Troubleshooting Tools

This chapter presents information about the wide variety of tools available to assist you in troubleshooting your internetwork.

This chapter consists of the following sections:

- Using Router Diagnostic Commands
- Using Cisco Network Management Tools
- Third-Party Troubleshooting Tools

Using Router Diagnostic Commands

Cisco routers provide numerous integrated commands to assist you in monitoring and troubleshooting your internetwork. This section, which describes the basic use of these commands, consists of the following sections:

- Using **show** Commands
- Using **debug** Commands
- Using the **ping** Command
- Using the **trace** Command

Using show Commands

The **show** commands are powerful monitoring and troubleshooting tools. You can use the **show** commands to perform a variety of functions:

- Monitor router behavior during initial installation
- Monitor normal network operation
- Isolate problem interfaces, nodes, media, or applications
- Determine when a network is congested
- Determine the status of servers, clients, or other neighbors

Following are some of the most commonly used **show** commands:

- **show interfaces**—Displays statistics for the network interfaces

  Some of the more frequently used **show interfaces** commands include the following:

  - **show interfaces ethernet**
  - **show interfaces tokenring**
  - **show interfaces fddi**
  - **show interfaces atm**
  - **show interfaces serial**

- **show controllers**—Displays statistics for interface card controllers

  Some of the more frequently used **show controllers** commands include the following:

  - **show controllers token**
  - **show controllers cxbus**
  - **show controllers t1**

- **show running-config**— Displays the router configuration currently running
- **show startup-config**—Displays the router configuration stored in nonvolatile RAM (NVRAM)
- **show flash**—Group of commands that display the layout and contents of Flash memory
- **show buffers**—Displays statistics for the buffer pools on the router
- **show memory**—Shows statistics about the router's memory, including free pool statistics
- **show processes**—Displays information about the active processes on the router
- **show stacks**—Displays information about the stack utilization of processes and interrupt routines, as well as the reason for the last system reboot
- **show version**—Displays the configuration of the system hardware, the software version, the names and sources of configuration files, and the boot images

There are hundreds of other **show** commands available. For details on using and interpreting the output of specific **show** commands, refer to the Cisco IOS command references.

Using debug Commands

The **debug** privileged EXEC commands can provide a wealth of information about the traffic being seen (or *not* seen) on an interface, error messages generated by nodes on the network, protocol-specific diagnostic packets, and other useful troubleshooting data.

Caution Exercise care when using **debug** commands. Many of these commands are processor intensive and can cause serious network problems (such as degraded performance or loss of connectivity) if they are enabled on an already heavily loaded router. When you finish using a **debug** command, remember to disable it with its specific **no debug** command (or use the **no debug all** command to turn off all debugging).

Use debug commands to isolate problems, not to monitor normal network operation. Because the high overhead of debug commands can disrupt router operation, you should use debug commands only when you are looking for specific types of traffic or problems and have narrowed your problems to a likely subset of causes.

Output formats vary with each **debug** command. Some generate a single line of output per packet, and others generate multiple lines of output per packet. Some generate large amounts of output, and others generate only occasional output. Some generate lines of text, and others generate information in field format.

To minimize the negative impact of using **debug** commands, follow this procedure:

Step 1 Use the **no logging console** global configuration command on your router. This command disables all logging to the console terminal.

Step 2 Telnet to a router port and enter the **enable** EXEC command.

Step 3 Use the **terminal monitor** command to copy **debug** command output and system error messages to your current terminal display.

This permits you to view **debug** command output remotely, without being connected through the console port.

Following this procedure minimizes the load created by using **debug** commands because the console port no longer has to generate character-by-character processor interrupts.

If you intend to keep the output of the **debug** command, spool the output to a file. The procedure for setting up such a **debug** output file is described in the *Debug Command Reference* publication.

This publication refers to specific **debug** commands that are useful when troubleshooting specific problems. Complete details regarding the function and output of **debug** commands are provided in the *Debug Command Reference* publication.

In many situations, third-party diagnostic tools can be more useful and less intrusive than using **debug** commands. For more information, see the section, “Third-Party Troubleshooting Tools,” later in this chapter.

Using the ping Command

To check host reachability and network connectivity, use the **ping** EXEC (user) or privileged EXEC command. This command can be used to confirm basic network connectivity on AppleTalk, ISO CLNS, IP, Novell, Apollo, VINES, DECnet, or XNS networks.

For IP, the **ping** command sends ICMP Echo messages. If a station receives an ICMP Echo message, it sends an ICMP Echo Reply message back to the source.

The extended command mode of the **ping** command permits you to specify the supported IP header options. This allows the router to perform a more extensive range of test options. To enter **ping** extended command mode, enter **yes** at the extended commands prompt of the **ping** command.

It is a good idea to use the **ping** command when the network is functioning properly to see how the command works under normal conditions and so you have something to compare against when troubleshooting.

For detailed information on using the **ping** and extended **ping** commands, refer to the Cisco IOS *Configuration Fundamentals Command Reference*.

Using the trace Command

The **trace** user EXEC command discovers the routes a router's packets follow when traveling to their destinations. The **trace** privileged EXEC command permits the supported IP header options to be specified, allowing the router to perform a more extensive range of test options.

The **trace** command works by using the error message generated by routers when a datagram exceeds its time-to-live (TTL) value. First, probe datagrams are sent with a TTL value of one. This causes the first router to discard the probe datagrams and send back "time exceeded" error messages. The **trace** command then sends several probes, and displays the round-trip time for each. After every third probe, the TTL is increased by one.

Each outgoing packet can result in one of two error messages. A "time exceeded" error message indicates that an intermediate router has seen and discarded the probe. A "port unreachable" error message indicates that the destination node has received the probe and discarded it because it could not deliver the packet to an application. If the timer goes off before a response comes in, **trace** prints an asterisk (*).

The **trace** command terminates when the destination responds, when the maximum TTL is exceeded, or when the user interrupts the trace with the escape sequence.

As with **ping**, it is a good idea to use the **trace** command when the network is functioning properly to see how the command works under normal conditions and so you have something to compare against when troubleshooting.

For detailed information on using the **trace** and extended **trace** commands, refer to the Cisco IOS *Configuration Fundamentals Command Reference*.

Using Cisco Network Management Tools

Cisco offers several network management products that provide design, monitoring, and troubleshooting tools to help you manage your internetwork.

The following three internetwork management tools are useful for troubleshooting internetwork problems:

- CiscoWorks Internetwork Management Software
- TrafficDirector RMON Application
- VlanDirector Switch Management Application

CiscoWorks Internetwork Management Software

CiscoWorks is a series of SNMP-based internetwork management software applications. CiscoWorks applications are integrated on several popular network management platforms and build on industry-standard platforms to provide applications for monitoring device status, maintaining configurations, and troubleshooting problems.

Following are some of the applications included in the CiscoWorks product that are useful for troubleshooting your internetwork:

- Device Monitor—Monitors specific devices for environmental and interface information
- Health Monitor—Displays information about the status of a device, including buffers, CPU load, memory available, and protocols and interfaces being used
- Show Commands—Enables you to view data similar to output from router **show** EXEC commands

- Path Tool—Displays and analyzes the path between two devices to collect utilization and error data
- Device Polling—Probes and extracts data about the condition of network devices
- CiscoView—Provides dynamic monitoring and troubleshooting functions, including a graphical display of Cisco devices, statistics, and comprehensive configuration information
- Offline Network Analysis—Collects historical network data for offline analysis of performance trends and traffic patterns
- CiscoConnect—Allows you to provide Cisco with debugging information, configurations, and topology information to speed resolution of network problems

CiscoWorks implements numerous other applications that are useful for administering, designing, and monitoring your internetwork. Refer to the *Cisco Systems Product Catalog* for more information.

TrafficDirector RMON Application

The TrafficDirector Remote Monitoring (RMON) console application gathers and displays information from RMON agents, providing a view of network activity and calling attention to potential problems.

Data can be captured from any remote LAN segment, ring, or switch link to assist in troubleshooting problems. TrafficDirector’s protocol analysis tool supports full seven-layer decodes for the AppleTalk, DECnet, IP, OSI, Novell, SNA, Sun NFS, Banyan VINES, and XNS protocol suites.

VlanDirector Switch Management Application

The VlanDirector switch management application simplifies VLAN port assignment and offers other management capabilities for VLANs. VlanDirector offers the following features for network administrators:

- Accurate representation of the physical network for VLAN design and configuration verification
- Capability to obtain VLAN configuration information on a specific device or link interface
- Discrepancy reports on conflicting configurations
- Ability to troubleshoot and identify individual device configurations that are in error with system-level VLANs
- Quick detection of changes in VLAN status of switch ports
- User authentication and write protection security

Third-Party Troubleshooting Tools

In many situations, third-party diagnostic tools can be more useful than commands that are integrated into the router. For example, enabling a processor-intensive **debug** command can be disastrous in an environment experiencing excessively high traffic levels. However, attaching a network analyzer to the suspect network is less intrusive and is more likely to yield useful information without interrupting the operation of the router.

The following are some typical third-party troubleshooting tools used for troubleshooting internetworks:

- Volt-Ohm Meters, Digital Multimeters, and Cable Testers
- TDRs and OTDRs
- Breakout Boxes, Fox Boxes, and BERTs/BLERTs
- Network Monitors
- Network Analyzers

Volt-Ohm Meters, Digital Multimeters, and Cable Testers

Volt-ohm meters and digital multimeters are at the lower end of the spectrum of cable testing tools. These devices measure parameters such as AC and DC voltage, current, resistance, capacitance, and cable continuity. They are used to check physical connectivity.

Cable testers (scanners) also enable you to check physical connectivity. Cable testers are available for shielded twisted pair (STP), unshielded twisted pair (UTP), 10BaseT, and coaxial and twinax cables. A given cable tester might be able to perform any of the following functions:

- Test and report on cable conditions, including near-end crosstalk (NEXT), attenuation, and noise
- Perform time domain reflectometer (TDR), traffic monitoring, and wire map functions
- Display media access control (MAC) layer information about LAN traffic, provide statistics such as network utilization and packet error rates, and perform limited protocol testing (for example, TCP/IP tests such as **ping**)

Similar testing equipment is available for fiber-optic cable. Due to the relatively high cost of fiber cable and its installation, fiber-optic cable should be tested both before installation (on-the-reel testing) and after installation. Continuity testing of the fiber requires either a visible light source or a reflectometer. Light sources capable of providing light at the three predominant wavelengths, 850 nanometers (nm), 1300 nm, and 1550 nm, are used with power meters that can measure the same wavelengths and test attenuation and return loss in the fiber.

TDRs and OTDRs

At the top end of the cable testing spectrum are time domain reflectometers (TDRs). These devices can quickly locate open and short circuits, crimps, kinks, sharp bends, impedance mismatches, and other defects in metallic cables.

A TDR works by “bouncing” a signal off the end of the cable. Opens, shorts and other problems reflect the signal back at different amplitudes, depending on the problem. A TDR measures how much time it takes for the signal to reflect and calculates the distance to a fault in the cable. TDRs can also be used to measure the length of a cable. Some TDRs can also calculate the propagation rate based on a configured cable length.

Fiber-optic measurement is performed by an optical time domain reflectometer (OTDR). OTDRs can accurately measure the length of the fiber, locate cable breaks, measure the fiber attenuation, and measure splice or connector losses. An OTDR can be used to take the “signature” of a particular installation, noting attenuation and splice losses. This baseline measurement can then be compared with future signatures when a problem in the system is suspected.

Breakout Boxes, Fox Boxes, and BERTs/BLERTs

Breakout boxes, fox boxes, and bit/block error rate testers (BERTs/BLERTs) are digital interface testing tools used to measure the digital signals present at PCs, printers, modems and CSU/DSUs, and other peripheral interfaces. These devices can monitor data line conditions, analyze and trap data, and diagnose problems common to data communication systems. Traffic from data terminal equipment (DTE) through data communications equipment (DCE) can be examined to help isolate problems, identify bit patterns, and ensure that the proper cabling has been installed. These devices cannot test media signals such as Ethernet, Token Ring, or FDDI.

Network Monitors

Network monitors continuously track packets crossing a network, providing an accurate picture of network activity at any moment, or a historical record of network activity over a period of time. They do not decode the contents of frames. Monitors are useful for baselining, in which the activity on a network is sampled over a period of time to establish a normal performance profile, or baseline.

Monitors collect information such as packet sizes, the number of packets, error packets, overall usage of a connection, the number of hosts and their MAC addresses, and details about communications between hosts and other devices. This data can be used to create profiles of LAN traffic as well as assist in locating traffic overloads, planning for network expansion, detecting intruders, establishing baseline performance, and distributing traffic more efficiently.

Network Analyzers

A network analyzer (also called a protocol analyzer) decodes the various protocol layers in a recorded frame and presents them as readable abbreviations or summaries, detailing which layer is involved (physical, data link, and so forth) and what function each byte or byte content serves.

Most network analyzers can perform many of the following functions:

- Filter traffic that meets certain criteria so that, for example, all traffic to and from a particular device can be captured
- Time-stamp captured data
- Present protocol layers in an easily readable form
- Generate frames and transmit them onto the network
- Incorporate an “expert” system in which the analyzer uses a set of rules, combined with information about the network configuration and operation, to diagnose and solve, or offer potential solutions, to network problems