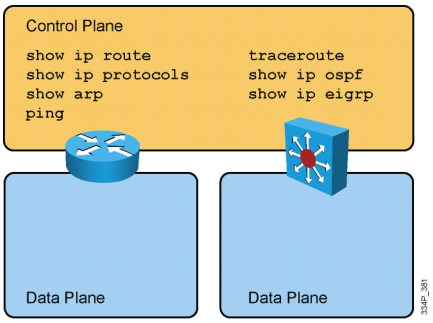
The following are the differences between multilayer switching and routing:

- Routers support a wide variety of media and interfaces. Multilayer switches are typically Ethernet only (and if other media are supported on switches, it is usually only a small subset of the full range of technologies that is available on routers).

- Multilayer switches utilize specialized hardware to achieve wire-speed Ethernet-to-Ethernet IP packet switching. Low-end to midrange routers use multipurpose hardware to perform the packet-switching process. On average, the packet-switching throughput of routers is lower than the packet-switching throughput of multilayer switches.
- Routers usually support a wider range of features, because switches need specialized hardware to be able to support certain data plane features or protocols, whereas on routers, you can often add features through a software update.

Control Plane Versus Data Plane

Troubleshooting the control plane is similar on routers and multilayer switches.

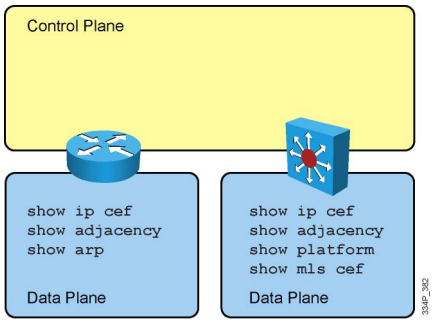


So what does this mean from a troubleshooting perspective?

The process of troubleshooting the control plane is exactly the same for routers and multilayer switches. There is no difference between troubleshooting Open Shortest Path First (OSPF) or Enhanced Interior Gateway Routing Protocol (EIGRP) on a multilayer switch compared to troubleshooting OSPF or EIGRP on a router. This means that you can use exactly the same toolkit of Cisco IOS commands on both routers and multilayer switches.

Control Plane Versus Data Plane (Cont.)

Different tools are available for troubleshooting the data plane on routers and multilayer switches.



Troubleshooting of data plane problems (like performance problems) is different, due to differences in the implementation of the packet-switching process.

Routers use Cisco Express Forwarding as the main packet switching mechanism. The Cisco Express Forwarding Forwarding Information Base (FIB) and adjacency table are both stored in the router's main memory and are consulted by the router to forward packets using the Cisco Express Forwarding switching method. The router builds the Cisco Express Forwarding data structures by combining information from a number of control plane data structures like the routing table and Address Resolution Protocol (ARP) cache. Therefore, the information in the Cisco Express Forwarding data structures should accurately reflect the information in the control plane data structures. Under normal circumstances, checking the control plane data structures should be sufficient when you are troubleshooting IP routing. However, if you run into a situation where all control plane information looks correct, but packets are not being forwarded as expected, you might need to check the Cisco Express Forwarding data structures and verify that they accurately reflect the control plane information.

There are two main commands to check the Cisco Express Forwarding data structures:

- **show ip cef** shows you the content of the Cisco Express Forwarding FIB. The content of this table largely reflects the content of the routing table, but also holds additional entries for directly connected hosts, the router's own IP addresses, and multicast and broadcast addresses.
- **show adjacency** shows you the content of the adjacency table. This table contains the complete Layer 2 frame headers that are used by the router for packets that are switched through each of these adjacencies.

Switches also use Cisco Express Forwarding as the Layer 3 packet-switching mechanism. Just like routers, they build the Cisco Express Forwarding FIB and adjacency table data structures in the main memory of the Route Processor (RP). The difference with routers is that the switches do not simply use these tables for packet forwarding, but they compile and download the information contained in the Cisco Express Forwarding and adjacency table into specialized ternary content addressable memory (TCAM) and then use specialized hardware to forward packets at high speeds based on the information contained in the TCAM memory. The exact process of compiling, downloading, and utilizing the Cisco Express Forwarding information in TCAMs is specific to each of the different types of switch hardware architectures.

Although the exact process and the way information is stored in the TCAMs is dependent on the switch hardware architecture, it is generally possible to gather information about the state of the content of the TCAMs through Cisco IOS commands.

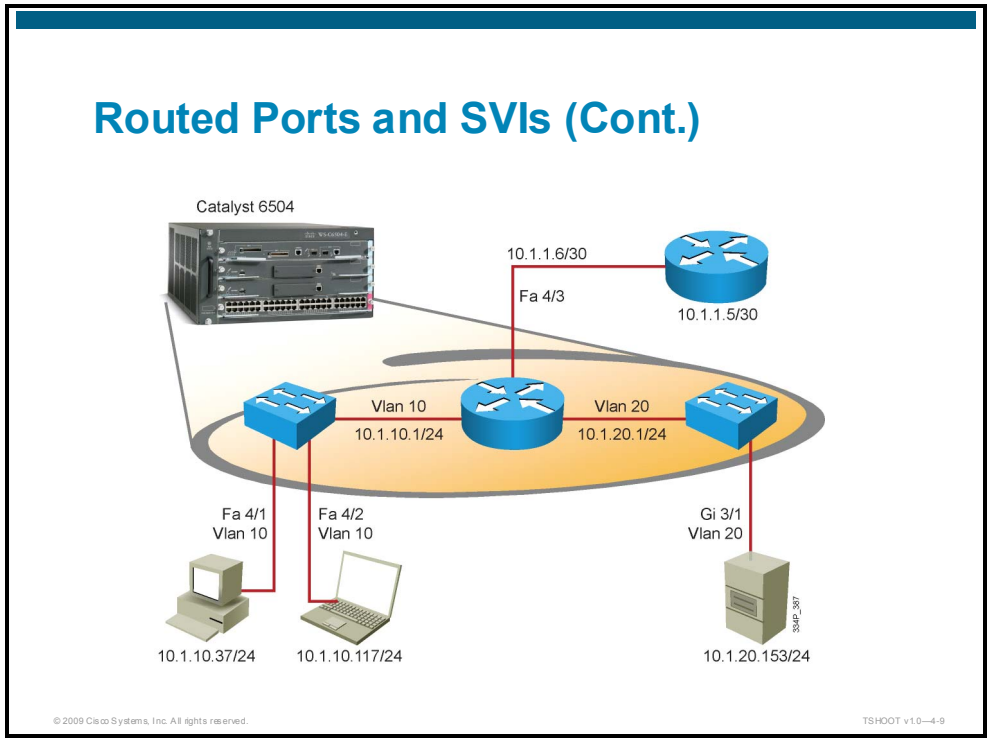
To extract information about the forwarding behavior of switches from the TCAMs on some of the common Cisco Catalyst series switches you can use the following commands:

- On the Catalyst 3560, 3750 and 4500 platforms, the **show platform** family of commands can be used to obtain detailed information about the forwarding behavior of the hardware.
- On the Catalyst 6500 platform, the **show mls cef** family of commands can be used to obtain detailed information about the forwarding behavior of the hardware.

Note	For a more detailed explanation and sample output of the commands referenced in this lesson, see the “Sample Troubleshooting Flows” section in the course lab guide and the Cisco IOS Command References on http://www.cisco.com.

Switched Virtual Interfaces and Routed Ports

This topic describes the differences between the two major types of Layer 3 ports on multilayer switches.



A multilayer switch essentially provides three different core functions in a single device:

- **Layer 2 switching within each VLAN:** This is accomplished by assigning physical ports to VLANs and setting up trunks between this switch and connected switches. No IP configuration is necessary.
- **Routing and multilayer switching between the VLANs connected to the switch:** To provide Layer 3 switching between VLANs connected to a switch, SVIs need to be configured. In the Cisco IOS command language, these interfaces are referred to as **interface vlan** *vlan-id*. You need to configure each of these VLAN interfaces with an IP address for its corresponding subnet, which will serve as the default gateway for hosts on that VLAN. Globally, you need to enable **ip routing** to start the multilayer switching process.

- **Routing and multilayer switching between the VLANs and the outside world:** There are two different ways to accomplish this:
  - One method is to select an additional VLAN to use between the multilayer switch and the port connected to the external router. The physical port connected to the router is then assigned to that VLAN. This is essentially using the same mechanism that is used to route between VLANs; therefore, an SVI corresponding to the VLAN that was selected needs to be configured with an IP address and subnet mask that corresponds to the subnet that was reserved for the link between the router and switch.
  - The second method is to change the physical interface connected to the router into a "routed port" by using the **no switchport** command at the interface level. This changes this port into an interface that does not belong to any particular VLAN, similar to an Ethernet interface on a router (although the switch will internally assign a special VLAN to the port). No corresponding SVI needs to be configured, and the IP address and subnet mask can be configured directly on the physical interface itself.

What are the differences between the routed port and SVI methods? The main difference is that a routed port is not a Layer 2 port, which means that it does not run the typical protocols that are normally enabled by default on Layer 2 ports, such as spanning tree and Dynamic Trunking Protocol (DTP).

Another difference has to do with link status. For a routed port, there is a direct relationship between the status of the port and the availability of the corresponding directly connected subnet. When the port goes down, the corresponding connected route is immediately removed from the routing table.

However, when you are using an SVI, the subnet is not associated to any particular physical interface, but instead is associated to the VLAN interface, which is a virtual interface. So the question is, under what conditions does an SVI go down?

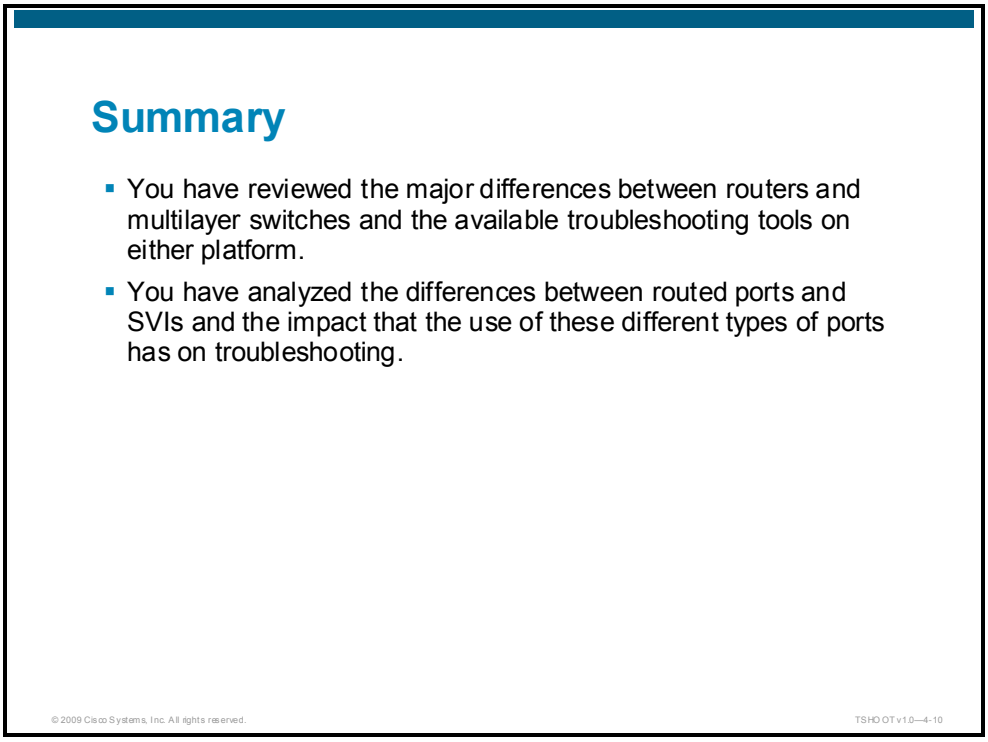
What if there are five ports assigned to a particular VLAN? Should the corresponding SVI go down when any port in that VLAN goes down? When three out of five go down? When the last port goes down?

The rule that Cisco Catalyst LAN Switches use to determine the status of an SVI is the following: an SVI is considered to be up as long as there is at least one port associated to the corresponding VLAN that is up and in the spanning-tree forwarding state. Note that this rule considers both access ports and trunks that have this VLAN in their allowed VLAN list.

This means that an SVI will go down (and the corresponding connected subnet will be removed from the routing table) only when the last active interface in the VLAN goes down or loses its spanning-tree forwarding status.

Summary

This topic summarizes the key points that were discussed in this lesson.



Lesson 5

Troubleshooting FHRPs

Overview

An essential element in building highly available networks is the implementation of a First Hop Redundancy Protocol (FHRP). Even if you have multiple routers or multilayer switches connected to a subnet, the clients and servers will still point to a single default gateway, and therefore they will lose connectivity if that gateway fails, even if a second gateway is available. FHRPs such as the Hot Standby Router Protocol (HSRP), Virtual Router Redundancy Protocol (VRRP), and Gateway Load Balancing Protocol (GLBP) can solve this issue by providing redundant default gateway functionality in a way that is transparent to the end hosts.

In this lesson, you will review the operation of the common FHRPs and you will learn how to use Cisco IOS commands to diagnose and resolve problems that might occur while using these protocols.

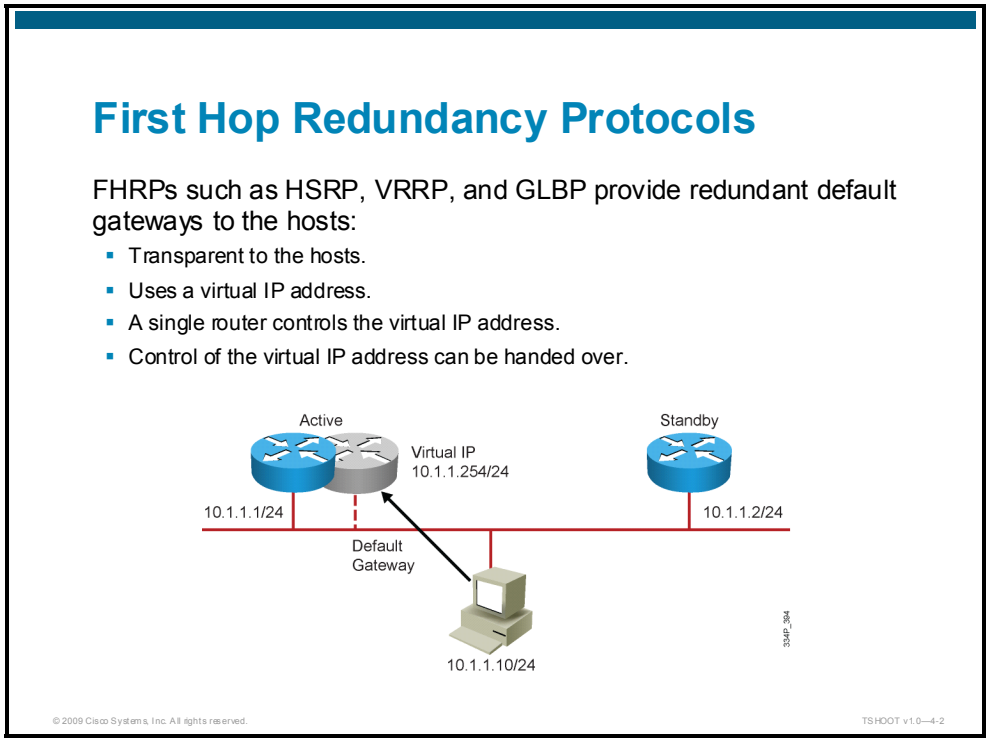
Objectives

Upon completing this lesson, you will be able to diagnose and resolve problems related to FHRPs such as HSRP, VRRP, and GLBP. This ability includes being able to meet these objectives:

- Understand the HSRP election process and packet forwarding via HSRP routers
- Verify the operation of HSRP using Cisco IOS commands
- Understand the similarities and major differences between HSRP, VRRP, and GLBP

Using HSRP for First-Hop Redundancy

This topic describes the operation of HSRP and how it can provide redundant default gateway functionality to clients.



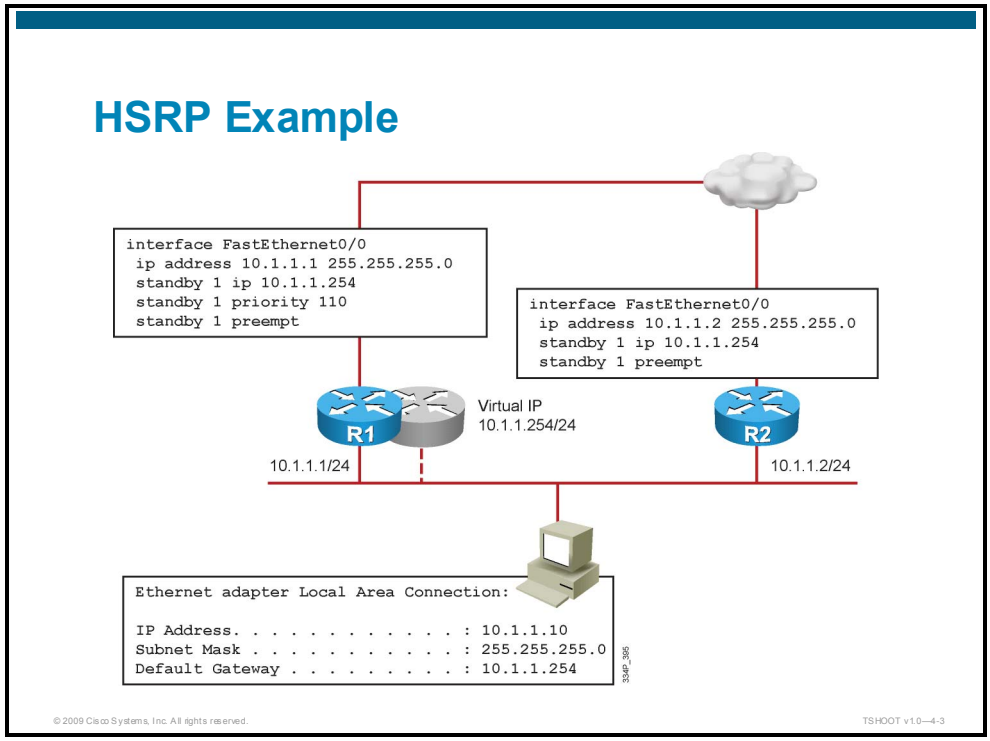
FHRPs such as HSRP, VRRP, and GLBP all serve the same purpose: they provide a redundant default gateway to hosts on a subnet and do this in such a way that the mechanism is entirely transparent to the hosts.

The basic mechanism employed by these protocols is to provide a virtual IP address (and corresponding virtual MAC address) that can be used as the default gateway by the hosts on the subnet. This virtual IP address is not bound to any particular router, but it can be controlled by any router in a group of routers participating in the protocol.

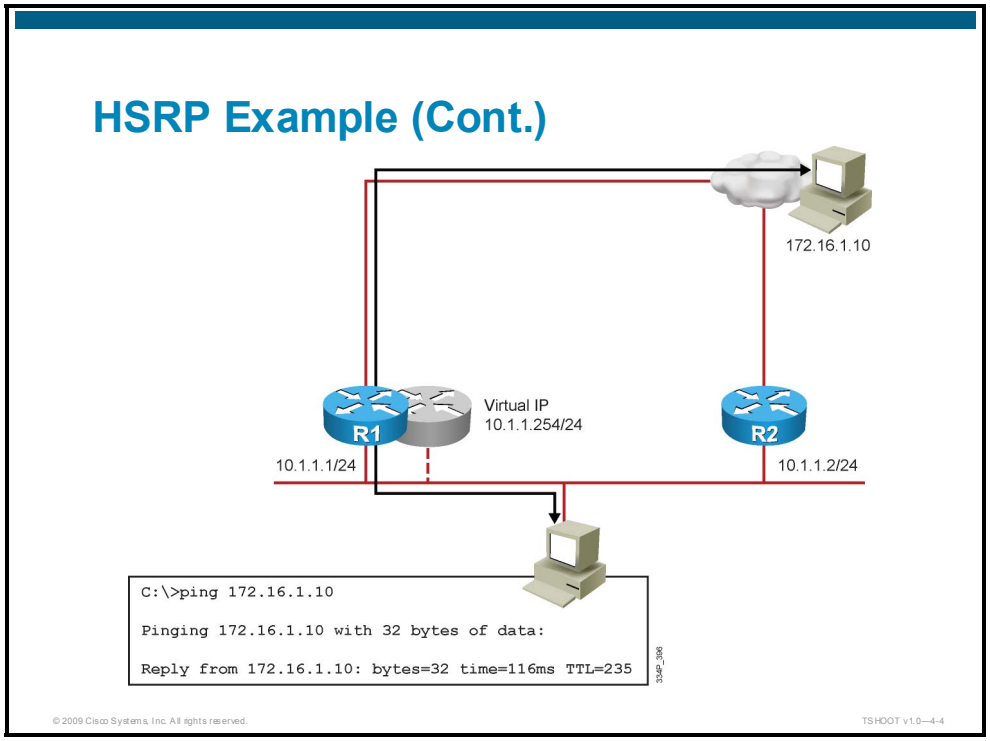
Under normal circumstances, at any given moment only one router, the active router, should have control over the virtual IP address. As a consequence, most of the mechanisms of these protocols revolve around the following functions:

- Electing a single router that controls the virtual IP address
- Tracking availability of the active router
- Determining under which conditions control of the virtual IP address should be handed over to another router

To look at a specific example that illustrates these principles, see the next figure.



Both routers have been configured for the same HSRP group and virtual IP address. Both routers have been configured for preemption. This will allow either of them to take over the role of active router when its priority is the highest of the routers in the group. R1 has been configured with a priority of 110, which is higher than the default priority of 100. This will cause R1 to be elected as the active router, while R2 will be elected as the standby router. This means that R1 will be in control of the virtual IP address and will forward packets sent to the virtual router's IP and MAC address.



When a failure occurs (for example, when the Fast Ethernet interface of R1 fails), the end hosts that are using the virtual router IP address as their default gateway lose connectivity to destinations outside of their own subnet. For a short period of time, there is no active router, so packets destined for the virtual router IP and MAC address will be dropped. After detecting the loss of hello packets from R1, router R2 concludes that R1 has stopped functioning, and R2 assumes the active role and starts forwarding packets for the virtual router IP and MAC address. Connectivity for all hosts on the subnet is restored.

Discussion: HSRP Convergence

- How long will it take for router R2 to take over the active role and start forwarding packets after the router R1 interface fails?
- What will happen when the router R1 interface is re-enabled?
- Will any packets be dropped when router R1 is re-enabled?

What effect does this process have on the network connectivity of the hosts on the subnets?

How long will it take R2 to discover that R1 is not active any longer? How long before it will take over the packet-forwarding role?

What will happen when R1 comes back? Will R1 take over the active router role? In addition, if so, how long will it take for router R1 to take over? Will any packets be dropped when R1 comes back?

Take a little time to answer these questions by yourself before reading the explanations below.

What will happen exactly when the active router fails?

First, take a look at the cause of the failure. In case of a physical failure, the only way for the standby router to detect failure of the active router is through the loss of hello packets. By default, both the active and standby router send hello packets every 3 seconds. If hellos are not received for 10 seconds (the default hold time), assumes that the active router has disappeared, and the standby router takes on the active role. This means that for a period of 10 seconds (or whatever period has been configured for the hold time), the hosts will lose connectivity, because there is no active router to forward packets. If the failure is caused by administrative actions (for example, a shutdown of an interface or reload of the router), the active HSRP router will send a “resign” message, causing the standby router to assume the active role immediately. This means that the 10-second hold time does not come into play.

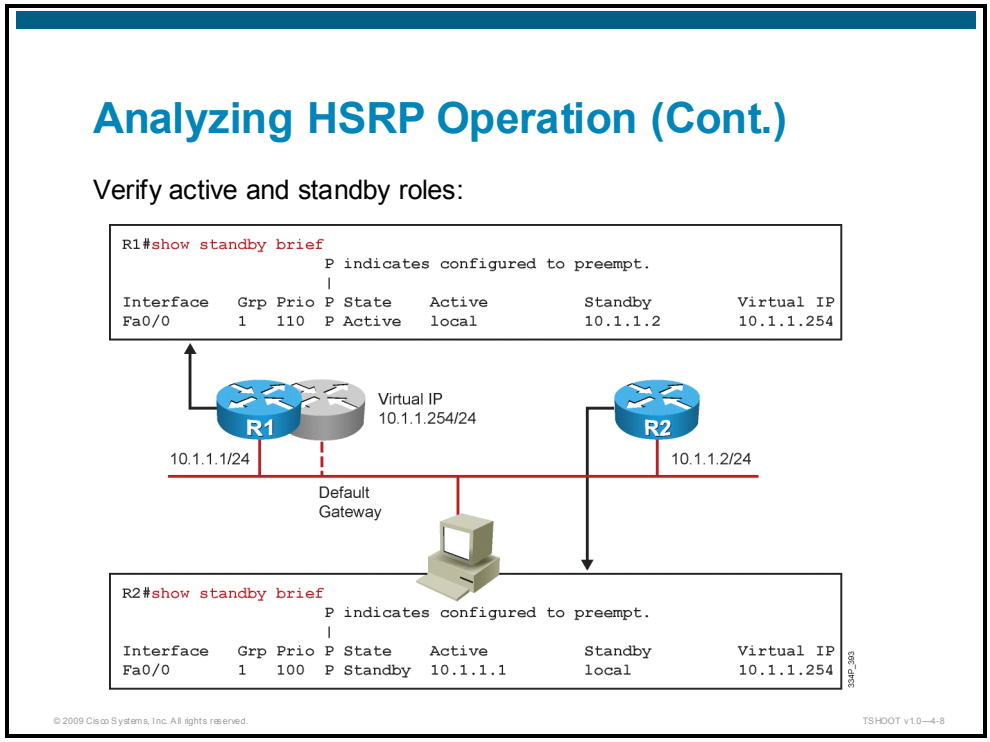
Second, you need to take into account the convergence for the return path. HSRP is only involved in the convergence of outbound packets that the host sends via the default gateway. Convergence for packets returning to the host is not governed by HSRP but by routing convergence. So typically, the overall convergence is as fast as the protocol that is the slowest to converge, which could be either HSRP or the routing protocol.

What will happen when a router that has a higher priority than the current active router is added to the HSRP group? First, this depends on whether that router has been configured to preempt or not. If it has not been configured for preemption, nothing happens. The router that is currently active will stay active, and if a standby router is already present, that router will also keep its standby role. (If no standby router is present, the new router will take on the standby role.) If, however, the new router has a higher priority and has been configured to preempt, it will take over the active role immediately. It will send out a “coup” message, telling the current active router that it will take over the active role due to its higher priority.

Will this cause any packet loss? The HSRP coup mechanism, in itself, does not cause packet loss. At all times, there will be an active router on the segment. Again, you need to consider routing convergence. If the new active router has not finished its routing convergence, it might not have a complete routing table yet, causing it to drop packets. Therefore, it is important that the new router does not assume the active role until it has fully built its routing table.

Verifying HSRP Operation

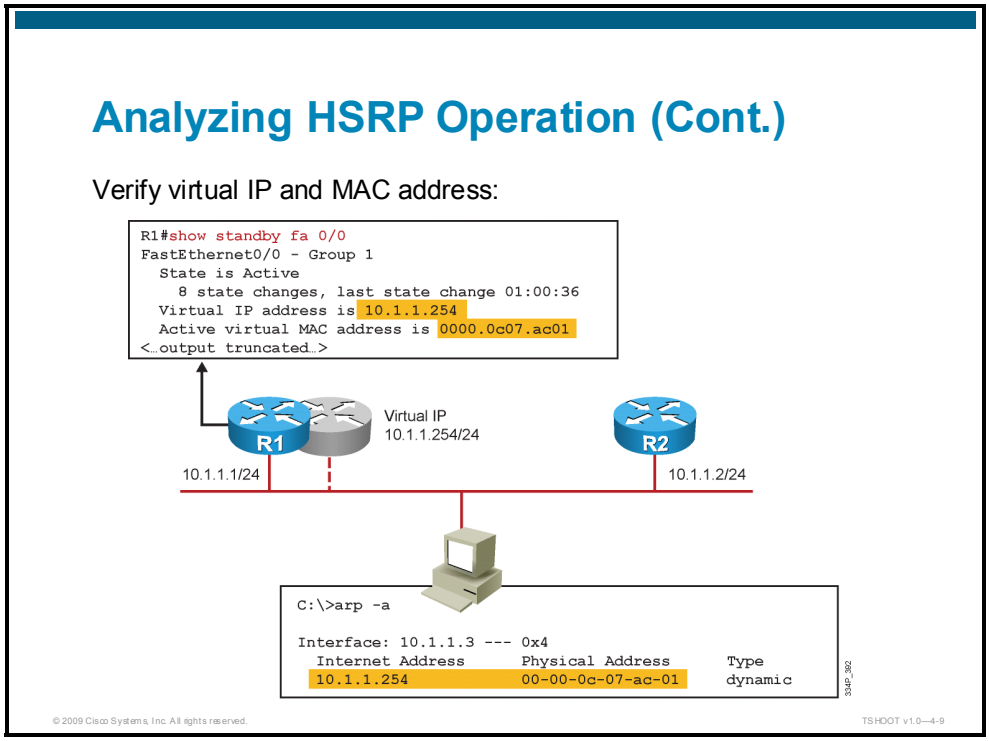
This topic describes the Cisco IOS commands that you can use to verify proper operation of HSRP and to diagnose potential problems.



What Cisco IOS commands can we use to verify proper operation of HSRP and which commands can we use to analyze the HSRP process?

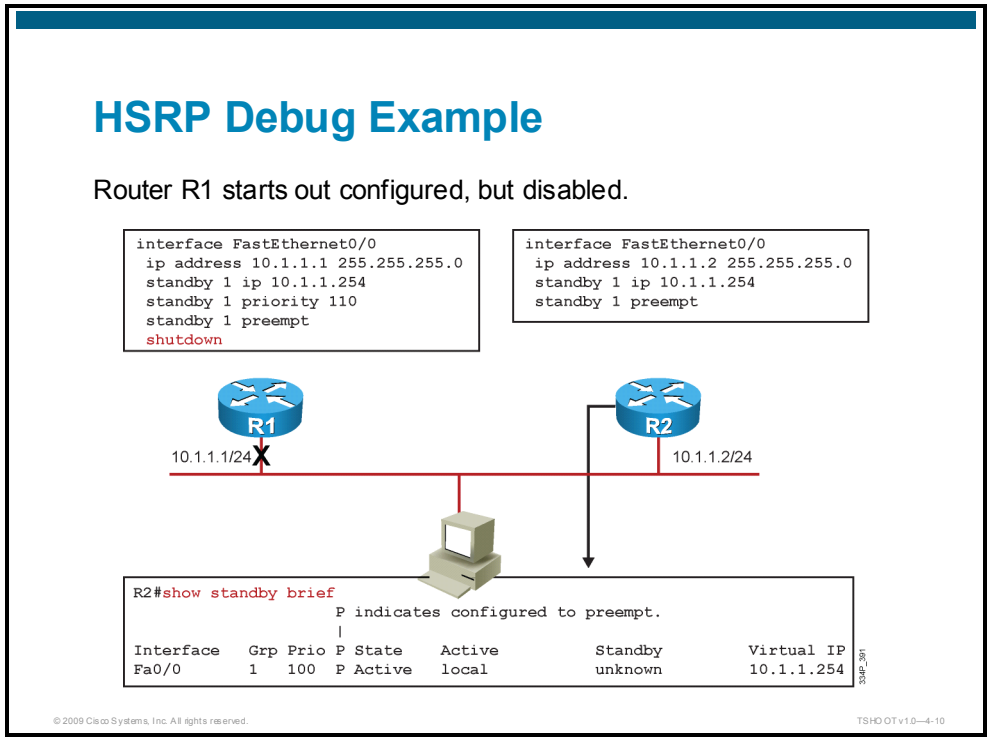
Note	In this topic, you will examine the most commonly used commands to verify and analyze HSRP operation. For additional details, consult the Cisco IOS command references on http://www.cisco.com.

The best way to get a quick overview of the actual HSRP status on the network is to use the **show standby brief** command. For each interface, this shows you the configured HSRP group, the IP addresses for the active and standby router, the virtual IP address, configured priority, and preemption.



If you want to see more-detailed information, such as configured timers, the virtual MAC address for the HSRP group, and information about recent HSRP state changes, you can use the **show standby** *interface-id* command.

Specifically, the virtual MAC address used for the standby group is relevant to know when you are troubleshooting HSRP-related problems because it can be used to verify the correct operation of the Address Resolution Protocol (ARP) and the Layer 2 connectivity between the end host and the active HSRP router. In many cases, HSRP related problems are not really, at the root, caused by HSRP itself, but by problems in the underlying switched network. For example, a typical symptom of a broadcast storm would be that you start seeing very frequent HSRP state changes on the Layer 3 switches that are connected to the affected VLANs.

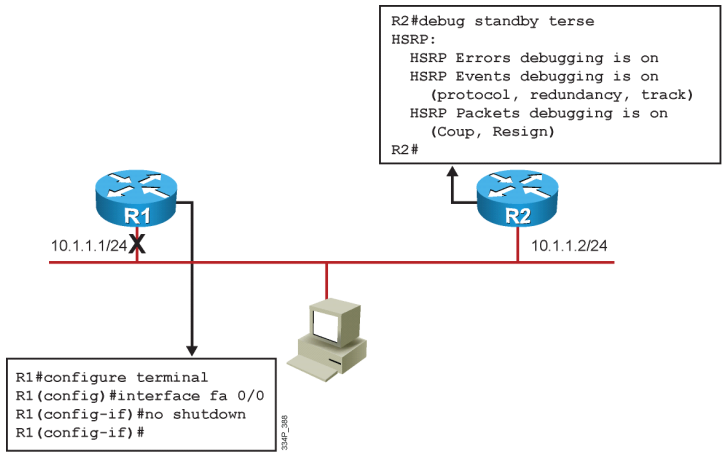


While the **show** family of commands is very useful for verification and initial diagnosis, there might be times where you want to observe the operation of HSRP in real time to gather clues about the underlying reasons that might cause HSRP to malfunction.

The situation shown in the figure illustrates what information you can learn by using **debug** commands. Both routers are configured for HSRP, but the Fast Ethernet interface on router R1 is currently shut down. Router R2 is now the active router because it is the only router on the segment.

HSRP Debug Example (Cont.)

- Debugging is started on router R2.
- The interface on router R1 is enabled.



On router R2, debugging is enabled by use of the **debug standby terse** command. This command is a good debugging command to start with because it includes most of the relevant messages but excludes the HSRP hellos. This keeps the output of the debugging commands limited and readable.

The interface on R1 is enabled and you watch the HSRP process on R2.

HSRP Debug Example (Cont.)

- Coup message sent by R1: Priority 110 is higher than R2 (100).
- R2 backs off and goes to Speak state.
- R2 is elected as the standby router and moves to Standby state.

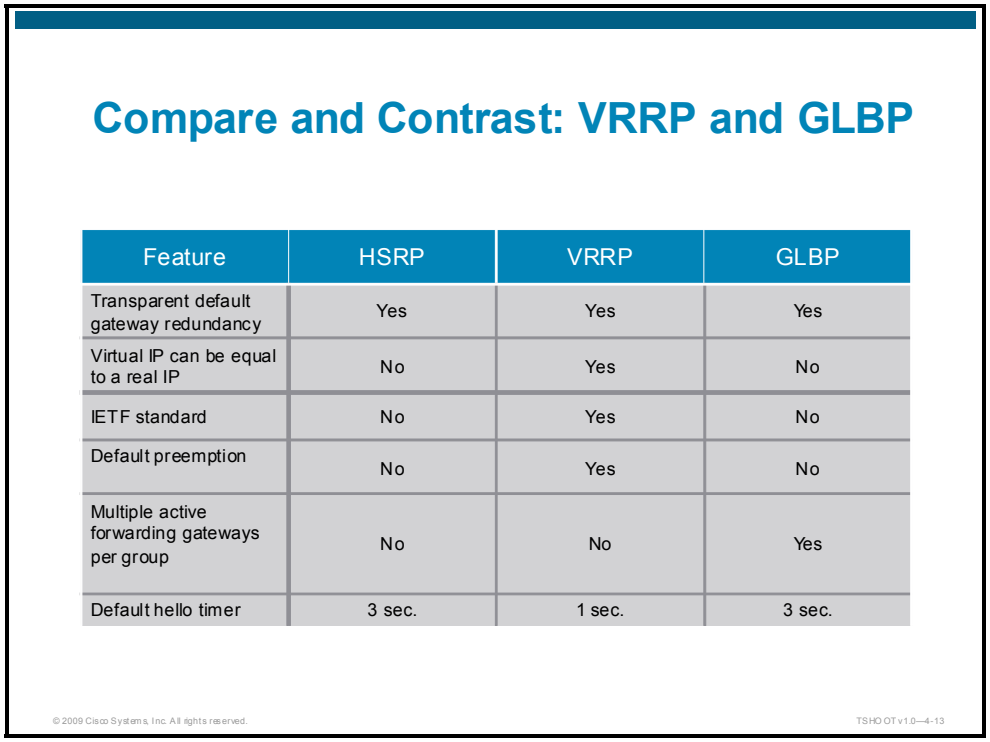
```
R2#
*Mar  1 00:16:23.555: HSRP: Fa0/0 Grp 1 Coup   in  10.1.1.1 Listen  pri 110 vIP
10.1.1.254
*Mar  1 00:16:23.555: HSRP: Fa0/0 Grp 1 Active: j/Coup rcvd from higher pri
router (110/10.1.1.1)
*Mar  1 00:16:23.555: HSRP: Fa0/0 Grp 1 Active router is 10.1.1.1, was local
*Mar  1 00:16:23.555: HSRP: Fa0/0 Grp 1 Active -> Speak
*Mar  1 00:16:23.555: %HSRP-5-STATECHANGE: FastEthernet0/0 Grp 1 state Active ->
Speak
*Mar  1 00:16:23.555: HSRP: Fa0/0 Grp 1 Redundancy "hsrp-Fa0/0-1" state Active -
> Speak
*Mar  1 00:16:33.555: HSRP: Fa0/0 Grp 1 Speak: d/Standby timer expired (unknown)
*Mar  1 00:16:33.555: HSRP: Fa0/0 Grp 1 Standby router is local
*Mar  1 00:16:33.555: HSRP: Fa0/0 Grp 1 Speak -> Standby
*Mar  1 00:16:33.555: %HSRP-5-STATECHANGE: FastEthernet0/0 Grp 1 state Speak ->
Standby
*Mar  1 00:16:33.559: HSRP: Fa0/0 Grp 1 Redundancy "hsrp-Fa0/0-1" state Speak ->
Standby
R2#
```


As you can see in the figure, the debugging command output clearly shows the subsequent steps that HSRP goes through:

1. R1 comes up on the segment and, because it has a higher priority than the current active router and is configured to preempt, it sends out a “coup” message to take over the active role.
2. R2 loses its active role, causing it to step back to the role of a nonactive, nonstandby HSRP router. Because there is currently no standby router on the segment, R2 moves to the “speak” state to announce its eligibility for the standby role.
3. R2 does not see another (better) candidate for the role of standby router for 10 seconds and thus promotes itself to the standby role.

Using VRRP and GLBP as Alternatives to HSRP

This topic describes the major differences and the similarities between HSRP and two other FHRPs, VRRP, and GLBP.



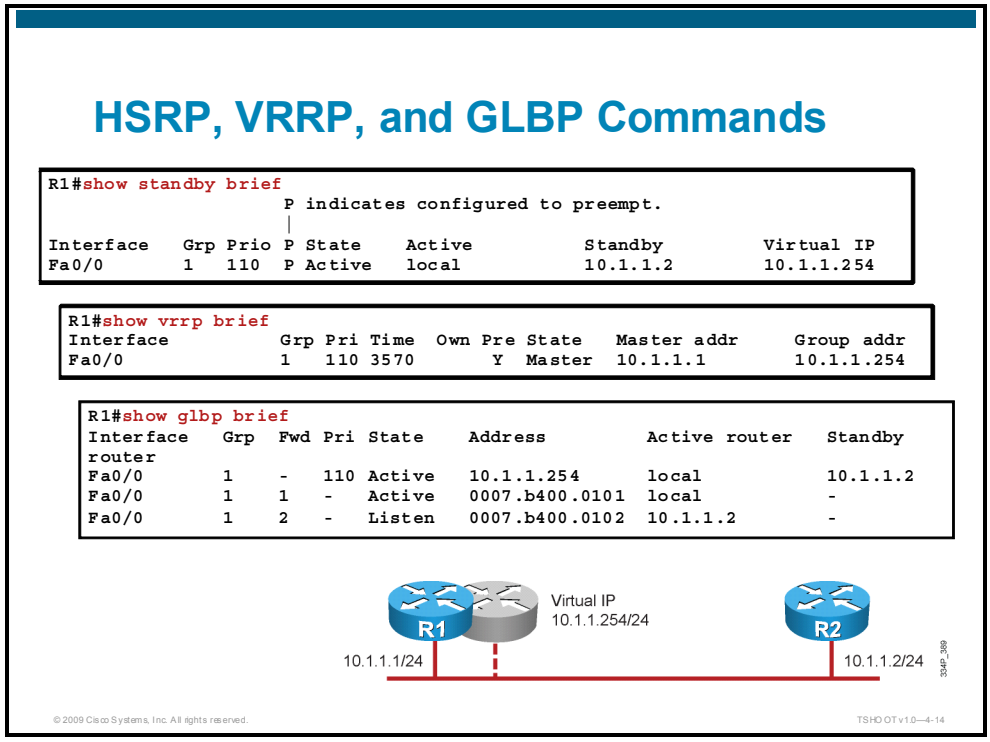
Besides HSRP, two other protocols provide first-hop redundancy, VRRP and GLBP. Because they provide a very similar service, they are also quite similar in operation and, from a troubleshooting perspective, the methods that you would use to troubleshoot these protocols are almost identical.

The Cisco IOS commands used to troubleshoot these protocols are also very similar in style to the HSRP commands. Replace the **standby** keyword with **vrrp** or **glbp** and you can use many of the same commands that you know from experience with troubleshooting HSRP.

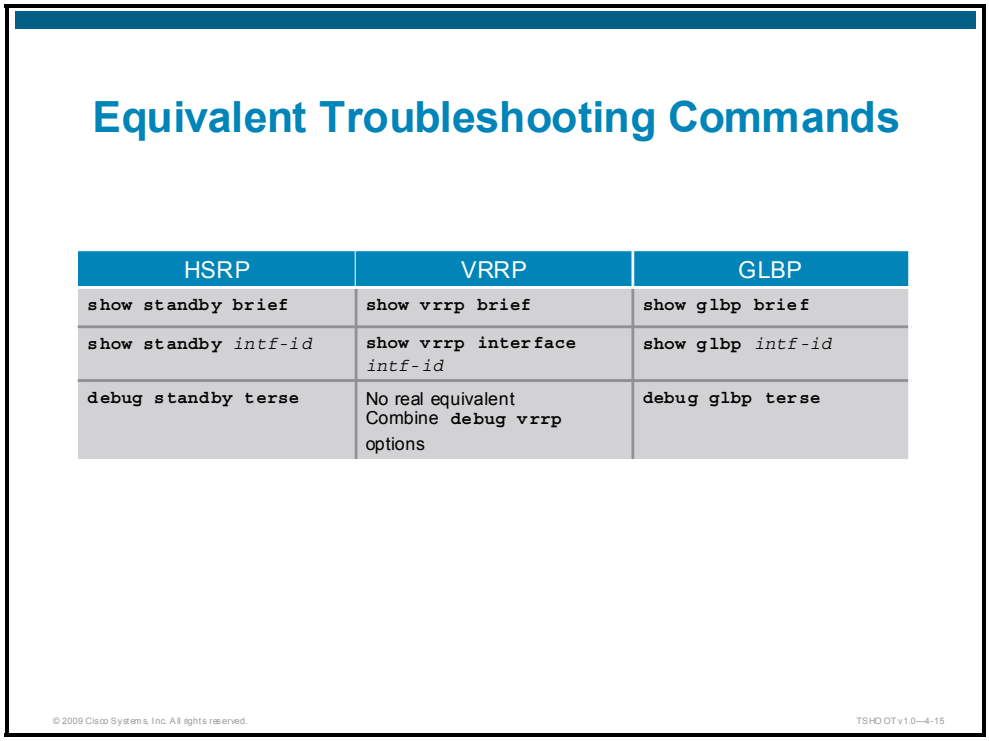
The following are the major differences between these protocols:

- HSRP and GLBP always require an additional IP address to function as the virtual IP address. With VRRP, you can use one of the router's assigned IP addresses as the VRRP IP address. Consequently, this router will then always be the master router for that IP address when it is up, even if another router has a higher priority.
- VRRP is an IETF standard (RFC 3768), which makes it suitable for multivendor environments.
- HSRP and GLBP do not preempt by default. If you want a higher-priority router to take over when it comes up on the segment, you must configure the preemption option. In VRRP, it is the opposite. Higher priority routers will preempt by default. If you do not want this, you should disable the preempt option.

- GLBP can have multiple routers forwarding traffic for a single virtual IP address. It achieves this by using multiple virtual MAC addresses for a single virtual address. There is still a single router that is in control of the virtual IP address and answers ARP requests for that IP address. This router effectively balances the load over the different forwarding routers.
- Default hello timers are different (3 seconds for HSRP, 1 second for VRRP, and 3 seconds for GLBP).



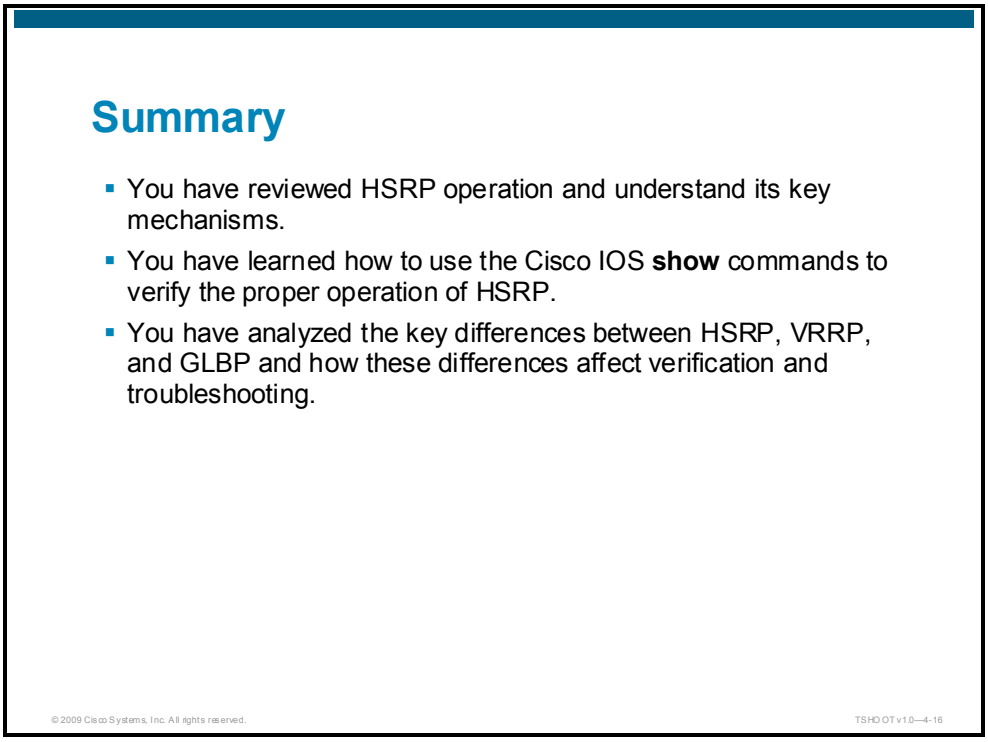
As you can see in the figure, the structure of the major commands is very similar, and if you know how to interpret the commands for one of these protocols, it is quite easy to understand the output of the other commands.



The figure shows the typical commands that you can use to troubleshoot HSRP, VRRP, and GLBP protocols. The most significant difference is that there is no **terse** debug option for VRRP, which means that you will need to select the debug options that you are interested in manually.

Summary

This topic summarizes the key points that were discussed in this lesson.



Lesson 6

Lab 4-2 Debrief

Overview

In this lab, you have practiced troubleshooting Layer 3 connectivity problems related to switched virtual interfaces (SVI), multilayer switching, and first-hop redundancy protocols (FHRPs) such as the Hot Standby Router Protocol (HSRP) and Gateway Load Balancing Protocol (GLBP). Because some of the first-hop redundancy problems were caused by underlying Layer 2 problems, you have also had an additional opportunity to practice troubleshooting Layer 2 problems related to VLANs and trunks. You have created a troubleshooting plan to coordinate the work between the team members. You have methodically researched the problems and kept a log of your actions and findings. In addition, you have finally resolved the problems and documented your solutions.

During the lab debrief, the instructor will lead a group discussion during which you can present your solutions. You will get an opportunity to verify your solution against a number of checkpoints provided by the instructor and compare your solution to other students' solutions. The instructor will discuss alternate solutions and their benefits and drawbacks.

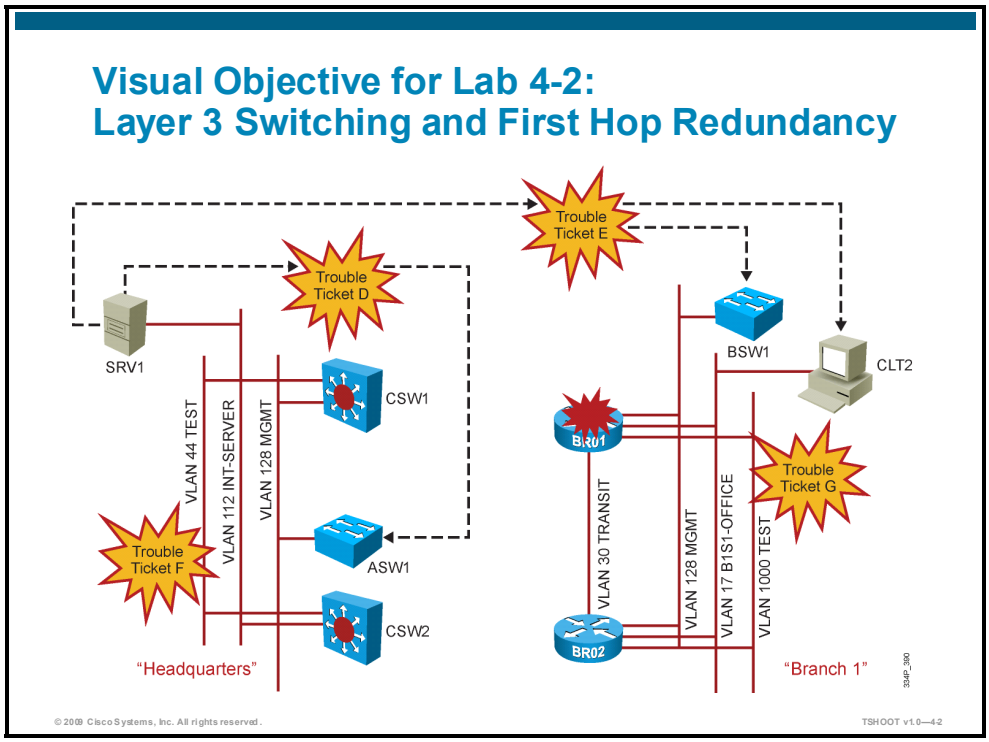
Objectives

Upon completing this lesson, you will be able to diagnose and resolve problems involving multilayer switching and FHRPs using a structured, methodical approach. This ability includes being able to meet these objectives:

- Compare your solution, your findings, and your action log against a set of checkpoints provided by the instructor and identify common and alternate solutions
- Consolidate the lessons learned during the review discussions into a set of best-practice methods and commands to aid you in future troubleshooting procedures

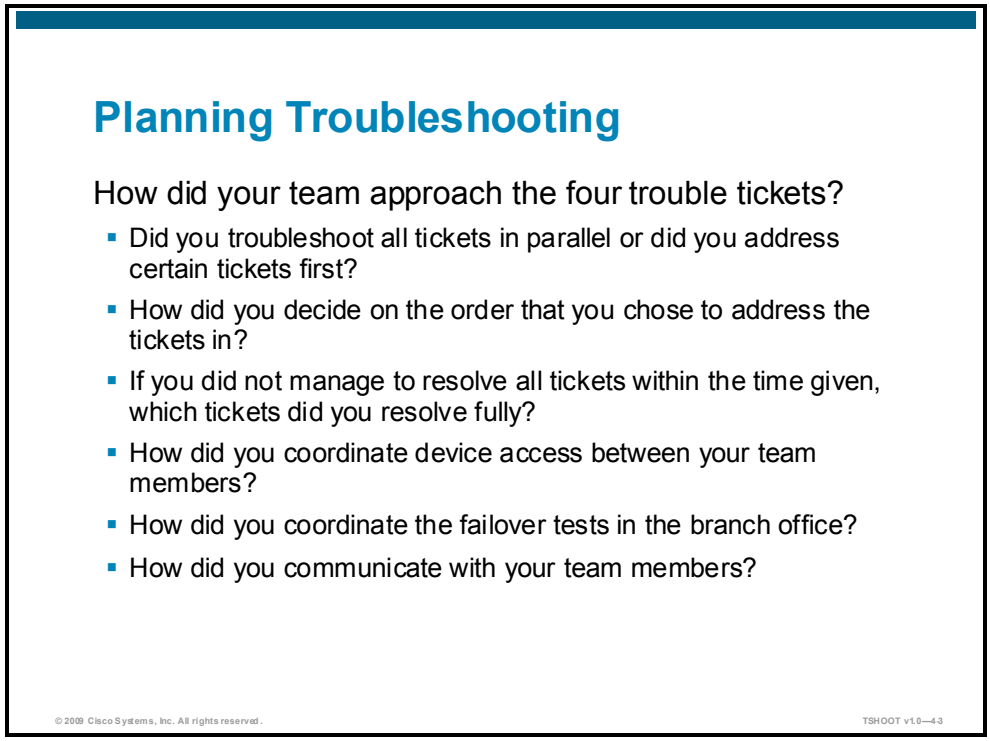
Review and Verification

This topic describes the problems that were introduced in Lab 4-2, asks how you can verify that you have solved the problems, and gives an example of a troubleshooting process that will allow you to find and resolve the issues.



This lab consists of four different trouble tickets: Trouble Ticket D is reported as a problem between access switch ASW1 and server SRV1. Trouble Ticket E involves the connectivity between server SRV1 at headquarters and switch BSW1 and client CLT2 in the branch office during a failure of router BRO1. Trouble Ticket F involves a test of authentication for HSRP on switches CSW1 and CSW2. The final ticket, Trouble Ticket G, involves another test of HSRP functionality, this time in the branch office.

These trouble tickets can be worked on in parallel, but it is important that team members working on one ticket do not disrupt the process of other team members. If you choose to work on trouble tickets sequentially, it should be noted that the impact of Trouble Tickets D and E on the production environment is higher than it is for Trouble Tickets F and G. Therefore, Trouble Tickets D and E should be addressed before moving on to Trouble Tickets F and G.



Note	This lesson provides sample answers and solutions to the questions posed during the debrief discussions. Keep in mind that alternate solutions and methods might be just as valid and that the solutions provided here only highlight a sample approach and solution.

Assessing the urgency of problems is an important step in troubleshooting and network maintenance processes. At least two of the four trouble tickets, F and G, have no operational urgency, because they concern tests that need to be executed, but are not related to any current operational issues. It could be argued that Trouble Ticket F has operational impact, because it concerns a potential security issue. However, based on the information in the ticket, the network-wide rollout of HSRP authentication will not take place for at least two weeks, until your colleague returns from vacation. Trouble Ticket D has operational impact, because the switch is unmanageable at this moment and even if users do not currently have a problem accessing the network via switch ASW1, having an unmanageable device can quickly turn into a problem, because it will be excluded from most of your regular maintenance procedures until the access problems are resolved. In a similar way, the failure between server SRV1 and the branch office is not an immediate problem in the sense that users are currently affected, but the resiliency of the network is affected, and not addressing this problem might cause unnecessary outages later. Another reason to tackle Trouble Ticket E directly and not wait is that according to the ticket you have permission to run failover tests today, because most employees are not in the office. Waiting to resolve this ticket could eventually result in postponing its resolution until a later scheduled maintenance window.

Therefore, based on urgency, Trouble Tickets D and E have a higher priority and should be addressed first if possible. Taking this into account, a sample plan for four team members could be:

- Team member 1 and 2 will work together on Trouble Tickets D and F, resolving Trouble Ticket D first, before moving on to Trouble Ticket F.
- Team member 3 and 4 will work together on Trouble Tickets E and G, addressing Trouble Ticket E first. However, given that both tickets require failover tests to be run and the tickets state that disruption to the users needs to be minimized, a reasonable strategy would be to work on Trouble Ticket E until testing becomes necessary and then switch to Trouble Ticket G. If Trouble Ticket E can be resolved quickly, the failover tests can be combined. If it turns out that Trouble Ticket G takes more time than expected, you might need to decide to give it up and resolve Trouble Ticket E before time runs out.
- Device access is distributed as follows: Team member 1 controls switch ASW1, and team member 2 controls switches CSW1 and CSW2. Team member 3 controls routers BRO1 and switch BSW1. Team member 4 controls router BRO2. Control of the other devices will be assigned as needed during the exercise.

As the communication and teamwork within your team improves, you can decide to work without fixed device assignments. Instead, you could assign device access as needed and hand over control from one team member to another when necessary during the exercise, if that improves process efficiency.

Trouble Ticket D: Switch ASW1 Cannot Be Managed from Server SRV1

When you came into the office this morning, you found a trouble ticket in the system. The text introducing this trouble ticket was the following:

"Switch ASW1 has been showing CRC errors on a group of eight ports for several days. Hardware was suspected to be the cause. During yesterday evening's maintenance window, the switch was swapped with a similar switch from a test lab. The configuration was pasted in on the console. After this replacement, clients could connect and no errors were shown on the ports. However, making a backup to server SRV1 did not work, nor is the switch reachable via Telnet or SSH from server SRV1. Unfortunately, there was no time for further research yesterday, but because there is no impact to the users, it was decided to leave the switch in place and pick up this issue the next day. Please follow up."

Your task is to diagnose the issue and restore connectivity between switch ASW1 and server SRV1. After resolving the problem, make a backup of the configuration to server SRV1.

Trouble Ticket D: Review and Discuss

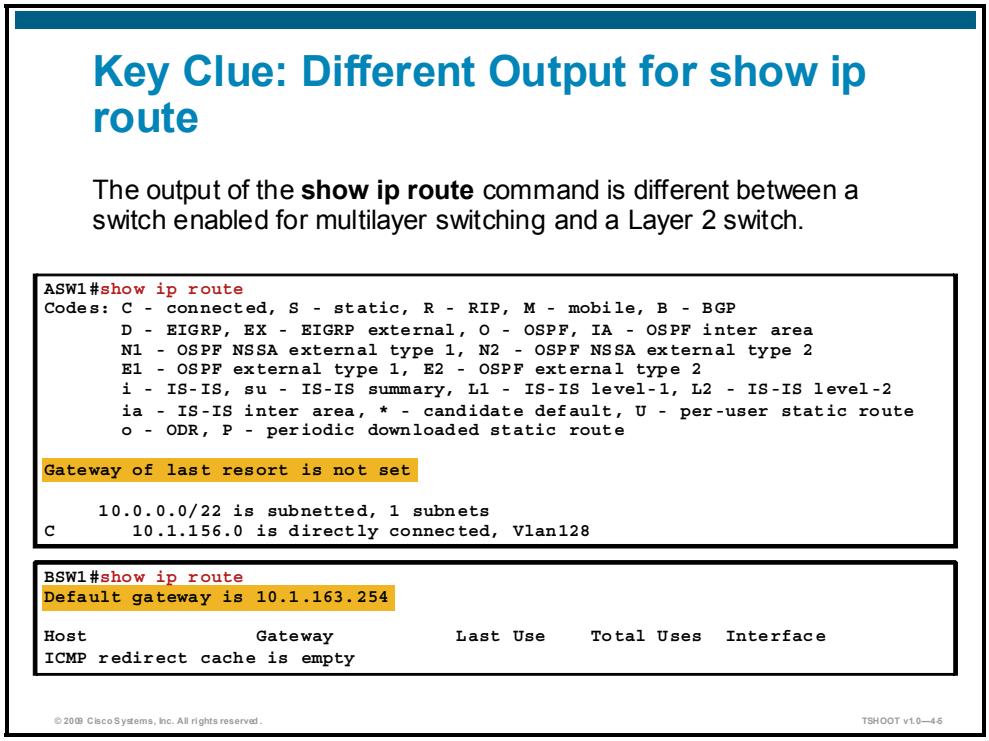
How did your team approach Trouble Ticket D?

- What method did you use to solve this ticket?
- Which tools did you use?
- Which problems did you find?
- Did you solve these problems and if so, how?
- Did you accomplish the goal of restoring management functions to switch ASW1?
- How did you verify that you accomplished your goal?

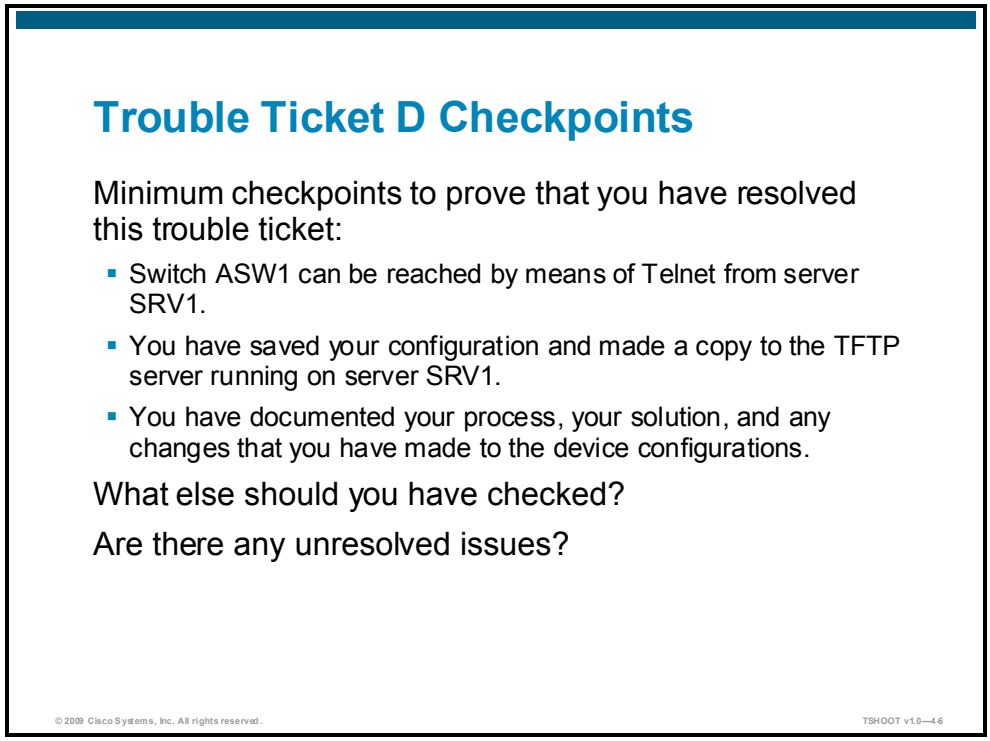
The main problem in Trouble Ticket D is that IP routing has been enabled on switch ASW1, causing it to act as a multilayer switch instead of a Layer 2 switch. As a result, it does not use the configured default gateway to route its own packets any longer, but relies on its routing table instead. Because there is no configured default route, it can no longer access anything outside its own subnet.

A troubleshooting log is listed here to provide you with an example of how this problem could have been resolved.

Trouble Ticket	Actions and results
Ticket D	ASW1: Verified problem. Cannot ping to IP address of SRV1 (10.1.152.1).
	ASW1: Verified SVI for management VLAN 128: IP address and mask are correct. Interface is up.
	ASW1: Checked **show ip route**: Gateway of last resort is not listed.
	ASW1: Verified default gateway configuration: Default gateway is present in the configuration and the IP address is correct.
	ASW1: Tried to ping the default gateway: ping is successful.
	BSW1: Spot differences between ASW1 and BSW1: Verified SVI, IP address, mask and default gateway configuration. Verified **show ip route**: Output is entirely different on BSW1. Default gateway is listed, but no directly connected routes or keywords for all routing protocols.
	Researched **ip default-gateway** command: according to documentation the default gateway is only used when IP routing is disabled.
	BSW1 & ASW1: Compared configurations. ASW1 lists **ip routing**, BSW does not.
	ASW1: Configuration changed: Disabled IP routing. Both prechange and postchange configurations were saved in flash on the switch. Communicated to team members.
	ASW1: Ping to SRV1 is successful.
	ASW1: Copied configuration to TFTP on SRV1.
	SRV1: Verified that using Telnet from SRV1 to ASW1 is working again.
	Problem fixed. Documented changes. Closed ticket.



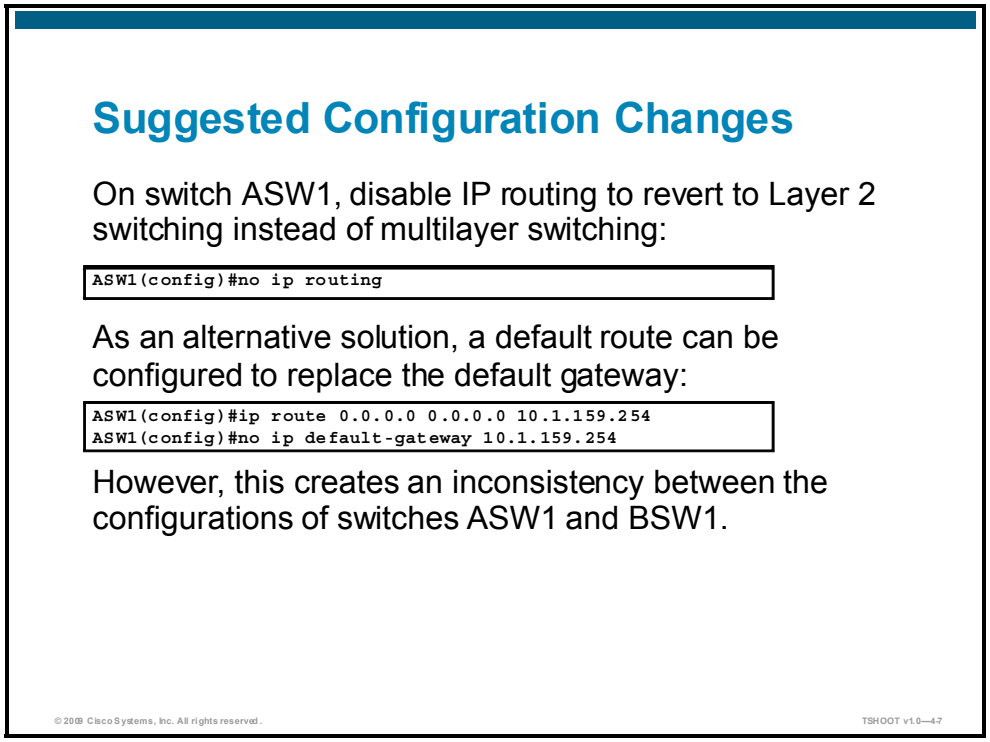
The primary clue in this troubleshooting flow is that the **show ip route** command on switch ASW1 does not reflect that a default gateway has been configured. If you compare it to the output on switch BSW1, which is deployed as a Layer 2 switch, you can see that the default gateway is listed, but all the information about routing protocols and also the directly connected subnet is not listed on switch BSW1. By researching the **ip default-gateway** command in the command reference on http://www.cisco.com, you can find that the **ip default-gateway** command is only used when IP routing is disabled. In other words, it is not used when multilayer switching is enabled on a switch. According to the design, switch ASW1 is not acting as a multilayer switch and therefore IP routing should be disabled.



The figure shows a list of minimum requirements that you should meet to consider this trouble ticket to be resolved.

Suggested Solution

Correcting the switch configuration as follows can solve the problem introduced in Trouble Ticket D.



The commands shown in the figure will fix the problems that were introduced in this trouble ticket and will return the configuration of switch ASW1 to its original baseline. Other solutions, such as the second solution shown in the figure, might also be valid. If you have a different solution, bring it to the attention of the group and discuss the possible advantages and disadvantages of your solution.

The preferred solution for this ticket is to disable IP routing on switch ASW1. Technically, the stated objectives can also be accomplished by configuring a default route on switch ASW1. However, this creates an inconsistency between the configurations of switches ASW1 and BSW1, which both serve as Layer 2 switches. If you decide to enable IP routing and configure a default route on a switch that is capable of multilayer switching, the change should be documented and the templates that you use for this type of switch need to be updated. Afterwards, all switches in a similar role should be updated to reflect the new baseline configuration.

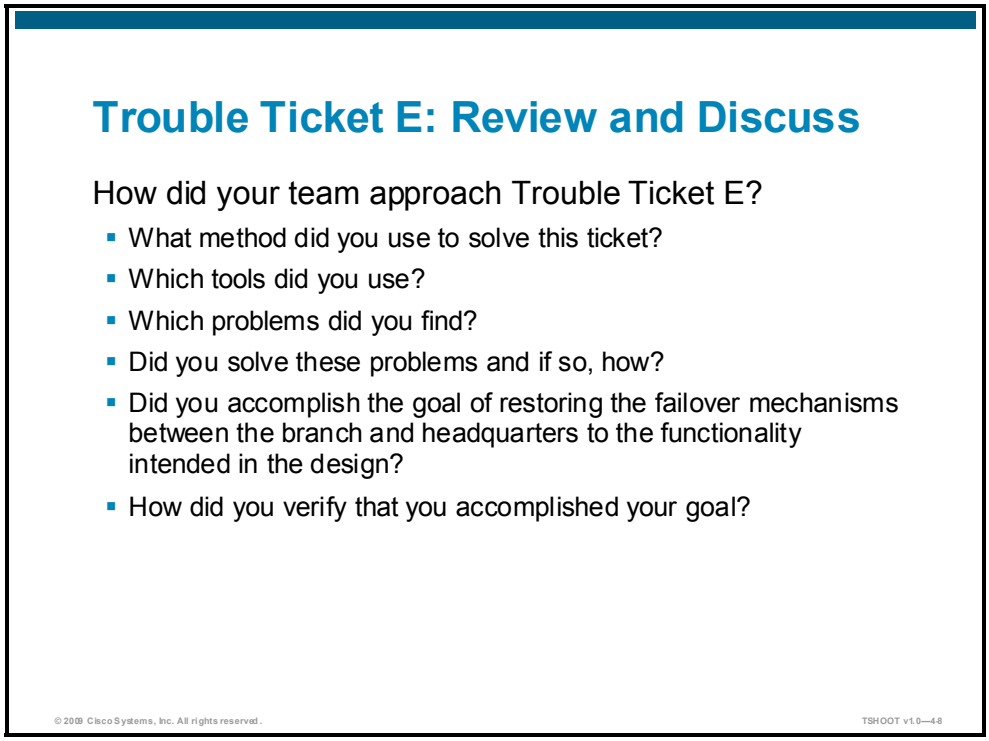
Trouble Ticket E: Failover Not Functioning as Expected

The text introducing this trouble ticket was the following:

"During last Friday's maintenance window, a series of failover tests between headquarters and the branch offices was executed. As a result of these tests, it was discovered that, during a reboot of router BRO1, connectivity between clients in the VLAN B1S1-OFFICE and hosts in the headquarters LANs is lost. After router BRO1 comes back online, the clients regain connectivity. In addition, management connectivity between server SRV1 and switch BSW1 on VLAN 128 is also lost during the failover. This is not the expected behavior, because the network is fully redundant and both a routing protocol (Enhanced Interior Gateway Routing Protocol [EIGRP]) and FHRPs (HSRP in the headquarters and GLBP in the branch) have been configured to ensure correct failover during outages.

Most of the users in the branch office are out of the office to attend training, so although it is not an official maintenance window, you have been authorized to run necessary failover tests during office hours. However, the disruption to the remaining branch office users should be kept to a minimum."

Your task is to diagnose this issue and restore the functionality of the failover mechanisms as intended in the design.



This ticket consists of two separate problems: The problem between the clients in the VLAN B1S1-OFFICE and headquarters, and the problem between the management VLAN of switch BSW1 and headquarters.

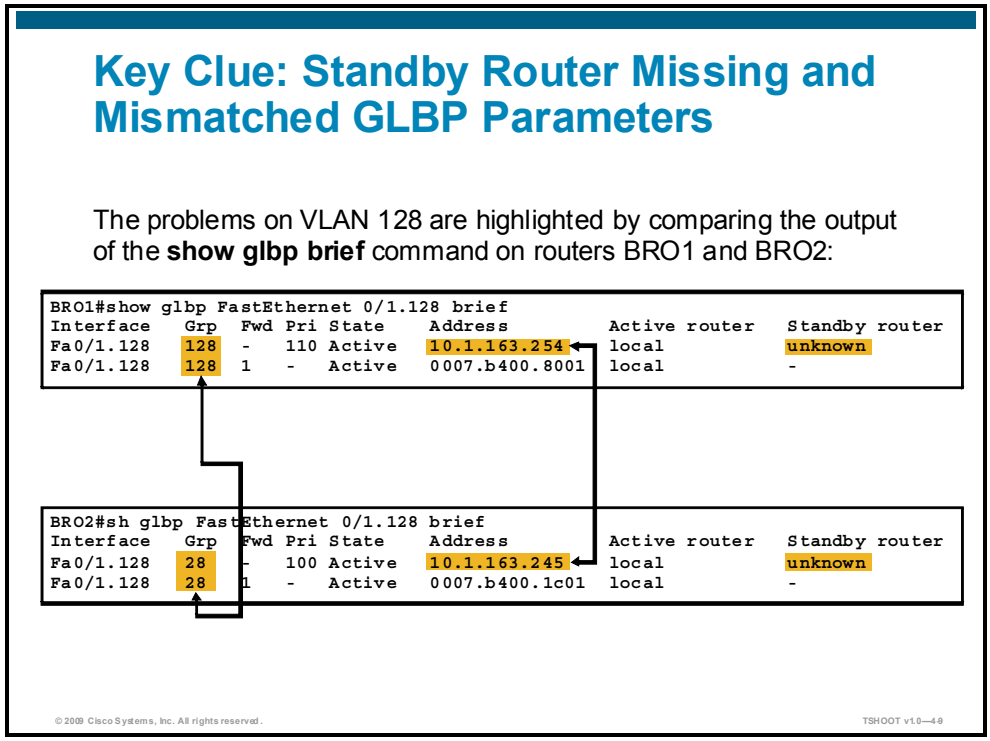
The first problem is not really a problem with FHRP, but is caused by the clients not using the correct default gateway. Instead of using the virtual GLBP IP address, they are using the IP address of router BRO1.

The second problem is a problem with GLBP, caused by the fact that both the GLBP group number and the GLBP IP address are misconfigured for VLAN 128 on router BRO2.

A troubleshooting log is listed here to provide you with an example of how these two problems could have been resolved. This log follows the flow outlined in the “Sample Troubleshooting Flows” section in the lab guide.

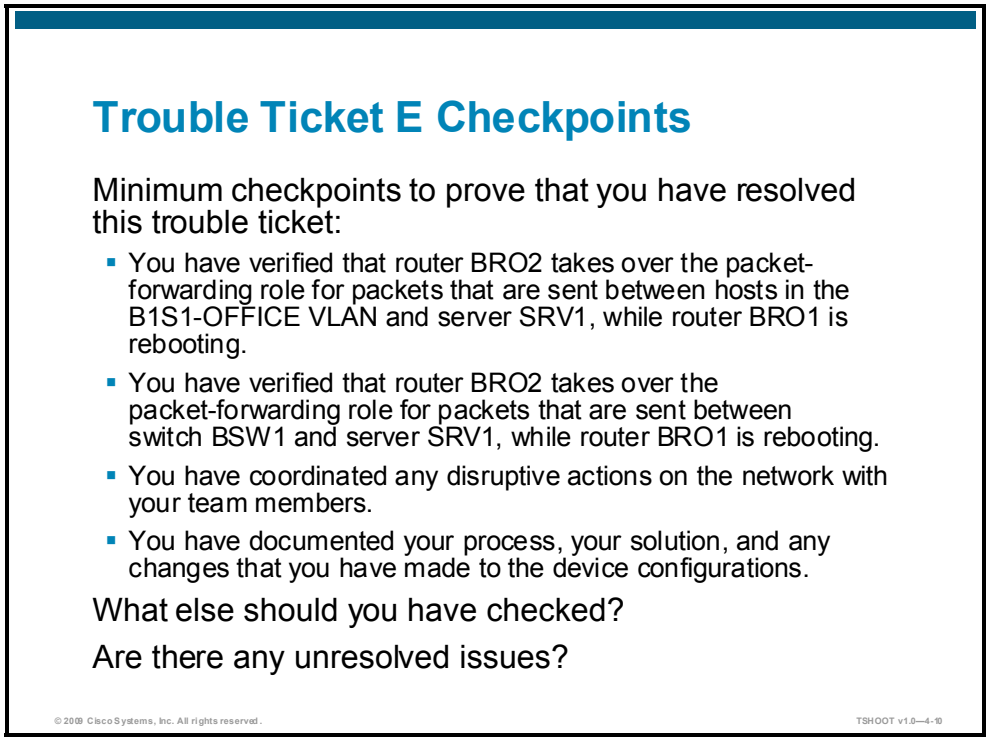
Trouble Ticket	Actions and results
Ticket E	Cannot verify problem without testing failover. Because testing will cause disruption to the users, we will attempt to troubleshoot without testing first. We will first verify the FHRP, then routing. If that does not deliver sufficient clues, we will execute failover tests to isolate the problem.
	Identified two subproblems. VLAN 17 (B1S1-OFFICE) and VLAN 128 (MGMT). We will address VLAN 17 first and then move on to VLAN 128.
	BRO1 and BRO2: Verified virtual and real IP addresses for GLBP group 17: BRO1 is 10.1.160.124, BRO2 is 10.1.160.125 and the GLBP virtual IP address is 10.1.160.126.
	CLT2: Default gateway is set to 10.1.160.124 (real IP address of BRO1).
	Check baseline: DHCP for VLAN B1S1-OFFICE is provided by BRO1.
	BRO1: Verify DHCP configuration for pool B1S1-OFFICE: Default router is listed as 10.160.1.124.
	Likely cause of problem for VLAN 17 discovered.
	BRO1: Configuration changed: Changed default router for DHCP pool B1S1-OFFICE to 10.1.160.126. Both prechange and postchange configurations were saved in flash on the switch. Communicated to team members.
	CLT2: Released and renewed IP address.
	CLT2: Ping to SRV1 succeeds. Confirmed that no connectivity was lost after the change.
	Postpone failover test to confirm problem hypothesis until after troubleshooting VLAN 128 (MGMT).
	BSW1: Default gateway is set to 10.1.163.254. Ping to the address succeeds and the MAC address registered in the ARP cache is 0007.b400.8001.
	BRO1: IP address 10.1.163.254 is equal to the configured IP address for the standby group 128 and the MAC address 0007.b400.8001 equals the GLBP MAC address of BRO1.
	BRO1: Noticed that no standby router is displayed for GLBP group 128.
	BRO1: Ping IP address of BRO2 for VLAN 128 (10.1.163.253): Ping succeeds.
	BRO2: Verify GLBP operation. This router also considers it self to be the active router, no standby router is listed. Noticed misconfigured GLBP group 28 instead of 128.
	BRO2: Configuration changed: Changed GLBP group to 128. Copied all other parameters. Both pre- and post-change configurations were saved in flash on the switch. Communicated to team members.
	BRO2: Received log message **%GLBP-4-DIFFVIP1: FastEthernet0/1.128 Grp 128 active routers virtual IP address 10.1.163.254 is different to the locally configured address 10.1.163.245**
	BRO2: Verify configured GLBP address for group 128. Turns out to be misconfigured.
	BRO2: Configuration changed: Changed GLBP IP address to 10.1.163.254. Copied all other parameters. Both prechange and postchange configurations were saved in flash on the switch. Communicated to team members.
	BSW1: Ping to SRV1 succeeds. Confirmed that no connectivity was lost after the change.
	Worked on ticket G before testing. See log for ticket G.
	CLT2: Started continuous pings to 10.1.160.124 (BRO1), 10.1.160.125 (BRO2), 10.1.160.126 (GLBP Virtual IP) and 10.1.152.1 (SRV1)
	BSW1: Started ping of 10000 packets to SRV1.
	BRO1: Reload router.

Trouble Ticket	Actions and results
	CLT2: Pings to BRO2 and GLBP address did not lose packets. Ping to SRV1 lost 1 packet. Ping to BRO1 lost all packets during reload. No packets were lost when BRO1 came back.
	BSW1: One packet was lost during reload of BRO1.
	BRO2: Reload router.
	CLT2: Pings to BRO1 and GLBP address did not lose packets. Ping to SRV1 lost 1 packet. Ping to BRO2 lost all packets during reload. No packets were lost when BRO2 came back.
	BSW1: One packet was lost during reload of BRO2.
	Test results confirm that failover functions as intended. Problem fixed. Documented test results and changes. Backed up configurations. Closed ticket.



For the problem in VLAN 17 B1S1-OFFICE there is no primary command worth listing. Verifying the default gateway that is used on the client reveals that the IP address of router BRO1 is used instead of the GLBP IP address for GLBP group 17.

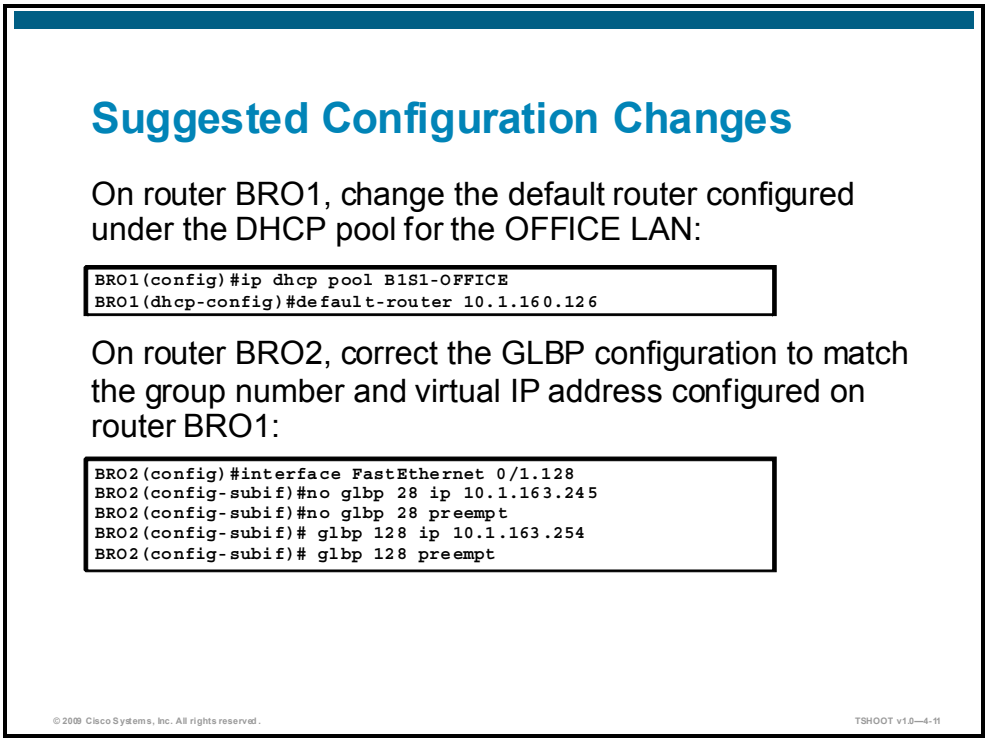
The cause of the problem in VLAN 128 is uncovered when you compare the output of the **show glbp brief** command for interface FastEthernet 0/1.128 between routers BRO1 and BRO2. This reveals a mismatch between both the configured GLBP group number and the virtual IP address. Because both routers are configured to operate in a different group, they will each list themselves as the active router for their respective groups, and they do not list the other router as a standby router.



The figure shows a list of minimum requirements that you should meet to consider this trouble ticket to be resolved.

Suggested Solution

Correcting the configurations of routers BRO1 and BRO2 as follows can solve the problems introduced in Trouble Ticket E.



The commands shown in the figure will fix the problems that were introduced in this trouble ticket. Other solutions might also be valid. If you have a different solution, bring it to the attention of the group and discuss the possible advantages and disadvantages of your solution.

As an alternative to making a change to the DHCP pool, you could configure client CLT2 with a fixed IP address, at least for initial testing. However, this is only a workaround, not a solution. It addresses the symptoms, but not the underlying problem. Therefore, the only real solution is to change the DHCP configuration.

For the second problem, you could also configure router BRO1 to match router BRO2 instead of the opposite. However, that creates an inconsistency with the implementation conventions that have been used in this network. In this network, the virtual IP address is always equal to the highest IP address in the subnet and the FHRP group number always matches the VLAN number. So technically the solution would work, but from a network maintenance and troubleshooting standpoint, it is bad practice because it goes against the established conventions for this network.

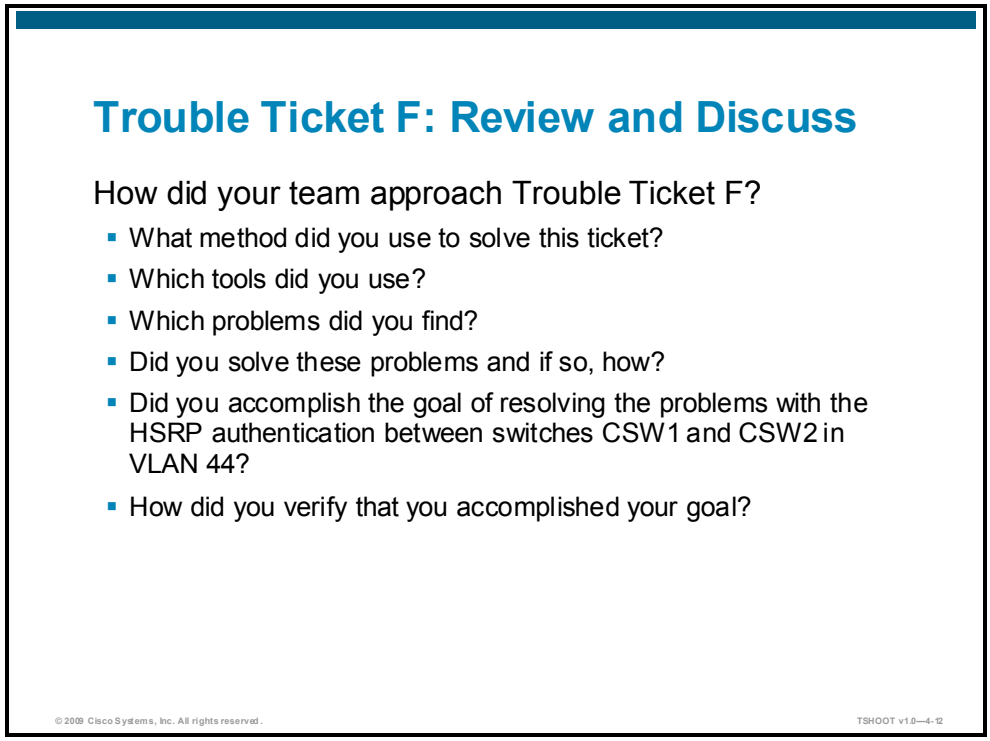
Trouble Ticket F: Verify HSRP Authentication

The text introducing this trouble ticket was the following:

"Several weeks ago, an external company performed a security audit on the network. One of the exposed attack vectors—or weaknesses— was that a denial of service (DoS) attack could be launched against HSRP. The recommended solution was to use Message Digest 5 (MD5)-based authentication between the HSRP routers. One of your colleagues has been too busy to implement this in a test-VLAN in the headquarters' LAN (VLAN 44) before rolling it out on all LANs.

Yesterday, just before this colleague left for a two-week vacation, she asked you to see if somebody else could finalize the tests and to guarantee that it can be rolled out as soon as she returns."

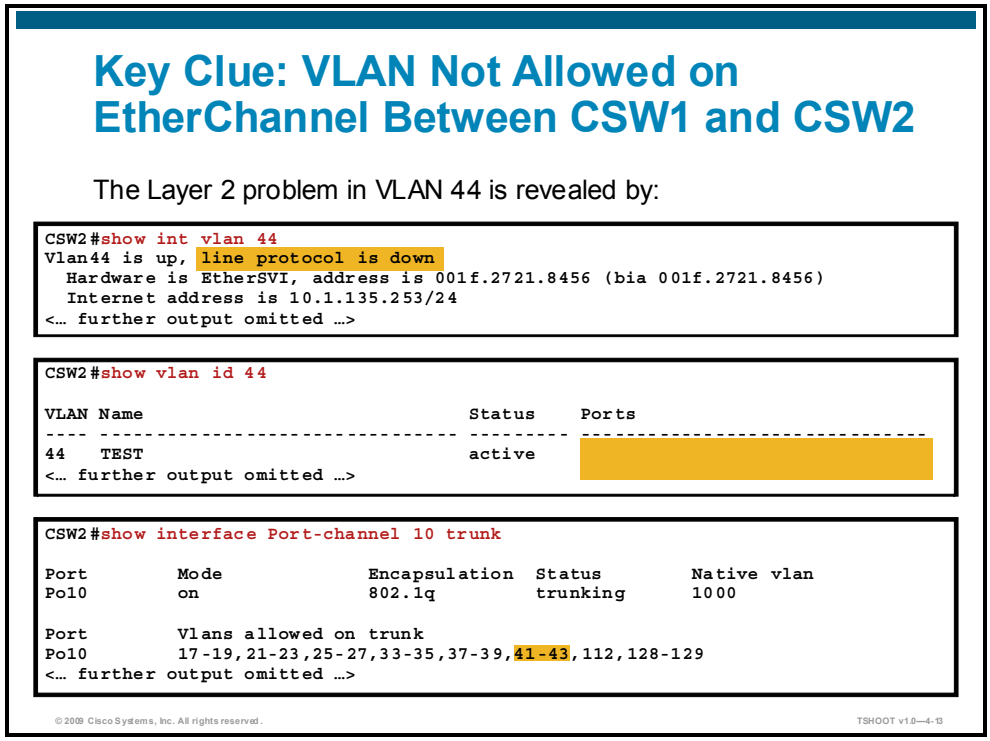
Your task is to review and verify the implementation of HSRP authentication in VLAN 44 and fix any issues that might remain.



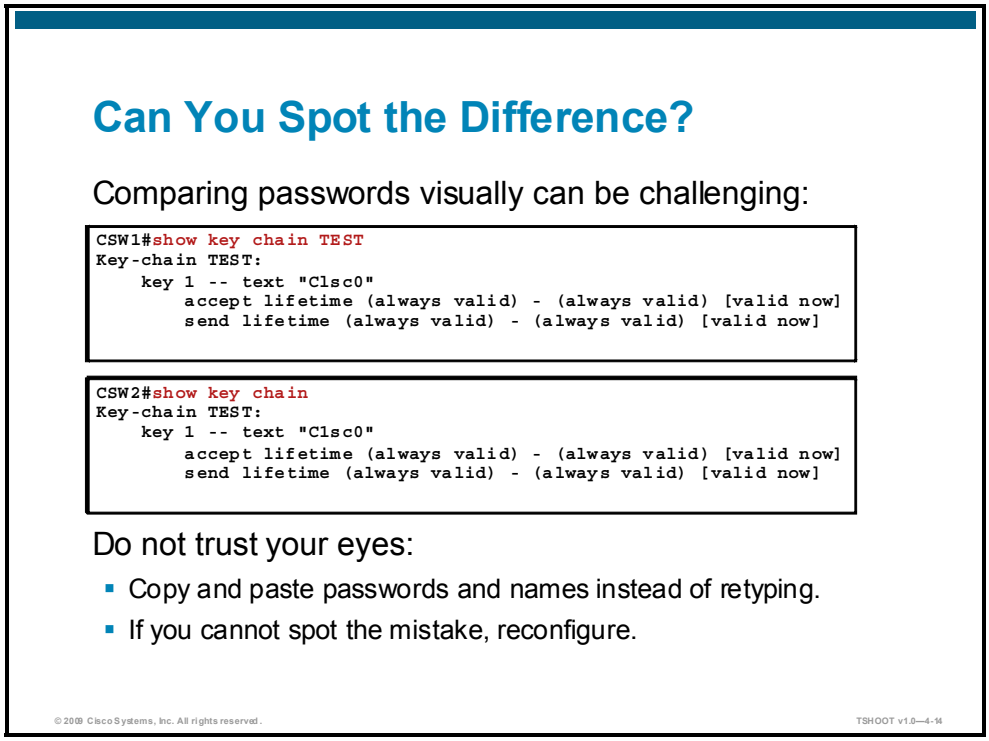
Trouble Ticket F actually consists of two problems. First, there is a Layer 2 problem in the newly created VLAN 44. The VLAN is not added on the trunk between switches CSW1 and CSW2, and it is not passed through via switch ASW1. Secondly, there is a problem with the passwords. Although they look very similar, there is a slight mismatch. One password uses the number "1" to represent the "i" in the word "Cisco," and the other password uses the lowercase letter "l" to represent the "i." This is hard to spot visually, so it calls for careful analysis and interpretation of the output of the various HSRP **show** commands and log messages.

A troubleshooting log is listed here to provide you with an example of how this problem could have been resolved.

Trouble Ticket	Actions and results
Ticket F	Scope of the problem: Review and verify the implementation and operation of HSRP authentication in the TEST VLAN 44. Client side failover testing is not required. The objective is to create a proof of concept of HSRP authentication, to be used in the creation of templates for the future rollout.
	CSW1: Verify operational state of the standby group in VLAN 44: CSW1 claims to be the active router for group 44 and IP address 10.1.135.254.
	CSW2: Verify operational state of the standby group in VLAN 44: CSW2 is in state “Init.”
	CSW2: Interface VLAN 44 is in state “up, line protocol is down.” Troubleshoot SVI status.
	CSW2: VLAN 44 exists but has no associated ports.
	Reference baseline: EtherChannel Port-channel 10 should carry all VLANs between CSW1 and CSW2.
	CSW2: Verify trunk status for Port-channel 10: VLAN 44 is not allowed on the trunk.
	CSW2: Configuration changed: Added VLAN 44 to the list of allowed VLANs on Port-channel 10. Both prechange and post change configurations were saved in flash on the switch. Communicated this information to team members.
	CSW2: Started receiving: **%HSRP-4-BADAUTH: Bad authentication from 10.1.135.252, group 44, remote state Active** messages
	CSW2: SVI for VLAN 44 is up. Layer 3 connectivity is confirmed through a succesful ping to 10.1.135.252.
	CSW1 & CSW2: Both routers claim to be active for group 44 and IP address 10.1.135.254.
	Log message indicates authentication failure.
	CSW1 and CSW2: Both switches are configured to use a key-chain named “TEST”, a key-id of 1, and a key-string of “C1sc0”.
	CSW1 and CSW2: Enabled **debug standby error**. Shows “MD5 auth failed” on both switches. Decision: Remove authentication and rebuild.
	CSW1 and CSW2: Configuration changed: Removed HSRP authentication for VLAN 44 and removed key-chain TEST on CSW1 and CSW2. Both prechange and postchange configurations were saved in flash on the switch. Communicated to team members.
	CSW1 and CSW2: Verify standby state: CSW1 is now active and CSW2 is standby as expected.
	CSW1 and CSW2: Configuration changed: Created new key-chain named TEST on CSW1, using key 1 and key-string “C1sc0.” Copied the configuration from the running configuration of CSW1 and pasted to CSW2. Enabled HSRP authentication for VLAN 44 on CSW1 and CSW2. Both prechange and postchange configurations were saved in flash on the switch. Communicated to team members.
	CSW1 and CSW2: Verify standby state: CSW1 is active and CSW2 is standby as expected.
	CSW1 and CSW2: Problem solved. Saved configurations for reference in flash and on TFTP server. Documented changes and conclusions and communicated to reporting colleague.
	CSW1 and CSW2: Reviewed changes made by team members. Conclusion: No changes made. Rolled back configuration to last known good configuration before test of HSRP authentication. Verified and updated documentation. Communicated to team members. Closed ticket.

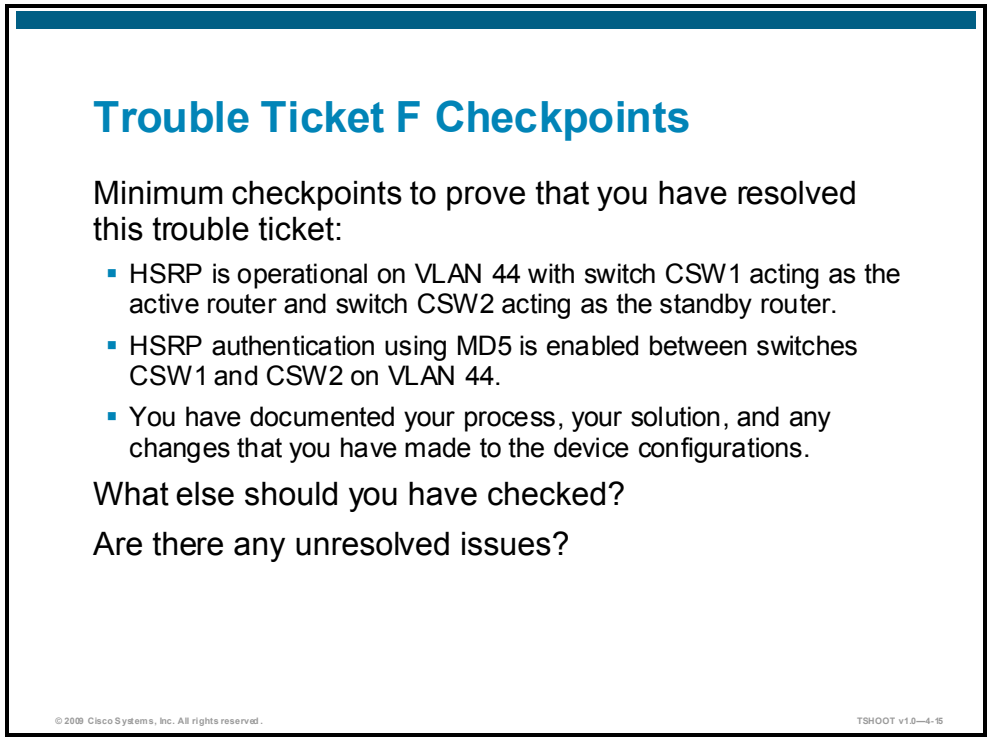


The primary commands that reveal the Layer 2 problem in VLAN 44 are listed in the figure. As soon as you see that the SVI for VLAN 44 is down, you verify the status of the underlying VLAN by use of the **show vlan id *vlan-id*** command. This reveals that no ports, including trunks, are assigned to the VLAN. Verification of the allowed VLAN list for the EtherChannel between switches CSW1 and CSW2 shows that VLAN 44 is missing on that list. After adding VLAN 44 to the allowed VLAN list on Port-channel 10, the SVI comes up and communication between switches CSW1 and CSW2 on VLAN 44 is restored.



For the HSRP authentication problem there is no single command that clearly shows the problem. Theoretically, you could spot the issue by comparing passwords, but in this case, that is particularly hard because the strings look so similar. Even when simpler key-strings are used, it is very easy to overlook a typing error.

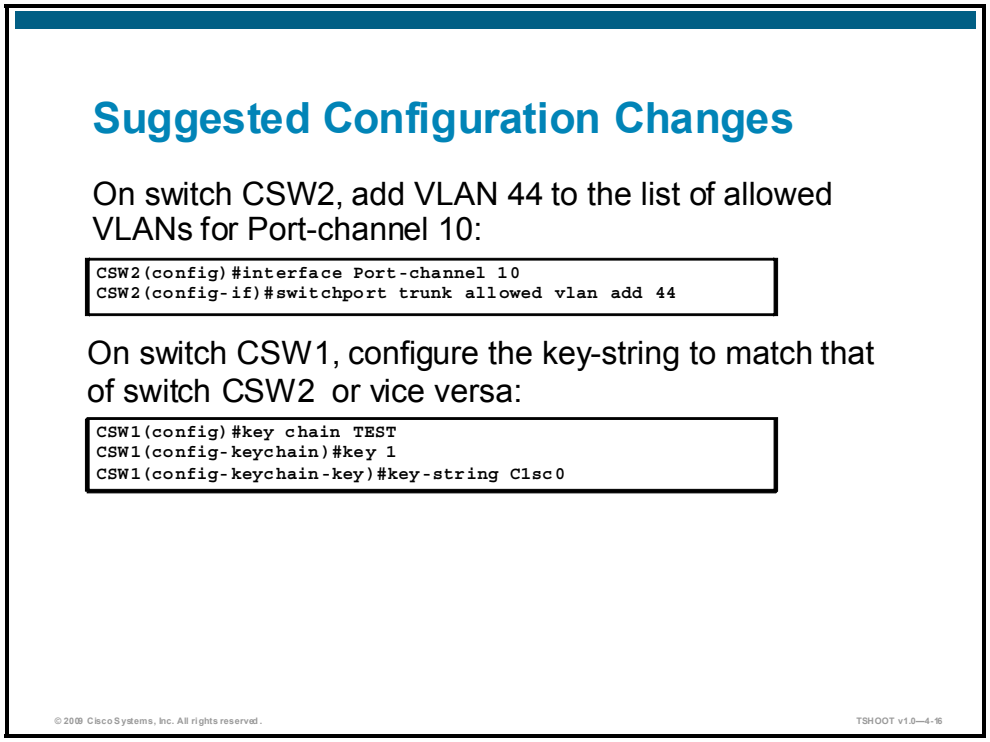
A good method to avoid this problem entirely (if you have the opportunity to configure both sides of an authenticated session) is to simply reconfigure the authentication part and copy and paste the key-string or password from the configuration on one side to the other.



The figure shows a list of minimum requirements that you should meet to consider this trouble ticket to be resolved.

Suggested Solution

Correcting the configurations of switches CSW1 and CSW2 as follows can solve the problems introduced in Trouble Ticket F.



The commands shown in the figure will fix the problems that were introduced in this trouble ticket. Other solutions might also be valid. If you have a different solution, bring it to the attention of the group and discuss the possible advantages and disadvantages of your solution.

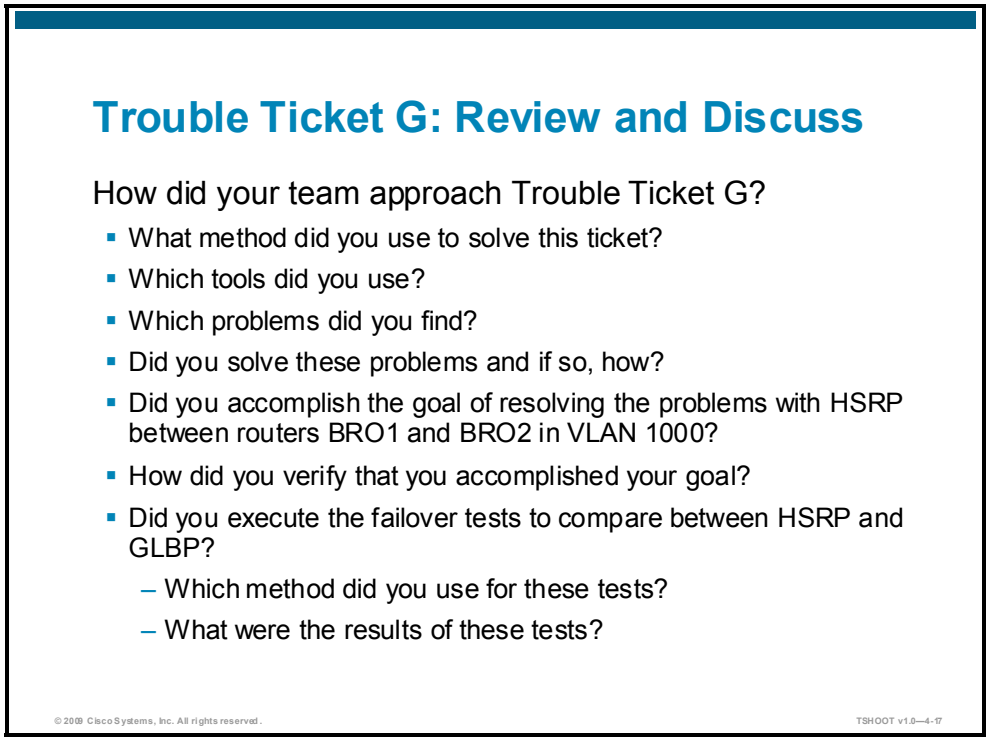
One of the additional steps that you might want to take after implementing and verifying HSRP authentication is to store the configurations and roll back the configurations in order to restore switches CSW1 and CSW2 to the standard baseline without the additional test VLAN. Even if proof-of-concept testing is done in the operational environment (and it could be argued that this should never be done), you must make sure that the implemented test scenarios do not become part of the operational environment.

Trouble Ticket G: HSRP and GLBP Comparison

The text introducing this trouble ticket was the following:

"The failover tests that were executed last Friday (as mentioned in Trouble Ticket E) have caused another scenario to be implemented and tested. One of the network engineers who works at Branch Office 1 has always said that it would be better to use HSRP instead of GLBP. The fact that the failover tests did not work out as expected has now caused this engineer to push for a good comparative test of the failover behavior of the two protocols and revert to HSRP, unless it can be proven that GLBP functions at least as well as HSRP where failover is concerned. You receive a phone call from the engineer in which he asks you to look at the configuration because it is frustrating him. Somehow, he cannot get HSRP to work in his test-VLAN (VLAN 1000), and now that he has pushed for this test, he needs to make it work. You offer to have a look and help the engineer run the tests."

Your task is to diagnose and resolve the problems with HSRP in the newly configured VLAN 1000 on routers BRO1 and BRO2, and to execute failover tests to compare the behavior of GLBP and HSRP. To minimize the disruption on the network, these tests should be coordinated with the rest of the team, specifically with the team members that are working on Trouble Ticket D.

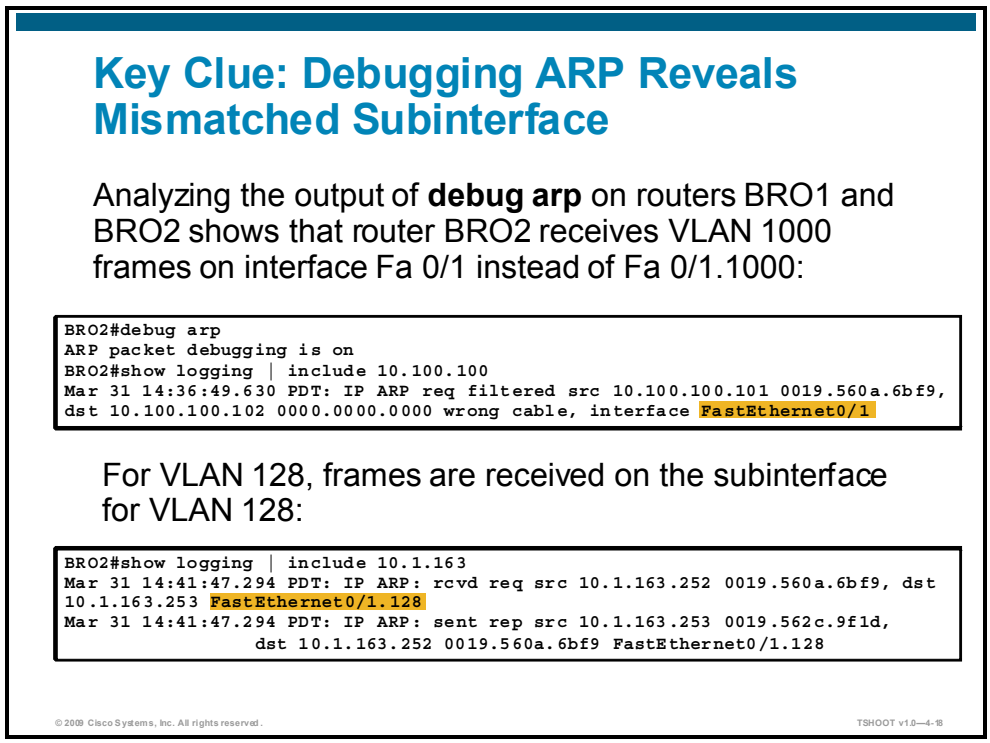


The main problem in this ticket was not related to HSRP directly, but was caused by a Layer 2 problem in the underlying VLAN. VLAN 1000 is configured as the native VLAN for the trunks to routers BRO1 and BRO2 on switch BSW1. This configuration is not matched on routers BRO1 and BRO2, causing switch BSW1 to send untagged frames routers BRO1 and BRO2. Routers BRO1 and BRO2 then discard these frames, because they associate them to the main interface instead of the subinterface for VLAN 1000.

A troubleshooting log is listed here to provide you with an example of how this problem could have been resolved.

Trouble Ticket	Actions and results
Ticket G	Scope of the problem: Review and verify the implementation and operation of HSRP in the TEST VLAN 1000. After that, execute failover tests to compare HSRP and GLBP. Coordinate with ticket E.
	Ticket B has been resolved to the point where tests can be executed. See log for ticket E. Will now start troubleshooting ticket G.
	BRO1: Verify problem: Operational status is that BRO1 considers itself to be the active router for standby group 100 using virtual IP address 10.100.100.100. Configured interface IP address is 10.100.100.101.
	BRO2: Verify problem: Operational status is that BRO2 also considers itself to be the active router for standby group 100 using virtual IP address 10.100.100.100. Configured interface IP address is 10.100.100.102.
	BRO1: Verify Layer 3 connectivity between BRO1 and BRO2: Ping to 10.100.100.102 (BRO2 in VLAN 1000) fails. Start troubleshooting Layer 2. ARP entry for 10.100.100.102 is displayed as “incomplete.”
	BSW1: The virtual MAC address for HSRP group 100 (0000.0c07.ac64) is learned alternately on interface Fa 0/3 and Fa 0/4 in VLAN 1000. The physical MAC address of BRO1 is learned on interface Fa 0/3 in VLAN 1000 and the physical MAC address of BRO2 is learned on interface Fa 0/4 in VLAN 1000.
	Conclusion: VLAN 1000 frames can be received from both BRO1 and BRO2 by BSW1. Problem seems to be either with transmission of frames to BRO1 and BRO2 or with the reception of frames on BRO1 and BRO2.
	BRO1 and BRO2: No input errors are recorded on interface Fa 0/1.
	BRO1 and BRO2: **show vlans 1000** does not show input packets on BRO1 and BRO2.
	BRO1 and BRO2: Enabled debug arp.
	BRO1: Ping 10.100.100.102 (BRO2 in VLAN 1000).
	BRO1: Debug displays **IP ARP: sent req src 10.100.100.101 0019.560a.6bf9, dst 10.100.100.102 0000.0000.0000 FastEthernet0/1.1000**
	BRO2: Debug displays “IP ARP req filtered src 10.100.100.101 0019.560a.6bf9, dst 10.100.100.102 0000.0000.0000 wrong cable, interface FastEthernet0/1”
	Compare: Frame leaves on subinterface Fa 0/1.1000 on BRO1, but is received on Fa 0/1 on BRO2.
	BRO1: Clear ARP entry for IP address 10.1.163.253 then ping to 10.1.163.253 (BRO2 in VLAN 128)
	BRO1: Debug displays “IP ARP: sent req src 10.1.163.252 0019.560a.6bf9, dst 10.1.163.253 0019.562c.9f1d FastEthernet0/1.128 and IP ARP: rcvd rep src 10.1.163.253 0019.562c.9f1d, dst 10.1.163.252 FastEthernet0/1.128”
	BRO2: Debug displays “IP ARP: rcvd req src 10.1.163.252 0019.560a.6bf9, dst 10.1.163.253 FastEthernet0/1.128 and IP ARP: sent rep src 10.1.163.253 0019.562c.9f1d, dst 10.1.163.252 0019.560a.6bf9 FastEthernet0/1.128”
	Compare: For VLAN 128 frames leave and enter on Fa0/1.128. Conclusion: Frames from VLAN 1000 are received on Fa 0/1 instead of Fa 0/1.1000.
	BRO1 and BRO2: Turned off all debugging.
	BSW1: Verify VLAN 1000. VLAN 1000 is named NATIVE and is configured as the native VLAN for the trunks to BRO1 and BRO2.
	Research native VLAN. Discovered that native VLAN can be added on the encapsulation dot1q command.

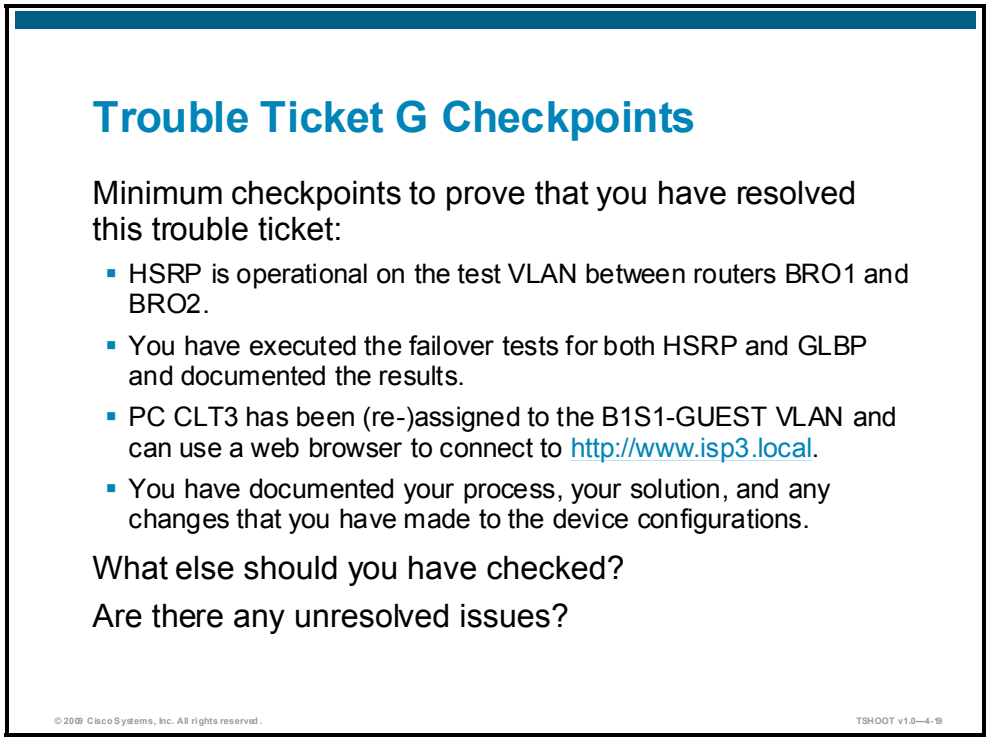
Trouble Ticket	Actions and results
	BRO1 and BRO2: Configuration changed: Changed to encapsulation dot1q 1000 native on sub-interface Fa 0/1.1000. Both prechange and postchange configurations were saved in flash on the switch. Communicated this information to team members.
	BRO1: Ping to 10.100.100.102 succeeds.
	BRO1: Verified HSRP state. BRO1 is now listed as standby router and BRO2 as active router.
	CLT3: Configuration changed: IP address changed to 10.100.100.150/24. Default gateway set to 10.100.100.100. Communicated to team members. Note: Roll back after testing.
	BSW1: Configuration changed: Access VLAN of port Fa 0/8 changed to VLAN 1000. Communicated to team members. Note: Roll back after testing.
	CLT3: Start continuous ping to 10.100.100.100.
	Execute failover test. Coordinated with other tests.
	BRO2: Failover behavior seems to be equal to GLBP. No packets lost.
	CLT3 and BSW1: Restore CLT3 to GUEST VLAN.
	Problem solved. Documented changes and test results and communicated to reporting colleague.



Until you spot that VLAN 1000 is the native VLAN for the trunks, this ticket is very hard to troubleshoot. MAC addresses of both routers are learned on the switch, but still Layer 3 connectivity is broken. Other VLANs share the same trunk but do not have a problem. Eventually, you should realize that there must be something special about VLAN 1000 and spot that it is the native VLAN for the trunks.

One of the commands referenced in the troubleshooting log is **debug arp**, which reveals the problem (although you need to pay close attention to the output to spot it).

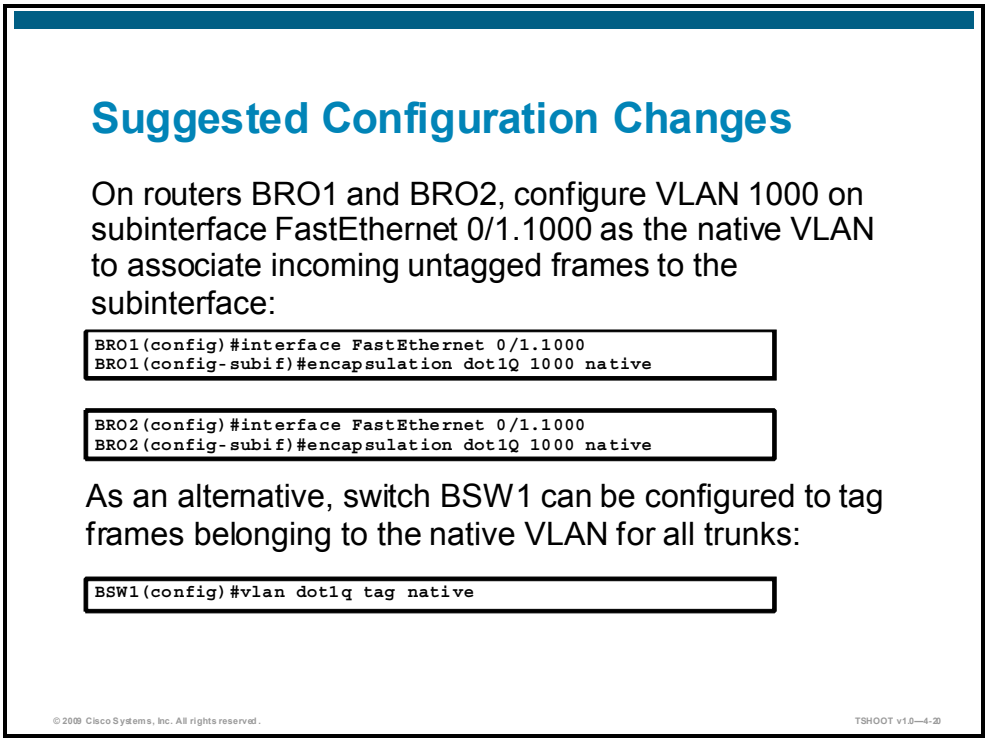
By comparing the output of the ARP requests received by router BRO2 for VLAN 128 and VLAN 1000, you can see that frames for VLAN 1000 are not associated to the subinterface, but to the main interface. For VLAN 128, frames are received on the correct subinterface. This points to a trunk problem, and by carefully examining the trunk configuration on switch BSW1 you discover that VLAN 1000 is the native VLAN and that this crucial difference between VLAN 1000 and other VLANs is the cause of the problem.



The figure shows a list of minimum requirements that should be met to consider this trouble ticket to be resolved.

Suggested Solution

Correcting the configurations of routers BRO1 and BRO2 as follows can solve the problems introduced in Trouble Ticket G.



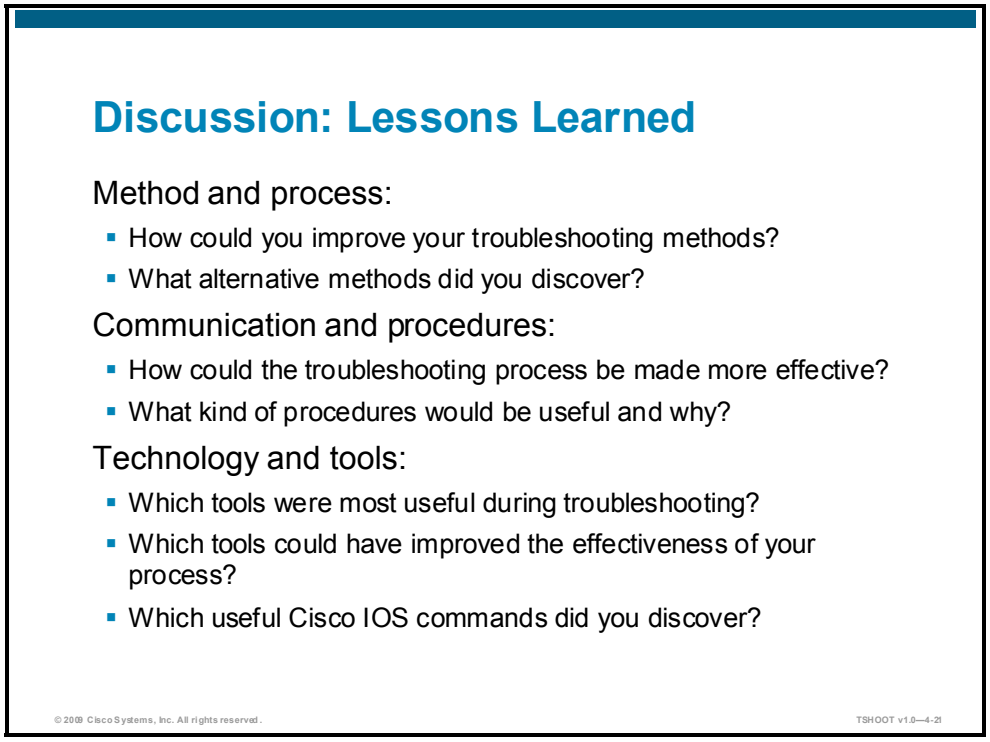
The commands shown in the figure will fix the problems that were introduced in this trouble ticket. Other solutions might also be valid. If you have a different solution, bring it to the attention of the group and discuss the possible advantages and disadvantages of your solution.

Both solutions technically solve the problem. Because the configuration is intended only as a temporary setup to execute failover tests, the choice between the two is not very important. The configurations will be removed after the test and will not be integrated into the regular configurations. If this were to be integrated in the design of the network, consider the following points:

- Security best practices recommend that the native VLAN of a trunk should not be used to carry user traffic, to prevent VLAN-hopping attacks. Electing to use VLAN 1000 as a data-carrying VLAN goes against current best practices.
- If you decide to implement the second solution, configuring switch BSW1 to tag all native VLAN traffic, you should consider configuring this command on all switches to keep configurations consistent between switches.

Consolidation

This topic describes the primary lessons that could be learned from the lab exercise.

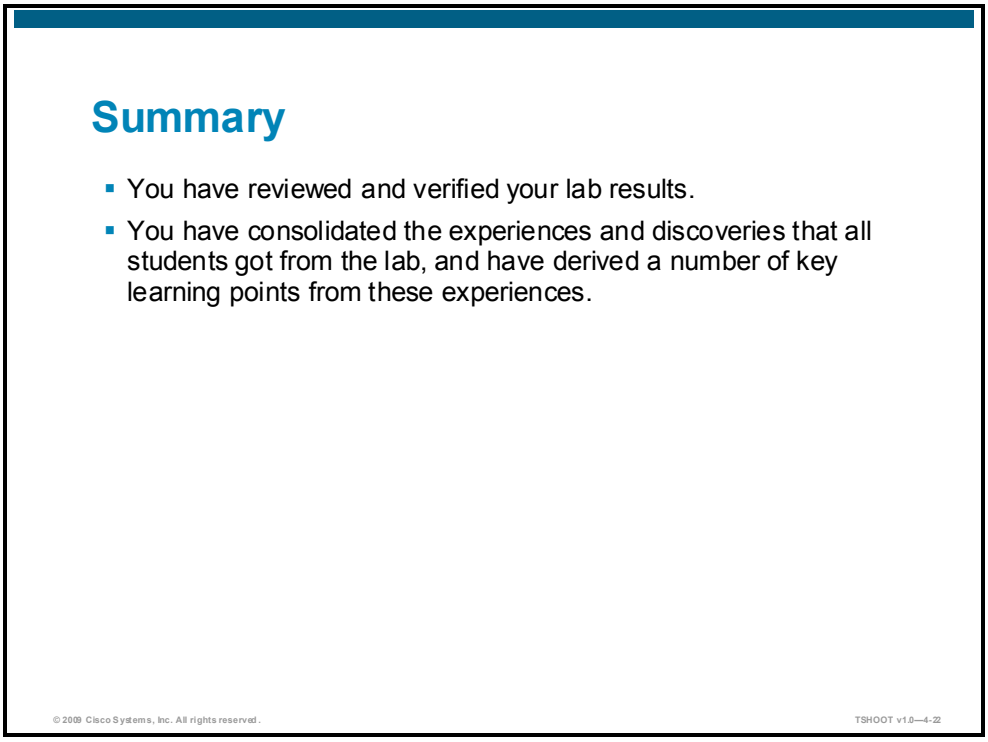


Think about all the things that you learned during the lab itself and during the debrief discussions. There is room to write down primary learning points in the Lab Debrief Notes section of the lab guide.

In addition to thinking of the methods, processes, and tools as they were used in the lab, reflect on how these would apply to your own organization.

Summary

This topic summarizes the key points that were discussed in this lesson.



Lesson 7

Troubleshooting Performance Problems on Switches

Overview

Performance problems are notoriously hard to troubleshoot, because they are defined in terms of expectations. When users report that an application is not performing as they expect, this can be a real problem at the business level. Employees are not as productive as they could be or they might not be able to service their customers correctly. However, if this is also a technical performance problem it still needs to be assessed. Does the network cause the problem or does the client, a server, or the application cause the problem? Even if you determine that the application performance is degraded because the network performance does not meet the expectations, you still need to establish whether this expectation was reasonable and the switches are underperforming, or whether the expectations are unrealistic, given the design and the traffic patterns.

It is important to be able to determine if switches are not performing as expected. It is also important to identify and possibly resolve the cause of the observed difference between the expected performance and the actual performance.

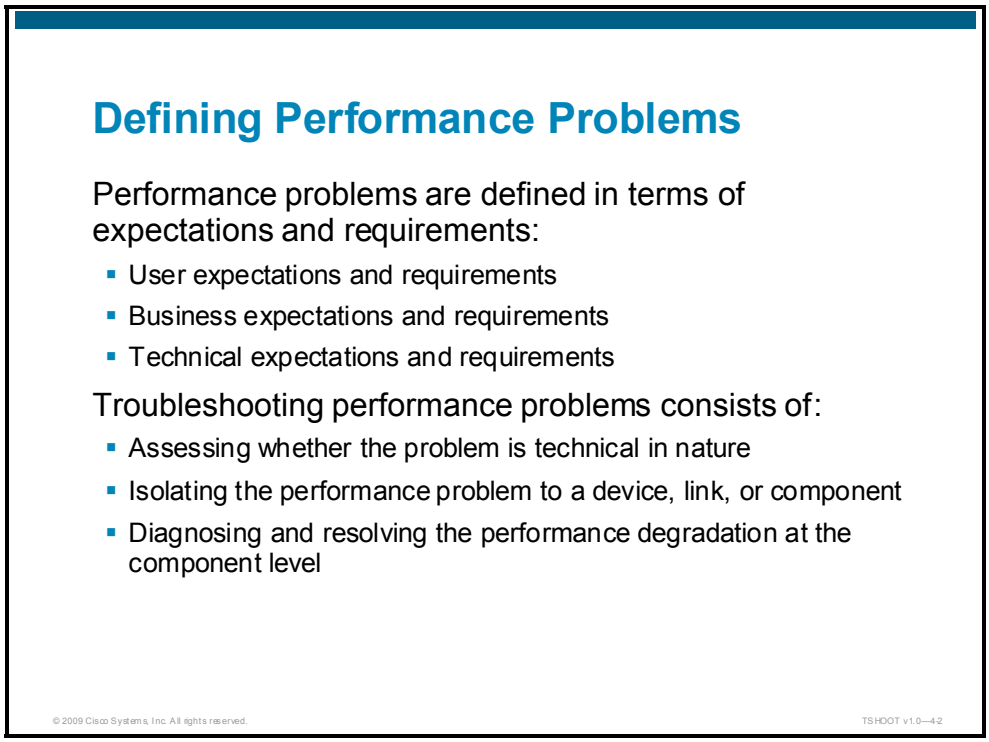
Objectives

Upon completing this lesson, you will be able to diagnose performance problems on Cisco Catalyst LAN switches. This ability includes being able to meet these objectives:

- Use Cisco IOS commands to diagnose physical and data link layer problems on switch ports
- Use Cisco IOS commands to analyze TCAM utilization on switches to determine the underlying cause of TCAM allocation failures
- Use Cisco IOS commands to determine the underlying cause of high CPU usage on a switch

Troubleshooting Physical and Data Link Layer Problems

This topic describes how to diagnose physical and data link layer problems that cause performance degradation to traffic entering or leaving a switch.



What exactly is a performance problem? In general, a network performance problem can be defined as a situation where the observed traffic handling of traffic on the network does not meet certain expected standards.

For this reason, performance problems are hard to troubleshoot, because they are defined in terms of "expected behavior," which is subjective to an extent. Is the problem caused by devices that are not performing according to predefined requirements or is the problem just a matter of mismatched perceptions and expectations? In other words, is this problem at the business level or at the technical level? Problems where something is simply not working are much clearer. All you need to ask yourself is whether the application or functionality is expected to work according to the requirements and design of the network. If the answer to that question is affirmative, and things are not working, this is clearly a technical problem.

The role of expectations in performance problems can be illustrated by the following example: Suppose there is a switch with 20 users, each connected to a 100-Mb/s port. A file server is connected to the same switch on a 1-Gb/s port. On average, users access the file server at different times, transferring files of various sizes. As long as not more than half of them are transferring files at the same time, they will experience transfer rates of up to the full 100 Mb/s that is available to each user.

Imagine that at one point in time all users need to transfer a file from the server at the exact same moment. Because the server only has a total bandwidth of 1 Gb/s, the average transfer rate of the users will be 50 Mb/s, while they are all transferring the file. Is this a performance problem or not? If users have come to expect transfer rates of 100 Mb/s, they might experience this as a performance problem, but from a technical standpoint, the network performs as expected.

On the other hand, if one of the users never gets transfer rates higher than 50 Mb/s, even if that user is the only person transferring files, then this is a performance problem from a technical standpoint. (Although in this case the user might not even experience this as a performance problem, because the user has not come to expect transfer rates higher than 50 Mb/s.)

In general, troubleshooting performance problems is a three-step process:

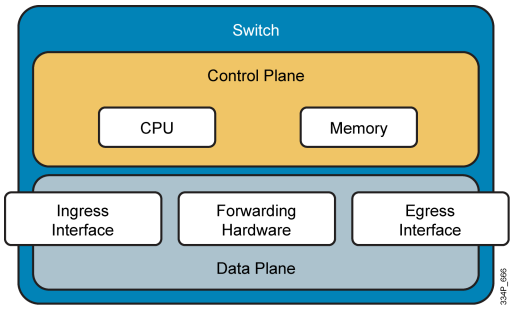
- Evaluate whether the behavior experienced by the user matches the technical expectations. If it turns out that the performance is technically within the boundaries of what can be expected of the network (as specified by the design and requirements), but the user still experiences it as a problem, then this is not a technical problem. The business requirements might need to be reevaluated. Eventually, this could result in a change of the technical requirements and the network design.
- If it is determined that the observed behavior is not within the boundaries of what should be expected from a technical standpoint, the problem needs to be isolated to a particular device, link, or component. A common approach to eliminate components from the chain of links and devices between the endpoints is comparing the behavior of different devices along different segments of the network path and spotting the differences or similarities in the behavior.
- After you have determined that a particular component is likely to be the limiting factor in the chain between the endpoints, you need to investigate that component for symptoms that confirm that this component is causing the performance problems. When this has been confirmed, the next step is to diagnose the underlying cause and, if possible, resolve the problem.

This third step is the focus for this lesson. How can you diagnose and resolve performance problems using Cisco IOS tools on a switch, when you have determined that a switch or a link connected to a switch could be the cause of performance degradation on the network?

Switch Performance

Switch performance is dependent on the performance of:

- Data plane:
  - Ingress interface
  - Forwarding hardware
  - Egress interface
- Control plane:
  - CPU
  - Memory



Although the specific details of the hardware architectures of the various Cisco Catalyst Switch families can be quite different, all switches essentially contain the following components:

- Interfaces to receive and transmit frames.
- Forwarding hardware that consists of two elements: hardware that implements the decision-making logic that is necessary to rewrite a frame and forward it to the correct interface, and a backplane to carry frames from the ingress interface to the egress interface.
- Control plane hardware to execute the processes that are part of the operating system.

Which of these components affect the performance of the switch?

Traffic flowing through the switch enters on an ingress interface, is forwarded by the forwarding hardware, and then leaves through the egress interface. So clearly, the performance of these components directly influences switch performance.

The control plane CPU and memory is not involved in switching traffic. Therefore, the control plane hardware does not have a direct effect on switch performance. However, the control plane is responsible for updating the information in the forwarding hardware, so it does have an indirect effect on the forwarding capability of the platform. If the control plane is consistently running at very high load, this could eventually start affecting the forwarding behavior of the platform.

In addition, the control plane hardware handles any traffic that cannot be handled by the forwarding hardware. Therefore, a high load on the control plane hardware could also be an indication that the forwarding hardware either has reached its maximum capacity or is not handling traffic as it should.

Checking for Interface Errors

Switches support specific command options for **the show interface** command to display packet and error counters:

```
ASW1#show interfaces FastEthernet 0/1 counters

Port            InOctets    InUcastPkts    InMcastPkts    InBcastPkts
Fa0/1          647140108         499128           4305              0

Port           OutOctets   OutUcastPkts   OutMcastPkts   OutBcastPkts
Fa0/1           28533484         319996             52              3

ASW1#show interfaces FastEthernet 0/1 counters errors

Port        Align-Err      FCS-Err    Xmit-Err     Rcv-Err  UnderSize  OutDiscards
Fa0/1               0        12618           0       12662          0            0

Port      Single-Col  Multi-Col   Late-Col  Excess-Col  Carri-Sen       Runts      Giants
Fa0/1              0          0          0           0          0           0          44
```


One of the first things to check in case of suspected performance problems are the interfaces. If the physical cabling is bad, this will cause packet loss, and packet loss can cause various performance problems.

TCP-based applications can survive a certain amount of packet loss, because the TCP protocol has retransmission capabilities that allow it to recover lost packets. However, TCP also has flow control mechanisms, and the way these mechanisms work is that TCP will slow down its transmission rate based on packet loss, because the most common cause of packet loss is (temporary) congestion on the network. As a result, packet loss caused by bad cables or interfaces will result in slow TCP-based connections across an interface.

User Datagram Protocol (UDP) does not have inherent retransmission mechanisms, and therefore the result of packet loss on UDP-based applications depends on the way that the application deals with packet loss. For real-time traffic such as voice or video, high percentages of packet loss directly affect the quality of the voice or video.

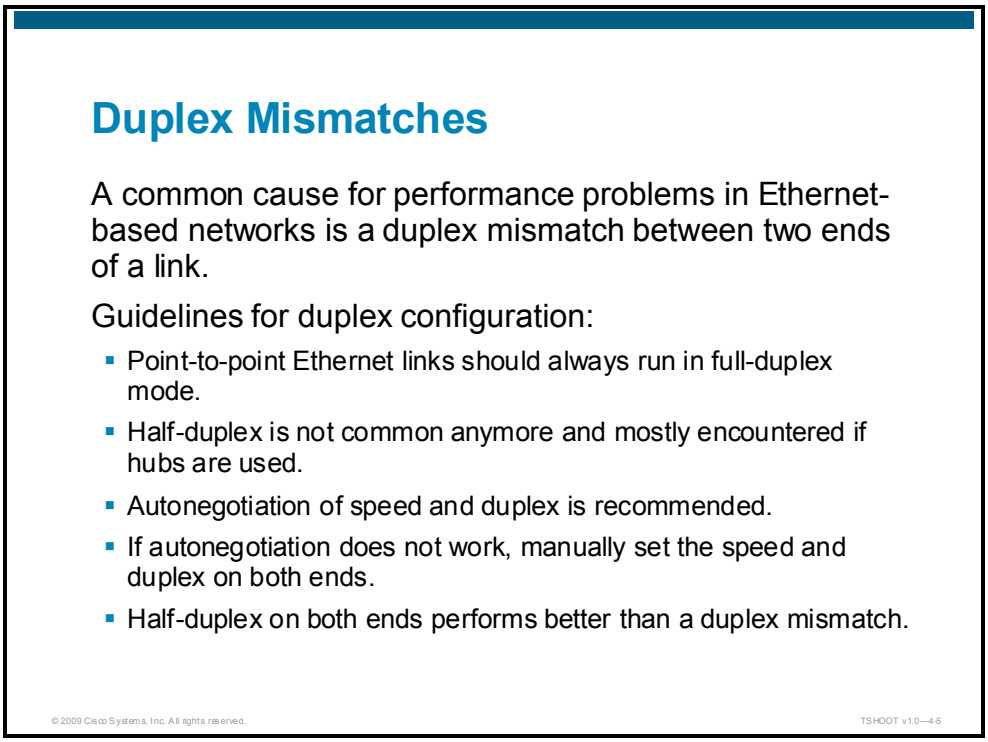
Where can you find indications of packet loss on a switch?

The first place to look is usually the **show interface** command. This will show packet statistics including various error counters. On switches, two additional command options are supported that are not available on routers. The **show interfaces counters** command displays the total numbers of input and output unicast, multicast, and broadcast packets and the total input and output byte counts. The **show interfaces counters errors** command displays the error statistics for the interface. It lists the following categories of errors:

- **Align-Err:** This is the number of frames with alignment errors, which are frames that do not end with an even number of octets and have a bad cyclic redundancy check (CRC), received on the port. These usually indicate a physical problem—for example, cabling, a bad port, or a bad NIC—but can also indicate a duplex mismatch. When the cable is first connected to the port, some of these errors can occur. Also, if there is a hub connected to the port, collisions between other devices on the hub can cause these errors.

- **FCS-Err:** The number of valid size frames with frame check sequence (FCS) errors but no framing errors. This is typically a physical issue—for example, cabling, a bad port, or a bad network interface card (NIC)—but can also indicate a duplex mismatch.
- **Xmit-Err and Rcv-Err:** This indicates that the internal port transmit (Tx) or receive (Rx) buffers are full. A common cause of Xmit-Err is traffic from a high-bandwidth link that is switched to a lower-bandwidth link, or traffic from multiple inbound links that is switched to a single outbound link. For example, if a large amount of bursty traffic comes in on a gigabit port and is switched out to a 100-Mb/s port, the Xmit-Err field can increment on the 100-Mb/s port. This is because the port output buffer is overwhelmed by the excess traffic due to the speed mismatch between the incoming and outgoing bandwidths.
- **Undersize:** The frames received that are smaller than the minimum IEEE 802.3 frame size of 64 bytes long (which excludes framing bits, but includes FCS octets) are otherwise well formed, so they have a valid CRC. Check the device that sends out these frames.
- **Single-Col:** The number of times one collision occurs before the port transmits a frame to the media successfully. Collisions are normal for ports operating in half-duplex mode, but should not be seen on ports operating in full-duplex mode. If collisions are increasing dramatically, this indicates a highly utilized link or possibly a duplex mismatch with the attached device.
- **Multi-Col:** This is the number of times multiple collisions occur before the port transmits a frame to the media successfully. Collisions are normal for ports operating in half-duplex mode, but should not be seen on ports operating in full-duplex mode. If collisions increase dramatically, this indicates a highly utilized link or possibly a duplex mismatch with the attached device.
- **Late-Col:** This is the number of times that a collision is detected on a particular port late in the transmission process. For a 10-Mb/s port, this is later than 512 bit-times into the transmission of a packet. Five hundred and twelve bit-times corresponds to 51.2 microseconds on a 10-Mb/s system. This error can indicate a duplex mismatch, among other things. For the duplex mismatch scenario, the late collision is seen on the half-duplex side. As the half-duplex side transmits, the full-duplex side does not wait its turn and transmits simultaneously, which causes a late collision. Late collisions can also indicate an Ethernet cable or segment that is too long. Collisions should not be seen on ports configured as full-duplex.
- **Excess-Col:** This is a count of frames transmitted on a particular port, which fail due to excessive collisions. An excessive collision occurs when a packet has a collision 16 times in a row. The packet is then dropped. Excessive collisions are typically an indication that the load on the segment needs to be split across multiple segments, but can also point to a duplex mismatch with the attached device. Collisions should not be seen on ports configured as full-duplex.
- **Carri-Sen:** This occurs every time an Ethernet controller wants to send data on a half-duplex connection. The controller senses the wire and checks if it is not busy before transmitting. This is normal on a half-duplex Ethernet segment.
- **Runts:** The frames received that are smaller than the minimum IEEE 802.3 frame size (64 bytes for Ethernet), and with a bad CRC. This can be caused by a duplex mismatch and physical problems, such as a bad cable, port, or NIC on the attached device.
- **Giants:** These are frames that exceed the maximum IEEE 802.3 frame size (1518 bytes for non-jumbo Ethernet), and have a bad FCS. Try to find the offending device and remove it from the network. In many cases, it is the result of a bad NIC.

It is important to relate any error statistics to the total number of received frames in case of receive errors (such as FCS errors) or the total number of transmitted frames in case of transmit errors (such as collisions). For example, in the figure, we can see that there are 12618 FCS errors on a total of 499128 + 4305 + 0 = 503433 received frames, which translates to 2.5 percent of the received traffic on the interface. In general, getting more than one FCS error in a million frames on average is reason to investigate, so this percentage of errors is clearly too high.



A common cause for interface errors is a mismatched duplex mode between two ends of an Ethernet link. In many Ethernet-based networks, point-to-point connections are now the norm, and the use of hubs and the associated half-duplex operation is becoming less common. This means that most Ethernet links today operate in full-duplex mode, and while collisions were seen as normal for an Ethernet link, collisions today often indicate that duplex negotiation has failed and the link is not operating in the correct duplex mode.

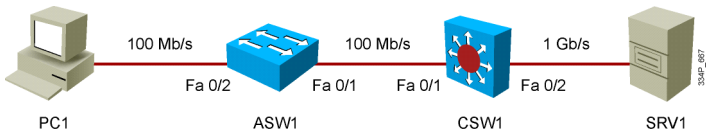
The IEEE 802.3ab Gigabit Ethernet standard mandates the use of auto negotiation for speed and duplex. In addition, although it is not strictly mandatory, practically all Fast Ethernet NICs also use auto negotiation by default. The use of auto negotiation for speed and duplex is the current recommended practice.

However, if duplex negotiation fails for some reason, it might be necessary to set the speed and duplex manually on both ends. Typically, this would mean setting the duplex mode to full-duplex on both ends of the connection. However, if even this cannot be made to work, running half-duplex on both ends is always to be preferred over a duplex mismatch.

Case Study: Duplex Problem

The user on PC1 regularly needs to transfer large files (several gigabytes each) between his PC and the server SRV1. He complains that transfers on this PC take hours instead of minutes.

- Initially you think that this might be caused by congestion on the network. But when you verify the average bandwidth usage, you notice that that none of the links in the path has been loaded above 50 percent in the past few hours.
- You start checking the links in the path for errors.



The case study in the figure shows you what the effect of a duplex mismatch can be and which observed symptoms can point to a duplex mismatch as the cause of a performance problem.

The user on PC1 has complained that transferring large files to SRV1 is much slower than he expects. First, you need to verify whether this is really a technical problem, or if the performance is within the boundaries of what you could reasonably expect in this situation. After you have determined the traffic path between the client and the server, you conclude that the maximum throughput that this user can expect is 100 Mb/s.

You calculate the number of seconds that it would take to transfer 1 GB of data at the full rate of 100 Mb/s. 1 GB equals 1,073,741,824 bytes, which equals 8,589,934,592 bits. 8,589,934,592 bits divided by 100,000,000 bits per second yields 86 seconds. This calculation does not take any overhead into account, but it is easy enough to see that even with added overhead the transfers should still only take minutes, not hours.

So, this leaves us with two potential explanations. Either congestion on the network causes this user to get only a small portion of the available 100 Mb/s, or this is caused by underperforming hardware or software on the client, network, or server. When you verify the load on the links in the path in your performance management system, you notice that the average load has not been higher than 50 percent over the last few hours. This rules out congestion as a cause of the problem. One approach to this problem is to start comparative tests from several different points in the network, but you decide to first verify the physical path between the client and server to see if the network might be the cause of the problem.

Duplex Mismatch: Full-Duplex Side

The typical symptoms of a duplex mismatch are:

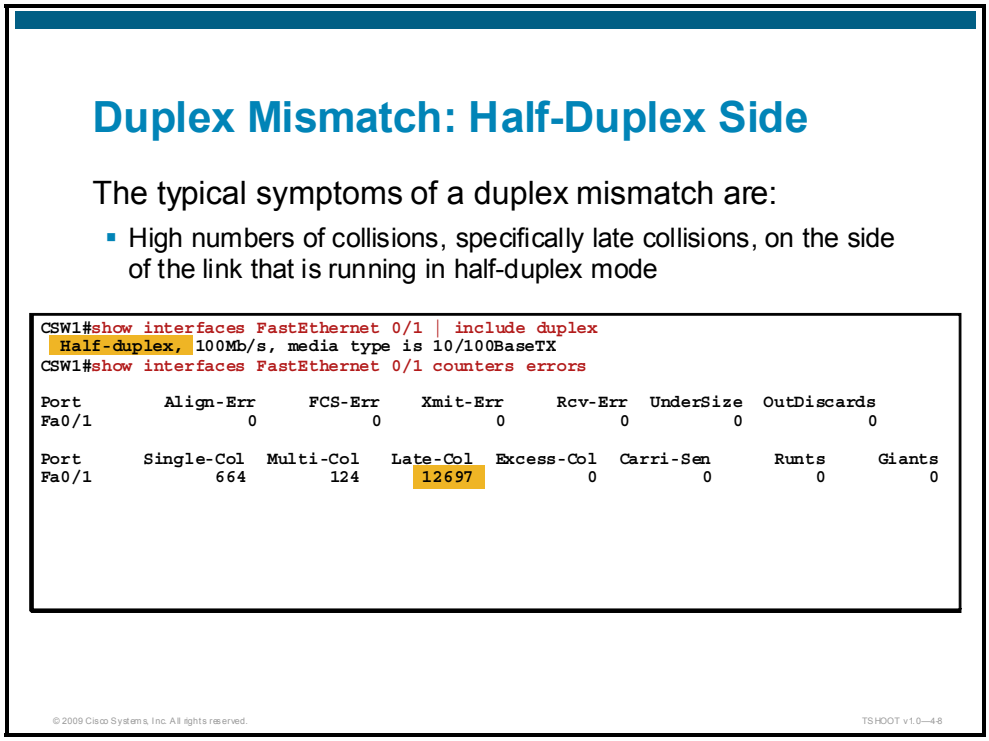
- High numbers of FCS errors on the side of the link that is running in full-duplex mode

```
ASW1#show interface FastEthernet 0/1 | include duplex
  Full-duplex, 100Mb/s, media type is 10/100BaseTX

ASW1#show interfaces FastEthernet 0/1 counters errors

Port        Align-Err     FCS-Err     Xmit-Err     Rcv-Err  UnderSize  OutDiscards
Fa0/1               0       12618            0       12662          0            0

Port      Single-Col  Multi-Col   Late-Col  Excess-Col  Carri-Sen      Runs     Giants
Fa0/1              0          0          0           0          0         0         44
```


Interface FastEthernet 0/2 on ASW1, which leads to the client, does not show a significant number of errors. However, when you verify interface FastEthernet 0/1, which leads to CSW1, you notice a high percentage of FCS errors.

Although FCS errors can have various other causes, such as bad cabling or bad interface hardware, they are also a symptom associated with a duplex mismatch.

If one side of an Ethernet link is running in full-duplex mode and the other side is running in half-duplex, you will typically see the following symptoms. On the side of the connection that is running in full-duplex mode, you will see FCS errors rapidly increasing. This happens because a NIC that is operating in full-duplex mode will not listen for carrier, but simply start transmitting whenever it has a frame to transmit. If the other side happens to be transmitting at that same moment, it will see the transmission coming in and detect a collision as a result. It will immediately stop its own transmission. This in turn causes only a partial frame to be received by the full-duplex side of the connection, which is recorded as an FCS error.



When you connect to switch CSW1 and verify the same connection on that side, you notice high numbers of collisions, specifically late collisions. This is an important clue, because you only see collisions on links that run in half-duplex mode. The fact that you see high numbers of collisions on this side and no collisions on the other side tells you that this side is running in half-duplex mode and the other side is running in full-duplex mode.

However, even if you would not have access to switch ASW1, the counters for interface FastEthernet 0/1 on CSW1 strongly suggest a duplex problem. If this link is supposed to run in half-duplex mode, a certain number of collisions is considered normal. However, late collisions should not happen and always indicate a problem.

The reason that you see late collisions on the half-duplex side of a duplex mismatch is the same behavior that causes the FCS errors on the full-duplex side. In normal half-duplex Ethernet operation, collisions will only happen during the first 64 bytes of a transmission. In case of a duplex mismatch, however, the full-duplex side will start transmitting its frames without listening for carrier, and this can be at any point during a transmission by the half-duplex side.

After considering all these symptoms, you conclude that the duplex mismatch is likely the cause of the performance problem. You verify the settings on both switches, and it turns out that somehow a mismatched manual speed and duplex configuration has caused this mismatch. You configure both sides for autonegotiation. You confirm that the negotiation results in full duplex operation on both ends, and you clear the counters for the interfaces.

Together with the user, you test by transferring some files. The transfers now only take a couple of minutes. You verify on the switches that the FCS and collision counters do not increase. You make a backup of the configuration and document the change.

Auto-MDIX and Duplex

- The automatic media-dependent interface crossover (Auto-MDIX) feature can detect the required cable connection type, straight-through or crossover, for a connection and automatically configure it appropriately.
- This feature allows you to use either crossover or straight-through cables to connect devices, and the interface will automatically correct for incorrect cabling.
- This feature is enabled by default on switches that support it.
- Auto-MDIX is dependent on auto negotiation for speed and duplex. If speed and duplex negotiation is disabled, auto-MDIX will be disabled as well.

Automatic medium-dependent interface crossover (auto-MDIX) is a feature supported on many switches and NICs. This feature automatically detects the required cable connection type (straight-through or crossover) for a connection. As long as one of the two sides of a connection supports auto-MDIX, you can use a crossover or a straight-through Ethernet cable and the connection will work. However, this feature is dependent on the speed and duplex autonegotiation feature, and disabling speed and duplex negotiation will also disable auto-MDIX for an interface.

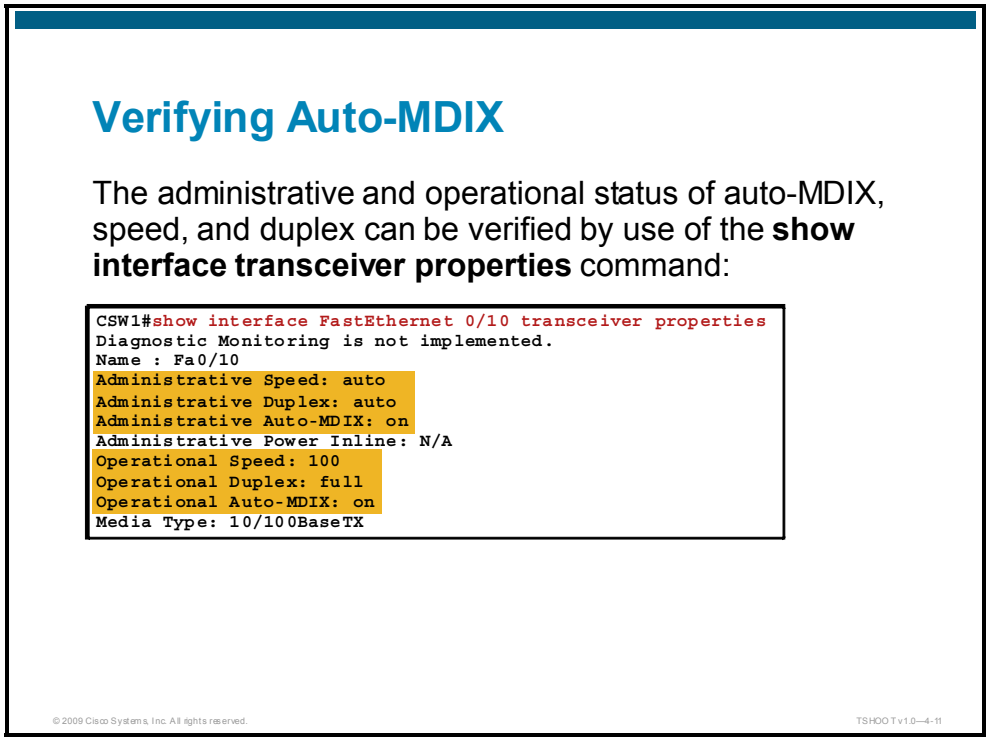
Configuring Auto-MDIX

This example shows how to enable auto-MDIX for an interface:

```
CSW1(config)#interface FastEthernet 0/10
CSW1(config-if)#mdix auto
CSW1(config-if)#speed auto
CSW1(config-if)#duplex auto
```

- Starting with Cisco IOS Software release 12.2(20)SE, the **mdix auto** command is enabled by default.
- Speed and duplex need to be set to auto for auto-MDIX to work.

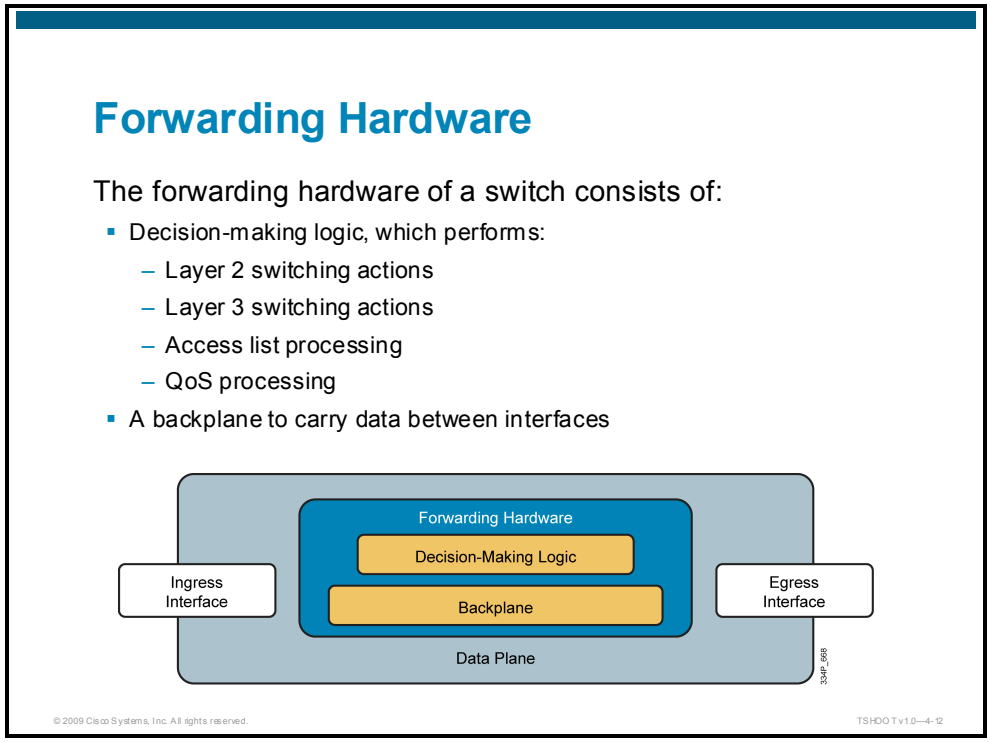
The default setting for auto-MDIX was changed from disabled to enabled starting from Cisco IOS Software release 12.2(20)SE. Therefore, auto-MDIX does not specifically need to be enabled on most switches. However, if you have a switch that supports auto-MDIX but is running older software, you can enable the feature manually by use of the **mdix auto** command. Be aware that this will enable auto-MDIX only if speed and duplex autonegotiation is enabled as well.



To verify the status of auto-MDIX, speed, and duplex for an interface you can use the **show interface transceiver properties** command.

Troubleshooting TCAM Problems

This topic describes the typical symptoms and performance degradation that can occur when the limits of the ternary content addressable memory (TCAM) on a switch are reached.



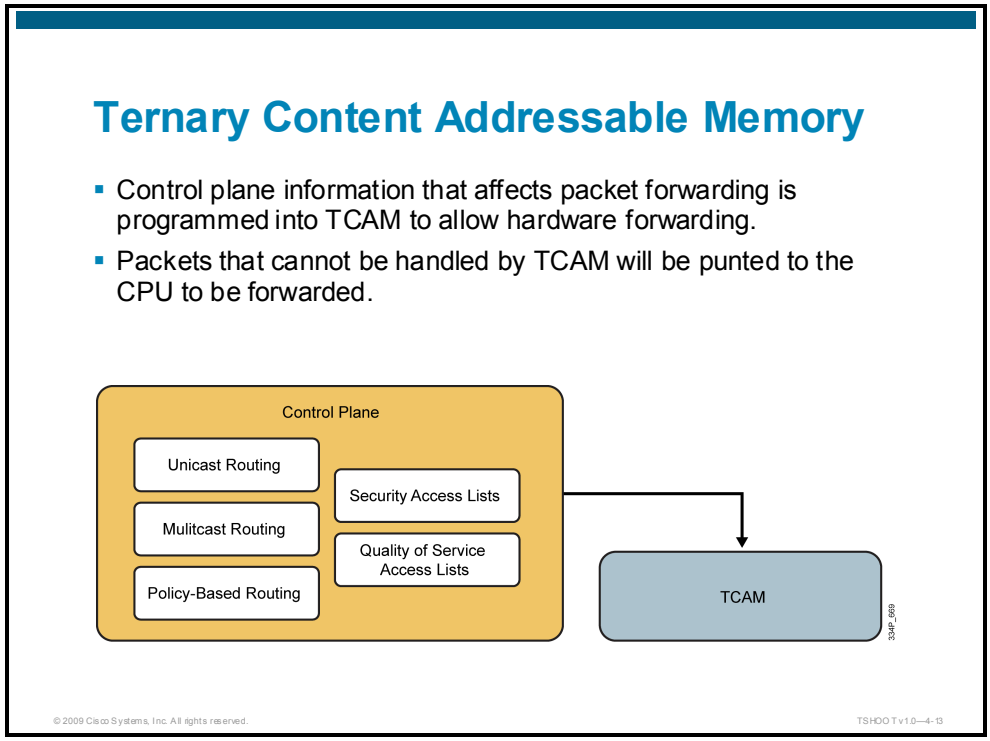
After considering the impact of ingress and egress interfaces on switch performance, this topic will look at the components of the forwarding hardware that are involved in switching the frames from the ingress interface to the egress interface and the effect that they have on the performance of the switch. Essentially the forwarding hardware always consists of two major components:

- **Backplane:** The backplane carries traffic between interfaces. There are many different types of backplane architectures. The hardware of a switch backplane can be based on a ring, bus, shared memory, or crossbar fabric, or a combination of these elements.
- **Decision-making logic:** For each incoming frame, the decision-making logic makes the decision to either forward the frame or discard it and, if it is forwarded, decides how it will be forwarded. It will provide the information that is necessary to rewrite and forward the frame and take other actions, such as the processing of access lists or quality of service (QoS) features.

The effect of the backplane on switch performance is limited. The backplane of a switch is designed for very high switching capacity. In most cases, the limiting factor in throughput on a switched network is the capacity of the links between the devices, not the capacity of the backplanes of the switches. Still, in certain specific cases the backplane can become a bottleneck and must be taken into account to correctly compute the maximum total throughput between a number of devices. For example, a number of ports might share a certain amount of bandwidth to the switch backplane. If that shared bandwidth is lower than the total bandwidth of all the ports combined, the ports are oversubscribed.

This situation is very similar to the situation where you have an access switch with 24 FastEthernet ports and a single 1-Gb/s uplink. The total aggregate bandwidth for the 24 FastEthernet ports is 2.4 Gb/s, and if they all need to send at full speed across the uplink, congestion will occur and frames will be dropped. However, in most cases, the 24 ports will not be transmitting at full speed at the same time and their combined load will easily fit the 1-Gb/s uplink.

Similarly, there could be instances where the hardware architecture of the switch limits the total throughput between a number of ports, and in some cases, it helps to rearrange the devices that are affected by this and connect them to different ports. However, to make these decisions you need to have detailed knowledge of both the expected traffic patterns and the specific switch hardware architecture.



The decision-making logic of a switch can have a big impact on switch performance. The decision-making logic consists of specialized high-performance lookup memory: TCAM. The control plane information that is necessary to make forwarding decisions, such as MAC address tables, routing information, access list information, and quality of service information, is programmed into the TCAMs. The TCAMs then take all the necessary decisions to forward a frame at speeds that are high enough so that the full capacity of the backplane can be utilized and the decision-making process will not affect the forwarding performance of the switch.

Seen in that light, TCAM performance itself does not limit the switch performance. However, if for some reason frames cannot be forwarded by the TCAM, they will be handed off (punted) to the CPU for processing. The CPU can forward only a limited amount of traffic, and the same CPU is used to execute the control plane processes, so if a large amount of traffic is punted to the CPU this will affect the throughput for the traffic concerned, and it can have an adverse affect on the control plane processes.

What could cause the TCAM to punt packets to the CPU for forwarding?

The fundamental answer to this question is that the TCAM will punt any frames to the CPU for forwarding that it cannot forward itself. (Note that this does not include frames that are explicitly dropped, such as frames dropped by an access list, frames that are dropped because the inbound port is in the spanning tree blocking state, frames that are dropped because their VLAN is not allowed on a trunk, and so on.

Some examples of packets that will be punted or copied to the CPU are:

- Packets destined for any of the switch IP addresses, for example Telnet, Secure Shell (SSH), or Simple Network management Protocol (SNMP) packets destined for one of the switch IP addresses.

- Multicasts and broadcasts from control plane protocols such as the Spanning Tree Protocol (STP) or routing protocols. Routing protocol broadcasts and multicasts will be copied to the CPU in addition to being flooded to all ports in the VLAN that the frame was received in as usual.
- Packets that cannot be forwarded by the TCAM because a feature is not supported in hardware. For example, Generic Routing Encapsulation (GRE) tunnels can be configured on a Cisco Catalyst 3560 switch, but this is not a supported configuration because the packets will not be forwarded in hardware.
- Packets that cannot be forwarded in hardware because the TCAM could not hold the necessary information. The TCAM has a limited capacity, and when entries cannot be programmed into the TCAM, the packets associated to those entries will need to be punted to the CPU to be forwarded. So, for example, if you have too many IP routes or too many access list entries, some of them will not be installed in the TCAM and associated packets cannot be forwarded in hardware.

The last item is the most likely to cause performance problems on a switch. The CPU always handles control plane packets in software, and the volume of this type is relatively low. However, the volume of traffic that flows through a switch is substantial, and if even a fraction of this traffic is forwarded to the CPU, it will quickly cause performance degradation. The traffic itself will suffer, because the CPU will not be able to handle high volumes of traffic and control plane processes will suffer, because the packet-switching process will consume a large share of the available CPU cycles.

Troubleshooting TCAM Utilization

The commands to verify TCAM utilization are platform dependent.

The examples here are for the Catalyst 3560 and 3750 switches and illustrate the principles of troubleshooting problems related to TCAM utilization.

The tools that you can be use to verify TCAM utilization and troubleshoot TCAM related problems are, by their nature, platform- and hardware-dependent. However, the principles of troubleshooting TCAM-related problems are similar. This topic will use commands that are available on the Cisco Catalyst 3560 and Cisco Catalyst 3750 platforms to illustrate these principles.

Consult the documentation of the platforms that you are working on to find the relevant commands to troubleshoot TCAM problems on that platform.

Verifying TCAM Utilization

- The **show platform tcam utilization** command can be used to determine the current utilization of the TCAM:

```
CSW1#show platform tcam utilization

CAM Utilization for ASIC# 0                    Max              Used
                                             Masks/Values     Masks/values

 Unicast mac addresses:                       784/6272          23/99
 IPv4 IGMP groups + multicast routes:         144/1152           6/26
 IPv4 unicast directly-connected routes:      784/6272          23/99
 IPv4 unicast indirectly-connected routes:    272/2176          30/175
 IPv4 policy based routing aces:                0/0              0/0
 IPv4 qos aces:                               768/768          260/260
 IPv4 security aces:                         1024/1024          27/27

Note: Allocation of TCAM entries per feature uses
a complex algorithm. The above information is meant
to provide an abstract view of the current TCAM utilization
```

- Comparing the "Used" column against the "Max" column gives an indication of the total TCAM usage by type.

The **show platform tcam utilization** command gives you an indication of how close the current TCAM utilization is to the platform limits. The TCAM is carved into separate areas that contain entries associated with a particular usage. Each of these areas has its associated limits.

Note	On the Cisco Catalyst 3560 and Cisco Catalyst 3750 series switches, the allocation of TCAM space for specific uses is based on a switch database manager (SDM) template. Templates other than the default can be selected to change the allocation of TCAM resources to better fit the role of the switch in the network. For more information, consult the SDM section of the configuration guide for the Cisco Catalyst 3560 or Cisco Catalyst 3750 series switches.

As you can see in the figure, the maximum number of masks and values that can be assigned to IP version 4 not directly connected routes are 272 masks and 2176 values. Currently, 30 masks and 175 values are in use. This means that this switch is still far from reaching its maximum capacity. As the **show** command tells you, the exact algorithm to allocate TCAM entries for a particular feature is complex, and you cannot simply tell how many IPv4 routes can be added to the routing table before the TCAM will reach its maximum. However, when you see the values in the Used column getting close to the values in the Max column, you might start experiencing extra load on the CPU due to failed allocation of TCAM resources.

TCAM Allocation Failures

- For certain types of TCAM entries, it is possible to see whether the TCAM allocation has failed.
- For example, the **show platform ip unicast counts** command shows whether the TCAM allocation has failed for prefixes of a particular length:

```
CSW1#show platform ip unicast counts
# of HL3U fibs 141
# of HL3U adjs 9
# of HL3U mpaths 2
# of HL3U covering-fibs 0
# of HL3U fibs with adj failures 0
Fibs of Prefix length 0, with TCAM fails: 0
Fibs of Prefix length 1, with TCAM fails: 0
Fibs of Prefix length 2, with TCAM fails: 0
Fibs of Prefix length 3, with TCAM fails: 0
Fibs of Prefix length 4, with TCAM fails: 0
Fibs of Prefix length 5, with TCAM fails: 0
Fibs of Prefix length 6, with TCAM fails: 0
<… further output omitted …>
```


For some types of TCAM entries, it is possible to see if any TCAM allocation failures have occurred. For example, the **show platform ip unicast counts** command will show you if any TCAM allocation failures were experienced for IP version 4 prefixes and, if so, what the length of these prefixes was.

In general, TCAM allocation failures are rare, because switches have more than enough TCAM capacity for the roles that they are designed and positioned for. However, all networks are different, so it is good to be aware of the fact that TCAM allocation failures can be a possible cause of performance problems.

Traffic Forwarding to the CPU

- Traffic being punted to the CPU for forwarding constitutes indirect proof of TCAM allocation failures or use of unsupported features.
- The **show controllers cpu-interface** command displays the statistics for packets that are forwarded to the CPU:

```
CSW1#sh controllers cpu-interface
ASIC    Rxbiterr   Rxunder    Fwdctfix   Txbuflos   Rxbufloc   Rxbufdrain
------------------------------------------------------------------------
ASIC0     0          0          0          0          0          0


cpu-queue-frames  retrieved  dropped    invalid    hol-block  stray
----------------- ---------- ---------- ---------- ---------- ----------
rpc               1          0          0          0          0
stp               853663     0          0          0          0
ipc               0          0          0          0          0
routing protocol  1580429    0          0          0          0
L2 protocol       22004      0          0          0          0
remote console    0          0          0          0          0
sw forwarding     1380174    0          0          0          0
<… further output omitted …>
```

In some cases, TCAM allocation failures can be observed directly, as seen in the previous figure. Another way to spot potential TCAM allocation failures is by observing traffic that is being punted to the CPU for forwarding.

The command **show controllers cpu-interface** displays packet counts for packets that are forwarded to the CPU. If the retrieved packet counter in the "sw forwarding" row is rapidly increasing when you execute this command multiple times in a row, traffic is being switched in software by the CPU instead of in hardware by the TCAM. An increased CPU load usually accompanies this behavior.

So what does this mean when it comes to troubleshooting performance problems on switches?

It is important to understand that TCAM resources are limited and that TCAM allocation problems can lead to packets being switched by the CPU instead of the TCAM. If this causes the CPU to be loaded to nearly 100 percent it will affect not only the traffic that is being forwarded by the CPU, but also control plane processes.

As a result, whenever you observe performance problems for traffic passing through a switch, and the CPU of that switch is consistently running at a very high load, you should investigate whether this problem might be caused by traffic that is forwarded by the CPU, and in turn whether this is caused by exhaustion of TCAM resources.

Finally, how do you resolve these problems?

Given that the cause of the problem is the fact that the amount of control plane information is too large to allow all of it to be programmed into the TCAMs, the solution is to reduce the amount of control plane information, for example by implementing summarization or route filtering for IP routes or optimizing access lists. In general, TCAMs are not upgradable, so either the information that needs to be programmed into the TCAM needs to be reduced or you will need to upgrade to a better switch that can handle more TCAM entries.

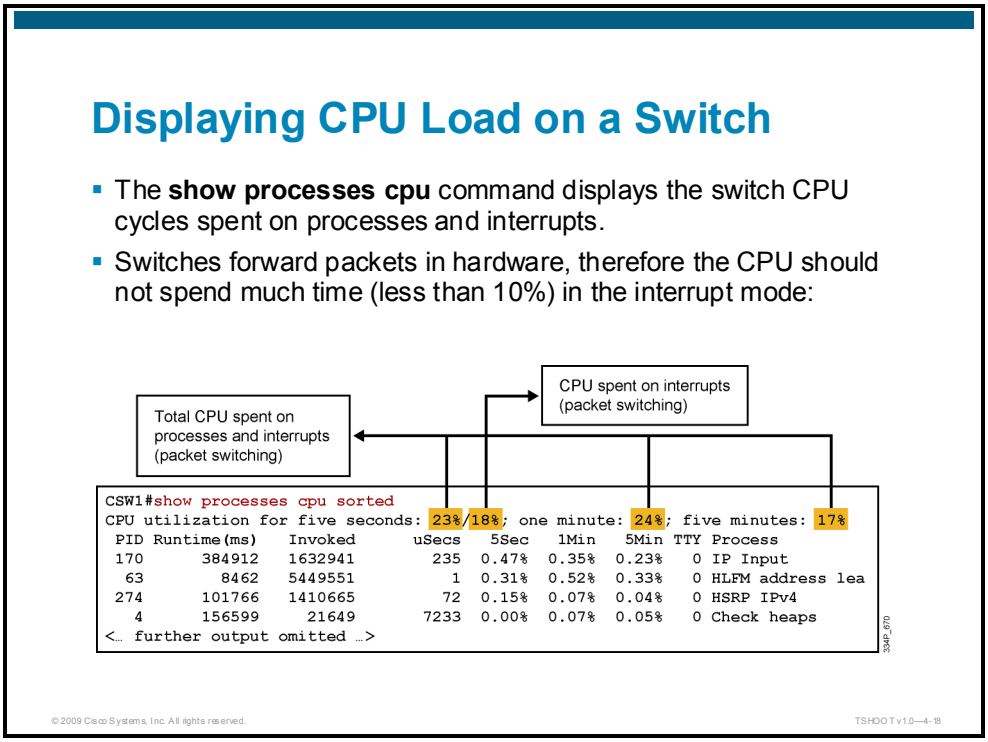
On some switches, such as the Cisco Catalyst 3560 and Cisco Catalyst 3750 series of switches, the allocation of TCAM space among the different features can be changed. For example, if you are deploying a switch at a point where it is almost exclusively involved in Layer 3 switching but hardly any Layer 2 switching, you can choose a different template that sacrifices TCAM space for MAC address entries in favor of IP route entries.

The TCAM allocation on the Cisco Catalyst 3560 and Cisco Catalyst 3750 series of switches is managed by the SDM. For more information, see the SDM section of the configuration guide for the Cisco Catalyst 3560 and Cisco Catalyst 3750 series of switches:

http://www.cisco.com/en/US/docs/switches/lan/catalyst3560/software/release/12.2_46_se/configuration/guide/swsdm.html

Troubleshooting High CPU Load on Switches

This topic describes how to diagnose and resolve performance degradation caused by high CPU load on switches.



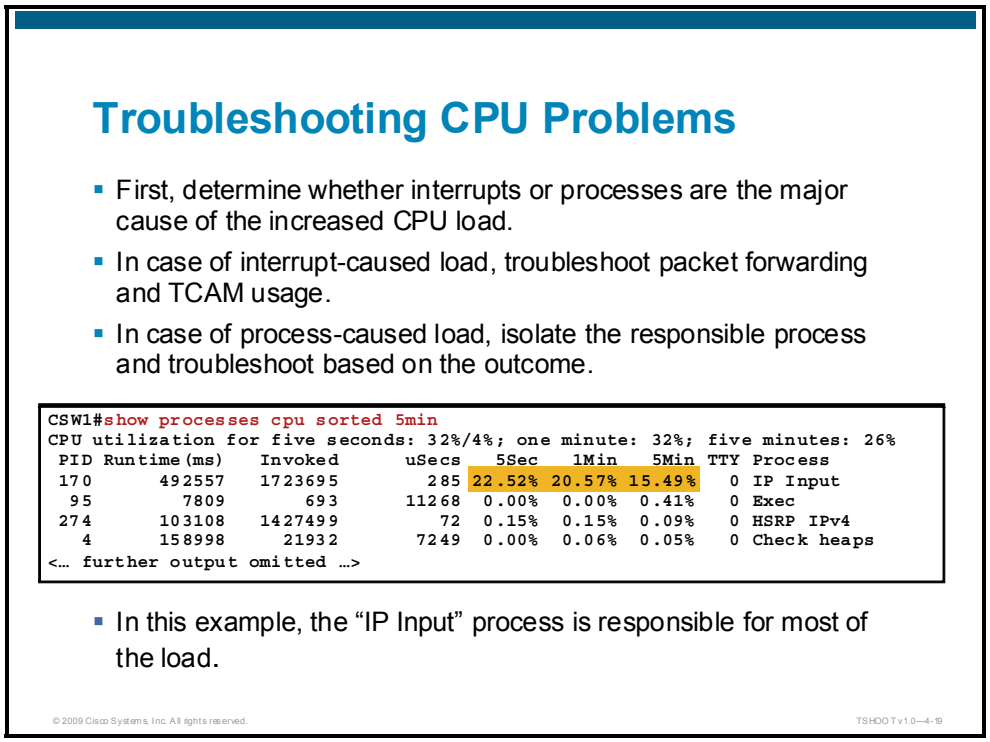
On a switch, the CPU load is directly related to the traffic load. Because the bulk of the traffic is switched in hardware by the TCAMs, the load of the CPU is often low even when the switch is forwarding large amounts of traffic. This is an important difference with routers. Low-range to midrange routers use the same CPU for packet forwarding that is also used for control plane functions, and therefore an increase in the traffic volume handled by the router will result in a corresponding increase in CPU load. On switches, this direct relationship between CPU load and traffic load does not exist.

The command to display the CPU load, **show processes cpu**, is the same on switches as it is on routers, and the interpretation of the fields is the same. However, the conclusions that you can draw from the output of the command are different, due to the different implementation of the packet-switching process.

If you look at the output shown in the figure, you can see that over the past five seconds the switch consumed 23 percent of the available CPU cycles. Eighteen percent of the CPU time was spent on interrupt processing, while only 5 percent was spent on the handling of control plane processes. For a router this would be perfectly acceptable and no reason for alarm. The CPU is forwarding packets and using the CPU to do so, and the total CPU usage is not high enough to warrant further investigation. Of course, this also depends on the normal baseline level of the CPU, but by itself, the levels are no reason for concerns about the performance of the router.

However, on a switch, this same output is a reason to investigate. A switch should not spend a significant amount of CPU time on interrupt processing, because the TCAMs should forward the bulk of the traffic and the CPU should not be involved. A percentage between 0 and 5 percent of CPU load spent on interrupts is considered normal, and a percentage between 5 and 10 percent is considered acceptable, but when CPU time spent in interrupt mode is above 10 percent, you should start to investigate what might be the cause.

If the CPU time spent in interrupt mode is high, this means that the switch is forwarding part of the traffic in software instead of the TCAM handling it. As discussed in the previous topic, the most likely reasons for this are TCAM allocation failures or configuration of unsupported features that cannot be handled in hardware.



To troubleshoot CPU problems effectively, it is important to have baseline measurements to compare. In general, an average CPU load of 50 percent is not problematic and temporary bursts to 100 percent are not problematic, as long as there is a reasonable explanation for the observed peaks.

The following events could all be responsible for spikes in the CPU utilization:

- **Processor-intensive Cisco IOS commands:** These might be, for example, **show tech-support**, **debug**, or even **show running-configuration**, **copy running-config startup-config,** and **write memory**.
- **Routing protocol update processing:** If the switch is acting as a Layer 3 switch and participating in a routing protocol, it might experience peaks in CPU usage when many routing updates are received at the same time and need to be processed.
- **SNMP polling:** During SNMP discoveries or other bulk transfers of SNMP information by a network management system, the CPU can temporarily peak to 100 percent. If the SNMP process is constantly using a high percentage of the available CPU cycles on a switch, you should investigate the settings on the network management station that is polling the device.

If you are observing CPU load spikes that cannot be explained by known events or if you are seeing that the CPU load is high for long periods, further investigation is warranted.

First, you need to decide whether the load is caused by interrupts or by processes. If the load is mainly caused by interrupts, you should investigate the packet-switching behavior of the switch and look for possible TCAM allocation problems. If the high load is mainly caused by processes, you should identify the responsible process or processes and see how these can be explained.

In the example in the figure, you can see that the "IP Input" process is responsible for most of the CPU load. The "IP Input" process is responsible for all IP traffic that is not handled by the TCAM or forwarded in interrupt mode. This includes the transmission of ICMP messages such as unreachable or echo reply packets.

Other processes that can be responsible for high CPU load are the following:

- **IP ARP**: This process is involved in processing Address Resolution Protocol (ARP) requests.
- **SNMP Engine**: This process is responsible for answering SNMP requests.
- **IGMPSN**: This process is responsible for Internet Group Management Protocol (IGMP) snooping and processes IGMP packets.

Another item you should be aware of is that a high CPU load for control plane protocols, such as routing protocols, first-hop redundancy protocols, and ARP, might be caused by a broadcast storm in the underlying Layer 2 network. In that case, they are not the underlying cause of the problem, but their behavior is a symptom of the underlying problem.

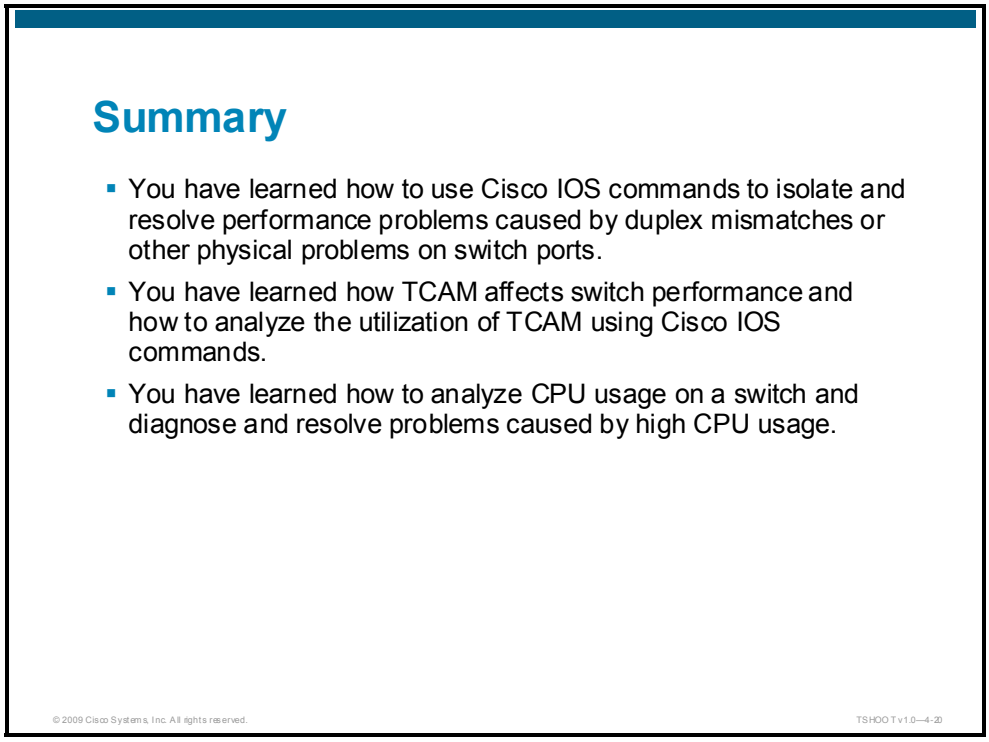
What can you do to resolve these problems?

Obviously, the answer to this question depends on the process. If routing protocol updates are causing the high CPU load, you should investigate the underlying cause of the routing protocol activity. This should always be the initial approach. Try to find the underlying cause of the problem and address it instead of only implementing workarounds that alleviate the symptoms but do not address the underlying cause.

For example, you might notice that a switch is running at 100 percent CPU, because protocols such as the Hot Standby Router Protocol (HSRP), Open Shortest Path First (OSPF), ARP, and the IEEE Spanning Tree Protocol are all using many CPU cycles as a result of a broadcast storm in the switched network. In this case, you could consider implementing broadcast and multicast storm control to limit the effect of the excessive broadcasts and multicasts generated by the broadcast storm. However, this is only a workaround; it will help you make the switch more manageable, but it does not solve the underlying spanning-tree problem. After implementing this workaround, you still must diagnose and resolve the underlying problem that caused the broadcast storm.

Summary

This topic summarizes the key points that were discussed in this lesson.



Lesson 8

References to Additional Campus Switching Technologies in E-Learning

Overview

The *Troubleshooting and Maintaining Cisco IP Networks* (TSHOOT) v1.0 instructor-led training (ILT) course is a comprehensive learning experience. Among the learning tools available to you, it includes e-learning modules that complement the classroom instructor-led content and add to your experience with self-paced materials and demonstrations. Upon accessing the e-learning content, you will be able to reinforce the knowledge acquired in class and witness real-life troubleshooting scenarios demonstrated in real routers and switches. The content structure is flexible, and you can navigate it at your own pace, at the time of your choosing, and at the depth you want according to your level of experience.

This lesson presents additional topics in switching technologies, and overviews these e-learning modules: “Troubleshooting Performance Problems on Switches”; “Troubleshooting Wireless Integration”; “Troubleshooting Voice over IP Integration”; and “Troubleshooting Video Integration.”

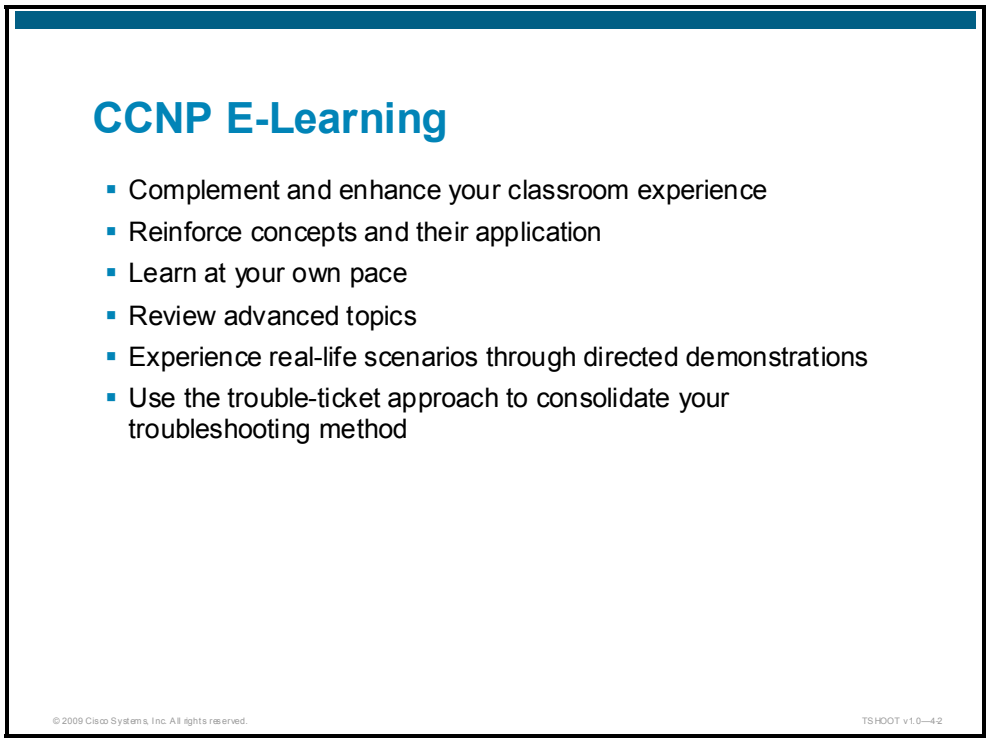
Objectives

Upon completing this lesson, you will be able to review the e-learning modules that pertain to additional topics in switching technologies. This ability includes being able to meet these objectives:

- Describe the content of the “Troubleshooting Performance Problems on Switches” e-learning module
- Describe the content of the “Troubleshooting Wireless Integration” e-learning module
- Describe the content of the “Troubleshooting Voice over IP Integration” e-learning module
- Describe the content of the “Troubleshooting Video Integration” e-learning module
- Understand the process of accessing and using e-learning content

Preview of E-Learning on Campus Switching Technologies

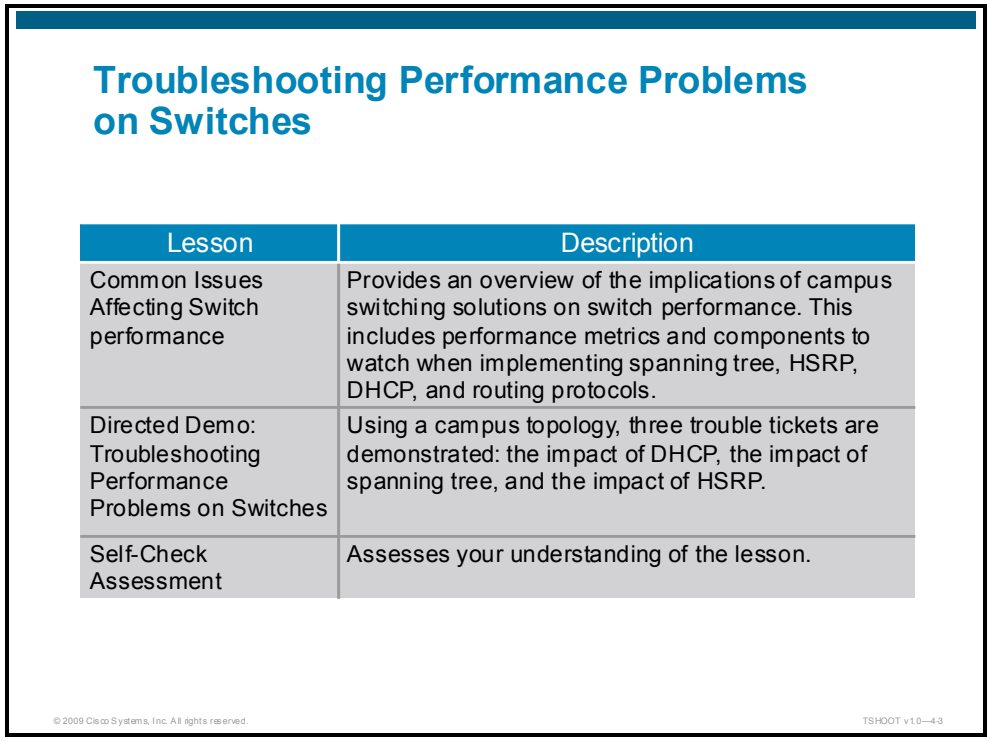
This topic describes e-learning products that teach configuring of implementation of additional campus switching technologies.



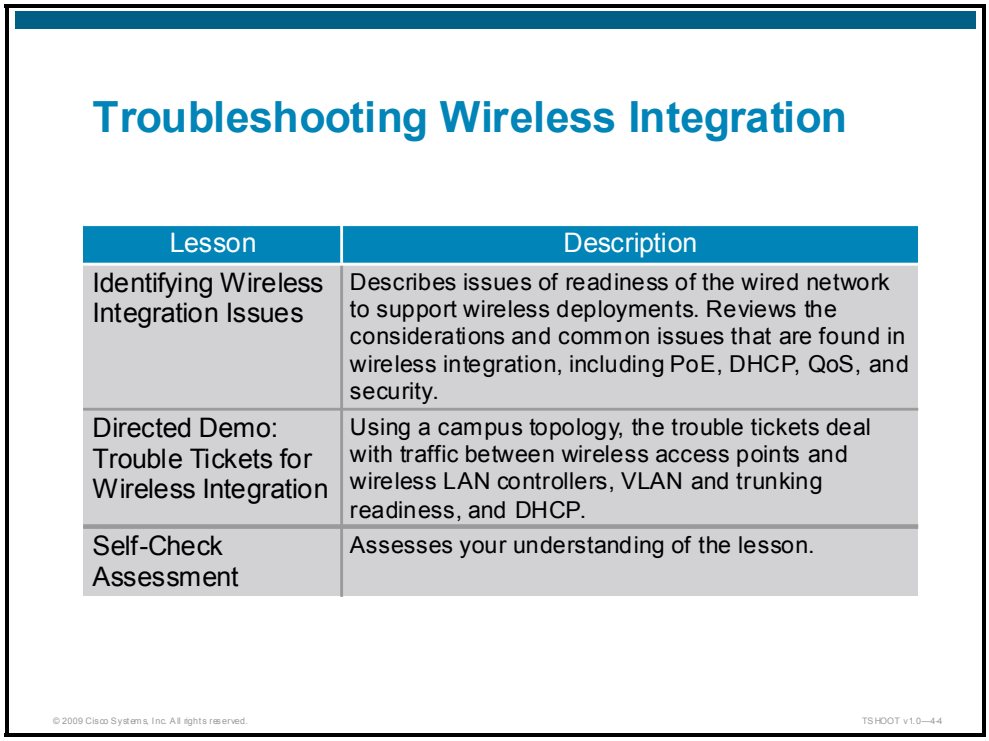
One of the main ideas behind the design of the Cisco CCNP® curriculum is the understanding that there is no one best method of learning for every student. Some students prefer individual labs, and others prefer one-on-one tutoring, hands-on sessions, self-paced computer-assisted instruction, direct learning, discovery learning, or cooperative learning, among others.

E-learning solutions offer new approaches that complement and enhance classroom-based learning. Designed with flexibility and learning effectiveness in mind, the e-learning modules presented in this lesson are based on knowledge that you acquire during your *Troubleshooting and Maintaining Cisco IP Networks* (TSHOOT) v1.0 instructor-led training course.

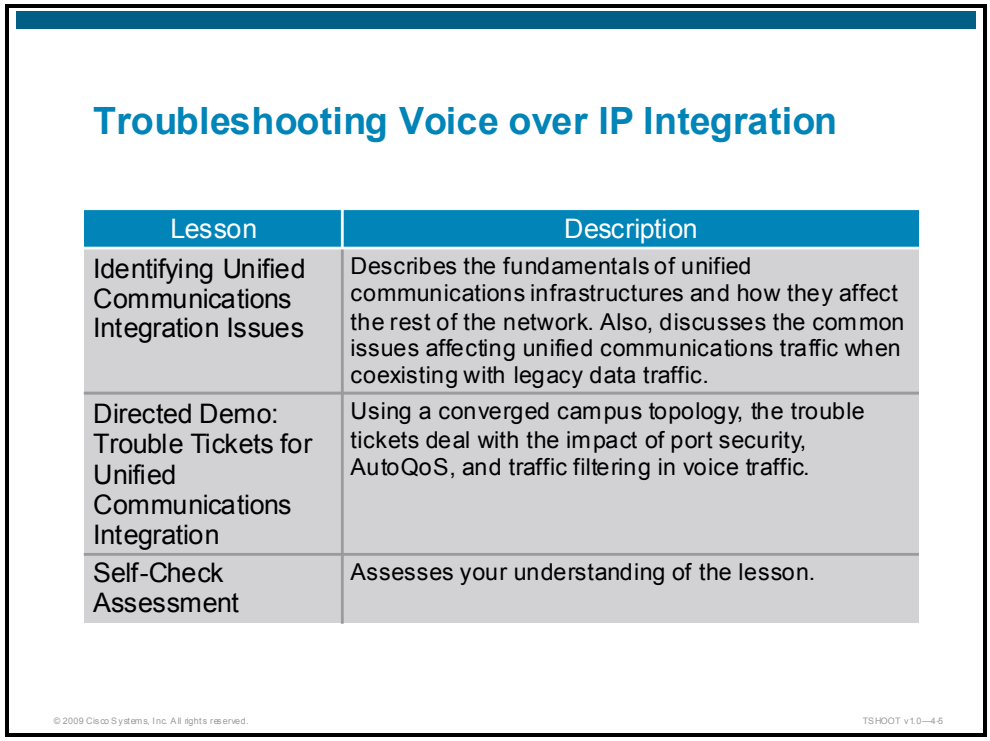
All e-learning modules described in this lesson use the trouble-ticket approach that you know very well by now: a case is presented, using certain assumptions and baseline information. The demonstrations show one or more of the possible paths to gathering information, identifying symptoms, prioritizing hypotheses, and gathering conclusions. The content is as much about the technical aspects presented as it is about applying a troubleshooting method to a specific scenario, controlled by certain variables.



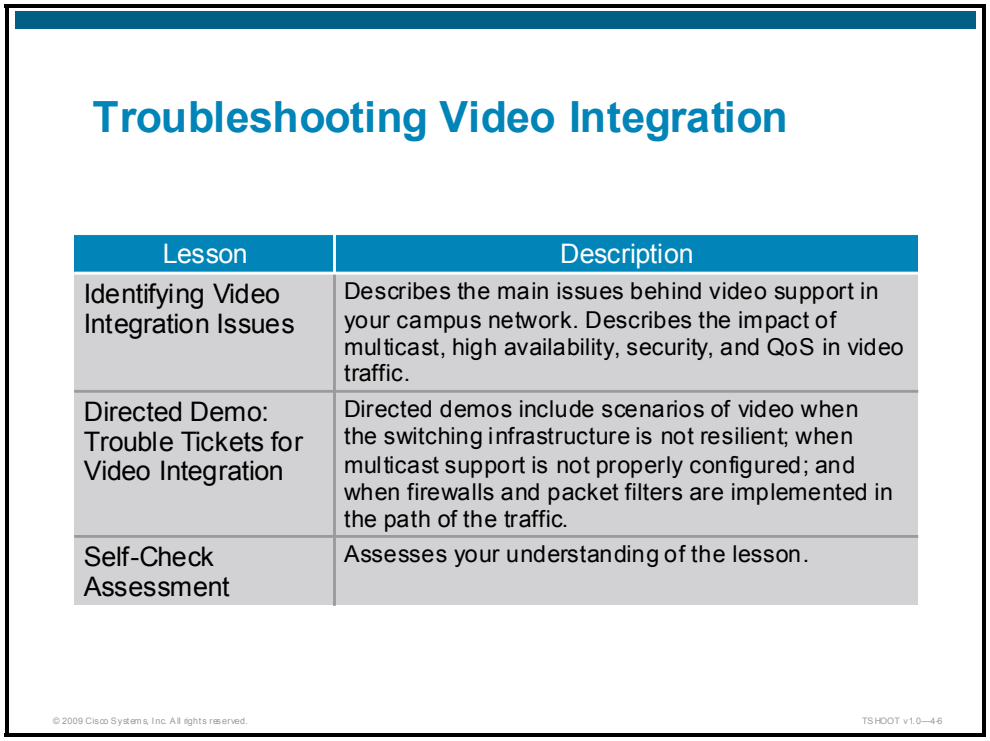
The "Troubleshooting Performance Problems on Switches" e-learning module covers topics that complement the classroom module you are reviewing. A brief introductory lesson discusses potential implementation issues of switching technologies in switch devices. The trouble-ticket lesson allows you to follow directed demonstrations of real-life scenarios that combine multiple switching technologies and measure their effect on switch performance. The focus is placed on Spanning Tree Protocol (STP), DHCP, and Hot Standby Router Protocol (HSRP). However, the objective is to identify performance variables and affected components of the switch architecture, identify the issue as a performance issue, and lay out a plan to resolve it.



The "Troubleshooting Wireless Integration" e-learning module covers topics that complement the classroom module you are reviewing. A brief introductory lesson discusses potential issues of wireless technologies when integrated into the wired Campus network. The trouble-ticket lesson allows you to follow directed demonstrations of real-life scenarios. The focus is placed on access point to wireless LAN controller interaction, and the effect of network services such as traffic filtering, DHCP, and the switching infrastructure. Remember, the objective is to use a troubleshooting methodology to approach the problem, identify critical variables and symptoms, and lay out a plan to resolve the problem.

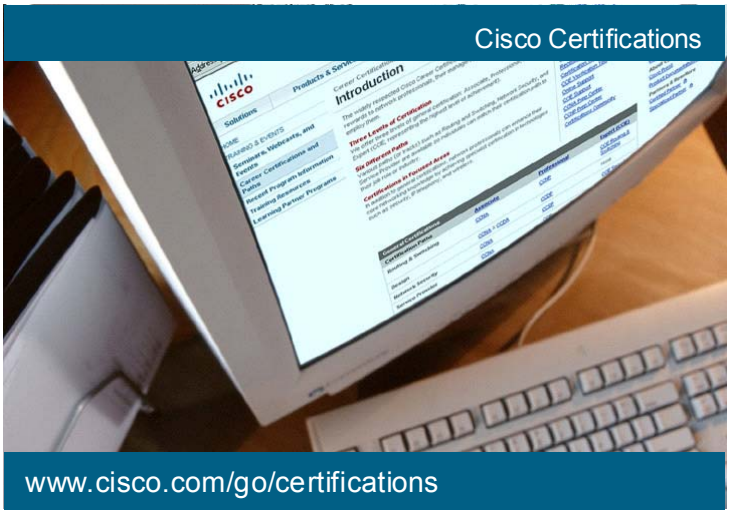


The "Troubleshooting Voice over IP Integration" e-learning module covers topics that complement the classroom module you are reviewing. It deals with the readiness of a campus network to support converged unified communications services. It also deals with the impact of converged traffic in a campus and of potential changes in the traditional network that will result in very interesting troubleshooting scenarios. The module starts with an overview lesson, summarizing the issues you need to consider. The trouble tickets are related to quality of service (QoS); rapid and automatic provisioning of a network environment for unified communications endpoints; and support services such as DHCP and Network Time Protocol (NTP).



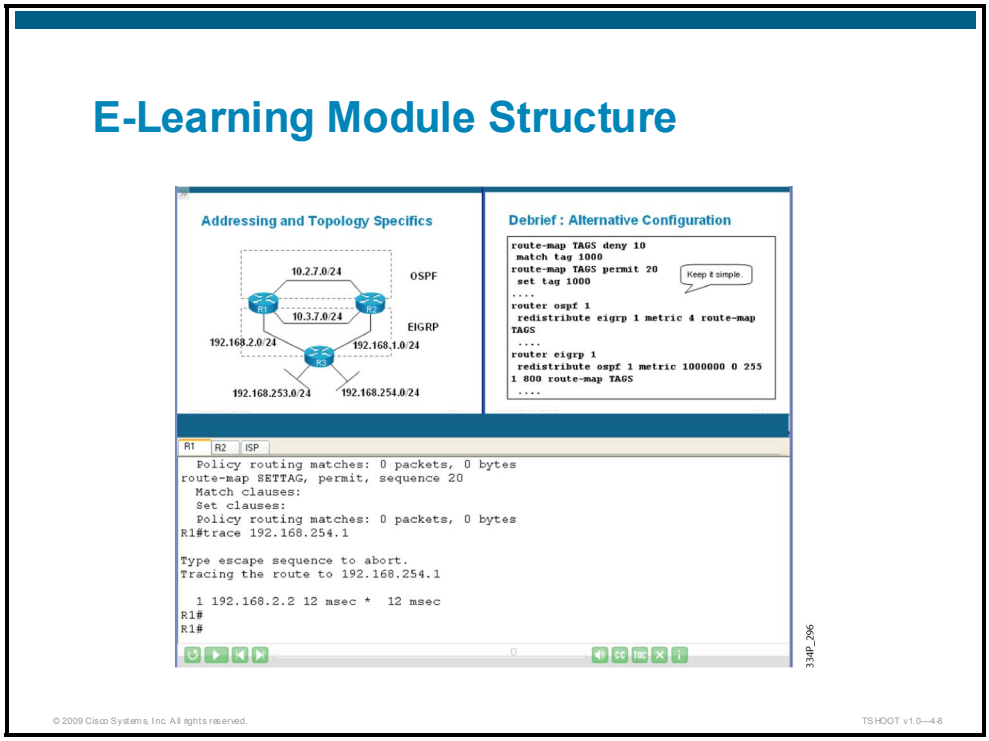
The "Troubleshooting Video Integration" e-learning module covers topics that complement the classroom module you are reviewing. The content addresses the challenge of troubleshooting network infrastructures supporting video and rich media traffic. A brief introductory lesson outlines the considerations, requirements, and common issues found in video integration. The trouble-ticket lesson demonstrates the troubleshooting method that can be used to approach some of the potential problems you might face in your video-ready network. It includes the effect of network performance issues on video traffic, and the impact of suboptimal routing, multicast technologies, and firewalling.

Where to Find E-Learning Modules



The e-learning modules presented in this lesson are available on a CD as part of your classroom materials. Contact your instructor if you have questions about finding and accessing the CD.

The objectives of the module are part of your CCNP certification exam and should be reviewed by candidates to the CCNP certification.

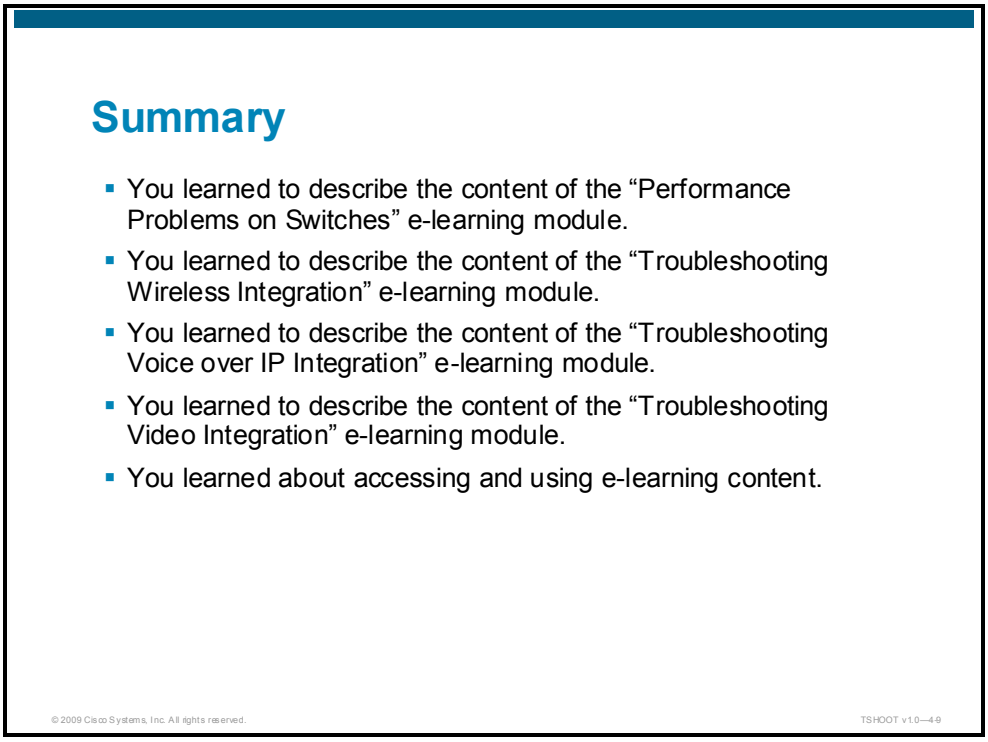


Utilizing a combination of lecture sessions, animated content, lab demonstrations, and assessments, each e-learning module presents a hierarchical structure of lessons and topics. You can navigate that structure by using an intuitive graphical user interface that includes playback controls, slide selection, and lesson and topic selection. Using those tools, you will be able to navigate the content at your own pace.

The three-panel screen presented in the directed demonstrations allows you to focus on the device console demonstrations, while having the command syntax and the topology diagram in sight for verification and additional information.

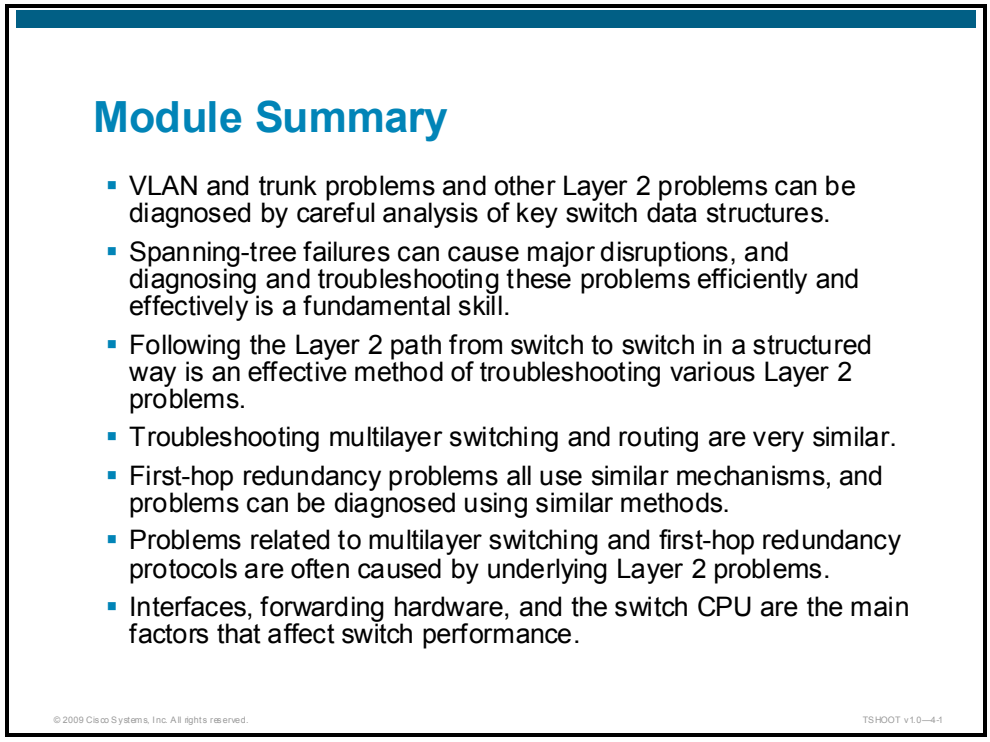
Summary

This topic summarizes the key points that were discussed in this lesson.



Module Summary

This topic summarizes the key points that were discussed in this module.



Diagnosing and resolving problems in a multilayer-switched campus network has many aspects. A good understanding of Layer 2 switching processes and careful analysis of major switch data structures, such as the MAC address table and VLAN database, is necessary to isolate, diagnose, and resolve Layer 2 switching problems. A common method of diagnosing Layer 2 problems is to follow the path from switch to switch between the affected hosts. To find the correct path through the network, a good understanding of the IEEE Spanning Tree Protocol and how it affects switch forwarding is essential.

On the control plane, multilayer switching processes are similar to routing processes, and therefore similar troubleshooting methods and tools can be used. The implementation of the forwarding process in hardware and the combination of Layer 2 switching and routing in a single chassis differentiate multilayer switching from routing. As a result, true multilayer switching problems are often related to the interaction between Layer 2 and Layer 3 or to the forwarding hardware performance.

First-hop redundancy protocols (FHRPs) are an essential element of any highly available network, and because the first hop in many modern campus networks is a multilayer switch, diagnosing and resolving problems related to first-hop redundancy is a common task in campus networks. Different protocols, such as the Hot Standby Router Protocol (HSRP), Virtual Router Redundancy Protocol (VRRP), and Gateway Load Balancing Protocol (GLBP), can be implemented to achieve first-hop redundancy, but the method of troubleshooting these protocols is very similar, because of the similarity in the mechanisms that they use.

Interfaces, forwarding hardware, and the switch CPU are the major components that affect switch performance, and information about these elements needs to be gathered, interpreted, and analyzed to determine whether a switch could be the cause of a network performance problem.

References

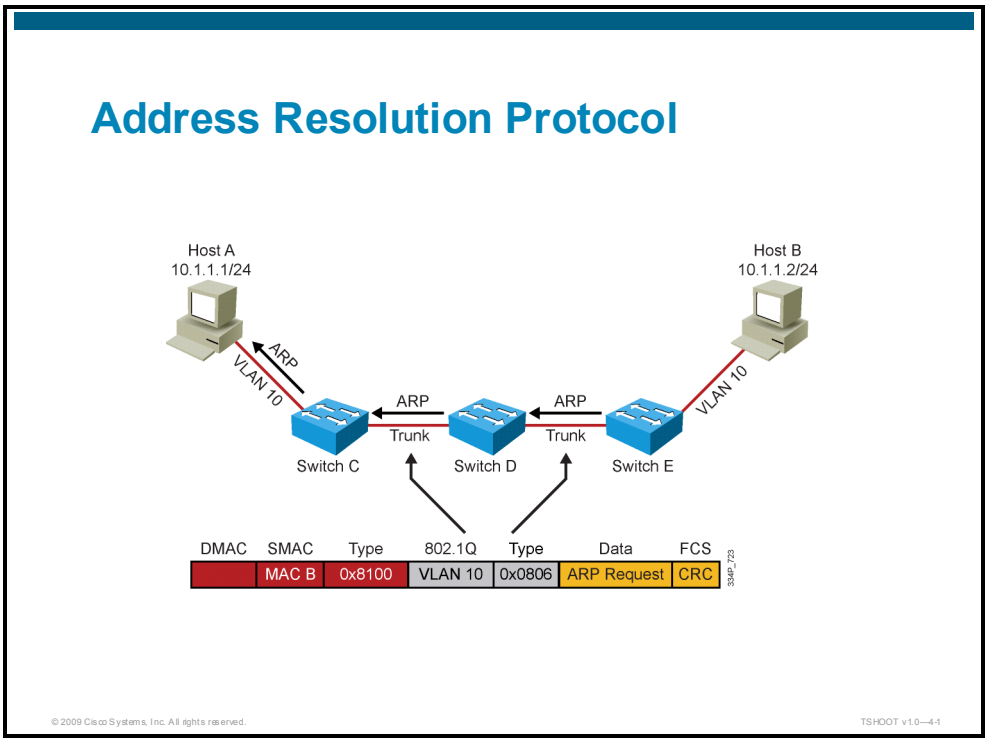
For additional information, refer to these resources:

- Cisco Systems Inc., *Understanding EtherChannel Load Balancing and Redundancy on Catalyst Switches:* http://www.cisco.com/en/US/tech/tk389/tk213/technologies_tech_note09186a0080094714.shtml
- Cisco Systems Inc., *Understanding EtherChannel Inconsistency Detection:* http://www.cisco.com/en/US/tech/tk389/tk213/technologies_tech_note09186a008009448d.shtml
- Cisco Systems Inc., *Understanding and Troubleshooting HSRP Problems in Catalyst Switch Networks:* http://www.cisco.com/en/US/tech/tk648/tk362/technologies_tech_note09186a0080094afd.shtml
- Cisco Systems Inc., *Troubleshooting Spanning Tree PVID- and Type-Inconsistencies:* http://www.cisco.com/en/US/tech/tk389/tk621/technologies_tech_note09186a00801d11a0.shtml
- Cisco Systems Inc., *Troubleshooting Switch Port and Interface Problems:* http://www.cisco.com/en/US/products/hw/switches/ps700/products_tech_note09186a008015bfd6.shtml
- Cisco Systems Inc., *Configuring and Troubleshooting Ethernet 10/100/1000Mb Half/Full Duplex Auto-Negotiation:* http://www.cisco.com/en/US/tech/tk389/tk214/technologies_tech_note09186a0080094781.shtml
- Cisco Systems Inc., *Troubleshooting High CPU Utilization:* https://www.cisco.com/en/US/docs/switches/lan/catalyst3750/software/troubleshooting/cpu_util.html
- Cisco Systems Inc., *Cisco Catalyst 6500 Architecture White Paper:* http://www.cisco.com/en/US/prod/collateral/switches/ps5718/ps708/prod_white_paper0900aecd80673385.html
- Cisco Systems Inc., *Configuring SDM Templates:* http://www.cisco.com/en/US/docs/switches/lan/catalyst3560/software/release/12.2_46_se/configuration/guide/swsdm.html

Module Self-Check

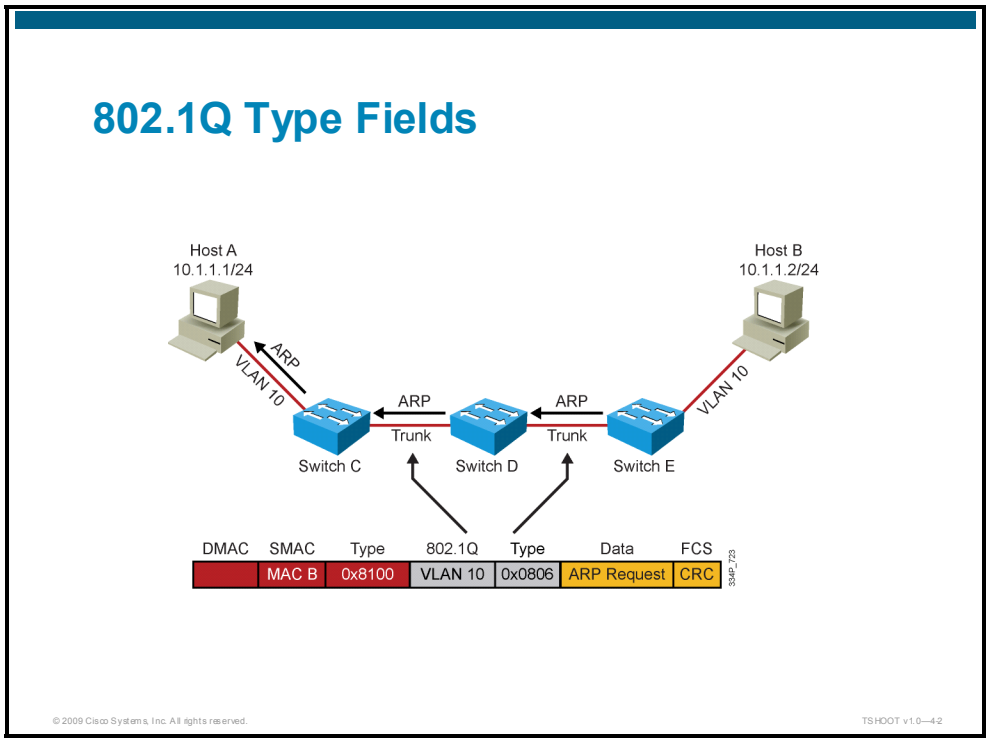
Use the questions here to review what you learned in this module. The correct answers and solutions are found in the Module Self-Check Answer Key.

Q1) The figure shows a frame containing an ARP reply from Host B in response to a request from Host A as it travels on the 802.1Q trunks between the switches. What will be the destination MAC address for that frame? (Source: Troubleshooting VLANs)



A) the MAC address of Host A
B) the MAC address of Host B
C) the broadcast MAC address ffff.ffff.ffff
D) the 801.1Q multicast MAC address 0180.C200.0000

Q2) The figure shows a frame containing an ARP reply from Host B in response to a request from Host A as it travels on the 802.1Q trunks between the switches. Which two items do the values 0x0806 and 0x8100 in the type fields represent? (Choose two.) (Source: Troubleshooting VLANs)



A) The value 0x0806 indicates that this frame is an 802.1Q frame.
B) The value 0x8100 indicates that this frame is an 802.1Q frame.
C) The value 0x0806 indicates that the data inside this frame belongs to the ARP protocol.
D) The value 0x8100 indicates that the data inside this frame belongs to the ARP protocol.

Q3) Which three items are recorded in the MAC address table of a switch? (Choose three.) (Source: Troubleshooting VLANs)

A) MAC address
B) switch port
C) IP address
D) VLAN
E) trunk or access port status
F) type

Q4) Which command combines information from the **show vlan** and **show interface trunk** commands, in addition to other VLAN-related information, such as the voice VLAN of an interface? (Source: Troubleshooting VLANs)

Q5) The _____ of a switch can provide proof that frames from a particular host have passed through the switch. (Source: Troubleshooting VLANs)

Q6) What is the correct order in which IEEE 802.1D STP evaluates the following criteria to select a root port? (Source: Troubleshooting Spanning Tree)

A) the lowest bridge ID in the received BPDUs
B) the lowest port ID in the received BPDUs
C) the lowest root path cost in the received BPDUs

_____ 1. The switch first evaluates this criterion.

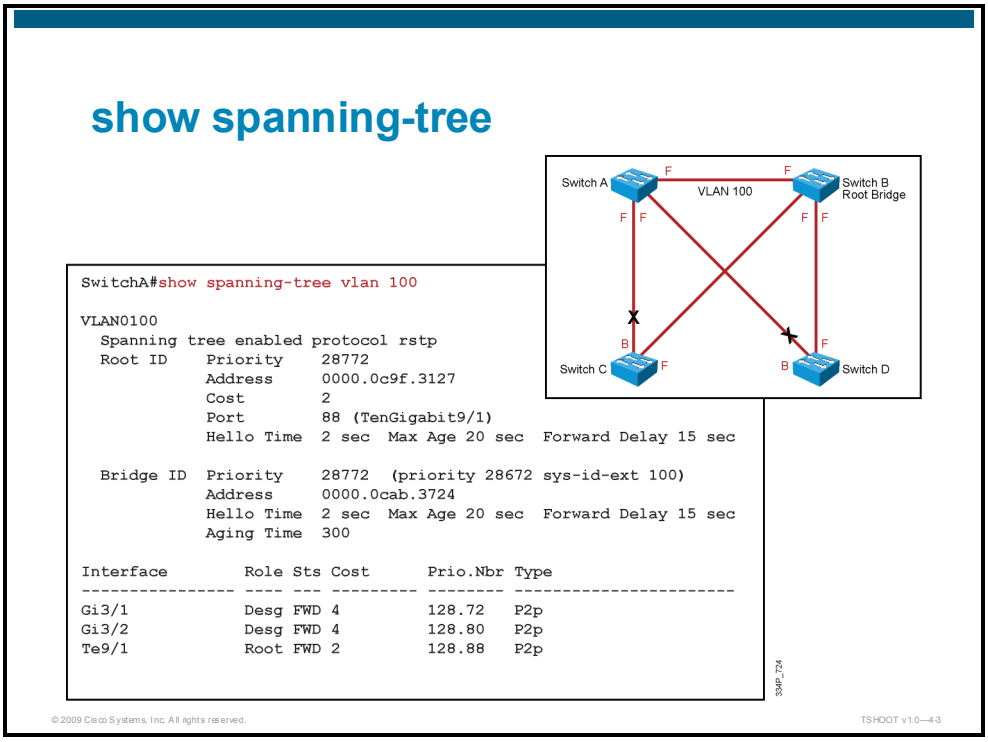
_____ 2. If the decision cannot be made based on the previous criterion, the switch evaluates this criterion.

_____ 3. If the decision cannot be made based on both previous criteria, the switch evaluates this criterion.

Q7) Which two port roles will be transitioned to the forwarding state by STP? (Choose two.) (Source: Troubleshooting Spanning Tree)

A) alternate port
B) root port
C) backup port
D) designated port

Q8) Based on the output of the **show** command in the figure, what was the reason that Switch B was preferred over Switch A during the root-bridge election process? (Source: Troubleshooting Spanning Tree)



A) Switch B has a better (lower) priority than switch A.
B) Switch B has a better port value for the 10 Gigabit Ethernet port than switch A.
C) Switch B has a better (lower) bridge ID than switch A
D) Switch B has a better (lower) cost for the 10 Gigabit Ethernet port than switch A.

Q9) Which three symptoms indicate that a bridging loop might exist in the network? (Choose three.) (Source: Troubleshooting Spanning Tree)

A) The CPU load of the switches approaches 100 percent utilization.
B) MAC addresses flap frequently between ports of the switches.
C) TTL expired messages are received by the hosts.
D) The load on the WAN links in the network approaches 100 percent utilization.
E) Layer 3 switches report frequent HSRP state changes.

Q10) What can you do to restore the network to a stable condition when you have determined that a broadcast storm might be happening on the network and you have lost all remote connectivity to the switches? (Source: Troubleshooting Spanning Tree)

Q11) Which condition could cause a %SPANTREE-2-CHNL_MISCFG log message to be generated by a switch? (Source: Troubleshooting Spanning Tree)

A) The switch has detected an inconsistency between two of its physical ports that are members of an EtherChannel.
B) The switch has detected a misconfiguration of the spanning-tree channel.
C) The switch has detected an attempt of a neighboring switch to promote itself to the designated switch on an EtherChannel link.
D) The switch has detected that the neighboring switch is not configured to bundle a number of physical interfaces in an EtherChannel, while this switch is configured to bundle the links.

Q12) Which two are differences between routers and multilayer switches? (Choose two.) (Source: Troubleshooting Switched Virtual Interfaces and Inter-VLAN Routing)

A) Routers support a wider variety of media than multilayer switches.
B) Routers can achieve higher throughput than multilayer switches.
C) Routers are less energy efficient than switches.
D) Routers support a wider range of features than multilayer switches.

Q13) Which command can be used on multilayer switches, but not on routers, to troubleshoot Layer 3 forwarding problems? (Source: Troubleshooting Switched Virtual Interfaces and Inter-VLAN Routing)

A) **show ip cef**
B) **show ip route**
C) **show adjacency**
D) **show platform**
E) None of the above. All commands necessary to troubleshoot Layer 3 forwarding are available on both routers and multilayer switches.

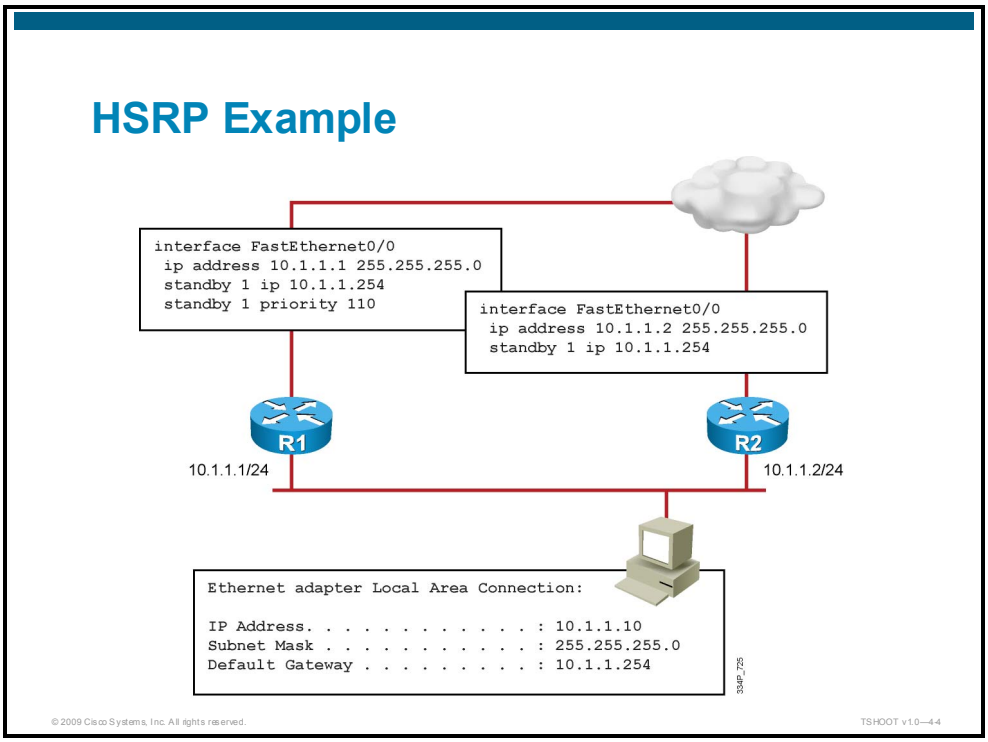
Q14) SVIs and routed ports are both capable of running the Dynamic Trunking Protocol. (Source: Troubleshooting Switched Virtual Interfaces and Inter-VLAN Routing)

A) true
B) false

Q15) Which two conditions need to be met for an SVI to be up? (Choose two.) (Source: Troubleshooting Switched Virtual Interfaces and Inter-VLAN Routing)

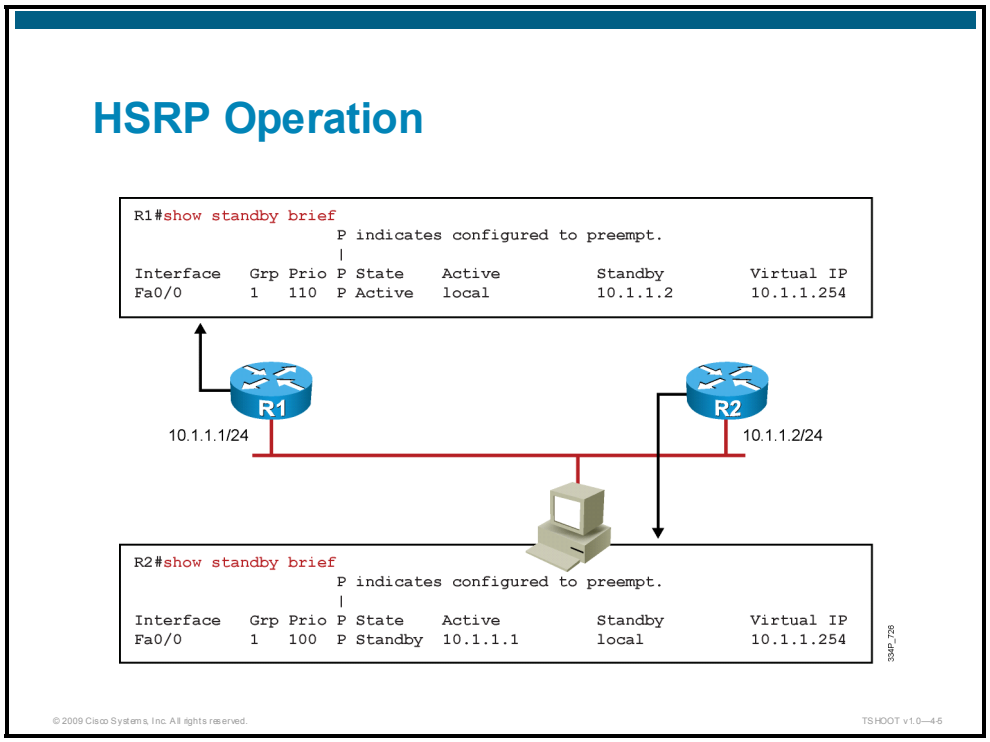
A) The VLAN corresponding to the SVI must exist in the VLAN database of the switch.
B) There must be at least one access port assigned to the VLAN corresponding to the SVI.
C) There must be at least one interface in the spanning-tree forwarding state for the VLAN corresponding to the SVI.
D) The VLAN corresponding to the SVI must have been advertised by the VTP.

Q16) Based on the configurations shown in the figure, which router do you predict will be the active router for standby group 1 and for which reason? (Source: Troubleshooting First Hop Redundancy Protocols)



A) Router R1 will be the active router, because it has the lowest IP address.
B) Router R2 will be the active router, because it has the highest IP address.
C) Router R1 will be the active router, because it has the highest priority.
D) Router R2 will be the active router, because it has the highest priority, the default priority of 200.
E) There is not enough information to answer the question. It will depend on the order in which the routers are enabled.

Q17) Based on the output of the commands displayed in the figure, which router is the active router for standby group 1 and for which reason? (Source: Troubleshooting First Hop Redundancy Protocols)



A) Router R1 will be the active router, because it has the lowest IP address.
B) Router R2 will be the active router, because it has the highest IP address.
C) Router R1 will be the active router, because it has the highest priority.
D) Router R2 will be the active router, because it has the lowest priority.
E) Router R1 will be the active router, because it was enabled first.

Q18) Which FHRP supports multiple active forwarding gateways per group? (Source: Troubleshooting First Hop Redundancy Protocols)

A) HSRP
B) VRRP
C) GLBP

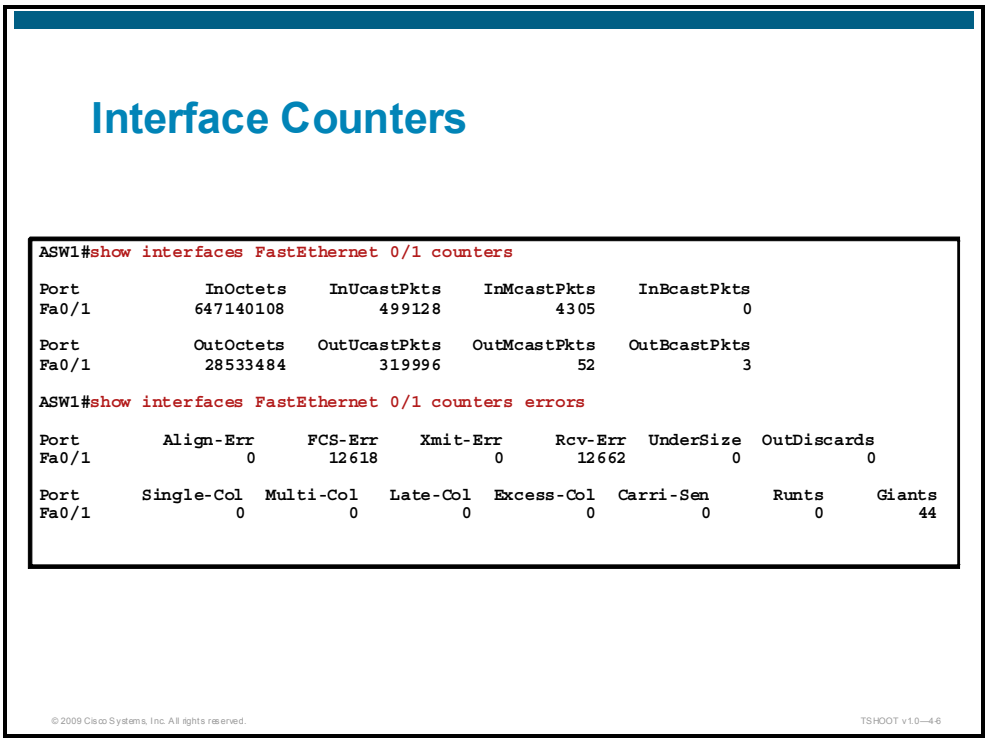
Q19) Which FHRP is based on an IETF standard? (Source: Troubleshooting First Hop Redundancy Protocols)

A) HSRP
B) VRRP
C) GLBP

Q20) For which FHRP is preemption the default behavior? (Source: Troubleshooting First Hop Redundancy Protocols)

A) HSRP
B) VRRP
C) GLBP

Q21) Based on the output of the commands displayed in the figure, which statement is true? (Source: Troubleshooting Performance Problems on Switches)



A) There is a problem, because the number of FCS errors in relation to the number of input packets is too high.
B) There is a problem, because the number of FCS errors in relation to the number of output packets is too high.
C) You cannot tell if there is a problem without knowing the duplex mode. FCS errors are normal on half-duplex Ethernet.
D) You cannot tell if there is a problem without knowing when the counters were cleared.
E) There is no problem. The number of FCS errors is within acceptable boundaries.

Q22) You have connected a switch that supports auto-MDIX to another switch that also supports auto-MDIX. A straight-through cable has been used to connect the switches, and both switches are configured for 100 Mb/s full-duplex. Will the link come up and work without problems? (Source: Troubleshooting Performance Problems on Switches)

A) yes
B) no

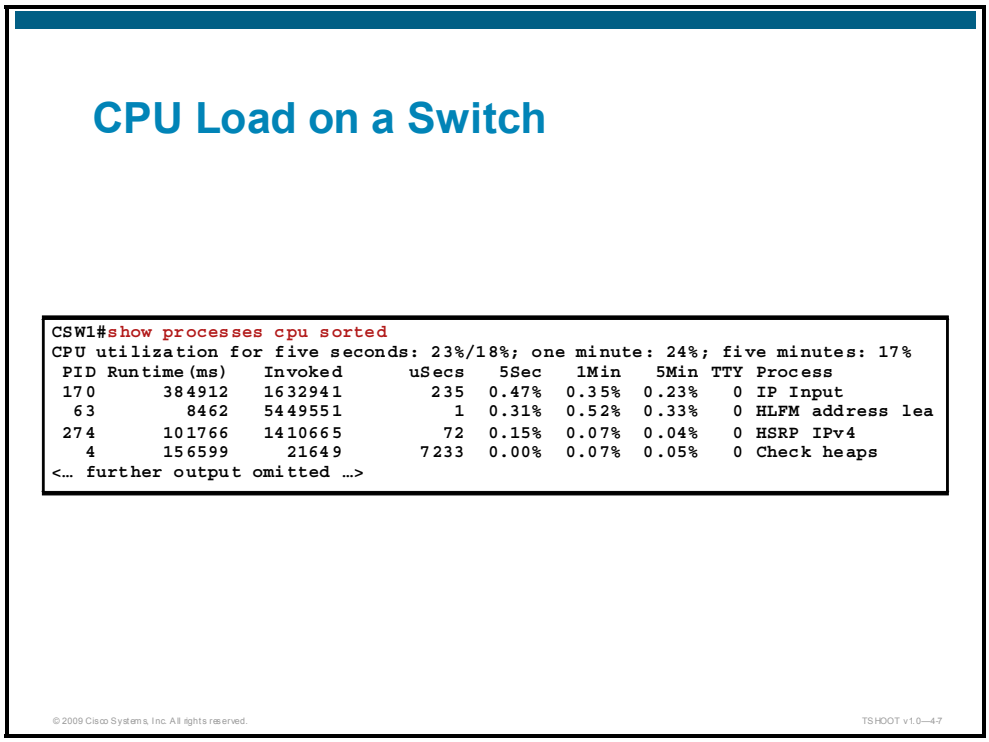
Q23) Which two hardware components can affect the switch packet forwarding performance? (Choose two.) (Source: Troubleshooting Performance Problems on Switches)

A) DRAM
B) flash
C) TCAM
D) console
E) interface hardware

Q24) Which three types of information can be programmed into a switch TCAM? (Choose three.) (Source: Troubleshooting Performance Problems on Switches)

A) IPv4 unicast routes
B) QoS access lists
C) IPS signatures
D) IPv4 multicast routes
E) NAT rules

Q25) Based on the output of the commands displayed in the figure, which statement is true? (Source: Troubleshooting Performance Problems on Switches)



A) There is a potential problem on this switch because the one-minute average should not be above 20 percent.
B) There is a potential problem on this switch, because the "IP Input" process is ranked the highest when the processes are sorted.
C) There is a potential problem on this switch because the second value in the 5-second average is higher than 10 percent.
D) You cannot determine whether there is a problem on this switch, because the baseline values are unknown.
E) There is no indication of problems on this switch.

Q26) Which three events could cause spikes in the CPU load of a switch? (Choose three.) (Source: Troubleshooting Performance Problems on Switches)

A) An engineer executes the **show tech-support** command.
B) A network management station polls the switch using SNMP to obtain the content of the routing table.
C) The switch startup configuration is copied from NVRAM to a TFTP server.
D) The OSPF process executes its SPF algorithm.
E) An engineer logs in to the switch using Telnet.

Module Self-Check Answer Key

Q1) A

Q2) B, C

Q3) A, B, D

Q4) **show interface switchport**

Q5) MAC address table

Q6) 1-C
2-A
3-B

Q7) B, D

Q8) C

Q9) A, B, E

Q10) Physically remove redundant links until all loops are eliminated from the topology.

Q11) D

Q12) A, D

Q13) D

Q14) B

Q15) A, C

Q16) E

Q17) C

Q18) C

Q19) B

Q20) B

Q21) B

Q22) B

Q23) C, E

Q24) A, B, D

Q25) C

Q26) A, B, D

