CHAPTER 5

Managing the System

This chapter describes the basic tasks that you can do to manage the general system (or nonprotocol-specific) features. Our system management features are supported via the Simple Network Management Protocol (SNMP). This chapter describes the tasks needed to configure SNMP support on the router. A part of SNMP is the Management Information Base (MIB). MIBs provide variables that can be set or read to change parameters or provide information on network devices and interfaces. Cisco supports several MIBs, including the Internet standard MIB II, and also provides its own Cisco MIB. For information on the Cisco MIB, see the *Cisco Management Information Base (MIB) User Quick Reference*.

Cisco Systems also provides CiscoWorks, a feature-rich network management software product that is integrated with Sun Microsystems' SunNet Manager product running on a Sun SPARCstation platform. CiscoWorks provides a graphical user interface that is menu driven and provides support for all five areas of network management. (See the next section, "Understanding System Management," for details.) With CiscoWorks, you can create network maps and set up automated network performance monitors and fault tests, such as pinpointing connectivity problems. Test results can be displayed in several graph formats. For more information, see the *CiscoWorks User Guide*.

The Cisco Configuration Builder application, based on an MS-Windows graphical user interface, also streamlines the process of configuring Cisco routers. The Cisco Configuration Builder offers a guided configuration capability, so that as you complete the first configuration step, you are automatically led through the correct order of remaining steps needed to complete configuration of each feature or protocol. In addition, the Cisco Configuration Builder provides an on-line help system for easy access to information. See the *Cisco Configuration Builder Getting Started Guide* for more information.

For a list of recommended books on network management, refer to Appendix A in the *Router Products Command Reference* publication.

For a complete description of the commands mentioned in this chapter, refer to the "System Management Commands" chapter of the *Router Products Command Reference* publication.

Understanding System Management

This chapter describes the tasks you can perform to manage the router and its performance on the network. In general, system or network management falls into the following categories. The categories are described in this chapter unless specified otherwise.

- Configuration management (page 5-3)

  The configuration of network devices determines the network's behavior. To manage device configurations, you need to list and compare configuration files on running devices, store configuration files on network servers for shared access, and perform software installations and upgrades. These configuration management tasks are described in the chapter "Loading System Images, Microcode Images, and Configuration Files."

  Other configuration management tasks include naming the router, setting router time services, and configuring SNMP support. These tasks are described in this chapter.

- Security management (page 5-15)

  To manage security on the network, you need to restrict access to the system. You can do so on several different levels.

  — You can assign passwords (and encrypt them) to restrict access to router terminal lines, login connections, or privileged EXEC mode.

  — You can establish Terminal Access Controller Access Control System (TACACS) protection for network servers that have shared access.

  — You can restrict login connections to specific users with a username authentication system.

  — You can control access on serial interfaces with Challenge Handshake Authentication Protocol (CHAP).

  — You can create access lists to filter traffic to and from specific destinations. Subsequent chapters that describe the routing protocols in detail define access lists. This section provides general guidelines for creating access lists.

  — You can create security labels for Internet Protocol (IP) datagrams using the Internet Protocol Security Option (IPSO), as described in the chapter "Configuring IP."

- Fault management (page 5-24)

  To manage network faults, you need to discover, isolate, and fix the problems. You can discover problems with the system's monitoring commands, isolate problems with the system's test commands, and resolve problems with other commands, including **debug**.

  This section introduces basic fault management commands. For detailed troubleshooting procedures and a variety of scenarios, see the *Troubleshooting Internetworking Systems* publication. For complete details on all **debug** commands, see the *Debug Command Reference* publication.

- System performance management (page 5-33)

  To manage system performance, you need to monitor and determine response time, error rates, and availability. Once these factors are determined, you can perform load balancing and modify system parameters to enhance performance. For example, priority queuing allows you to prioritize traffic order. You can configure fast and autonomous switching to improve network throughput, as described in the "Configuring Interfaces" chapter of this manual.

  See the *Internetwork Design Guide* for additional information.

- Accounting management (page 5-39)

  Accounting management allows you to track both individual and group usage of network resources. You can then reallocate resources as needed. For example, you can change the system timers and configure TCP keepalives. See also the IP accounting feature in the "Configuring IP" chapter of this manual.

Configuration Management

You can complete any of the following tasks to perform configuration management functions:

- Set the router name
- Set the router time services
- Monitor time services
- Configure SNMP support

The following sections summarize these tasks. Other configuration management tasks are described in Chapter 3.

Set the Router Name

One of the first basic tasks is to name your router. The name of the router is considered the host name and is the name that is displayed by the system prompt. If no name is configured, the system default router name is *Router*. You can name the router while in global configuration mode as follows:

Task	Command
Set the host name.	**hostname** *name*

Set the Router Time Services

All of our router and communication server products provide an array of time-of-day services. These services allow the products to keep track of the current time and date to a high degree of accuracy, to synchronize multiple products to the same time, and to provide time services to other systems.

The heart of the time service is the system clock. This clock runs from the moment the system starts up and keeps track of the current date and time. The system clock can be set from a number of sources, and in turn can be used to distribute the current time through various mechanisms to other systems. When the system is initialized, the system clock is set based on the time in the Cisco 7000 hardware; on other router models, the system clock it is set to midnight on March 1, 1993. The system clock can then be set from the following sources:

- Network Time Protocol (NTP)
- VINES Time Service
- Manual configuration

The system clock can provide time to the following services:

- Network Time Protocol
- VINES Time Service
- User **show** commands
- logging and debugging messages

The system clock internally keeps track of time based on Coordinated Universal Time (UTC), also known as Greenwich Mean Time (GMT). You can configure information about the local time zone and summer time (daylight savings time) so that the time is displayed correctly relative to the local time zone.

The system clock keeps track of whether the time is "authoritative" or not (that is, whether it has been set by a time source considered to be authoritative). If it is not authoritative, the time will be available only for display purposes and will not be redistributed.

Network Time Protocol

The Network Time Protocol (NTP) is a protocol designed to time-synchronize a network of machines. NTP runs over UDP, which in turn runs over IP. NTP is documented in RFC 1305.

An NTP network usually gets its time from an authoritative time source, such as a radio clock or an atomic clock attached to a time server. NTP then distributes this time across the network. NTP is extremely efficient; no more than one packet per minute is necessary to synchronize two machines to within a millisecond of one another.

NTP uses the concept of a "stratum" to describe how many NTP "hops" away a machine is from an authoritative time source. A "stratum 1" time server has a radio or atomic clock directly attached, a "stratum 2" time server receives its time via NTP from a "stratum 1" time server, and so on. A machine running NTP will automatically choose as its time source the machine with the lowest stratum number that it is configured to communicate with via NTP. This strategy effectively builds a self-organizing tree of NTP speakers.

NTP is careful to avoid synchronizing to a machine whose time may not be accurate. It avoids doing so in two ways. First of all, NTP will never synchronize to a machine that is not in turn synchronized itself. Secondly, NTP will compare the time reported by several machines, and will not synchronize to a machine whose time is significantly different than the others, even if its stratum is lower.

The communications between machines running NTP (known as "associations") are usually statically configured; each machine is given the IP address of all machines with which it should form associations. Accurate timekeeping is made possible by exchanging NTP messages between each pair of machines with an association. However, in a LAN environment, NTP may be configured to use IP broadcast messages instead. This alternative reduces configuration complexity, because each machine can simply be configured to send or receive broadcast messages. However, the accuracy of timekeeping is marginally reduced, because the information flow is one-way only.

The time kept on a machine is a critical resource, so we strongly recommend that you use the security features of NTP to avoid the accidental or malicious setting of incorrect time. Two mechanisms are available: an access list-based restriction scheme and an encrypted authentication mechanism.

Our implementation of NTP does not support stratum 1 service; in other words, it is not possible to connect a radio or atomic clock to this router. It is recommended that time service for your network be derived from the public NTP servers available in the IP Internet. If the network is isolated from the Internet, our implementation of NTP allows a machine to be configured so that it acts as though it is synchronized via NTP, when in fact it has determined the time using other means. Other machines will then synchronize to that machine via NTP.

When multiple sources of time (VINES, Cisco 7000 calendar, manual configuration) are available, NTP is always considered to be more authoritative. NTP time will override the time set by any other method.

A number of manufacturers include NTP software for their host systems, and a publicly available version for systems running UNIX and its various derivatives is also available. This software allows host systems to be time-synchronized as well.

VINES Time Service

Time service is also available when Banyan VINES is configured. This protocol is a standard part of VINES. Our implementation allows the VINES time service to be used in two ways. First, if the system has learned the time from some other source, it can act as a VINES time server and provide time to other machines running VINES. It also can use the VINES time service to set the system clock if no other form of time service is available.

Cisco 7000 Calendar

The Cisco 7000 contains a battery-powered calendar system that tracks the date and time across system restarts and power outages. This calendar system is always used to initialize the system clock when the system is restarted. It can also be considered to be an authoritative source of time and be redistributed via NTP or VINES time service if no other source is available. Furthermore, if NTP is running, the Cisco 7000 calendar can be periodically updated from NTP, compensating for the inherent drift in the calendar time.

Configure NTP

NTP services are enabled on all interfaces by default. There are a number of optional subtasks you could perform:

- Configure NTP authentication
- Configure NTP associations
- Configure NTP broadcast service
- Configure NTP access restrictions
- Configure the source IP address for NTP packets
- Configure the system as an authoritative NTP server
- Configure NTP to update the Cisco 7000 calendar

Configure NTP Authentication

If you want to authenticate the associations with other systems for security purposes, perform the tasks that follow. The first task enables the NTP authentication feature. The second task defines each of the authentication keys. Each key has a key number, a type, and a value. Currently the only key type supported is **md5**. Third, a list of "trusted" authentication keys is defined. If a key is trusted, then this system will be willing to synchronize to a system that uses this key in its NTP packets.

To configure NTP authentication, perform the following tasks in global configuration mode:

Task		Command
Step 1	Enable the NTP authentication feature.	**ntp authenticate**
Step 2	Define the authentication keys.	**ntp authentication-key** *number* **md5** *value*
Step 3	Define trusted authentication keys.	**ntp trusted-key** *number*

Configure NTP Associations

If you want to form an NTP association with another system, perform the task in global configuration mode:

Task	Command
Form an NTP association with another system.	**ntp peer** *ip address* [**version** *number*] [**key** *keyid*] [**source** *interface*] [**prefer**]
	or
	ntp server *ip address* [**version** *number*] [**key** *keyid*] [**source** *interface*] [**prefer**]

The association can be a peer association (meaning that this system is willing to either synchronize to the other system or to allow the other system to synchronize to it), or it can be a server association (meaning that this system will only synchronize to the other system, and not the other way around). Note that only one end of an association needs to be configured; the other system will automatically establish the association.

See an example of the **ntp server** command at the end of this chapter.

Configure NTP Broadcast Service

The system can either send broadcast packets or listen to them on an interface-by-interface basis. The estimated round-trip delay for broadcast packets can also be configured. Perform one or more of these tasks in global configuration mode if you want to use NTP's broadcast feature:

Task	Command
Send NTP broadcast packets.	**ntp broadcast** [**version** *number*] [**key** *nn*]
Receive NTP broadcast packets.	**ntp broadcast client**
Adjust estimated delay.	**ntp broadcastdelay** *delay*

See an example of the **ntp broadcast** command at the end of this chapter.

Configure NTP Access Restrictions

You can control NTP access on two levels by completing the following tasks:

- Create an access group and assign a basic IP access list to it
- Disable NTP services on specific interfaces

Create an Access Group and Assign a Basic IP Access List to It

To control access to NTP services, you can create an NTP access group and apply a basic IP access list to it. To do so, perform the following task in global configuration mode:

Task	Command
Create an access group and apply a basic IP access list to it.	**ntp access-group** {**query-only** | **serve-only** | **serve** | **peer**} *number*

The access group options are scanned in the following order from least restrictive to most restrictive:

1 Peer

Allows time requests and NTP control queries and allows the system to synchronize itself to a system whose address passes the access list criteria.

2 Serve

Allows time requests and NTP control queries, but does not allow the system to synchronize itself to a system whose address passes the access list criteria.

3 Serve-only

Allows only time requests from a system whose address passes the access list criteria.

4 Query-only

Allows only NTP control queries from a system whose address passes the access list criteria.

If the source IP address matches the access lists for more than one access type, the first type is granted. If no access groups are specified, all access types are granted to all systems. If any access groups are specified, only the specified access types will be granted.

For details on NTP control queries, see RFC 1305 (NTP version 3).

Disable NTP Services on a Specific Interface

NTP services are enabled on all interfaces by default. You can disable NTP packets from being received through an interface by performing the following task in interface configuration mode:

Task	Command
Disable NTP services on a specific interface.	**ntp disable**

Configure the Source IP Address for NTP Packets

When the system sends an NTP packet, the source IP address is normally set to the address of the interface through which the NTP packet is sent. Perform the following task in global configuration mode if you want to configure a specific interface from which the IP source address will be taken:

Task	Command
Configure an interface from which the IP source address will be taken.	**ntp source** *interface*

This interface will be used for the source address for all packets sent to all destinations. If a source address is to be used for a specific association, use the **source** parameter on the **ntp peer** or **ntp server** command shown earlier in this chapter.

Configure the System as an Authoritative NTP Server

Perform the following task in global configuration mode if you want the system to be an authoritative NTP server, even if the system is not synchronized to an outside time source:

Task	Command
Make the system an authoritative NTP server.	**ntp master** [*stratum*]

See an example of the **ntp master** command at the end of this chapter.

> **Caution** Use this command with **extreme** caution. It is very easy to override valid time sources using this command, especially if a low stratum number is configured. Configuring multiple machines in the same network with the **ntp master** command can cause instability in timekeeping if the machines do not agree on the time.

See an example of the **ntp master** command at the end of this chapter.

Configure NTP to Update the Cisco 7000 Calendar

Perform the following task in global configuration mode if the system is synchronized to an outside time source via NTP and you want the Cisco 7000 calendar to be periodically synchronized to NTP time:

Task	Command
Configure NTP to update the Cisco 7000 calendar.	**ntp update-calendar**

See an example of the **ntp update-calendar** command at the end of this chapter.

Configure VINES Time Service

Perform the following task in global configuration mode if you want to distribute the system clock to other VINES systems:

Task	Command
Distribute the system clock to other VINES systems.	**vines time use-system**

To receive VINES time service to control the system clock, perform the following task in global configuration mode:

Task	Command
Receive VINES time service.	**vines time set-system**

The two preceding commands are described in Chapter 13 of the *Router Products Command Reference* publication.

Configure Time and Date Manually

If no other source of time is available, you can manually configure the current time and date after the system is restarted. The time will remain accurate until the next system restart. We recommend that you use manual configuration only as a last resort.

To set up time services, complete the following tasks. If you have an outside source to which the router can synchronize, you do not need to manually set the system clock.

- Configure the time zone
- Configure summer time (daylight savings time) if applicable
- Set the system clock (if no other time source is available)
- Set the calendar on a Cisco 7000

Configure the Time Zone

Complete the following task in global configuration mode to manually configure the time zone used by the router:

Task	Command
Set the router time zone.	**clock timezone** *name hours* [*minutes*]

See an example of the **clock timezone** command at the end of this chapter.

Configure Summer Time

To configure summer time (daylight savings time) in areas where it starts and ends on a particular day of the week each year, use the following form of the command in global configuration mode:

Task	Command
Configure summer time.	**clock summer-time** *name* **recurring** [*week-number day month hh:mm week-number day month hh:mm* [*offset*]]

If summer time in your area does not follow this pattern, you can configure the exact date and time of the next summer time events using one of the following commands in global configuration mode:

Task	Command
Configure summer time.	**clock summer-time** *name* **date** *month day year hh:mm month day year hh:mm* [*offset*] or **clock summer-time** *name* **date** *day month year hh:mm day month year hh:mm* [*offset*]

See an example of the **clock summer-time** command at the end of this chapter.

Set the System Clock

If you have an outside source on the network that provides time services (such as an NTP server or VINES time service), you do not need to manually set the system clock.

However, if you have do not have any time service source, complete the following task in EXEC mode to set the system clock:

Task	Command
Set the system clock.	**clock set** *hh:mm:ss day month year* or **clock set** *hh:mm:ss month day year*

Set the Cisco 7000 Calendar

In addition to a system clock, the Cisco 7000 hardware provides a system calendar that can set the system time and control the system clock, as well as enable the Cisco 7000 to act as a time service for the network.

You can complete the following tasks to enable the Cisco 7000 calendar capabilities:

- Set the system calendar
- Set the Cisco 7000 as a network time source
- Set the system clock from the calendar
- Set the calendar from the system clock

Set the System Calendar

The Cisco 7000 calendar maintains time separately from the system clock. It continues to run when the system is restarted or power is turned off. Typically, it will only need to be manually set once, when the system is first installed. If time is available from an external source using NTP, the calendar can be updated from the system clock instead.

If you do not have an external time source, complete the following task in EXEC mode to set the system calendar:

Task	Command
Set the Cisco 7000 calendar.	**calendar set** *hh:mm:ss day month year* or **calendar set** *hh:mm:ss month day year*

Set the Cisco 7000 as a Network Time Source

Although the system clock is always initialized from the Cisco 7000 calendar when the system is restarted, by default it is not considered to be authoritative and so will not be redistributed with NTP or VINES Time Service. To make the Cisco 7000 calendar be authoritative, complete this task:

Task	Command
Enable the Cisco 7000 to act as a valid time source to which network peers can synchronize.	**clock calendar-valid**

See an example of the **clock calendar-valid** command at the end of this chapter.

Set the System Clock from the Cisco 7000 Calendar

You can set the system clock to the new calendar setting by completing the following task in global configuration mode:

Task	Command
Set the system clock from the calendar.	**clock read-calendar**

Set the Cisco 7000 Calendar from the System Clock

You can update the calendar with the new clock setting by performing the following task in EXEC mode:

Task	Command
Set the calendar from the system clock.	**clock update-calendar**

Monitor Time Services

You can monitor clock, calendar, and NTP EXEC services by completing the following tasks in EXEC mode:

Task	Command
Display the current 7000 calendar time.	**show calendar**
Display the current system clock time.	**show clock** [**detail**]
List NTP statistics.	**show ntp associations** [**detail**] or **show ntp status**

Configure Simple Network Management Protocol (SNMP) Support

The Simple Network Management Protocol (SNMP) provides a way for network management client and server applications to communicate. It does this by providing a message format for sending information between an SNMP manager and an SNMP agent.

The SNMP agent contains Management Information Base (MIB) variables that the SNMP manager can request or change. The SNMP agent can also send traps, or messages alerting the SNMP manager to a condition on the network. Traps can indicate improper user authentication, restarts, link status (up or down), closing of a TCP connection, or loss of connection to a neighbor router.

Our implementation of SNMP supports all MIB II variables (as described in RFC 1213 and SNMP traps (as described in RFC 1215). Cisco also supports the definition of management information described in RFCs 1155, 1157, and 1213, and supports some or all variables in the MIBs described in the following RFCs: 1156, 1212, 1231, 1285, 1286, 1315, 1381, and 1382.

Cisco also provides its own MIB with every system. With the current software release, the Cisco MIB provides a new chassis MIB variable that enables the SNMP manager to gather data on system card descriptions, serial numbers, hardware and software revision levels, and slot locations.

See the *Cisco Management Information Base (MIB) User Quick Reference* for a detailed description of each Cisco MIB variable and SNMP trap.

You can perform the following tasks to configure SNMP support on the router:

- Enable SNMP and define access control
- Define SNMP trap operations
- Define the maximum SNMP packet size
- Enable the SNMP server shutdown mechanism
- Establish the contact, location, and serial number for the SNMP server
- Disable the SNMP server
- Monitor SNMP status

These tasks are described in the following sections.

Enable SNMP and Define Access Control

You can enable SNMP server operation and specify which hosts can send requests to the router by performing the following tasks in global configuration mode:

Task		Command
Step 1	Enable the SNMP server and define the community access string.	**snmp-server community** [*string* [**RO**	**RW**] [*list*]]
Step 2	Specify the access list that determines which hosts can send requests to the network server.	**snmp-server access-list** *list*

Define SNMP Trap Operations

The SNMP trap operations allow a system administrator to configure the router to send information to a network management application when a particular event occurs. You can specify the following features for SNMP server trap operations:

- Source interface
- Recipient
- Trap operation authentication
- Retransmission interval
- Message (packet) queue length for each trap host

Perform these tasks in global configuration mode, as needed, to define traps for your system configuration:

Task	Command
Specify the source interface (and hence IP address) of the trap message.	**snmp-server trap-source** *interface*
Specify the recipient of the trap message.	**snmp-server host** *address community-string* [**snmp**] [**tty**]
Establish trap message authentication.	**snmp-server trap-authentication**
Define how often to resend trap messages on the retransmission queue.	**snmp-server trap-timeout** *seconds*
Establish the message queue length for each trap host.	**snmp-server queue-length** *length*

Define the Maximum SNMP Packet Size

You can set the maximum packet size permitted when the SNMP server is receiving a request or generating a reply. To do so, perform the following task in global configuration mode:

Task	Command
Establish the maximum packet size.	**snmp-server packetsize** *bytes*

Enable the SNMP Server Shutdown Mechanism

Using SNMP packets, a network management tool can send messages to users on virtual terminals and the console. This facility operates in a similar fashion to the EXEC **send** command; however, the SNMP request that causes the message to be issued to the users also specifies the action to be taken after the message is delivered. One possible action is a shutdown request. Because the ability to cause a reload from the network is a powerful feature, it is protected by this global configuration command:

Task	Command
Use this SNMP message reload feature and request a system shutdown message.	**snmp-server system-shutdown**

To understand how to use this feature with SNMP requests, read the document *mib.txt* available by anonymous **ftp** from *ftp.cisco.com.* See the beginning of this chapter for more information.

Establish the Contact, Location, and Serial Number of the SNMP Server

You can set the system contact, location, and serial number of the SNMP server so that these descriptions can be accessed via the configuration file. To do so, perform the following tasks in global configuration mode:

Task	Command
Set the system contact string.	**snmp-server contact** *text*
Set the system location string.	**snmp-server location** *text*
Set the system serial number.	**snmp-server chassis-id** *text*

Disable the SNMP Server

Once the SNMP server has been enabled with the **snmp-server community** command, you can specifically disable it by performing the following task in global configuration mode:

Task	Command
Disable SNMP server operation.	**no snmp-server**

Monitor SNMP Status

To monitor SNMP input and output statistics, including number of illegal community string entries, errors, requested variables, and so on, complete the following task in EXEC mode:

Task	Command
Monitor SNMP status.	**show snmp**

Security Management

To set up security features, you need to identify sensitive information, find the network access points to that information, secure these access points, and maintain the secure access points.

This section describes the following optional tasks used to control access to the system:

- Establish password protection
- Disable password protection
- Recover a lost password
- Create access lists
- Establish Terminal Access Controller Access Control System (TACACS) protection schemes
- Establish username authentication
- Enable Challenge Handshake Authentication Protocol (CHAP)
- Enable Password Authentical Protocol (PAP)

Other chapters in this guide provide information on protocol-specific security features. Chapter 6 provides information on CHAP, an additional authentication feature. Another example is the IP Security Option (IPSO) feature described in Chapter 15. Finally, see the separate protocol chapters for information about how to create access lists.

Establish Password Protection

Complete the following tasks to establish password protection:

- Protect access to terminals on individual lines
- Protect access to privileged EXEC commands, and thus, to the system configuration file
- Encrypt passwords so that they cannot be read in the configuration file with the **show configuration** EXEC command or with a protocol analyzer

Protect Access to Terminal Lines

You can provide access control on a terminal line by entering the password and establishing password checking. To do so, perform the following tasks in line configuration mode:

Task		Command
Step 1	Assign a password to a terminal or other device on a line.	**password** *text*
Step 2	Enable password checking.	**login**

The password checker is case sensitive. The password *Secret* is different than the password *secret*, for example, and the password *two words* is an acceptable password.

For a complete description of the **password** command, see Chapter 4 of the *Router Products Command Reference* publication.

Protect Access to Privileged EXEC Commands

You can control access to the system by setting a password that must be entered to gain access to the privileged-level prompt, and therefore to the system configuration. Perform the following task in global configuration mode:

Task	Command
Establish a password for the privileged command level.	**enable password** *password*

Encrypt the Passwords

You can increase access security to your router by configuring it to encrypt passwords, because protocol analyzers can examine packets. Encryption prevents the password from being visible in the configuration file.

Configure the router to encrypt passwords by performing the following task in global configuration mode:

Task	Command
Encrypt a password.	**service password-encryption**

It is not possible to recover a lost encrypted password.

Disable Password Protection

You can disable line password verification by disabling password checking. To do so, perform the following task in line configuration mode:

Task	Command
Disable password checking or allow access to a line without password verification.	**no login**

Recover a Lost Password

If your server has the nonvolatile memory option, you can accidentally lock yourself out if you enable password checking on the console terminal line and then forget the line password. To recover a lost password, force the server into factory diagnostic mode and then follow these steps:

Step 1 You will be asked if you want to set the manufacturers' addresses. Respond by typing **Yes**. You will then see the following prompt:

```
TEST-SYSTEM>
```

Step 2 Type the **enable** command to get the privileged prompt:

```
TEST-SYSTEM> enable
```

Step 3 Type the **show configuration** command to review the system configuration and find the password. Do not change anything in the factory diagnostic mode.

```
TEST-SYSTEM> show configuration
```

Step 4 To resume normal operation, restart the server and/or reset the configuration register.

Step 5 Log into the server with the password that was shown in the configuration file.

Note All debugging capabilities are turned on during diagnostic mode.

See the hardware installation and maintenance publication for your product for specific information about configuring the processor configuration register for factory diagnostic mode. Table 5-1 summarizes the hardware or software settings required by the various products to set factory diagnostic mode.

Table 5-1 Factory Diagnostic Mode Settings for the Configuration Register

Platform	Setting
Modular products	Set jumper in bit 15 of the processor configuration register, then restart; remove jumper when finished.
Older IGS	Set jumper in bit 7 of the processor configuration register, then restart; remove jumper when finished.
Later IGS, Cisco 3000, Cisco 4000	Use the **config register** command to set the processor configuration register to 0x8000, then **initialize** and **boot** the system. Use the **reload** command to restart and set the processor configuration register to 0x2102 when finished.

Create Access Lists

This section summarizes the protocols for access lists. The general guidelines for access lists vary from protocol to protocol. See the appropriate chapter in this guide for detailed task information on each protocol-specific access list. To control SNMP access, see "Enable SNMP and Define Access Control" earlier in this chapter. Also refer to the appropriate protocol-specific chapters of the *Router Products Command Reference* publication.

Table 5-2 provides the protocols that have access lists specified by names.

Table 5-2 Protocols that have Access Lists Specified by Names

Protocol
Apollo Domain
ISO CLNS
Source-Route Bridging NetBIOS
NetBIOS IPX

Table 5-3 provides the protocols that have access lists specified by numbers and provides the corresponding numerical ranges.

Table 5-3   Protocols that have Access Lists Specified by Numbers

Protocol	Range
IP	1–99
Extended IP	100–199
Transparent Bridging (protocol type)	200–299
Transparent Bridging (vendor code)	700–799
Extended Transparent Bridging	1100–1199
DECnet and Extended DECnet	300–399
XNS	400–499
Extended XNS	500–599
AppleTalk	600–699
Source-Route Bridging (protocol type)	200–299
Source-Route Bridging (vendor code)	700–799
IPX	800–899
Extended IPX	900–999
IPX SAP	1000–1099
Standard VINES	1–100
Extended VINES	101–200
Simple VINES	201–300

Establish Terminal Access Control

You can configure the router to use a special TCP/IP protocol called Terminal Access Controller Access Control System (TACACS). TACACS provides an additional level of control over servers running on a timesharing system. The Defense Data Network (DDN) developed TACACS to control access to its TAC servers; Cisco patterned its TACACS support after the DDN application.

The TACACS security program allows you to set these features:

- Set TACACS password protection and login authentication at both the user and privileged EXEC levels with "last resort" and optional password options

- Set separate TACACS server host names with custom retransmit and timeout intervals

- Establish a limit on login attempts

- Enable an extended TACACS mode that supports system accounting and logging applications

The following sections describe these tasks.

Note   If you require additional protection using TCP/IP access lists, see Chapter 16 for more information.

You can establish TACACS password protection on both user and privileged levels of the system EXEC. The following features are available with TACACS login and password protection feature:

- Set TACACS user ID and password checking at the user level

- Disable checking at the user level and guarantee "last resort" login

- Set optional password verification

- Set TACACS user ID and password checking at the privileged level

- Disable checking at the privileged level and guarantee "last resort" login

- Set notification messages when the user makes a connection, accesses the privileged EXEC level, or logs out

- Set authentication when the user makes a connection or accesses the privileged EXEC level

Set TACACS Password Protection at the User Level

You can enable password checking at login by performing the following task in line configuration mode:

Task	Command
Set the TACACS-style user ID and password-checking mechanism.	**login tacacs**

Disable Password Checking at the User Level

If a TACACS server does not respond to a login request, the router will deny the request by default. However, you can prevent that login failure in one of two ways. You can allow a user to access privileged EXEC mode if that user enters the password set by the **enable** command. To do so, specify **tacacs-server last-resort password**.

Alternatively, you can ensure a successful login by configuring **tacacs-server last-resort succeed**. In this case, the user can access the privileged EXEC mode without further question.

To specify one of these features, perform the following task in global configuration mode:

Task	Command
Set "last resort" options for logins.	**tacacs-server last-resort** {**password**	**succeed**}

Set Optional Password Verification

You can specify that the first TACACS request to a TACACS server is made without password verification. To do so, perform the following task in global configuration mode:

Task	Command
Set TACACS password as optional.	t**acacs-server optional-passwords**

When the user types in the login name, the login request is transmitted with the name and a zero-length password. If accepted, the login procedure completes. If the TACACS server refuses this request, the terminal server prompts for a password and tries again when the user supplies a password. The TACACS server must support authentication for users without passwords to make use of this feature. This feature supports all TACACS requests such as login, SLIP, and enable.

Set TACACS Password Protection at the Privileged Level

You can set the TACACS protocol to determine whether a user can access the privileged EXEC level. To do so, perform the following task in global configuration mode:

Task	Command
Set the TACACS-style user ID and password-checking mechanism at the privileged command level.	**enable use-tacacs**

When you use this command, the EXEC **enable** command will ask for both a new user name and a password. This information is then passed to the TACACS server for authentication. If you are using the extended TACACS, it will also pass any existing UNIX user identification code to the server.

Caution If you use the **enable use-tacacs** command, you must also specify **tacacs-server authenticate enable**, or else you will be locked out of the router.

Note When used without extended TACACS, this command allows anyone with a valid user name and password to access the privileged command level, creating a potential security problem. This is because the TACACS query resulting from entering the **enable** command is indistinguishable from an attempt to log in without extended TACACS.

Disable Password Checking at the Privileged Level

You can specify what happens if the TACACS servers used by the **enable** command do not respond. To invoke this "last resort" login feature, perform the following task in global configuration mode:

Task	Command
Set "last resort" options for logins at the privileged prompt.	**enable last-resort** {**password**	**succeed**}

The default action is that the **enable** command fails. By specifying **enable last-resort password**, you are allowed to enable by entering the privileged command level password. Alternatively, by specifying **enable last-resort succeed**, you are allowed to enable without further question.

Set Notification of User Actions

You can cause a message to be transmitted to the TACACS server when a user either makes a TCP connection, enters the **enable** command, or logs out. To do so, perform the following task in global configuration mode:

Task	Command
Set server notification of user actions.	**tacacs-server notify** {**connect**	**enable**	**logout**}

The retransmission of the message is performed by a background process for up to five minutes. The terminal user, however, receives an immediate response allowing access to the terminal.

Set Authentication of User Actions

You can specify that if a user tries to make a TCP connection or enters the **enable** command, the router requires a response from the network or communication server indicating whether the user can perform the action. You can specify authentication of either one action or the other. To do so, perform the following task in global configuration mode:

Task	Command
Set server authentication of user actions.	**tacacs-server authenticate** {**connect**	**enable**}

Establish the TACACS Server Host and Response Times

You can specify the names of the IP host or hosts maintaining a TACACS server. The software searches for the hosts in the order specified, so this feature can be useful for setting up a list of preferred servers.

You can also modify the number of times the system software searches the list of TACACS servers (from the default of two times) and the interval it waits for a reply (from the default of 5 seconds).

Perform the following tasks in global configuration mode, as needed, for your system configuration:

Task	Command
Specify a TACACS host.	**tacacs-server host** *name*
Specify the number of times the server will search the list of TACACS server hosts before giving up.	**tacacs-server retransmit** *retries*
Set the interval the server waits for a TACACS server host to reply.	**tacacs-server timeout** *seconds*

Set Limits on Login Attempts

You can set controls on the number of login attempts that can be made on a line set up for TACACS by performing the following task in global configuration mode:

Task	Command
Control the number of login attempts that can be made on a line set for TACACS verification.	**tacacs-server attempts** *count*

Enable the Extended TACACS Mode

Extended TACACS mode provides information about the terminal requests to help set up UNIX auditing trails and accounting files for tracking use of protocol translators, communication servers, and routers. The information includes responses from these network devices and validation of user requests.

An unsupported, extended TACACS server is available from Cisco Systems using **ftp** for UNIX users who want to create the auditing programs (see the README file in the *ftp.cisco.com* directory).

Extended TACACS differs from standard TACACS in that standard TACACS provides only username and password information.

To enable extended TACACS mode, perform the following task in global configuration mode:

Task	Command
Enable an extended TACACS mode.	**tacacs-server extended**

Establish Username Authentication

You can create a username-based authentication system, which is useful for the following reasons:

- To provide a TACACS-like username and encrypted password authentication system for those networks that cannot support TACACS

- To provide special case logins, for example, access list verification, no password verification, autocommand execution at login, and "no escape" situation.

Perform the following tasks in global configuration mode, as needed for your system configuration:

Task	Command
Establish username authentication with encrypted passwords or by access list.	**username** *name* [**no password** | **password** *encryptiontype* **password**] **username** *name* [**access-class** *number*]
Specify a command to automatically execute.	**username** *name* [**autocommand** *command*]
Set a "no escape" login environment.	**username** *name* [**noescape**] [**nohangup**]

The keyword **noescape** prevents users from using escape characters on the hosts to which they are connected.

Enable Challenge Handshake Authentication Protocol (CHAP)

Access control using Challenge Handshake Authentication Protocol (CHAP) or Password Authentication Protocol (PAP) is available on all serial interfaces. The authentication feature reduces the risk of security violations on your router. You can configure either CHAP or PAP for the interface.

Note To use CHAP or PAP, you must be running PPP encapsulation.

When CHAP is enabled on an interface, the local router sends a CHAP packet and the remote device (a PC, workstation, or server) attempting to connect to the router is requested, or *challenged*, to respond. The required response is an encrypted version of a secret password, or *secret*, plus a random value and the name of the remote device.

By transmitting this response, the secret is never transmitted, preventing other devices from stealing it and gaining illegal access to the system. Without the proper response, the remote device cannot connect to the local router.

CHAP transactions occur only at the time a link is established. The local router does not request a password during the rest of the call. (The local router can, however, respond to such requests from other devices during a call.)

When PAP is enabled, the remote router attempting to connect to the local router is required to send an authentication request. Unlike CHAP, the remote router initiates the sequence. If the username and password specified in the authentication request are accepted, the router sends an authentication acknowledgment.

To use CHAP or PAP, perform the following tasks:

- Enable CHAP or PAP on the interface. Once you have done so, the local router requires a password from remote devices that are calling in. If the remote device does not support the authentication method, no traffic will be passed to that device.

- Configure server host-name authentication. Configure the secret or password for each remote system for which authentication is required.

To enable CHAP or PAP on an interface configured for PPP encapsulation, perform one of the following tasks in interface configuration mode:

Task	Command
Enable CHAP on an interface.	**ppp authentication chap** [**if-needed**]
Enable PAP on an interface.	**ppp authentication pap** [**if-needed**]

To specify the password to be used in CHAP or PAP caller identification, perform the following task in global configuration mode:

Task	Command
Configure authentication.	**username** *name* **password** *secret*

CHAP and PAP are specified in the IETF RFC 1334 "The PPP Authentication Protocols" by Brian Lloyd of Lloyd and Associates and William A. Simpson of Computer Systems Consulting Services. CHAP is specified as an additional authentication phase of the PPP Link Control Protocol.

Enable Password Authentication Protocol (PAP)

Access control using the Password Authentication Protocol (PAP) is available on all serial interfaces. The authentication feature reduces the risk of security violations on your router.

Note PAP is supported only on lines that use PPP encapsulation.

To use PAP, perform the following task in interface configuration mode:

Task	Command
Enable PAP on the interface.	**ppp authentication pap**

Fault Management

To perform general fault management, complete the following tasks:

- Display system information
- Test network connectivity
- Limit TCP transactions
- Trace packet routes
- Test interfaces and memory
- Log system error messages
- Enable debug operations

Most chapters in this guide include fault management tasks in a monitoring and maintaining section. For example, Chapter 6 provides a section on interface loopback testing. Another example is the information on Internet Control Messages Protocol (ICMP) support described in Chapter 15.

Display System Information

To provide information about system processes, the software includes an extensive list of EXEC commands that begin with the word **show**, which, when executed, display detailed tables of system information. Following is a list of the more common system management **show** commands. Perform these tasks in EXEC mode to display the information described:

Task	Command
On the AGS+, display temperature and voltage information. On the 7000, display a message indicating whether an environmental warning condition currently exists.	**show environment**
Display the temperature and voltage information on the console (for the Cisco 7000 only).	**show environment all**
Display the last measured value from each of the six test points stored in nonvolatile memory.	**show environment last**

Task	Command
Display environmental specifications (for the Cisco 7000 only).	**show environment table**
Display information about all active processes.	**show processes**
Display the configured protocols.	**show protocols**

Look for specific **show** commands in the tables of configuration tasks found throughout the chapters in this guide. See the *Router Products Command Reference* publication for detailed descriptions of the commands.

The following sections describe the EXEC commands you can use to monitor and troubleshoot the voltage and temperature of your system environment.

Receiving Automatic Warning Messages

On the Cisco 7000 only, the environmental monitor is built into the route processor (RP). If a measurement exceeds acceptable margins, a warning message is printed to the system console. The system software queries the RP for measurements once every 60 seconds, but warnings for a given test point are printed at most once every four hours. If the temperature measurements are out of specification more than the shutdown margin, the software will shut the router down but the fan will stay on. The router has to be manually turned off and on after such a shutdown. You can query the RP using the **show environment** command at any time to determine whether a measurement is out of tolerance.

Refer to the *System Error Messages* publication for a description of environmental monitor warning messages.

Receiving the Automatic Shutdown Message

If the RP detects that any of its temperature test points have exceeded maximum margins, it performs the following steps in this order:

Step 1 Saves the last measured values from each of the six test points to internal nonvolatile memory.

Step 2 Interrupts the system software and causes a shutdown message to be printed on the system console.

Step 3 Shuts off the power supplies after a few milliseconds of delay.

Example

The following is the message the system displays if temperatures exceed maximum margins, along with a message indicating the reason for the shutdown:

```
Router#
%ENVM-1-SHUTDOWN: Environmental Monitor initiated shutdown
%ENVM-2-TEMP: Inlet temperature has reached SHUTDOWN level at 64(C)
```

Refer to the hardware installation and maintenance publication for your router for more information about environmental specifications.

Test Network Connectivity

Complete the following tasks to test basic network connectivity:

- Set up TCP keepalive packet service
- Test connections with the **ping** command
- Trace packet routes

Set up TCP Keepalive Packet Service

The TCP keepalive capability allows a router to detect when the host with which it is communicating experiences a system failure, even if data stops being transmitted (in either direction). This is most useful on incoming connections. For example, if a host failure occurs while talking to a printer, the router may never notice, since the printer does not generate any traffic in the opposite direction. If keepalives are enabled, they are sent once every minute on otherwise idle connections. If five minutes pass and no keepalives are detected, the connection is closed. The connection will also be closed if the host replies to a keepalive packet with a reset packet. This will happen if the host crashes and comes back up again.

To set up the TCP keepalive packet service, perform the following task in global configuration mode:

Task	Command
Generate TCP keepalive packets on idle network connections, either incoming connections initiated by a remote host, or outgoing connections initiated by a user.	s**ervice tcp-keepalives** {**in**	**out**}

Test Connections with the Ping Command

As an aid to diagnosing basic network connectivity, many network protocols support an echo protocol. The protocol involves sending a special datagram to the destination host, then waiting for a reply datagram from that host. Results from this echo protocol can help in evaluating the path-to-host reliability, delays over the path, and whether the host can be reached or is functioning.

To use the echo protocol, perform the following task in either user EXEC or privileged EXEC mode:

Task	Command
Invoke a diagnostic tool for testing connectivity.	**ping** [*protocol*] {*host*	*address*}

Look for specific **ping** commands in the tables of configuration tasks found throughout the chapters in this guide. See the *Router Products Command Reference* publication for detailed descriptions of the command.

Trace Packet Routes

You can discover the routes that packets will actually take when traveling to their destinations. To do so, perform one of the following tasks in EXEC mode:

Trace packet routes through the network (privileged level).	**trace** [*protocol*] [*destination*]
Trace packet routes through the network (user level).	**trace** [*protocol*] [*destination*]

Limit TCP Transactions

When using a standard TCP implementation to send keystrokes between machines, TCP tends to send one packet for each keystroke typed. On larger networks, many small packets use up bandwidth and contribute to congestion.

John Nagle's algorithm (RFC-896) helps alleviate the small-packet problem in TCP. In general, it works this way: The first character typed after connection establishment is sent in a single packet, but TCP holds any additional characters typed until the receiver acknowledges the previous packet. Then the second, larger packet is sent, and additional typed characters are saved until the acknowledgement comes back. The effect is to accumulate characters into larger chunks, and pace them out to the network at a rate matching the round-trip time of the given connection. This method is usually a good for all TCP-based traffic. However, do not use the **service nagle** command if you have XRemote users on X Window sessions.

By default, the Nagle algorithm is not enabled. To invoke the Nagle algorithm and thereby reduce TCP transactions, perform the following task in global configuration mode:

Task	Command
Enable the Nagle slow packet avoidance algorithm.	**service nagle**

Test Memory and Interfaces

You can test the status of the following items:

- Flash memory
- System memory
- Interfaces

Note We do not recommend using these commands; they are intended to aid manufacturing personnel in checking system functionality.

Test Flash

To test the status of Flash memory, perform the following task in privileged EXEC mode:

Task	Command
Test Flash memory on MCI and envm Flash EPROM interfaces	**test flash**

Test System Memory

To test the status of system memory, perform the following task in privileged EXEC mode:

Task	Command
Diagnose Multibus memory, including nonvolatile memory.	**test memory**

Test Interfaces

To test the status of the interfaces, perform the following task in privileged EXEC mode:

Task	Command
Check network interfaces. This test is not intended for diagnosing problems with an operational server.	**test interfaces**

Log System Error Messages

By default, the network servers send the output from the EXEC command **debug** and system error messages to the console terminal. You can redirect these messages, as well as output from asynchronous events such as interface transition, to other destinations. These destinations include virtual terminals, internal buffers, and UNIX hosts running a syslog server; the syslog format is compatible with 4.3 BSD UNIX.

Additionally, you can set the severity level of the messages to control the type of messages displayed. You can also have log messages timestamped to enhance real-time debugging and management.

With the current software release, there are three new syslog messages at LOG_NOTICE syslog level that make it easier to check the status of how the system provides address resolution. An example follows:

```
%LINK-5-BOOTP: Ethernet0 address 131.108.160.24, resolved by 131.108.1.111
%LINK-5-RARP: Ethernet0 address 131.108.160.24, resolved by 131.108.1.111
%LINK-5-SLARP: Ethernet0 address 131.108.160.24, resolved by 131.108.1.111
```

There are also new startup messages that help you identify NVRAM problems:

```
Warning: NVRAM device not found
Warning: NVRAM invalid, possibly due to write erase
```

The following level 4 LOG_WARNING message has been added for FDDI status information:

```
%FDDISTAT-4-STATUS: FDDI state indication detected on interface variable
```

The possible values for *indication* are listed in the next paragraph. The variable will be replaced with something like fddi0, for example.

Changes in status reflect interface connectivity or cabling problems (or fixes). The possible status reports include the following indications:

```
isolated
wrap A
wrap B
wrap a-B
thru A
thru B
thru A-B
```

Log Errors to a UNIX Syslog Daemon

To set up the syslog daemon on a 4.3 BSD UNIX system, include a line such as the following in the file */etc/syslog.conf*:

```
local7.debugging /usr/adm/logs/cisco.log
```

The **local7** keyword specifies the logging facility to be used; see Table 5-5 for a list of other keywords. The **debugging** keyword specifies the syslog level; see Table 5-4 for a list of other keywords.

The syslog daemon sends messages at this level or a more severe level to the file specified in the next field. The file must already exist, and the syslog daemon must have permission to write to it.

Enable Message Logging

To enable message logging, perform the following task in global configuration mode:

Task	Command
Enable message logging.	**logging on**

Set the Error Message Display Device

By default, error messages are directed to the system console. To direct messages to other devices, perform one of the following tasks in global configuration mode:

Task	Command
Log messages to an internal buffer.	**logging buffered**
Log messages to a UNIX syslog server host.	**logging** *host*
Redirect messages to the system console.	**no logging on**

The **logging buffered** command copies logging messages to an internal buffer instead of writing them to the console terminal. The buffer is circular, so newer messages overwrite older messages. To display the messages that are logged in the buffer, use the **show logging** EXEC command. The first message displayed is the oldest message in the buffer.

The EXEC command **terminal monitor** locally accomplishes the task of displaying the system error messages to a nonconsole terminal.

The **logging** command identifies a syslog server host to receive logging messages. The argument *host* is the name or Internet address of the host. By issuing this command more than once, you build a list of syslog servers that receive logging messages. The **no logging** command deletes the syslog server with the specified address from the list of syslogs.

Define the Error Message Severity Level and Facilities

You can limit messages displayed to the selected device by specifying the severity level of the error message. To do so, perform one of the following tasks in global configuration mode:

Task	Command
Limit messages logged to the console.	**logging console** *level*
Limit messages logged to the terminal lines.	**logging monitor** *level*
Limit messages logged to the syslog servers.	**logging trap** *level*

The **logging console** command limits the logging messages displayed on the console terminal to messages with a level number at or below the specified severity level, which is specified by the *level* argument. Table 5-4 lists the error message *level* keywords and corresponding UNIX syslog definitions in order from the most severe level to the least severe level.

Table 5-4   Error Message Logging Keywords

Level Keyword	Level	Description	Syslog Definition
emergencies	0	System unusable	LOG_EMERG
alerts	1	Immediate action needed	LOG_ALERT
critical	2	Critical conditions	LOG_CRIT
errors	3	Error conditions	LOG_ERR
warnings	4	Warning conditions	LOG_WARNING
notifications	5	Normal but significant condition	LOG_NOTICE
informational	6	Informational messages only	LOG_INFO
debugging	7	Debugging messages	LOG_DEBUG

The **no logging console** command disables logging to the console terminal.

The default is to log messages to the console at the **debugging** level and those level numbers that are lower, which means all levels. The **logging monitor** command defaults to **debugging** also. The **logging trap** command defaults to **informational**.

To display logging messages on a terminal, use the **terminal monitor** EXEC command.

Current software generates four categories of error messages:

- Error messages about software or hardware malfunctions, displayed at levels **warnings** through **emergencies**.

- Output from the **debug** commands, displayed at the **debugging** level

- Interface up/down transitions and system restart messages, displayed at the **notifications** level

- Reload requests and low-process stack messages, displayed at the **informational** level

Define the Syslog Facility

You can also configure the syslog facility in which error messages are sent, by performing the following task in global configuration mode:

Task	Command
Configure system log facilities.	**logging facility** *facility-type*

Table 5-5 lists the logging facility types and their descriptions.

Table 5-5   Logging Facility Types

Facility Type	Description
auth	Indicates the authorization system.
cron	Indicates the cron facility.
daemon	Indicates the system daemon.
kern	Indicates the Kernel.
local0–7	Reserved for locally defined messages.
lpr	Indicates line printer system.
mail	Indicates mail system.
news	Indicates USENET news.
sys9	Indicates system use.
sys10	Indicates system use.
sys11	Indicates system use.
sys12	Indicates system use.
sys13	Indicates system use.
sys14	Indicates system use.
syslog	Indicates the system log.
user	Indicates user process.
uucp	Indicates UNIX-to-UNIX copy system.

Refer also to your syslog manual pages.

The EXEC command **show logging** displays the addresses and levels associated with the current logging setup, as well as any other logging statistics. To display this information, perform the following task in EXEC mode:

Task	Command
Display the state of syslog error and event logging, including host addresses and whether console logging is enabled.	**show logging**

Enable Timestamps on Log Messages

By default, log messages are not timestamped. You can enable timestamping of log messages by performing the following task in global configuration mode:

Task	Command
Enable log timestamps.	**service timestamps log uptime** or **service timestamps log datetime** [**msec**] [**localtime**] [**show-timezone**]

Enable Debug Operations

Your router includes hardware and software to aid in tracking down problems with the router or with other hosts on the network. The privileged **debug** EXEC commands start the console display of several classes of network events. The following tasks describe in general the system debug message feature. Refer to the *Debug Command Reference* publication for all information regarding **debug** commands. Also refer to the *Troubleshooting Internetworking Systems* publication.

Task	Command
Display the state of each debugging option.	**show debugging**
Display a list and brief description of all the **debug** command options.	**debug ?**
Begin message logging for the specified **debug** command.	**debug** *command*
Turn message logging off for the specified **debug** command.	**no debug** *command*

You can configure timestamping of system debug messages. Timestamping enhances real-time debugging by providing the relative timing of logged events. This information is especially useful when customers send debugging output to your technical support personnel for assistance. To enable timestamping of system debug messages, perform the following task in global configuration mode:

Task	Command
Enable timestamping of system debug messages.	**service timestamps debug uptime** or **service timestamps debug datetime** [**msec**] [**localtime**] [**show-timezone**]

Normally, the messages are displayed only on the console terminal. See the section "Set the Error Message Display Device" earlier in this chapter to change the output device.

Note The system gives high priority to debugging output. For this reason, debugging commands should be turned on only for troubleshooting specific problems or during troubleshooting sessions with technical support personnel. Excessive debugging output can render the system inoperable.

System Performance Management

This section describes how to manage general system performance by completing the following tasks:

- Configure switching and scheduling priorities
- Establish queuing strategies
- Modify the system buffer size
- Delay EXEC startup
- Handle idle Telnet connection

In addition, most chapters in this guide include performance tasks specific to the chapter content, and the *Internetworking Design Guide* includes detailed information on performance issues that arise when designing a network.

Configure Switching and Scheduling Priorities

The normal operation of the network server allows the switching operations to use as much of the central processor as is required. If the network is running unusually heavy loads that do not allow the processor the time to handle the routing protocols, you may need to give priority to the system process scheduler. To do so, perform the following task in global configuration mode:

Task	Command
Define the maximum amount of time that can elapse without running the lowest-priority system processes.	**scheduler-interval** *milliseconds*

Establish Queuing Strategies

We provide two types of queuing strategies for prioritizing network traffic:

- Priority queuing

- Custom queuing

You can configure both priority queuing and custom queuing, but you can only assign *either* a priority group or a custom queue to an interface.

Priority Queuing

Priority output queuing is a mechanism that allows the administrator to set priorities on the type of traffic passing through the network. Packets are classified according to various criteria, including protocol and subprotocol type, and then queued on one of four output queues (high, medium, normal, and low).

When the server is ready to transmit a packet, it scans the priority queues in order, from highest to lowest, to find the highest-priority packet. After that packet is completely transmitted, the server scans the priority queues again. If a priority output queue fills up, packets will be dropped and, for IP, quench indications will be sent to the original transmitter.

Although you can enable priority output queuing for any interface, the intended application was for low-bandwidth, congested serial interfaces. Our priority output queuing mechanism allows traffic control based on protocol or interface type. You can also set the size of the queue and defaults for what happens to packets that are not defined by priority output queue rules.

The priority output queuing mechanism can be used to manage traffic from all networking protocols. Additional fine-tuning is available for IP and for setting boundaries on the packet size.

Note Priority queuing introduces extra overhead that is acceptable for slow interfaces, but may not be acceptable for higher-speed interfaces such as Ethernet.

Note   Priority queuing does not operate over X.25

The four priority queues—high, medium, normal, and low—are listed in order from highest to lowest priority. Keepalives sourced by the network server are always assigned to the high-priority queue; all other management traffic (such as IGRP updates) must be configured. Packets that are not classified by the priority list mechanism are assigned to the normal queue.

A priority list is a set of rules that describes how packets should be assigned to priority queues. A priority list might also describe a default priority or the queue size limits of the various priority queues.

Custom Queuing

Priority queuing introduces a fairness problem in that packets classified to lower-priority queues may not get serviced in a timely manner, or at all, depending upon the bandwidth used by packets sent from the higher-priority output queues.

With custom output queuing, a "weighted fair" queuing strategy is implemented for the processing of interface output queues. You can control the percentage of an interface's available bandwidth that is used by a particular kind of traffic. When custom queuing is enabled on an interface, the system maintains 11 output queues for that interface that can be used to modify queuing behavior.

For queue numbers 1 through 10, the system cycles through the queues sequentially, delivering packets in the current queue before moving on to the next. Associated with each output queue is a configurable byte count, which specifies how many bytes of data the system should deliver from the current queue before it moves on to the next queue. When a particular queue is being processed, packets are sent until the number of bytes sent exceed the queue byte count or the queue is empty. Bandwidth used by a particular queue can only be indirectly specified in terms of byte count and queue length.

Queue number 0 is a system queue; it is emptied before any of the queues numbered 1 through 10 are processed. The system enqueues high-priority packets, such as keepalive packets, to this queue. Other traffic cannot be configured to use this queue.

Note   With custom or priority queueing enabled, the system takes longer to switch packets, because the packets are classified by the processor card.

Queuing Task List

You can set up both priority queuing and custom queuing on your network, but you can assign only one or the other to an interface.

Following is a list of the priority-setting tasks that you can choose from, depending upon the needs of your network:

- Set priority by protocol type

- Assign a default priority

- Set priority by interface type

- Specify the maximum packets and bytes in the priority queues

- Assign priority by STUN address

- Assign a priority list or a custom queue to an interface

- Monitor the priority and custom queue lists

See the following sections for more information about these tasks.

Set Priority by Protocol Type

You can establish queuing priorities based upon the protocol type by using one of the following commands in global configuration mode. All Cisco-supported protocols are allowed.

Task	Command
Establish queuing priorities based upon the protocol type.	**priority-list** *list* **protocol** *protocol-name* {**high**	**medium**	**normal**	**low**} *queue-keyword keyword-value*
	or
	queue-list *list* **protocol** *protocol-name queue-number queue-keyword keyword-value*

Queue keywords provide additional options including byte-count, TCP service and port number assignments, and AppleTalk, IP, IPX, VINES, or XNS access list assignments. See the **priority-list** and **queue-list** command syntax descriptions in Chapter 5 of the *Router Products Command Reference* publication.

Assign a Default Priority

You can assign a queue for those packets that did not match any other rule in the list. To do so, perform one of the following tasks in global configuration mode:

Task	Command
Assign a priority queue for those packets that do not match any other rule in the priority list.	**priority-list** *list* **default** {**high**	**medium**	**normal**	**low**}
Assign a queue number for those packets that do not match any other rule in the custom queue list.	**queue-list** *list* **default** *queue-number*

Set Priority by Interface Type

You can establish queuing priorities on packets entering from a specific interface by performing one of the following tasks in global configuration mode:

Task	Command
Establish queuing priorities on packets entering from a given interface.	**priority-list** *list* **interface** *interface-type interface-number* {**high**	**medium**	**normal**	**low**}
Establish custom queuing based on packets entering from a given interface.	**queue-list** *list* **interface** *interface-type interface-number queue-number*

Specify the Maximum Packets and Bytes in the Priority Queues

You can specify the maximum number of packets that may be waiting in each of the priority queues.
To do so, perform one of the following tasks in global configuration mode:

Task	Command
Specify the maximum number of packets that can be waiting in each of the priority queues	**priority-list** *list* **queue-limit** *high-limit medium-limit normal-limit low-limit*
	or
	queue-list *list* **queue** *queue-number* **limit** *limit-number*
Designate the byte size allowed per queue.	**queue-list** *list* **queue** *queue-number* **byte-count** *byte-count number*

Both **limit** and **byte-count** keywords may appear as arguments to the **queue-list** *list* **queue**
command.

Assign Priority by STUN Address

You can establish queuing priorities based on the address of a serial link on a STUN connection. To
do so, perform one of the following tasks in global configuration mode:

Task	Command
Establish queuing priorities based on the address of the serial link on a STUN connection.	**priority-list** *list* **stun** {**high** | **medium** | **normal** | **low**} **address** *group-number address-number*
	or
	queue-list *list* **stun** *queue-number* **address** *group-number address-number*

Assign a Priority Group or a Custom Queue to an Interface

You can assign a priority list number to an interface. Only one list can be assigned per interface. To
assign an priority group or custom queue to an interface, perform one of the following tasks in
interface configuration mode:

Task	Command
Assign a priority list number to the interface.	**priority-group** *list*
Assign a custom queue list number to the interface.	**custom-queue-list** *list*

See Chapter 6 of the *Router Products Command Referenc*e for syntax descriptions of the
priority-group and **custom-queue-list** interface configuration commands.

Monitor the Priority and Custom Queuing Lists

You can display information about the input and output queues when priority queuing is enabled on an interface. To do so, perform one of the following tasks in EXEC mode:

Task	Command
Show the status of the priority queuing lists.	**show queueing priority**
Show the status of the custom queuing lists.	**show queueing custom**

If you enter the **show queueing** command without any keywords, the router displays status on both custom and priority queue lists. See Chapter 6 of the *Router Products Command Reference* for the syntax description on the **show queueing** command.

Modify the System Buffer Size

You can adjust initial buffer pool settings and the limits at which temporary buffers are created and destroyed. To do so, perform the following tasks in global configuration mode:

Task	Command
Adjust the system buffer sizes.	**buffers** {**small** | **middle** | **big** | **large** | **huge**} {**permanent** | **max-free** | **min-free** | **initial**} *number*
Dynamically resize all huge buffers to the value that you supply.	**buffers huge size** *number*

During normal system operation, there are several pools of different sized buffers. These pools grow and shrink based upon demand. Some buffers are temporary and are created and destroyed as needed. Other buffers are permanently allocated and cannot be destroyed. For examples of the **buffers** command, see the examples at the end of this chapter.

Note It is normally not necessary to adjust these parameters; *do so only after consulting with technical support personnel*. Improper settings could adversely impact system performance.

To display statistics about the buffer pool on the system, perform the following task in EXEC mode:

Task	Command
List statistics for the buffer pools. The server has one pool of queuing elements and five pools of packet buffers of different sizes. For each pool, the server keeps count of the number of buffers outstanding, the number of buffers in the free list, and the maximum number of buffers allowed in the free list.	**show buffers** [*interface*]

Delay EXEC Startup

You can delay the startup of the EXEC on noisy lines until the line has been idle for 3 seconds. To do so, perform the following task in global configuration mode:

Task	Command
Delay startup of the EXEC.	**service exec-wait**

This command is useful on noisy modem lines or when a modem attached to the line is configured to ignore MNP or V.42 negotiations, and MNP or V.42 modems may be dialing in. In these cases, noise or MNP/V.42 packets might be interpreted as usernames and passwords, causing authentication failure before the user can type a username/password. The command is not useful on non-modem lines or lines without some kind of login configured.

Handle Idle Telnet Connection

You can configure the router to set the TCP window to zero (0) when the Telnet connection is idle. To do so, perform the following task in global configuration mode:

Task	Command
Set the TCP window to 0 when the Telnet connection is idle.	**service telnet-zero-idle**

Normally, data sent to non-current Telnet connections is accepted and discarded. When **service telnet-zero-idle** is enabled, if a session is suspended (that is, some other connection is made active or the EXEC is sitting in command mode), the TCP window is set to zero. This action prevents the remote host from sending any more data until the connection is resumed. Use this command when it is important that all messages sent by the host be seen by the users and the users are likely to use multiple sessions. Do not use this command if your host will eventually time out and log out a TCP user whose window is zero.

Accounting Management

Accounting management allows you to track individual and group usage of network resources. You can then reallocate resources as needed.

Additional tasks for measuring system resources are covered in other chapters; for example, IP accounting tasks are described in Chapter 15.

Display Stack Utilization

You can display stack utilization of processes and interrupt routines, including the reason for the last system reboot. This feature is useful for analyzing system crashes. To display stack utilization, perform the following task in EXEC mode:

Task	Command
Display stack utilization of processes and interrupt routines.	**show stacks**

Display Memory Utilization

To display memory usage information, perform the following tasks in EXEC mode:

Task	Command
Display memory pool statistics including summary information about the activities of the system memory allocator and a block-by-block listing of memory use.	**show memory** [*type*] [**free**]
Display information about memory utilization.	**show processes memory**

System Management Examples

The following is a list of the examples in this section:

- System configuration file example (page 5-41)

- Clock, calendar, and NTP examples (page 5-41)

- Examples of modifying buffers (page 5-42)

- Username example (page 5-42)

System Configuration File Example

The following is an example of a typical system configuration file:

```
! Define line password
line 0 4
password secret
login
!
! Define privileged-level password
enable-password Secret Word
!
! Define a system hostname
hostname TIP
! Define host filenames
boot host host1-confg 131.108.1.111
boot host host2-confg 131.108.1.111
! Define system filenames
boot system sys1-system 131.108.13.111
boot system sys2-system 131.108.1.111
!
! Enable SNMP
snmp-server community
snmp-server trap-authentication
snmp-server host 131.108.1.27 public
snmp-server host 131.108.1.111 public
snmp-server host 131.108.2.63 public
!
! Define TACACS server hosts
tacacs-server host 131.108.1.27
tacacs-server host 131.108.13.33
tacacs-server host 131.108.1.33
!
! Define a message-of-the-day banner
banner motd ^C
The Information Place welcomes you

Please call 1-800-555-2222 for a login account, or enter
your password at the prompt.
^C
```

Clock, Calendar, and NTP Configuration Examples

In the following example, a Cisco 7000 has server associations with two other systems, transmits broadcast NTP packets, periodically updates the Cisco 7000 calendar, and redistributes time into VINES.

```
clock timezone PST -8
clock summer-time PDT recurring
ntp update-calendar
ntp server 131.108.13.57
ntp server 131.108.11.58
interface Ethernet 0/0
ntp broadcast
vines time use-system
```

In the following example, a Cisco 7000 has no outside time source, so it uses the calendar as an authoritative time source and distributes the time via NTP broadcast packets.

```
clock timezone MET 2
clock calendar-valid
ntp master
interface fddi 0/0
ntp broadcast
```

Examples of Modifying Buffers

In the following example, the system will try to keep at least 50 small buffers free:

```
buffers small min-free 50
```

In the following example, the system will try to keep no more than 200 medium buffers free:

```
buffers middle max-free 200
```

In the following example, the system will try to create one large temporary extra buffer, just after a reload:

```
buffers large initial 1
```

In the following example, the system will try to create one permanent huge buffer:

```
buffers huge permanent 1
```

Username Examples

The following sample configuration sets up secret passwords on routers A, B, and C, thus enabling the three routers to connect to each other.

To authenticate connections between routers A and B, enter the following commands:

On router A:

```
username B password a-b_secret
```

On router B:

```
username A password a-b_secret
```

To authenticate connections between routers A and C, enter the following commands:

On router A:

```
username C password a-c_secret
```

On router C:

```
username A password a-c_secret
```

To authenticate connections between routers B and C, enter the following commands:

On router B:

```
username C password b-c_secret
```

On router C:

```
username B password b-c_secret
```

When you specify an encryption type of 0 to enter an unencrypted password, the system displays the encrypted version of the password. For example, suppose you enter the following command:

username bill password westward

The system would display this command like this:

```
username bill password 7 21398211
```

The encrypted version of the password is 21398211. The password was encrypted by the Cisco-defined encryption algorithm, as indicated by the "7."

If you were to enter the following command, the system would assume that the password is already encrypted and would do no encryption. It would display the command exactly as you typed it:

username bill password 7 21398211
```
username bill password 7 21398211
```