Cisco IOS
Terminal Services
Command Reference
Release 12.2

Corporate Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 526-4100

Customer Order Number: DOC-7812094=
Text Part Number: 78-12094-02
THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB’s public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED "AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Cisco IOS Terminal Services Command Reference
All rights reserved.
About Cisco IOS Software Documentation vi

Using Cisco IOS Software xiii

Terminal Services Commands TR-1

INDEX
About Cisco IOS Software Documentation

This chapter discusses the objectives, audience, organization, and conventions of Cisco IOS software documentation. It also provides sources for obtaining documentation from Cisco Systems.

Documentation Objectives

Cisco IOS software documentation describes the tasks and commands necessary to configure and maintain Cisco networking devices.

Audience

The Cisco IOS software documentation set is intended primarily for users who configure and maintain Cisco networking devices (such as routers and switches) but who may not be familiar with the tasks, the relationship between tasks, or the Cisco IOS software commands necessary to perform particular tasks. The Cisco IOS software documentation set is also intended for those users experienced with Cisco IOS software who need to know about new features, new configuration options, and new software characteristics in the current Cisco IOS software release.

Documentation Organization

The Cisco IOS software documentation set consists of documentation modules and master indexes. In addition to the main documentation set, there are supporting documents and resources.

Documentation Modules

The Cisco IOS documentation modules consist of configuration guides and corresponding command reference publications. Chapters in a configuration guide describe protocols, configuration tasks, and Cisco IOS software functionality and contain comprehensive configuration examples. Chapters in a command reference publication provide complete Cisco IOS command syntax information. Use each configuration guide in conjunction with its corresponding command reference publication.
Figure 1 shows the Cisco IOS software documentation modules.

Note

The abbreviations (for example, FC and FR) next to the book icons are page designators, which are defined in a key in the index of each document to help you with navigation. The bullets under each module list the major technology areas discussed in the corresponding books.
Module DC/DR:
- Preparing for Dial Access
- Modern and Dial Shelf Configuration and Management
- ISDN Configuration
- Signalling Configuration
- Dial-on-Demand Routing Configuration
- Dial-Backup Configuration
- Dial-Related Addressing Services
- Virtual Templates, Profiles, and Networks
- PPP Configuration
- Callback and Bandwidth Allocation Configuration
- Dial Access Specialized Features
- Dial Access Scenarios

Module TC/TR:
- ARA
- LAT
- NASI
- Telnet
- TN3270
- X.25 PAD
- Protocol Translation

Module BC/B1R:
- Transparent Bridging
- SRB
- Token Ring Inter-Switch Link
- Token Ring Route Switch Module
- RSRB
- DLSw+
- Serial Tunnel and Block Serial Tunnel
- LLC2 and SDLC
- IBM Network Media Translation
- SNA Frame Relay Access
- NCIA Client/Server
- Airline Product Set

Module BC/B2R:
- DSPU and SNA Service Point
- SNA Switching Services
- Cisco Transaction Connection
- Cisco Mainframe Channel Connection
- CLAW and TCP/IP Offload
- CSNA, CMPC, and CMPC+
- TN3270 Server

Module VC/VR:
- Voice over IP
- Call Control Signalling
- Voice over Frame Relay
- Voice over ATM
- Telephony Applications
- Trunk Management
- Fax, Video, and Modem Support

Module QC/QR:
- Packet Classification
- Congestion Management
- Congestion Avoidance
- Policing and Shaping
- Signalling
- Link Efficiency Mechanisms
Master Indexes

Two master indexes provide indexing information for the Cisco IOS software documentation set: an index for the configuration guides and an index for the command references. Individual books also contain a book-specific index.

The master indexes provide a quick way for you to find a command when you know the command name but not which module contains the command. When you use the online master indexes, you can click the page number for an index entry and go to that page in the online document.

Supporting Documents and Resources

The following documents and resources support the Cisco IOS software documentation set:

- **Cisco IOS Command Summary (two volumes)**—This publication explains the function and syntax of the Cisco IOS software commands. For more information about defaults and usage guidelines, refer to the Cisco IOS command reference publications.

- **Cisco IOS System Error Messages**—This publication lists and describes Cisco IOS system error messages. Not all system error messages indicate problems with your system. Some are purely informational, and others may help diagnose problems with communications lines, internal hardware, or the system software.

- **Cisco IOS Debug Command Reference**—This publication contains an alphabetical listing of the `debug` commands and their descriptions. Documentation for each command includes a brief description of its use, command syntax, usage guidelines, and sample output.

- **Dictionary of Internetworking Terms and Acronyms**—This Cisco publication compiles and defines the terms and acronyms used in the internetworking industry.

- New feature documentation—The Cisco IOS software documentation set documents the mainline release of Cisco IOS software (for example, Cisco IOS Release 12.2). New software features are introduced in early deployment releases (for example, the Cisco IOS “T” release train for 12.2, 12.2(x)T). Documentation for these new features can be found in standalone documents called “feature modules.” Feature module documentation describes new Cisco IOS software and hardware networking functionality and is available on Cisco.com and the Documentation CD-ROM.

- Release notes—This documentation describes system requirements, provides information about new and changed features, and includes other useful information about specific software releases. See the section “Using Software Release Notes” in the chapter “Using Cisco IOS Software” for more information.

- Caveats documentation—This documentation provides information about Cisco IOS software defects in specific software releases.

- RFCs—RFCs are standards documents maintained by the Internet Engineering Task Force (IETF). Cisco IOS software documentation references supported RFCs when applicable. The full text of referenced RFCs may be obtained on the World Wide Web at http://www.rfc-editor.org/.

- MIBs—MIBs are used for network monitoring. For lists of supported MIBs by platform and release, and to download MIB files, see the Cisco MIB website on Cisco.com at http://www.cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml.
New and Changed Information

The *Cisco IOS Terminal Services Configuration Guide* and *Cisco IOS Terminal Services Command Reference* were extracted from the Release 12.1 of the *Cisco IOS Dial Services Configuration Guide: Terminal Services and Cisco IOS Dial Services Command Reference*. The basic sequence of chapters pertaining to terminal services seen in the *Cisco IOS Dial Services Configuration Guide: Terminal Services* book can still be seen in the *Cisco IOS Terminal Services Configuration Guide*.

Document Conventions

Within Cisco IOS software documentation, the term *router* is generally used to refer to a variety of Cisco products (for example, routers, access servers, and switches). Routers, access servers, and other networking devices that support Cisco IOS software are shown interchangeably within examples. These products are used only for illustrative purposes; that is, an example that shows one product does not necessarily indicate that other products are not supported.

The Cisco IOS documentation set uses the following conventions:

<table>
<thead>
<tr>
<th>Convention</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>^ or Ctrl</td>
<td>The ^ and Ctrl symbols represent the Control key. For example, the key combination ^D or Ctrl-D means hold down the Control key while you press the D key. Keys are indicated in capital letters but are not case sensitive.</td>
</tr>
<tr>
<td>string</td>
<td>A string is a nonquoted set of characters shown in italics. For example, when setting an SNMP community string to public, do not use quotation marks around the string or the string will include the quotation marks.</td>
</tr>
</tbody>
</table>

Command syntax descriptions use the following conventions:

<table>
<thead>
<tr>
<th>Convention</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>boldface</td>
<td>Boldface text indicates commands and keywords that you enter literally as shown.</td>
</tr>
<tr>
<td>italics</td>
<td>Italic text indicates arguments for which you supply values.</td>
</tr>
<tr>
<td>[x]</td>
<td>Square brackets enclose an optional element (keyword or argument).</td>
</tr>
<tr>
<td>l</td>
<td>A vertical line indicates a choice within an optional or required set of keywords or arguments.</td>
</tr>
<tr>
<td>[x</td>
<td>y]</td>
</tr>
<tr>
<td>{x</td>
<td>y}</td>
</tr>
</tbody>
</table>

Nested sets of square brackets or braces indicate optional or required choices within optional or required elements. For example:

<table>
<thead>
<tr>
<th>Convention</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[x {y</td>
<td>z}]</td>
</tr>
</tbody>
</table>
Examples use the following conventions:

<table>
<thead>
<tr>
<th>Convention</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>screen</td>
<td>Examples of information displayed on the screen are set in Courier font.</td>
</tr>
<tr>
<td>boldface screen</td>
<td>Examples of text that you must enter are set in Courier bold font.</td>
</tr>
<tr>
<td>&lt; &gt;</td>
<td>Angle brackets enclose text that is not printed to the screen, such as passwords.</td>
</tr>
<tr>
<td>!</td>
<td>An exclamation point at the beginning of a line indicates a comment line. (Exclamation points are also displayed by the Cisco IOS software for certain processes.)</td>
</tr>
<tr>
<td>[ ]</td>
<td>Square brackets enclose default responses to system prompts.</td>
</tr>
</tbody>
</table>

The following conventions are used to attract the attention of the reader:

⚠️ Caution
Means reader be careful. In this situation, you might do something that could result in equipment damage or loss of data.

📝 Note
Means reader take note. Notes contain helpful suggestions or references to materials not contained in this manual.

💡 Timesaver
Means the described action saves time. You can save time by performing the action described in the paragraph.

## Obtaining Documentation

The following sections provide sources for obtaining documentation from Cisco Systems.

### World Wide Web

The most current Cisco documentation is available on the World Wide Web at the following website:

http://www.cisco.com

Translated documentation is available at the following website:


### Documentation CD-ROM

Cisco documentation and additional literature are available in a CD-ROM package, which ships with your product. The Documentation CD-ROM is updated monthly and may be more current than printed documentation. The CD-ROM package is available as a single unit or through an annual subscription.
Ordering Documentation

Cisco documentation can be ordered in the following ways:

- Registered Cisco Direct Customers can order Cisco product documentation from the Networking Products MarketPlace:
  http://www.cisco.com/cgi-bin/order/order_root.pl
- Registered Cisco.com users can order the Documentation CD-ROM through the online Subscription Store:
  http://www.cisco.com/go/subscription
- Nonregistered Cisco.com users can order documentation through a local account representative by calling Cisco corporate headquarters (California, USA) at 408 526-7208 or, in North America, by calling 800 553-NETS(6387).

Documentation Feedback

If you are reading Cisco product documentation on the World Wide Web, you can submit technical comments electronically. Click Feedback in the toolbar and select Documentation. After you complete the form, click Submit to send it to Cisco.

You can e-mail your comments to bug-doc@cisco.com.

To submit your comments by mail, use the response card behind the front cover of your document, or write to the following address:

Cisco Systems, Inc.
Document Resource Connection
170 West Tasman Drive
San Jose, CA 95134-9883

We appreciate your comments.

Obtaining Technical Assistance

Cisco provides Cisco.com as a starting point for all technical assistance. Customers and partners can obtain documentation, troubleshooting tips, and sample configurations from online tools. For Cisco.com registered users, additional troubleshooting tools are available from the TAC website.

Cisco.com

Cisco.com is the foundation of a suite of interactive, networked services that provides immediate, open access to Cisco information and resources at anytime, from anywhere in the world. This highly integrated Internet application is a powerful, easy-to-use tool for doing business with Cisco.

Cisco.com provides a broad range of features and services to help customers and partners streamline business processes and improve productivity. Through Cisco.com, you can find information about Cisco and our networking solutions, services, and programs. In addition, you can resolve technical issues with online technical support, download and test software packages, and order Cisco learning materials and merchandise. Valuable online skill assessment, training, and certification programs are also available.
Customers and partners can self-register on Cisco.com to obtain additional personalized information and services. Registered users can order products, check on the status of an order, access technical support, and view benefits specific to their relationships with Cisco.

To access Cisco.com, go to the following website:
http://www.cisco.com

Technical Assistance Center

The Cisco TAC website is available to all customers who need technical assistance with a Cisco product or technology that is under warranty or covered by a maintenance contract.

Contacting TAC by Using the Cisco TAC Website

If you have a priority level 3 (P3) or priority level 4 (P4) problem, contact TAC by going to the TAC website:
http://www.cisco.com/tac

P3 and P4 level problems are defined as follows:

- **P3**—Your network performance is degraded. Network functionality is noticeably impaired, but most business operations continue.
- **P4**—You need information or assistance on Cisco product capabilities, product installation, or basic product configuration.

In each of the above cases, use the Cisco TAC website to quickly find answers to your questions.

To register for Cisco.com, go to the following website:
http://www.cisco.com/register/

If you cannot resolve your technical issue by using the TAC online resources, Cisco.com registered users can open a case online by using the TAC Case Open tool at the following website:
http://www.cisco.com/tac/caseopen

Contacting TAC by Telephone

If you have a priority level 1 (P1) or priority level 2 (P2) problem, contact TAC by telephone and immediately open a case. To obtain a directory of toll-free numbers for your country, go to the following website:

P1 and P2 level problems are defined as follows:

- **P1**—Your production network is down, causing a critical impact to business operations if service is not restored quickly. No workaround is available.
- **P2**—Your production network is severely degraded, affecting significant aspects of your business operations. No workaround is available.
Using Cisco IOS Software

This chapter provides helpful tips for understanding and configuring Cisco IOS software using the command-line interface (CLI). It contains the following sections:

- Understanding Command Modes
- Getting Help
- Using the no and default Forms of Commands
- Saving Configuration Changes
- Filtering Output from the show and more Commands
- Identifying Supported Platforms

For an overview of Cisco IOS software configuration, refer to the *Cisco IOS Configuration Fundamentals Configuration Guide*.

For information on the conventions used in the Cisco IOS software documentation set, see the chapter “About Cisco IOS Software Documentation” located at the beginning of this book.

Understanding Command Modes

You use the CLI to access Cisco IOS software. Because the CLI is divided into many different modes, the commands available to you at any given time depend on the mode you are currently in. Entering a question mark (?) at the CLI prompt allows you to obtain a list of commands available for each command mode.

When you log in to the CLI, you are in user EXEC mode. User EXEC mode contains only a limited subset of commands. To have access to all commands, you must enter privileged EXEC mode, normally by using a password. From privileged EXEC mode you can issue any EXEC command—user or privileged mode—or you can enter global configuration mode. Most EXEC commands are one-time commands. For example, `show` commands show important status information, and `clear` commands clear counters or interfaces. The EXEC commands are not saved when the software reboots.

Configuration modes allow you to make changes to the running configuration. If you later save the running configuration to the startup configuration, these changed commands are stored when the software is rebooted. To enter specific configuration modes, you must start at global configuration mode. From global configuration mode, you can enter interface configuration mode and a variety of other modes, such as protocol-specific modes.

ROM monitor mode is a separate mode used when the Cisco IOS software cannot load properly. If a valid software image is not found when the software boots or if the configuration file is corrupted at startup, the software might enter ROM monitor mode.
Table 1 describes how to access and exit various common command modes of the Cisco IOS software. It also shows examples of the prompts displayed for each mode.

### Table 1 Accessing and Exiting Command Modes

<table>
<thead>
<tr>
<th>Command Mode</th>
<th>Access Method</th>
<th>Prompt</th>
<th>Exit Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>User EXEC</td>
<td>Log in.</td>
<td><code>Router&gt;</code></td>
<td>Use the <code>logout</code> command.</td>
</tr>
<tr>
<td>Privileged EXEC</td>
<td>From user EXEC mode, use the <code>enable</code> EXEC command.</td>
<td><code>Router#</code></td>
<td>To return to user EXEC mode, use the <code>disable</code> command.</td>
</tr>
<tr>
<td>Global configuration</td>
<td>From privileged EXEC mode, use the <code>configure terminal</code> privileged EXEC command.</td>
<td><code>Router(config)#</code></td>
<td>To return to privileged EXEC mode from global configuration mode, use the <code>exit</code> or <code>end</code> command, or press Ctrl-Z.</td>
</tr>
<tr>
<td>Interface configuration</td>
<td>From global configuration mode, specify an interface using an <code>interface</code> command.</td>
<td><code>Router(config-if)#</code></td>
<td>To return to global configuration mode, use the <code>exit</code> command. To return to privileged EXEC mode, use the <code>end</code> command, or press Ctrl-Z.</td>
</tr>
<tr>
<td>ROM monitor</td>
<td>From privileged EXEC mode, use the <code>reload</code> EXEC command. Press the Break key during the first 60 seconds while the system is booting.</td>
<td><code>&gt;</code></td>
<td>To exit ROM monitor mode, use the <code>continue</code> command.</td>
</tr>
</tbody>
</table>

For more information on command modes, refer to the “Using the Command-Line Interface” chapter in the Cisco IOS Configuration Fundamentals Configuration Guide.

### Getting Help

Entering a question mark (?) at the CLI prompt displays a list of commands available for each command mode. You can also get a list of keywords and arguments associated with any command by using the context-sensitive help feature.

To get help specific to a command mode, a command, a keyword, or an argument, use one of the following commands:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>help</code></td>
<td>Provides a brief description of the help system in any command mode.</td>
</tr>
<tr>
<td><code>abbreviated-command-entry?</code></td>
<td>Provides a list of commands that begin with a particular character string. (No space between command and question mark.)</td>
</tr>
<tr>
<td><code>abbreviated-command-entry&lt;Tab&gt;</code></td>
<td>Completes a partial command name.</td>
</tr>
<tr>
<td><code>?</code></td>
<td>Lists all commands available for a particular command mode.</td>
</tr>
<tr>
<td><code>command ?</code></td>
<td>Lists the keywords or arguments that you must enter next on the command line. (Space between command and question mark.)</td>
</tr>
</tbody>
</table>
Example: How to Find Command Options

This section provides an example of how to display syntax for a command. The syntax can consist of optional or required keywords and arguments. To display keywords and arguments for a command, enter a question mark (?) at the configuration prompt or after entering part of a command followed by a space. The Cisco IOS software displays a list and brief description of available keywords and arguments. For example, if you were in global configuration mode and wanted to see all the keywords or arguments for the `arap` command, you would type `arap ?`.

The `<cr>` symbol in command help output stands for “carriage return.” On older keyboards, the carriage return key is the Return key. On most modern keyboards, the carriage return key is the Enter key. The `<cr>` symbol at the end of command help output indicates that you have the option to press Enter to complete the command and that the arguments and keywords in the list preceding the `<cr>` symbol are optional. The `<cr>` symbol by itself indicates that no more arguments or keywords are available and that you must press Enter to complete the command.

Table 2 shows examples of how you can use the question mark (?) to assist you in entering commands. The table steps you through configuring an IP address on a serial interface on a Cisco 7206 router that is running Cisco IOS Release 12.0(3).

**Table 2** How to Find Command Options

<table>
<thead>
<tr>
<th>Command</th>
<th>Comment</th>
</tr>
</thead>
</table>
| **Router> enable**  
Password: <password>  
Router# | Enter the `enable` command and password to access privileged EXEC commands. You are in privileged EXEC mode when the prompt changes to `Router#`. |
| **Router# configure terminal**  
Enter configuration commands, one per line. End with CNTL/Z.  
Router(config)# | Enter the `configure terminal` privileged EXEC command to enter global configuration mode. You are in global configuration mode when the prompt changes to `Router(config)#`. |
| **Router(config)# interface serial ?**  
<0-6> Serial interface number  
Router(config)# interface serial 4 ?  
/  
**Router(config)# interface serial 4/ ?**  
<0-3> Serial interface number  
Router(config)# interface serial 4/0  
Router(config-if)# | Enter interface configuration mode by specifying the serial interface that you want to configure using the `interface serial` global configuration command.  
Enter ? to display what you must enter next on the command line. In this example, you must enter the serial interface slot number and port number, separated by a forward slash.  
You are in interface configuration mode when the prompt changes to `Router(config-if)#`. |
**Table 2  How to Find Command Options (continued)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Comment</th>
</tr>
</thead>
</table>
| Router(config-if)# ?
Interface configuration commands:

- 
- 
- ip
  
  Interface Internet Protocol config commands
- keepalive
  
  Enable keepalive
- lan-name
  
  LAN Name command
- llc2
  
  LLC2 Interface Subcommands
- load-interval
  
  Specify interval for load calculation for an interface
- locaddr-priority
  
  Assign a priority group
- logging
  
  Configure logging for interface
- loopback
  
  Configure internal loopback on an interface
- mac-address
  
  Manually set interface MAC address
- mls
  
  mls router sub/interface commands
- mpoa
  
  MPOA interface configuration commands
- mtu
  
  Set the interface Maximum Transmission Unit (MTU)
- netbios
  
  Use a defined NETBIOS access list or enable name-caching
- no
  
  Negate a command or set its defaults
- nrrz-encoding
  
  Enable use of NRZI encoding
- ntp
  
  Configure NTP
- .
  
  .

Router(config-if)#

Enter ? to display a list of all the interface configuration commands available for the serial interface. This example shows only some of the available interface configuration commands.

<table>
<thead>
<tr>
<th>Command</th>
<th>Comment</th>
</tr>
</thead>
</table>
| Router(config-if)# ip ?
Interface IP configuration subcommands:

- access-group
  
  Specify access control for packets
- accounting
  
  Enable IP accounting on this interface
- address
  
  Set the IP address of an interface
- authentication
  
  authentication subcommands
- bandwidth-percent
  
  Set EIGRP bandwidth limit
- broadcast-address
  
  Set the broadcast address of an interface
- cgmp
  
  Enable/disable CGMP
- directed-broadcast
  
  Enable forwarding of directed broadcasts
- dvmrp
  
  DVMRP interface commands
- hello-interval
  
  Configures IP-EIGRP hello interval
- helper-address
  
  Specify a destination address for UDP broadcasts
- hold-time
  
  Configures IP-EIGRP hold time
- .
  
  .

Router(config-if)# ip

Enter the command that you want to configure for the interface. This example uses the ip command.
Enter ? to display what you must enter next on the command line. This example shows only some of the available interface IP configuration commands.
Almost every configuration command has a **no** form. In general, use the **no** form to disable a function. Use the command without the **no** keyword to reenable a disabled function or to enable a function that is disabled by default. For example, IP routing is enabled by default. To disable IP routing, use the **no ip routing** command; to reenable IP routing, use the **ip routing** command. The Cisco IOS software command reference publications provide the complete syntax for the configuration commands and describe what the **no** form of a command does.

Configuration commands also can have a **default** form, which returns the command settings to the default values. Most commands are disabled by default, so in such cases using the **default** form has the same result as using the **no** form of the command. However, some commands are enabled by default and
have variables set to certain default values. In these cases, the default form of the command enables the command and sets the variables to their default values. The Cisco IOS software command reference publications describe the effect of the default form of a command if the command functions differently than the no form.

**Saving Configuration Changes**

Use the `copy system:running-config nvram:startup-config` command to save your configuration changes to the startup configuration so that the changes will not be lost if the software reloads or a power outage occurs. For example:

```
Router# copy system:running-config nvram:startup-config
Building configuration...
```

It might take a minute or two to save the configuration. After the configuration has been saved, the following output appears:

```
[OK]
Router#
```

On most platforms, this task saves the configuration to NVRAM. On the Class A Flash file system platforms, this task saves the configuration to the location specified by the CONFIG_FILE environment variable. The CONFIG_FILE variable defaults to NVRAM.

**Filtering Output from the show and more Commands**

In Cisco IOS Release 12.0(1)T and later releases, you can search and filter the output of `show` and `more` commands. This functionality is useful if you need to sort through large amounts of output or if you want to exclude output that you need not see.

To use this functionality, enter a `show` or `more` command followed by the “pipe” character (`|`); one of the keywords `begin`, `include`, or `exclude`; and a regular expression on which you want to search or filter (the expression is case-sensitive):

```
command | {begin | include | exclude} regular-expression
```

The output matches certain lines of information in the configuration file. The following example illustrates how to use output modifiers with the `show interface` command when you want the output to include only lines in which the expression “protocol” appears:

```
Router# show interface | include protocol
FastEthernet0/0 is up, line protocol is up
Serial4/0 is up, line protocol is up
Serial4/1 is up, line protocol is up
Serial4/2 is administratively down, line protocol is down
Serial4/3 is administratively down, line protocol is down
```

For more information on the search and filter functionality, refer to the “Using the Command-Line Interface” chapter in the *Cisco IOS Configuration Fundamentals Configuration Guide*, Release 12.2.
Identifying Supported Platforms

Cisco IOS software is packaged in feature sets consisting of software images that support specific platforms. The feature sets available for a specific platform depend on which Cisco IOS software images are included in a release. To identify the set of software images available in a specific release or to find out if a feature is available in a given Cisco IOS software image, see the following sections:

- Using Feature Navigator
- Using Software Release Notes

Using Feature Navigator

Feature Navigator is a web-based tool that enables you to quickly determine which Cisco IOS software images support a particular set of features and which features are supported in a particular Cisco IOS image.

Feature Navigator is available 24 hours a day, 7 days a week. To access Feature Navigator, you must have an account on Cisco.com. If you have forgotten or lost your account information, e-mail the Contact Database Administration group at cdbadmin@cisco.com. If you do not have an account on Cisco.com, go to http://www.cisco.com/register and follow the directions to establish an account.

To use Feature Navigator, you must have a JavaScript-enabled web browser such as Netscape 3.0 or later, or Internet Explorer 4.0 or later. Internet Explorer 4.0 always has JavaScript enabled. To enable JavaScript for Netscape 3.x or Netscape 4.x, follow the instructions provided with the web browser. For JavaScript support and enabling instructions for other browsers, check with the browser vendor.

Feature Navigator is updated when major Cisco IOS software releases and technology releases occur. You can access Feature Navigator at the following URL:

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

Using Software Release Notes

Cisco IOS software releases include release notes that provide the following information:

- Platform support information
- Memory recommendations
- Microcode support information
- Feature set tables
- Feature descriptions
- Open and resolved severity 1 and 2 caveats for all platforms

Release notes are intended to be release-specific for the most current release, and the information provided in these documents may not be cumulative in providing information about features that first appeared in previous releases.
Terminal Services Commands

This chapter describes the commands used to configure, connect to, and maintain Cisco IOS terminal services. The commands are listed alphabetically. Some commands required for configuring terminal services may be found in other Cisco IOS command references. Use the command reference master index or search online to find these commands.

For interface configuration tasks and examples, refer to the Cisco IOS Terminal Services Configuration Guide, Release 12.2.
**absolute-timeout**

To set the interval for closing the connection, use the `absolute-timeout` command in line configuration mode. To restore the default, use the `no` form of this command.

```
absolute-timeout minutes

no absolute-timeout
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>minutes</code></td>
<td>Number of minutes after which the user session will be terminated.</td>
</tr>
</tbody>
</table>

**Defaults**

No timeout interval is automatically set.

**Command Modes**

Line configuration

**Command History**

- **Release** 11.0  
  Modification This command was introduced.

**Usage Guidelines**

Use the `absolute-timeout` command line configuration command to configure the EXEC to terminate when the configured number of minutes occurs on the virtual terminal (vty) line. The `absolute-timeout` command terminates the connection after the specified time period has elapsed, regardless of whether the connection is being used at the time of termination. You can specify an absolute-timeout value for each port. The user is given 20 seconds notice before the session is terminated. You can use this command along with the `logout-warning` command to notify users of an impending logout.

Cisco IOS software also provides the `session-timeout` and `exec-timeout` line configuration commands for releasing lines when they have been idle for too long.

You can set the `absolute-timeout` command and an AppleTalk Remote Access Protocol (ARAP) timeout for the same line; however, this command supersedes any timeouts set in ARAP. Additionally, ARAP users will receive no notice of any impending termination if you use this command.

**Examples**

The following example sets an interval of 60 minutes on line 5:

```
line 5
absolute-timeout 60
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>exec-timeout</td>
<td>Sets the interval that the EXEC command interpreter waits until user input is detected.</td>
</tr>
<tr>
<td>logout-warning</td>
<td>Sets and displays a warning for users about an impending forced timeout.</td>
</tr>
<tr>
<td>session-timeout</td>
<td>Sets the interval for closing the connection on a console or terminal line.</td>
</tr>
</tbody>
</table>
access-class (LAT)

To define restrictions on incoming and outgoing connections, use the `access-class` command in line configuration mode. To remove the access list number, use the `no` form of this command.

```
access-class access-list-number [in | out]
no access-class access-list-number
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>access-list-number</strong></td>
<td>Specifies an integer from 1 to 199 that defines the access list.</td>
</tr>
<tr>
<td><strong>in</strong></td>
<td>Controls which nodes can make local-area transport (LAT) connections into the server.</td>
</tr>
<tr>
<td><strong>out</strong></td>
<td>Defines the access checks made on outgoing connections. (A user who types a node name at the system prompt to initiate a LAT connection is making an outgoing connection.)</td>
</tr>
</tbody>
</table>

**Defaults**

Disabled

**Command Modes**

Line configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

This command defines access list numbers that will then be used with the `lat access-list` command to specify the access conditions.

The value supplied for the `access-list-number` argument is used for all protocols supported by the Cisco IOS software. If you are already using an IP access list, you must define LAT and possibly X.25 access lists permitting connections to all devices, to emulate the behavior of previous software versions.

When both IP and LAT connections are allowed from a terminal line and an IP access list is applied to that line with the `access-class` line configuration command, you must also create a LAT access list with the same number if you want to allow any LAT connections from that terminal. You can specify only one incoming and one outgoing access list number for each terminal line. When checking LAT access lists, if the specified list does not exist, the system denies all LAT connections.

**Examples**

The following example configures an incoming access class on vty line 4:

```
line vty 4
access-class 4 in
```
**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>lat access-list</code></td>
<td>Specifies access conditions to nodes on the LAT network.</td>
</tr>
</tbody>
</table>
arap dedicated

To configure a line to be used only as an AppleTalk Remote Access (ARA) connection, use the arap dedicated command in line configuration mode. To return the line to interactive mode, use the no form of this command.

```
arap dedicated
no arap dedicated
```

Syntax Description

This command has no arguments or keywords.

Defaults

Disabled

Command Modes

Line configuration

Command History

```
<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>
```

Examples

The following example configures line 3 to be used only for ARA connections:

```
line 3
arap dedicated
```
arap enable

To enable AppleTalk Remote Access (ARA) for a line, use the arap enable command in line configuration mode. To disable ARA, use the no form of this command.

    arap enable
    no arap enable

Syntax Description
This command has no arguments or keywords.

Defaults
Disabled

Command Modes
Line configuration

Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

Examples
The following example enables ARA on a line:

    line 3
    arap enable

Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>appletalk routing</td>
<td>Enables AppleTalk routing.</td>
</tr>
<tr>
<td>autoselect</td>
<td>Configures a line to start an ARA, PPP, or SLIP session.</td>
</tr>
</tbody>
</table>
arap net-access-list

To control Apple Macintosh access to networks, use the `arap net-access-list` command in line configuration mode. To return to the default setting, use the `no` form of this command.

`arap net-access-list net-access-list-number`

`no arap net-access-list net-access-list-number`

**Syntax Description**

- `net-access-list-number` One of the `list` values configured using the AppleTalk `access-list cable-range`, `access-list includes`, `access-list network`, `access-list other-access`, or `access-list within` commands.

**Defaults**

Disabled. The Macintosh has access to all networks.

**Command Modes**

Line configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

You can use the `arap net-access-list` command to apply access lists defined by the `access-list cable-range`, `access-list includes`, `access-list network`, `access-list other-access`, or `access-list within` commands.

You cannot use the `arap net-access-list` command to apply access lists defined by the `access-list zone` or `access-list additional-zones` commands.

**Examples**

In the following example, ARA is enabled on line 3 and the Macintosh will have access to the AppleTalk access list numbered 650:

```
line 3
arap enable
arap net-access-list 650
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>access-list cable-range</code></td>
<td>Defines an AppleTalk access list for a cable range (for extended networks only).</td>
</tr>
<tr>
<td><code>access-list includes</code></td>
<td>Defines an AppleTalk access list that overlaps any part of a range of network numbers or cable ranges (for both extended and nonextended networks).</td>
</tr>
<tr>
<td><code>access-list network</code></td>
<td>Defines an AppleTalk access list for a single network number (that is, for a nonextended network).</td>
</tr>
<tr>
<td>Command</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>access-list other-access</td>
<td>Defines the default action to take for subsequent access checks that apply to networks or cable ranges.</td>
</tr>
<tr>
<td>access-list within</td>
<td>Defines an AppleTalk access list for an extended or a nonextended network whose network number or cable range is included entirely within the specified cable range.</td>
</tr>
<tr>
<td>arap zonelist</td>
<td>Controls which zones the Apple Macintosh client sees.</td>
</tr>
</tbody>
</table>
**arap network**

To create a new network or zone and cause it to be advertised, use the `arap network` command in global configuration mode. To prevent a new network or zone from being advertised, use the `no` form of this command.

```
arap network [network-number] [zone-name]
no arap network
```

**Syntax Description**

- `network-number` (Optional) AppleTalk network number. The network number must be unique on your AppleTalk network. This network is where all AppleTalk Remote Access (ARAP) users appear when they dial in to the network.

- `zone-name` (Optional) AppleTalk zone name.

**Defaults**

A new network or zone is not created.

**Command Modes**

Global configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

This is a required command. ARAP does not run without it in Cisco IOS Release 10.2 and later.

**Examples**

The following example creates a new zone:

```
arap network 400 test zone
```
arap noguest

To prevent Apple Macintosh guests from logging in to the router, use the `arap noguest` command in line configuration mode. To remove this restriction, use the `no` form of this command.

```
arap noguest [if-needed]

no arap noguest
```

**Syntax Description**

<table>
<thead>
<tr>
<th>if-needed</th>
<th>(Optional) Does not authenticate if the user already provided authentication. This allows users to log in as guests if they have already been authenticated through a username or password.</th>
</tr>
</thead>
</table>

**Defaults**

Disabled

**Command Modes**

Line configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

A guest is a person who connects to the network without needing to give a name or a password.

**Note**

You should not use the `arap noguest` command if you are using modified Common Command Language (CCL) scripts and the `login tacacs` command.

**Examples**

The following example prohibits guests from logging in to the router:

```
line 3
    arap enable
    arap noguest
```


**arap require-manual-password**

To require users to enter their password manually at the time they log in, use the `arap require-manual-password` command in line configuration mode. To disable the manual password-entry requirement, use the `no` form of this command.

```
arap require-manual-password
no arap require-manual-password
```

**Syntax Description**

This command has no arguments or keywords.

**Defaults**

Disabled

**Command Modes**

Line configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>11.1</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

This command only works for AppleTalk Remote Access Protocol (ARAP) 2.0 connections.

**Examples**

The following example forces users to enter their passwords manually at the time they log in, rather than use a saved password:

```
arap require-manual-password
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>enable password</td>
<td>Sets a local password to control access to various privilege levels.</td>
</tr>
<tr>
<td>login (line)</td>
<td>Enables password checking at login and defines the method (local or TACACS+).</td>
</tr>
<tr>
<td>resume (setting X.3 PAD parameters)</td>
<td>Specifies an IP address, an address from a specific IP address pool, or an address from the DHCP mechanism to be returned to a remote peer connecting to this interface.</td>
</tr>
</tbody>
</table>
arap timelimit

To set the maximum length of an AppleTalk Remote Access (ARA) session for a line, use the `arap timelimit` command in line configuration mode. To return to the default of unlimited session length, use the no form of this command.

```
arap timelimit [minutes]
no arap timelimit
```

### Syntax Description

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>minutes</code></td>
<td>(Optional) Maximum length of time, in minutes, for a session.</td>
</tr>
</tbody>
</table>

### Defaults

Unlimited session length

### Command Modes

Line configuration

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

### Usage Guidelines

After the specified length of time, the session will be terminated.

### Examples

The following example specifies a maximum length of 20 minutes for ARA sessions:

```
line 3
  arap enable
  arap timelimit 20
```

### Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>arap warningtime</code></td>
<td>Sets when a disconnect warning message is displayed.</td>
</tr>
</tbody>
</table>
arap warningtime

To set when a disconnect warning message is displayed, use the `arap warningtime` command in line configuration mode. To disable this function, use the `no` form of this command.

```
arap warningtime [minutes]
no arap warningtime
```

**Syntax Description**

- `minutes` (Optional) Amount of time, in minutes, before the configured session time limit. At the configured amount of time before a session is to be disconnected, the router sends a message to the Apple Macintosh client, which causes a warning message to appear on the user screen.

**Defaults**

Disabled

**Command Modes**

Line configuration

**Command History**

- **Release** 10.0  This command was introduced.

**Usage Guidelines**

This command can only be used if a session time limit has been configured on the line.

**Examples**

The following example shows a line configured for 20-minute AppleTalk Remote Access (ARA) sessions, with a warning 17 minutes after the session is started:

```
line 3
arap enable
arap dedicated
arap timelimit 20
arap warningtime 3
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>arap timelimit</code></td>
<td>Sets the maximum length of an ARA session for a line.</td>
</tr>
</tbody>
</table>
**arap zonelist**

To control which zones the Apple Macintosh client sees, use the **arap zonelist** command in line configuration mode. To disable the default setting, use the **no** form of this command.

```
arap zonelist zone-access-list-number
no arap zonelist zone-access-list-number
```

**Syntax Description**

- **zone-access-list-number**  
  One of the list values configured using the AppleTalk **access-list zone** or **access-list additional-zones** command.

**Defaults**

Disabled. The Macintosh will see all defined zones.

**Command Modes**

Line configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

You can use the **arap zonelist** command to apply access lists defined by the **access-list zone** and **access-list additional-zones** commands.

You cannot use the **arap zonelist** command to apply access lists defined by the **access-list network** command.

Hiding a zone from users is not the same as preventing them from sending and receiving packets from the networks that make up that zone. For true security, an **arap net-access-list** command must be issued to prevent traffic to and from those networks.

**Examples**

The following example enables AppleTalk Remote Access (ARA) on line 3; the Macintosh will see only zones permitted by access list 650.

```
line 3
  arap enable
  arap zonelist 650
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>arap net-access-list</td>
<td>Controls Apple Macintosh access to networks.</td>
</tr>
</tbody>
</table>
async default ip address

The `async default ip address` command is replaced by the `peer default ip address` command. See the description of the `peer default ip address` command for more information.
autocommand

To automatically execute a command when a user connects to a particular line, use the `autocommand` command in line configuration mode. To disable the automatic execution, use the `no` form of this command.

```
autocommand command

no autocommand command
```

<table>
<thead>
<tr>
<th>Syntax Description</th>
<th><code>command</code></th>
<th>Any appropriate EXEC command, including the host name and any switches that occur with the EXEC command.</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Defaults</th>
<th>No commands are configured to automatically execute.</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Command Modes</th>
<th>Line configuration</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Command History</th>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Usage Guidelines</th>
<th>This command enables you to automatically execute an EXEC command when a user connects to a line.</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Examples</th>
<th>The following example forces an automatic connection to a host named host21 (which could be an IP address):</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>line vty 4</td>
</tr>
<tr>
<td></td>
<td>autocommand connect host21</td>
</tr>
</tbody>
</table>
To create a “host failed” message that displays when a connection fails, use the **busy-message** command in global configuration mode. To disable the “host failed” message from displaying on the specified host, use the **no** form of this command.

```console
busy-message host-name d message d
```

```console
no busy-message host-name
```

### Syntax Description

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>host-name</strong></td>
<td>Name of the host that cannot be reached.</td>
</tr>
<tr>
<td><strong>d</strong></td>
<td>Delimiting character of your choice—a pound sign (#), for example. You cannot use the delimiting character in the message.</td>
</tr>
<tr>
<td><strong>message</strong></td>
<td>Message text.</td>
</tr>
</tbody>
</table>

### Defaults

No message is displayed.

### Command Modes

Global configuration

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

### Usage Guidelines

This command applies only to Telnet connections.

Follow the **busy-message** command with one or more blank spaces and a delimiting character of your choice. Then enter one or more lines of text, terminating the message with the second occurrence of the delimiting character.

Defining a “host failed” message for a host prevents all Cisco IOS software-initiated user messages, including the initial message that indicates the connection is “Trying....” The **busy-message** command can be used in the **autocommand** command to suppress these messages.

### Examples

The following example sets a message that will be displayed on the terminal whenever an attempt to connect to the host named dross fails. The pound sign (#) is used as a delimiting character.

```console
busy-message dross #
Cannot connect to host. Contact the computer center.
#```
clear entry

To delete an entry from the list of queued host-initiated connections, use the `clear entry` command in EXEC mode.

```
clear entry number
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>number</code></td>
<td>An entry number obtained from the <code>show entry</code> EXEC command.</td>
</tr>
</tbody>
</table>

**Command Modes**

EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Examples**

The following example deletes pending entry number 3 from the queue:

```
Router# clear entry 3
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>show entry</code></td>
<td>Displays the list of queued host-initiated connections to a router.</td>
</tr>
</tbody>
</table>
connect

To log in to a host that supports Telnet, rlogin, or local-area transport (LAT), use the connect command in EXEC mode.

```
connect host [port] [keyword]
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>host</code></td>
<td>A host name or an IP address.</td>
</tr>
<tr>
<td><code>port</code></td>
<td>(Optional) A decimal TCP port number; the default is the Telnet router port (decimal 23) on the host.</td>
</tr>
<tr>
<td><code>keyword</code></td>
<td>(Optional) One of the keywords listed in Table 3.</td>
</tr>
</tbody>
</table>

**Command Modes**

EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced in a release prior to Cisco IOS Release 10.0.</td>
</tr>
<tr>
<td>12.1</td>
<td>The <code>/quiet</code> keyword was added.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Table 3 lists the optional connect command keywords.

**Table 3  connect Keyword Options**

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>/debug</code></td>
<td>Enables Telnet debugging mode.</td>
</tr>
<tr>
<td><code>/encrypt kerberos</code></td>
<td>Enables an encrypted Telnet session. This keyword is available only if you have the Kerberized Telnet subsystem. If you authenticate using Kerberos Credentials, the use of this keyword initiates an encryption negotiation with the remote server. If the encryption negotiation fails, the Telnet connection will be reset. If the encryption negotiation is successful, the Telnet connection will be established, and the Telnet session will continue in encrypted mode (all Telnet traffic for the session will be encrypted).</td>
</tr>
<tr>
<td><code>/line</code></td>
<td>Enables Telnet line mode. In this mode, the Cisco IOS software sends no data to the host until you press the Enter key. You can edit the line using the standard Cisco IOS software command editing characters. The <code>/line</code> keyword is a local switch; the remote router is not notified of the mode change.</td>
</tr>
<tr>
<td><code>/noecho</code></td>
<td>Disables local echo.</td>
</tr>
<tr>
<td><code>/quiet</code></td>
<td>Prevents onscreen display of all messages from the Cisco IOS software.</td>
</tr>
<tr>
<td><code>/route path</code></td>
<td>Specifies loose source routing. The <code>path</code> argument is a list of host names or IP addresses that specify network nodes and ends with the final destination.</td>
</tr>
<tr>
<td><code>/source-interface</code></td>
<td>Specifies the source interface.</td>
</tr>
</tbody>
</table>
### Terminal Services Commands

**connect**

**TR-20**

**Cisco IOS Terminal Services Command Reference**

---

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/stream</td>
<td>Turns on <em>stream</em> processing, which enables a raw TCP stream with no Telnet control sequences. A stream connection does not process Telnet options and can be appropriate for connections to ports running UNIX-to-UNIX Copy Program (UUCP) and other non-Telnet protocols.</td>
</tr>
<tr>
<td>port-number</td>
<td>Port number.</td>
</tr>
<tr>
<td>bgp</td>
<td>Border Gateway Protocol.</td>
</tr>
<tr>
<td>chargen</td>
<td>Character generator.</td>
</tr>
<tr>
<td>cmd rcmd</td>
<td>Remote commands.</td>
</tr>
<tr>
<td>daytime</td>
<td>Daytime.</td>
</tr>
<tr>
<td>discard</td>
<td>Discard.</td>
</tr>
<tr>
<td>domain</td>
<td>Domain Naming Service.</td>
</tr>
<tr>
<td>echo</td>
<td>Echo.</td>
</tr>
<tr>
<td>exec</td>
<td>EXEC.</td>
</tr>
<tr>
<td>finger</td>
<td>Finger.</td>
</tr>
<tr>
<td>ftp</td>
<td>File Transfer Protocol.</td>
</tr>
<tr>
<td>ftp-data</td>
<td>FTP data connections (used infrequently).</td>
</tr>
<tr>
<td>gopher</td>
<td>Gopher.</td>
</tr>
<tr>
<td>hostname</td>
<td>Host name server.</td>
</tr>
<tr>
<td>ident</td>
<td>Ident Protocol.</td>
</tr>
<tr>
<td>irc</td>
<td>Internet Relay Chat.</td>
</tr>
<tr>
<td>klogin</td>
<td>Kerberos login.</td>
</tr>
<tr>
<td>kshell</td>
<td>Kerberos shell.</td>
</tr>
<tr>
<td>login</td>
<td>Login (rlogin).</td>
</tr>
<tr>
<td>lpd</td>
<td>Printer service.</td>
</tr>
<tr>
<td>nntp</td>
<td>Network News Transport Protocol.</td>
</tr>
<tr>
<td>node</td>
<td>Connect to a specific LAT node.</td>
</tr>
<tr>
<td>pop2</td>
<td>Post Office Protocol v2.</td>
</tr>
<tr>
<td>pop3</td>
<td>Post Office Protocol v3.</td>
</tr>
<tr>
<td>port</td>
<td>Destination LAT port name.</td>
</tr>
<tr>
<td>smtp</td>
<td>Simple Mail Transport Protocol.</td>
</tr>
<tr>
<td>sunrpc</td>
<td>Sun Remote Procedure Call.</td>
</tr>
<tr>
<td>syslog</td>
<td>Syslog.</td>
</tr>
<tr>
<td>tacacs</td>
<td>Specify TACACS security.</td>
</tr>
<tr>
<td>talk</td>
<td>Talk.</td>
</tr>
<tr>
<td>telnet</td>
<td>Telnet.</td>
</tr>
<tr>
<td>time</td>
<td>Time.</td>
</tr>
<tr>
<td>uucp</td>
<td>UNIX-to-UNIX Copy Program.</td>
</tr>
</tbody>
</table>
**Table 3** connect Keyword Options (continued)

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/stream</td>
<td>Turns on <em>stream</em> processing, which enables a raw TCP stream with no Telnet control sequences. A stream connection does not process Telnet options and can be appropriate for connections to ports running UNIX-to-UNIX Copy Program (UUCP) and other non-Telnet protocols.</td>
</tr>
<tr>
<td>port-number</td>
<td>Port number.</td>
</tr>
<tr>
<td>bgp</td>
<td>Border Gateway Protocol.</td>
</tr>
<tr>
<td>chargen</td>
<td>Character generator.</td>
</tr>
<tr>
<td>cmd rcmd</td>
<td>Remote commands.</td>
</tr>
<tr>
<td>daytime</td>
<td>Daytime.</td>
</tr>
<tr>
<td>discard</td>
<td>Discard.</td>
</tr>
<tr>
<td>domain</td>
<td>Domain Naming Service.</td>
</tr>
<tr>
<td>echo</td>
<td>Echo.</td>
</tr>
<tr>
<td>exec</td>
<td>EXEC.</td>
</tr>
<tr>
<td>finger</td>
<td>Finger.</td>
</tr>
<tr>
<td>ftp</td>
<td>File Transfer Protocol.</td>
</tr>
<tr>
<td>ftp-data</td>
<td>FTP data connections (used infrequently).</td>
</tr>
<tr>
<td>gopher</td>
<td>Gopher.</td>
</tr>
<tr>
<td>hostname</td>
<td>Host name server.</td>
</tr>
<tr>
<td>ident</td>
<td>Ident Protocol.</td>
</tr>
<tr>
<td>irc</td>
<td>Internet Relay Chat.</td>
</tr>
<tr>
<td>klogin</td>
<td>Kerberos login.</td>
</tr>
<tr>
<td>kshell</td>
<td>Kerberos shell.</td>
</tr>
<tr>
<td>login</td>
<td>Login (rlogin).</td>
</tr>
<tr>
<td>lpd</td>
<td>Printer service.</td>
</tr>
<tr>
<td>nntp</td>
<td>Network News Transport Protocol.</td>
</tr>
<tr>
<td>node</td>
<td>Connect to a specific LAT node.</td>
</tr>
<tr>
<td>pop2</td>
<td>Post Office Protocol v2.</td>
</tr>
<tr>
<td>pop3</td>
<td>Post Office Protocol v3.</td>
</tr>
<tr>
<td>port</td>
<td>Destination LAT port name.</td>
</tr>
<tr>
<td>smtp</td>
<td>Simple Mail Transport Protocol.</td>
</tr>
<tr>
<td>sunrpc</td>
<td>Sun Remote Procedure Call.</td>
</tr>
<tr>
<td>syslog</td>
<td>Syslog.</td>
</tr>
<tr>
<td>tacacs</td>
<td>Specify TACACS security.</td>
</tr>
<tr>
<td>talk</td>
<td>Talk.</td>
</tr>
<tr>
<td>telnet</td>
<td>Telnet.</td>
</tr>
<tr>
<td>time</td>
<td>Time.</td>
</tr>
<tr>
<td>uucp</td>
<td>UNIX-to-UNIX Copy Program.</td>
</tr>
</tbody>
</table>
With the Cisco IOS software implementation of TCP/IP, you are not required to enter the `connect`, `telnet`, `lat`, or `rlogin` commands to establish a terminal connection. You can enter only the learned host name—as long as the host name is different from a command word in the Cisco IOS software. The Cisco IOS software assigns a logical name to each connection, and several commands use these names to identify connections. The logical name is the same as the host name, unless that name is already in use, or you change the connection name with the `name-connection` EXEC command. If the name is already in use, the Cisco IOS software assigns a null name to the connection. To display a list of the available hosts, use the `show hosts` command. To display the status of all TCP connections, use the `show tcp` command.

**Examples**

The following example establishes an encrypted Telnet session from a router to a remote host named host1:

```bash
Router> connect host1 /encrypt kerberos
```

The following example routes packets from the source system named host1 to kl.sri.com, then to 10.1.0.11, and finally back to host1:

```bash
Router> connect host1 /route:kl.sri.com 10.1.0.11 host1
```

The following example connects to a host with logical name host1:

```bash
Router> host1
```

The following example suppresses all onscreen messages from the Cisco IOS software during login and logout:

```bash
Router> connect host2 /quiet
```

The following example shows the limited messages displayed when connection is done using the optional `/quiet` keyword:

```bash
login: User2
Password:
Welcome to OpenVMS VAX version V6.1 on node CRAW
Last interactive login on Tuesday, 15-DEC-1998 11:01
Last non-interactive login on Sunday, 3-JAN-1999 22:32
Server3] logout
User2 logged out at 16-FEB-2000 09:38:27.85
```

---

**Table 3: connect Keyword Options (continued)**

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>whois</td>
<td>Nickname.</td>
</tr>
</tbody>
</table>
### Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>kerberos clients mandatory</strong></td>
<td>Causes the <code>rsh</code>, <code>rcp</code>, <code>rlogin</code>, and <code>telnet</code> commands to fail if they cannot negotiate the Kerberos Protocol with the remote server.</td>
</tr>
<tr>
<td><strong>l2f ignore-mid-sequence</strong></td>
<td>Specifies a connection to a particular LAT node that offers LAT services.</td>
</tr>
<tr>
<td><code>rlogin</code></td>
<td>Logs in to a UNIX host using rlogin.</td>
</tr>
<tr>
<td><code>telnet</code></td>
<td>Logs in to a host that supports Telnet.</td>
</tr>
</tbody>
</table>
flush-at-activation

To discard any data or noise characters that are sitting in the input buffer of the asynchronous line before the line is activated, use the `flush-at-activation` command in line configuration mode. To keep any data or noise characters that are sitting in the input buffer of the asynchronous line before the line is activated, use the `no` form of this command.

```
flush-at-activation
no flush-at-activation
```

**Syntax Description**
This command has no keywords or arguments.

**Defaults**
Enabled by default.

**Command Modes**
Line configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>11.1(5)</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**
For an incoming call on a line configured with modem control (using the `modem inout` and `modem dialin` commands), the line will be activated when the data set ready (DSR) signal goes high and will be dropped when the DSR signal goes low. While the line is idle, its input buffer may receive characters; for example, modem result codes such as “NO CARRIER” or “RING” or line noise. Such characters are not useful to the line application. Flushing the line input buffer when the DSR goes high using the `flush-at-activation` command is the preferred behavior.

To know whether the DSR signal is going high or low, use the `debug modem` command or the `show line` command. Output of these commands displays the status of DSR signal.

On most Cisco IOS platforms, there may be up to a one-second delay between when the DSR signal goes high and Cisco IOS activates the line. Therefore, some valid data received from the line may be discarded when you issue the `flush-at-activation` command. If it is important to process this valid data rather than discarding it and the application is tolerant of receiving bad data, configure the `no flush-at-activation` command.

The application that is used determines whether the system can differentiate the valid data from the bad data or the system is tolerant of receiving any data. For example, consider that the application used is TCP over IP over PPP. PPP uses a Frame Check Sequence (FCS) in a data frame format to verify the integrity of the received data. If an invalid data pattern is delivered to a PPP receiver, PPP will discard it as a framing or FCS error. So the bad data will not be delivered to the higher layers. Even if some data is delivered up to IP and TCP, TCP has its own FCS which will reject bad data. Therefore, the application is tolerant of receiving the bad data that the line delivers.
Consider another application where incoming character data received from the line is delivered as TCP payload to a server running a pager application. Unless the pager application has implemented its own protocol to verify data integrity, this bad data may cause the pager not to be delivered, or to deliver bad data within the message payload to the receiving pager. So the bad data should not be delivered as payload to the line.

Where an upper-layer framed protocol such as PPP or Serial Line Internet Protocol (SLIP) is always used (asynchronous mode dedicated), the framed protocol may reach link status more quickly when you issue the `no flush-at-activation` command. Since the framed protocol discards any erroneous data received, you do not have to use the `flush-at-activation` command.

If the line application is not tolerant of receiving bad data; for example, when you are using character-mode username/password authentication, always use the `flush-at-activation` command. Otherwise, the bad data may trigger an application failure.

Prior to Cisco IOS Release 12.2, the `no flush-at-activation` command was the default on AS5000 platforms with modem ISDN channel aggregation (MICA) and NextPort modems. However, from Cisco IOS Release 12.3 and later, there is no longer any significant delay between when the modem link reaches steady state (DSR high) and when the line is activated so you do not need to use the `no flush-at-activation` command.

The modem state STEADY_STATE is mapped to DSR high and TERMINATING is mapped to DSR low when asynchronous lines are the internal digital modem ports.

**Examples**

The following example shows how to configure lines 1/0 through 1/59 to flush any data in their input buffers when the lines are activated:

```
Router(config)# line 1/0 1/59
Router(config-line)# flush-at-activation
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>activation-character</td>
<td>Defines the character entered at a vacant terminal to begin a terminal session.</td>
</tr>
<tr>
<td>debug modem</td>
<td>Observes modem line activity on an access server.</td>
</tr>
<tr>
<td>modem dialin</td>
<td>Configures a line to enable a modem attached to the router to accept incoming calls only.</td>
</tr>
<tr>
<td>modem inout</td>
<td>Configures a line for both incoming and outgoing calls.</td>
</tr>
<tr>
<td>show line</td>
<td>Displays parameters of a terminal line.</td>
</tr>
</tbody>
</table>
ip alias

To assign an IP address to the service provided on a TCP port, use the ip alias command in interface configuration mode. To remove the specified address for the router, use the no form of this command:

```
ip alias ip-address tcp-port
no ip alias ip-address
```

<table>
<thead>
<tr>
<th>Syntax Description</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ip-address</td>
<td>Specifies the IP address for the service.</td>
</tr>
<tr>
<td>tcp-port</td>
<td>Specifies the number of the TCP port.</td>
</tr>
</tbody>
</table>

**Defaults**

No default behavior or values.

**Command Modes**

Interface configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

A user attempting to establish a connection is connected to the first free line in a rotary group using the Telnet protocol.

The IP address must be on the same network or subnet as the main address of the terminal server, and must not be used by another host on that network or subnet. Connecting to the IP address has the same effect as connecting to the main address of the router, using the argument tcp-port as the TCP port.

You can use the ip alias command to assign multiple IP addresses to the router. For example, in addition to the primary alias address, you can specify addresses that correspond to lines or rotary groups. Using the ip alias command in this way makes connection to a specific rotary group transparent to the user.

When asynchronous mode is implemented, the Cisco IOS software creates the appropriate IP aliases, which map the asynchronous addresses for the lines to which they are connected. This process is automatic and does not require configuration.

**Examples**

The following example configures connections to IP address 172.30.42.42 to act identically to connections made to the primary IP address of the server on TCP port 3001. In other words, a user is connected to the first free line on port 1 of the rotary group that uses the Telnet protocol.

```
ip alias 172.30.42.42 3001
```
**ipx nasi-server enable**

To enable NetWare Access Server Interface (NASI) clients to connect to asynchronous devices attached to your router, use the `ipx nasi-server enable` command in global configuration mode. To prevent NASI clients from connecting through a router, use the `no` form of this command.

```
ipx nasi-server enable
no ipx nasi-server enable
```

**Syntax Description**
This command has no arguments or keywords.

**Defaults**
NASI is not enabled.

**Command Modes**
Global configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>11.1</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**
When you issue this command, NASI clients can connect to any port on the router other than the console port to access network resources. When the user on the NASI client uses the Windows or DOS application to connect to the router, a list of available TTY and vty lines appears, beginning with TTY1. The user selects the desired outgoing TTY or vty port.

You can configure TACACS+ security on the router so that after the user selects a tty or vty port, a username and password prompt appear for authentication, authorization, and accounting purposes.

**Examples**
The following example shows a minimum configuration to enable NASI client dial-in access with TACACS+ authentication:

```
ipx routing
ipx internal-network ncs001
interface ethernet 0
ipx network 1
ipx nasi-server enable
! Enable TACACS+ authentication for NASI clients using the list name swami.
aaa authentication nasi swami tacacs+
line 1 8
modem inout
```
### Terminal Services Commands

#### ipx nasi-server enable

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>aaa authentication nasi</code></td>
<td>Specifies AAA authentication for NASI clients connecting through the access server.</td>
</tr>
<tr>
<td><code>nasi authentication</code></td>
<td>Enables AAA authentication for NASI clients connecting to a router.</td>
</tr>
<tr>
<td><code>show ipx nasi connections</code></td>
<td>Displays the status of NASI connections.</td>
</tr>
<tr>
<td><code>show ipx spx-protocol</code></td>
<td>Displays the status of the SPX protocol stack and related counters.</td>
</tr>
</tbody>
</table>
keymap

To define specific characteristics of keyboard mappings, use the keymap command in global configuration mode. To remove the named keymap from the current image of the configuration file, use the no form of this command.

    keymap keymap-name keymap-entry
    no keymap keymap-name

Syntax Description

<table>
<thead>
<tr>
<th>keymap-name</th>
<th>Name of the file containing the keyboard mappings. The name can be up to 32 characters long and must be unique.</th>
</tr>
</thead>
<tbody>
<tr>
<td>keymap-entry</td>
<td>Commands that define the keymap.</td>
</tr>
</tbody>
</table>

Defaults

VT100 keyboard emulation

Command Modes

Global configuration

Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

Usage Guidelines

The keymap command maps individual keys on a non-TN3270 keyboard to perform the function defined for the TN3270 keyboard. Use the show keymap EXEC command to test for the availability of a keymap.

Do not use the name default for a ttycap entry filename or the Cisco IOS software will adopt the newly defined entry as the default.

The guidelines for creating a keymap follow.

The Keymap Entry Structure

A keymap is a keyboard map file. A keymap consists of an entry for a keyboard. The first part of a keymap lists the names of the keyboards that use that entry. These names will often be the same as in the ttycaps (terminal emulation) file, and often the terminals from various ttycap entries will use the same keymap entry. For example, both 925 and 925vb (for 925 with visual bells) terminals would probably use the same keymap entry. There are other circumstances in which it is necessary to specify a keyboard name as the name of the entry (for example, if a user requires a custom key layout).

After the names, which are separated by vertical bars (|), comes an open brace ({), the text that forms the definitions, the a close brace (}), as follows:

    ciscodefault{
    clear = ‘^z’;\n    flinp = ‘^x’;\n    enter = ‘^m’;\n    delete = ‘^d’ | ‘^?’;\n    synch = ‘^r’;\n    

Each definition consists of a reserved keyword, which identifies the TN3270 function, followed by an equal sign (=), followed by the various ways to generate this particular function, followed by a semicolon (;), as follows:

```
pa1 = '^p1'; pa2 = '^p2'; pa3 = '^p3';
```

Each alternative way to generate the function is a sequence of ASCII characters enclosed inside single quotes (‘’); the alternatives are separated by vertical bars (|), as follows:

```
delete = '^d' | '^?';
```

Inside the single quotes, a few characters are special. A caret (^) specifies that the next character is a control (Ctrl) character. The two-character string caret-a (^a) represents Ctrl-a. The caret-A sequence (^A) generates the same code as caret-a (^a). To generate Delete (or DEL), enter the caret-question mark (^?) sequence.

The Ctrl-caret combination (Ctrl-^), used to generate a hexadecimal 1E, is represented as two caret symbols in sequence (^—)—not as a caret-backslash caret combination (\^). In addition to the caret, a letter can be preceded by a backslash (\). Because this sequence has little effect for most characters, its use is usually not recommended. In the case of a single quote (‘’), the backslash prevents that single quote from terminating the string. In the case of a caret (^), the backslash prevents the caret from having its special meaning. To include the backslash in the string, place two backslashes (\) in the keymap. Table 4 lists other supported special characters.

### Table 4: Special Characters Supported by TN3270 Keypad Capability

<table>
<thead>
<tr>
<th>Character</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>\E</td>
<td>Escape character</td>
</tr>
<tr>
<td>\n</td>
<td>Newline</td>
</tr>
<tr>
<td>\t</td>
<td>Tab</td>
</tr>
<tr>
<td>\r</td>
<td>Carriage return</td>
</tr>
</tbody>
</table>
Each character in a string needs not be enclosed within single quotes. For example, \E\E\E means three escape characters.

To enter a keymap, provide a unique name for it and explicitly define all special keys you intend to include in it within open and close braces. Also, except for the last line, each line must be terminated with a backslash symbol (\). The last line ends with the closing bracket (}) symbol and an end-of-line character.

**Keymap Restrictions**

When IBM-style TN3270 terminals are emulated, a mapping must be performed between sequences of keys pressed at an ASCII keyboard and the keys available on a TN3270 keyboard. For example, a TN3270 keyboard has a key labeled EEOF that erases the contents of the current field from the location of the cursor to the end. To accomplish this function, the terminal user and a program emulating a TN3270 keyboard must agree on which keys will be typed to invoke the function. The requirements for these sequences follow:

- The first character of the sequence must be outside of the standard ASCII printable characters.
- No sequence can be a complete subset of another sequence (although sequences can share partial elements).

Following are examples of acceptable keymap entries:

```plaintext
pfkl = '\E1';
pfk2 = '\E2';
```

Following are examples of unacceptable keymap entries:

```plaintext
pfkl = '\E1';
pfkll = '\E11';
```

In the acceptable example, the keymap entry for `pfkl` is not completely included in the keymap entry for `pfk2`. By contrast, in the unacceptable, or conflicting keymap pair, the sequence used to represent `pfkl` is a complete subset of the sequence used to represent `pfkll`. See the keymap entry provided later in the Examples section of how various keys can be represented to avoid this kind of conflict.

**Table 5** lists TN3270 key names that are supported in this keymap. Note that some of the keys do not exist on a TN3270 keyboard. An unsupported function will cause the Cisco IOS software to send a (possibly visual) bell sequence to the terminal.

<table>
<thead>
<tr>
<th>TN3270 Key Name</th>
<th>Functional Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LPRT</td>
<td>Local print¹</td>
</tr>
<tr>
<td>DP</td>
<td>Duplicate character</td>
</tr>
<tr>
<td>FM</td>
<td>Field mark character</td>
</tr>
<tr>
<td>CURSEL</td>
<td>Cursor select</td>
</tr>
<tr>
<td>CENTSIGN</td>
<td>EBCDIC cent sign</td>
</tr>
<tr>
<td>RESHOW</td>
<td>Redisplay the screen</td>
</tr>
<tr>
<td>EINP</td>
<td>Erase input</td>
</tr>
<tr>
<td>EEOF</td>
<td>Erase end of field</td>
</tr>
<tr>
<td>DELETE</td>
<td>Delete character</td>
</tr>
<tr>
<td>INSRT</td>
<td>Toggle insert mode</td>
</tr>
</tbody>
</table>
**Table 5  TN3270 Key Names Supported by Defaults Keymap (continued)**

<table>
<thead>
<tr>
<th>TN3270 Key Name</th>
<th>Functional Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TAB</td>
<td>Field tab</td>
</tr>
<tr>
<td>BTAB</td>
<td>Field back tab</td>
</tr>
<tr>
<td>COLTAB</td>
<td>Column tab</td>
</tr>
<tr>
<td>COLBAK</td>
<td>Column back tab</td>
</tr>
<tr>
<td>INDENT</td>
<td>Indent one tab stop</td>
</tr>
<tr>
<td>UNDENT</td>
<td>Undent one tab stop</td>
</tr>
<tr>
<td>NL</td>
<td>New line</td>
</tr>
<tr>
<td>HOME</td>
<td>Home the cursor</td>
</tr>
<tr>
<td>UP</td>
<td>Up cursor</td>
</tr>
<tr>
<td>DOWN</td>
<td>Down cursor</td>
</tr>
<tr>
<td>RIGHT</td>
<td>Right cursor</td>
</tr>
<tr>
<td>LEFT</td>
<td>Left cursor</td>
</tr>
<tr>
<td>SETTAB</td>
<td>Set a column tab</td>
</tr>
<tr>
<td>DELTAB</td>
<td>Delete a column tab</td>
</tr>
<tr>
<td>SETMRG</td>
<td>Set left margin</td>
</tr>
<tr>
<td>SETHOM</td>
<td>Set home position</td>
</tr>
<tr>
<td>CLRTAB</td>
<td>Clear all column tabs</td>
</tr>
<tr>
<td>APLON</td>
<td>Apl on¹</td>
</tr>
<tr>
<td>APOFF</td>
<td>Apl off¹</td>
</tr>
<tr>
<td>APLEND¹</td>
<td>Treat input as ASCII</td>
</tr>
<tr>
<td>PCON</td>
<td>Xon/xoff on¹</td>
</tr>
<tr>
<td>PCOFF</td>
<td>Xon/xoff off¹</td>
</tr>
<tr>
<td>DISC</td>
<td>Disconnect (suspend)³</td>
</tr>
<tr>
<td>INIT</td>
<td>New terminal type¹</td>
</tr>
<tr>
<td>ALTK</td>
<td>Alternate keyboard dvorak³</td>
</tr>
<tr>
<td>FLINP</td>
<td>Flush input</td>
</tr>
<tr>
<td>ERASE</td>
<td>Erase last character</td>
</tr>
<tr>
<td>WERASE</td>
<td>Erase last word¹</td>
</tr>
<tr>
<td>FERASE</td>
<td>Erase field</td>
</tr>
<tr>
<td>SYNCH</td>
<td>In synchronization with the user</td>
</tr>
<tr>
<td>RESET</td>
<td>Reset key–unlock keyboard</td>
</tr>
<tr>
<td>MASTER_RESET</td>
<td>Reset, unlock, and redisplay</td>
</tr>
<tr>
<td>XOFF</td>
<td>Please hold output¹</td>
</tr>
<tr>
<td>XON</td>
<td>Please give me output¹</td>
</tr>
<tr>
<td>WORDTAB</td>
<td>Tab to beginning of next word¹</td>
</tr>
<tr>
<td>WORDBACKTAB</td>
<td>Tab to beginning of current or last word¹</td>
</tr>
</tbody>
</table>
Table 5  
**TN3270 Key Names Supported by Defaults Keymap (continued)**

<table>
<thead>
<tr>
<th>TN3270 Key Name</th>
<th>Functional Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>WORDEND</td>
<td>Tab to end of current or next word¹</td>
</tr>
<tr>
<td>FIELDEND</td>
<td>Tab to last nonblank of current or next unprotected (writable) field¹</td>
</tr>
<tr>
<td>PA1</td>
<td>Program attention 1</td>
</tr>
<tr>
<td>PA2</td>
<td>Program attention 2</td>
</tr>
<tr>
<td>PA3</td>
<td>Program attention 3</td>
</tr>
<tr>
<td>CLEAR</td>
<td>Local clear of the TN3270 screen</td>
</tr>
<tr>
<td>TREQ</td>
<td>Test request</td>
</tr>
<tr>
<td>ENTER</td>
<td>Enter key</td>
</tr>
<tr>
<td>PFK1 to PFK30</td>
<td>Program function key 1 program function key 30</td>
</tr>
<tr>
<td>ATTN</td>
<td>Attention</td>
</tr>
<tr>
<td>SYSREQ</td>
<td>System request</td>
</tr>
</tbody>
</table>

¹ Not supported by the Cisco TN3270 implementation.

Table 6 lists the proper keys used to emulate each TN3270 function when default key mappings are used.

**Table 6  Keys Used to Emulate Each TN3270 Function with Defaults Keymap**

<table>
<thead>
<tr>
<th>Key Types</th>
<th>IBM TN3270 Key</th>
<th>Defaults Keys</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cursor movement keys</td>
<td>New Line</td>
<td>Ctrl-n or Home</td>
</tr>
<tr>
<td></td>
<td>Tab</td>
<td>Ctrl-i</td>
</tr>
<tr>
<td></td>
<td>Back Tab</td>
<td>Ctrl-b</td>
</tr>
<tr>
<td></td>
<td>Back Tab</td>
<td>Ctrl-b</td>
</tr>
<tr>
<td></td>
<td>Cursor Left</td>
<td>Ctrl-h</td>
</tr>
<tr>
<td></td>
<td>Cursor Right</td>
<td>Ctrl-l</td>
</tr>
<tr>
<td></td>
<td>Cursor Up</td>
<td>Ctrl-k</td>
</tr>
<tr>
<td></td>
<td>Cursor Down</td>
<td>Ctrl-j or LINE FEED</td>
</tr>
<tr>
<td>Edit control keys</td>
<td>Delete Char</td>
<td>Ctrl-d or RUB</td>
</tr>
<tr>
<td></td>
<td>Erase EOF</td>
<td>Ctrl-e</td>
</tr>
<tr>
<td></td>
<td>Erase Input</td>
<td>Ctrl-w</td>
</tr>
<tr>
<td></td>
<td>Insert Mode</td>
<td>ESC-Space¹</td>
</tr>
<tr>
<td></td>
<td>End Insert</td>
<td>ESC-Space</td>
</tr>
</tbody>
</table>
The following example is the default entry used by the TN3270 emulation software when it is unable to locate a valid keymap in the active configuration image. Table 5 lists the key names supported by the default Cisco TN3270 keymap.

ciscodefault{
clear = '^z';
flinp = '^x';
enter = '^m';
delete = '^d' | '^?';
synch = '^r';
reshow = '^v';
ebcdic_xx='string'
eeof = '^e';
tab = '^i';
btab = '^b';
nl = '^n';
left = '^h';
right = '^l';
up = '^k';
down = '^j';
einp = '^w';
}
reset = '^t';
ferase = '^u';
insrt = '\E ';
pa1 = '^p1'; pa2 = '^p2'; pa3 = '^p3';
pfk1 = '\E1'; pfk2 = '\E2'; pfk3 = '\E3'; pfk4 = '\E4';
pfk5 = '\E5'; pfk6 = '\E6'; pfk7 = '\E7'; pfk8 = '\E8';
pfk9 = '\E9'; pfk10 = '\E0'; pfk11 = '\E-'; pfk12 = '\E=';
pfk13 = '\E!'; pfk14 = '\E@'; pfk15 = '\E#'; pfk16 = '\ES';
pfk17 = '\ES'; pfk18 = '\E'; pfk19 = '\E&'; pfk20 = '\E*';
pfk21 = '\E('; pfk22 = '\E)'; pfk23 = '\E_'; pfk24 = '\E+';
}

The following keymap statement maps the “|” character to send EBCDIC 0x6A:

=ebcdic_6f='|'

<table>
<thead>
<tr>
<th>Related Commands</th>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>keymap-type</td>
<td>Specifies the keyboard map for a terminal connected to the line.</td>
</tr>
<tr>
<td></td>
<td>show keymap</td>
<td>Tests the availability of a keymap after a connection on a router takes place.</td>
</tr>
<tr>
<td></td>
<td>terminal-type</td>
<td>Specifies the type of terminal connected to a line.</td>
</tr>
</tbody>
</table>
keymap-type

To specify the keyboard map for a terminal connected to the line, use the `keymap-type` command in line configuration mode. To reset the keyboard type for the line to the default, use the `no` form of this command.

```
keymap-type keymap-name

no keymap-type
```

### Syntax Description

| keymap-name | Name of a keymap defined within the configuration file of the router. The TN3270 terminal-type negotiations use the specified keymap type when setting up a connection with the remote host. |

### Defaults

VT100

### Command Modes

Line configuration

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

### Usage Guidelines

This command must follow the corresponding `keymap` global configuration entry in the configuration file. The TN3270 terminal-type negotiations use the specified keymap type when setting up a connection with the remote host.

Setting the keyboard to a different keymap requires that a keymap be defined with the Cisco IOS software configuration either by obtaining a configuration file over the network that includes the keymap definition or by defining the keyboard mapping using the `keymap` global configuration command.

Use the command `show keymap` EXEC command to test for the availability of a keymap.

### Examples

The following example sets the keyboard mapping to a keymap named vt100map:

```
line 3
keymap-type vt100map
```

### Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>show keymap</code></td>
<td>Tests the availability of a keymap after a connection on a router takes place.</td>
</tr>
<tr>
<td><code>ttycap</code></td>
<td>Defines characteristics of a terminal emulation file.</td>
</tr>
</tbody>
</table>
**lat**

To connect to a local-area transport (LAT) host, use the `lat` command in EXEC mode.

```
lat name [node nodename | port portname | /debug]
```

### Syntax Description

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>LAT-learned service name.</td>
</tr>
<tr>
<td>node</td>
<td>(Optional) Specifies a connection to a particular LAT node that offers a</td>
</tr>
<tr>
<td>nodename</td>
<td>service. If you do not include the node name option, the node with the</td>
</tr>
<tr>
<td></td>
<td>highest rating offering the service is used. Use the <code>show lat nodes</code> EXEC</td>
</tr>
<tr>
<td></td>
<td>command to display information about all known LAT nodes.</td>
</tr>
<tr>
<td>port</td>
<td>(Optional) Specifies a destination LAT port name. This keyword is ignored</td>
</tr>
<tr>
<td>portname</td>
<td>in most time-sharing systems, but is used by routers and network access</td>
</tr>
<tr>
<td></td>
<td>servers offering reverse LAT services. Reverse LAT involves connecting to</td>
</tr>
<tr>
<td></td>
<td>one router from another, so that the target router runs the host portion of</td>
</tr>
<tr>
<td></td>
<td>the protocol. Enter the port name in the format of the remote system as the</td>
</tr>
<tr>
<td></td>
<td><code>portname</code> argument.</td>
</tr>
<tr>
<td>/debug</td>
<td>(Optional) Enables a switch to display parameter changes and other special</td>
</tr>
<tr>
<td></td>
<td>messages.</td>
</tr>
</tbody>
</table>

### Command Modes

EXEC

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>11.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

### Usage Guidelines

After entering the `lat` command, you can quit the connection by pressing Ctrl-C, or complete the connection by entering the password for a given service.

You can have several concurrent LAT sessions open and switch between them. To open a subsequent session, first enter the escape sequence (`Ctrl-Shift-6` then `x | Ctrl^x` by default) to suspend the current session. Then open a new session.

To list the available LAT services, use the `show lat services` EXEC command.

You can temporarily define the list of services to which you or another user can connect. To do so, use the `terminal lat out-group` command to define the group code lists used for connections from specific lines.

To exit a session, simply log off the remote system. Then terminate an active LAT session by entering the `exit` command.

If your preferred transport is set to `lat`, you can use the `connect` command in place of the `lat` command. Refer to the chapter “Configuring Terminal Operating Characteristics for Dial-In Sessions” in the Cisco IOS Terminal Services Configuration Guide for more information about configuring a preferred transport type. When your preferred transport is set to `none` or to another protocol, you must use the `lat` command to connect to a LAT host.
The following sample command and output shows a LAT connection from the router named Router_A to host eng2:

```
Router_A> lat eng2
Trying ENG2...Open
   ENG2 – VAX/VMS V5.2
Username: JSmith
Password:
   Welcome to VAX/VMS version V5.2 on node ENG2
   Last interactive login on Friday, 1-APR-1994 19:46
```

The system informs you of its progress by displaying the messages “Trying <system>...” and then “Open.” If the connection attempt is not successful, you receive a failure message.

The following sample command establishes a LAT connection from the router named Router_B to a device named our-modems and specifies port 24, which is a special modem:

```
Router_B> lat our-modems port 24
```

The following sample command establishes a LAT connection from the router named Router_C to a device named our-modems and specifies a node named eng:

```
Router_C> lat our-modems node eng
```

The following sample command and output shows the LAT session debugging capability:

```
Router_D> lat Eng2 /debug
Trying ENG2...Open
   ENG2 – VAX/VMS V5.2
Username: JSmith
Password:
   Welcome to VAX/VMS version V5.2 on node ENG2
   Last interactive login on Tuesday, 5-APR-1994 19:02
   [Set Flow out off, Flow in on, Format 8:none, Speed 9600/9600]
   [Set Flow out off, Flow in on, Format 8:none, Speed 9600/9600]
   $
   [Set Flow out off, Flow in on, Format 8:none, Speed 2400/2400]
```

A variety of LAT events are reported, including all requests by the remote system to set local line parameters. The messages within brackets ([ ]) are the messages produced by the remote system setting line characteristics to operating system defaults.

### Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ip alias</td>
<td>Assigns an IP address to the service provided on a TCP port.</td>
</tr>
<tr>
<td>show lat services</td>
<td>Displays information about learned LAT services in the Cisco IOS software.</td>
</tr>
</tbody>
</table>
lat access-list

To specify access conditions to nodes on the local-area transport (LAT) network, use the `lat access-list` command in global configuration mode. To remove a specified access list number, use the `no` form of this command.

```
lat access-list number {permit | deny} nodename

no lat access-list number
```

**Syntax Description**

<table>
<thead>
<tr>
<th>number</th>
<th>Specifies a number from 1 to 199 assigned to the line using the <code>access-class</code> line configuration command.</th>
</tr>
</thead>
<tbody>
<tr>
<td>permit</td>
<td>Allows any matching node name to access the line.</td>
</tr>
<tr>
<td>deny</td>
<td>Denies access to any matching node name.</td>
</tr>
<tr>
<td>nodename</td>
<td>Specifies the name of the LAT node, with or without regular expression pattern matching characters, with which to compare for access. The UNIX-style regular expression characters allow for pattern matching of characters and character strings in the node name.</td>
</tr>
</tbody>
</table>

**Defaults**

No access conditions are defined.

**Command Modes**

Global configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Regular expressions are case sensitive. Because LAT node names are always in all uppercase letters, make sure you use only all uppercase regular expressions.

Table 7 and Table 8 list pattern and character matching symbols and their use. A more complete description of the pattern matching characters is found in the “Regular Expressions” appendix in the *Cisco IOS Terminal Services Configuration Guide*.

**Table 7  Pattern Matching**

<table>
<thead>
<tr>
<th>Character</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>\0</td>
<td>Replaces the entire original address.</td>
</tr>
<tr>
<td>\1..9</td>
<td>Replaces the strings that match the first through ninth parenthesized part of X.121 address.</td>
</tr>
<tr>
<td>*</td>
<td>Matches 0 or more sequences of the regular expressions.</td>
</tr>
<tr>
<td>+</td>
<td>Matches 1 or more sequences of the regular expressions.</td>
</tr>
<tr>
<td>?</td>
<td>Matches the regular expression of the null string.</td>
</tr>
</tbody>
</table>
Table 8  Character Matching

<table>
<thead>
<tr>
<th>Character</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>^</td>
<td>Matches the null string at the beginning of the input string.</td>
</tr>
<tr>
<td>$</td>
<td>Matches the null string at the end of the input string.</td>
</tr>
<tr>
<td>\char</td>
<td>Matches char.</td>
</tr>
<tr>
<td>.</td>
<td>Matches any single character.</td>
</tr>
</tbody>
</table>

Examples

The following example permits all packets destined for any LAT node named WHEEL:

lat access-list 1 permit WHEEL

The following example denies all packets destined for any LAT node name beginning with the BLDG1-prefix:

lat access-list 2 deny ^BLDG1-

Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>accept dialin</td>
<td>Defines access list restrictions on incoming and outgoing connections.</td>
</tr>
</tbody>
</table>
lat enabled

To enable local-area transport (LAT), use the `lat enabled` command in interface configuration mode. To disable LAT, use the `no` form of this command.

```
lat enabled

no lat enabled
```

**Syntax Description**

This command has no arguments or keywords.

**Defaults**

Enabled

**Command Modes**

Interface configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Examples**

The following example enables LAT on Ethernet interface 0:

```
interface ethernet 0
lat enabled
```

The following example disables LAT on the same Ethernet interface:

```
interface ethernet 0
no lat enabled
```
**lat group-list**

To allow a name to be assigned to the group list, use the `lat group-list` command in global configuration mode. To remove the specified group list, use the `no` form of this command.

```
lat group-list groupname {number | range | all} [enabled | disabled]
no lat group-list groupname {number | range | all} [enabled | disabled]
```

**Syntax Description**

- `groupname` Specifies a group code name.
- `number` Specifies a group code number. You can enter both a group code name and group code numbers.
- `range` Specifies a hyphenated range of numbers.
- `all` Specifies the range from 0 to 255.
- `enabled` (Optional) Allows incremental changes to the list; that is, you can add a group code without retyping the entire command.
- `disabled` (Optional) Allows selective removal of a group code from the list.

**Defaults**

None. A group list is any combination of group names, numbers, or ranges. No group names are assigned to the list by default.

**Command Modes**

Global configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>11.1</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Specifying a name for a group list simplifies the task of entering individual group codes. In other words, a name makes it easier to refer to a long list of group code numbers. The group list must already exist. Use the `show lat groups` EXEC command to see a list of existing groups.

**Examples**

The following example creates the new group named stockroom and defines it to include the group numbers 71 and 99:

```
lat group-list stockroom 71 99
```

The following example adds group code 101 to the group named stockroom:

```
lat group-list stockroom 101 enabled
```

The following example deletes the group named Bldg-2:

```
no lat group-list Bldg-2
```
## Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>lat out-group</code></td>
<td>Defines a group list for the outgoing user-initiated connections for a line.</td>
</tr>
<tr>
<td><code>lat service-group</code></td>
<td>Specifies a group code mask to use when advertising all services for this node and to control incoming services.</td>
</tr>
<tr>
<td><code>show lat groups</code></td>
<td>Displays the groups that were defined in the Cisco IOS software.</td>
</tr>
</tbody>
</table>
**lat host-buffers**

To set the number of receive buffers that will be negotiated when the router is acting as a local-area transport (LAT) host, use the `lat host-buffers` command in global configuration mode. To return to the default of one receive buffer, use the `no` form of this command.

```
lat host-buffers receive-buffers

no lat host-buffers receive-buffers
```

**Syntax Description**

- receive-buffers: Integer from 1 to 128 that specifies the number of receive buffers that will be negotiated.

**Defaults**

One receive buffer

**Command Modes**

Global configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Before LAT Version 5.2, LAT allowed only one outstanding message at a time on a virtual circuit. This restriction could limit the performance of large routers. For example, only one Ethernet packet of data could be in transit at a time. With LAT Version 5.2, nodes can indicate that they are willing to receive more than one message at a time. During virtual circuit startup, each side communicates to the other how many outstanding messages it is willing to accept.

**Examples**

The following example enables LAT and configures the LAT host to negotiate 100 receive buffers:

```
lat enabled
lat host-buffers 100
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>lat server-buffers</td>
<td>Sets the number of receive buffers that will be negotiated when the router is acting as a LAT server.</td>
</tr>
</tbody>
</table>
**lat ka-timer**

To set the rate of the keepalive timer, use the `lat ka-timer` command in global configuration mode. To restore the default, use the `no` form of this command.

```
lat ka-timer seconds

no lat ka-timer
```

**Syntax Description**

- **seconds**: Timer rate, in seconds.

**Defaults**

20 seconds

**Command Modes**

Global configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

The keepalive timer sets the rate that messages are sent in the absence of actual traffic between the router and the remote node. The server uses keepalive messages to detect when communication with a remote node is disrupted or when the remote node has crashed.

**Examples**

The following example sets the keepalive timer rate to 5 seconds:

```
lat ka-timer 5
```
To change the local-area transport (LAT) node name without changing the system host name, use the `lat node` command in global configuration mode.

```
lat node node-name
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>node-name</code></td>
<td>Name of the LAT node.</td>
</tr>
</tbody>
</table>

**Defaults**

No default LAT node name

**Command Modes**

Global configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

This command allows you to give the server a node name that is different from the host name. Use the `show entry` EXEC command to determine which LAT hosts have queue entries for printers on the servers. Use the `clear entry` EXEC command to delete entries from the queue.

**Examples**

The following example specifies the LAT node name as DEC2:

```
lst node DEC2
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>hostname</code></td>
<td>Specifies or modifies the host name for the network server.</td>
</tr>
<tr>
<td><code>show entry</code></td>
<td>Displays the list of queued host-initiated connections to a router.</td>
</tr>
</tbody>
</table>
lat out-group

To define a group list for outgoing user-initiated connections on a line, use the **lat out-group** command in line configuration mode. To return to the default value, use the **lat out-group 0** command.

```
lat out-group {groupname number | range | all}
```

### Syntax Description

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>groupname</td>
<td>Group code name.</td>
</tr>
<tr>
<td>number</td>
<td>Group code number. You can also enter both a group code name and group code numbers.</td>
</tr>
<tr>
<td>range</td>
<td>Hyphenated range of numbers.</td>
</tr>
<tr>
<td>all</td>
<td>Range from 0 to 255.</td>
</tr>
</tbody>
</table>

### Defaults

The default group code number is 0.

### Command Modes

Line configuration

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

### Usage Guidelines

You can have values for one, two, or all three arguments. If the **all** keyword is specified, no arguments can be used. You can enter the **groupname**, **number**, and **range** arguments in any order.

Use the **show lat** EXEC command to display group numbers. If the host node and router do not share a common group number, the host services will not be displayed.

### Examples

The following example defines the services for lines 1 through 7, 10 through 17, and 20 through 24. Access to systems on the first set of lines is limited to groups 12 and 18 through 23; the second set is limited to group 12; the third set is limited to group codes 12, 18 through 23, and 44. All other lines use the default of group 0.

```plaintext
line 1 7
  lat out-group 12 18-23
line 10 17
  lat out-group 12
line 20 24
  lat out-group 12 18-23 44
```

### Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>lat group-list</td>
<td>Allows a name to be assigned to the group list, which is any combination of group names, numbers, or ranges.</td>
</tr>
</tbody>
</table>
lat remote-modification

To enable remote local-area transport (LAT) modification of line characteristics (for example, baud rate), use the `lat remote-modification` command in line configuration mode. To disable remote LAT modification of line characteristics, use the `no` form of this command.

```
lat remote-modification
no lat remote-modification
```

**Syntax Description**

This command has no arguments or keywords.

**Defaults**

Remote modification is disabled.

**Command Modes**

Line configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Enabling the line for remote modification allows the remote LAT node to change line characteristics (for example, baud rate, parity, and so on).

**Examples**

The following example enables remote LAT modification on line 4:

```
line 4
lat remote-modification
```
**lat retransmit-limit**

To set the number of times that local-area transport (LAT) resends a message before declaring the remote system unreachable, use the `lat retransmit-limit` command in global configuration mode. To restore the default retry value, use the `no` form of this command.

```
lat retransmit-limit number

no lat retransmit-limit
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>number</code></td>
<td>Number of retries; any number from 4 to 255.</td>
</tr>
</tbody>
</table>

**Defaults**

8 retries

**Command Modes**

Global configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Assigning larger values to the number of tries increases the robustness of the LAT service at the cost of longer delays when communications are disrupted. Because LAT generally resends messages once per second, the value is approximately the number of seconds that LAT connections will survive connection disruption.

If you bridge LAT, the retransmission limit should be set to at least 20 tries for LAT sessions to survive a worst-case spanning-tree reconfiguration, because the time for bridging spanning-tree reconfiguration to be completed can be up to 15 seconds.

**Examples**

The following example sets the retransmission limit to 30 tries, enough time to sustain the downtime incurred when the system must reconfigure a spanning-tree topology:

```
lat retransmit-limit 30
```
lat server-buffers

To set the number of receive buffers that will be negotiated when the router is acting as a local-area transport (LAT) server, use the `lat server-buffers` command in global configuration mode. To return to the default of one receive buffer, use the `no` form of this command.

```
lat server-buffers receive-buffers

no lat server-buffers receive-buffers
```

**Syntax Description**

| receive-buffers | Integer from 1 to 128 that specifies the number of receive buffers that will be negotiated. |

**Defaults**

1 receive buffer

**Command Modes**

Global configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Before LAT Version 5.2, LAT allowed only one outstanding message on a virtual circuit at a time. This restriction could limit the performance of large routers because only one Ethernet packet of data could be in transit at a time. With LAT Version 5.2, nodes can indicate that they are willing to receive more than one message at a time. During virtual circuit startup, each side communicates to the other how many outstanding messages it is willing to accept.

**Examples**

The following example enables LAT and configures the server to negotiate 25 receive buffers:

```
lat enabled
lat server-buffers 25
```
**lat service-announcements**

To reenable local-area transport (LAT) broadcast service announcements, use the `lat service-announcements` command in global configuration mode. To disable the sending of LAT service announcements, use the `no` form of this command.

```
lat service-announcements

no lat service-announcements
```

**Syntax Description**

This command has no arguments or keywords.

**Defaults**

Enabled

**Command Modes**

Global configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

If this command is enabled, the LAT code will periodically broadcast service advertisements. If the command is disabled, the LAT code will not send service announcements, so solicit information messages must be used to look up node information.

**Note**

You should only disable service announcements if all of the nodes on the LAN support the service responder feature.

**Examples**

The following example reenables the sending of broadcast service announcements:

```
lat service-announcements
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>lat service-responder</code></td>
<td>Configures a node to act as proxy for other nodes when a solicit-information multicast message is received.</td>
</tr>
</tbody>
</table>
**lat service enabled**

To enable inbound connections to the specified service and enable the advertisement of this service to routers on the network, use the `lat service enabled` command in global configuration mode. To delete the named service, use the `no` form of this command.

```
lat service service-name enabled

no lat service service-name enabled
```

**Syntax Description**

- `service-name`  Name of the service.

**Defaults**

No services are enabled.

**Command Modes**

Global configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

In the simplest form, this command creates a service that gives connecting users access to a vty port on the server.

Use the `lat service enabled` command after commands that define a service so that users do not connect to a service before all the parameters are set.

Deleting a service does not disconnect existing connections.

**Examples**

The following example enables inbound connections to the service named WHEEL:

```
lat service WHEEL enabled
```
**lat service-group**

To specify a group code mask to use when advertising all services for this node and to control incoming services, use the **lat service-group** command in global configuration mode. To remove the group code mask specified, use the **no** form of this command.

```
lat service-group { groupname | number | range | all } [ enabled | disabled ]
no lat service-group { groupname | number | range | all } [ enabled | disabled ]
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>groupname</code></td>
<td>Specifies a group code name.</td>
</tr>
<tr>
<td><code>number</code></td>
<td>Specifies a group code number.</td>
</tr>
<tr>
<td><code>range</code></td>
<td>Specifies a hyphenated range of numbers from 0 to 255.</td>
</tr>
<tr>
<td><code>all</code></td>
<td>Specifies the range from 0 to 255.</td>
</tr>
<tr>
<td><code>enabled</code></td>
<td>(Optional) Allows incremental changes to the list; that is, you can add a group code without retying the entire command.</td>
</tr>
<tr>
<td><code>disabled</code></td>
<td>(Optional) Allows selective removal of a group code from the list.</td>
</tr>
</tbody>
</table>

**Defaults**

If no service group is specified, the Cisco IOS software defaults to advertising to group 0.

**Command Modes**

Global configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

When this command is written to NVRAM (using the **write memory** EXEC command), the system looks for an exact match on a group code name. If it finds one, it uses that name in the command. Otherwise, it writes out a list of numbers, using the range syntax whenever possible.

**Examples**

The following example specifies groups 100 through 103, then defines engineering as the group code list to advertise:

```
lat group-list engineering 100-103
lat service-group engineering enabled
```

The following example specifies the groups 1, 5, 20 through 36, and 52:

```
lat service-group 1 5 20-36 52
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>lat group-list</strong></td>
<td>Allows a name to be assigned to the group list, which is any combination of group names, numbers, or ranges.</td>
</tr>
</tbody>
</table>
lat service ident

To set the local-area transport (LAT) service identification for a specified service, use the `lat service ident` command in global configuration mode. To remove the identification, use the `no` form of this command.

```
lat service service-name ident identification
no lat service service-name ident
```

**Syntax Description**

<table>
<thead>
<tr>
<th>service-name</th>
<th>Name of the service.</th>
</tr>
</thead>
<tbody>
<tr>
<td>identification</td>
<td>Descriptive name (text only) that identifies the service.</td>
</tr>
</tbody>
</table>

**Defaults**

No LAT service identification is set for specific services.

**Command Modes**

Global configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

The identification is advertised to other servers on the network and is displayed along with the list of name services on the LAN.

**Examples**

The following example specifies the identification “Welcome to Gateway-A” on the service named STELLA:

```
lat service STELLA ident Welcome to Gateway-A
```
**lat service password**

To set up a local-area transport (LAT) password for a service, use the `lat service password` command in global configuration mode. To remove the password, use the `no` form of this command.

```plaintext
lat service service-name password password
no lat service service-name password
```

**Syntax Description**

- `service-name` Name of the service.
- `password` Password used to gain access to the service.

**Defaults**

No default LAT service passwords

**Command Modes**

Global configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

The connecting user will be required to enter the password to complete the connection. The password is obtained through the LAT password mechanism.

**Examples**

The following example specifies a service named BLUE and the password secret:

```plaintext
lat service BLUE password secret
```
### lat service rating

To set a static service rating for the specified service, use the `lat service rating` command in global configuration mode. To remove the service rating, use the `no` form of this command.

```
lat service service-name rating static-rating
no lat service service-name rating
```

#### Syntax Description

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>service-name</td>
<td>Name of the service.</td>
</tr>
<tr>
<td>static-rating</td>
<td>Static service rating. The rating must be in the range from 1 to 255.</td>
</tr>
</tbody>
</table>

#### Defaults

Dynamic rating

#### Command Modes

Global configuration

#### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

#### Usage Guidelines

If this command is not entered, the Cisco IOS software calculates a dynamic rating based on the number of free ports that can handle connections to the service. Setting a static rating overrides this calculation and causes the specified value to be used.

#### Examples

The following example specifies a service rating of 84 on the service named WHEEL:

```
lat service WHEEL rating 84
```
**lat service-responder**

To configure a node to act as proxy for other nodes when a solicit-information multicast message is received, use the `lat service-responder` command in global configuration mode. To remove any proxy definition set up using the `lat service-responder` command, use the `no` form of this command.

```
lat service-responder
no lat service-responder
```

**Syntax Description**

This command has no arguments or keywords.

**Defaults**

Disabled

**Command Modes**

Global configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

The Cisco IOS software can be configured to support the service responder feature that is part of the latest LAT Version 5.2 specification.

Specifically, the DECserver90L+, which has less memory than other Digital Equipment Corporation servers, does not maintain a cache of learned services. Instead, the DECserver90L+ solicits information about services as they are needed.

LAT Version 5.2 nodes can respond for themselves; however LAT Version 5.1 nodes, for example, VMS Version 5.4 or earlier nodes, cannot respond for themselves. Instead, a LAT Version 5.2 node configured as a service responder must respond in proxy for the LAT Version 5.1 nodes.

The Cisco IOS software can be configured as a LAT service responder. If all your nodes are LAT Version 5.2 nodes, you need not enable the service responder features.

**Examples**

The following example configures a node to act as a proxy for a node when a solicit-information multicast message is received. The node configured with this command will respond to solicit messages.

```
lat service-responder
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>lat service-announcements</strong></td>
<td>Reenables LAT broadcast service announcements.</td>
</tr>
</tbody>
</table>
**lat service rotary**

To associate a rotary group with a service, use the `lat service rotary` command in global configuration mode. To remove the association, use the `no` form of this command.

```plaintext
lat service service-name rotary group-number
no lat service service-name rotary
```

**Syntax Description**

- **service-name** Name of the service.
- **group-number** Rotary group number.

**Defaults**

Disabled

**Command Modes**

Global configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Establish rotary groups using the `rotary` line configuration command.

When an inbound connection is received for this service, the router establishes a reverse local-area transport (LAT) connection to a terminal in that rotary group.

If the rotary option is not set, the connection will be to a virtual terminal session on the router.

**Examples**

The following example creates a service named MODEM to establish a rotary group:

```plaintext
lat services MODEM rotary 1
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>rotary</td>
<td>Defines a group of lines consisting of one of more lines.</td>
</tr>
</tbody>
</table>
**lat service-timer**

To adjust the time between local-area transport (LAT) service advertisements, use the `lat service-timer` command in global configuration mode. To return to the default setting, use the `no` form of this command.

```
lat service-timer interval

no lat service-timer
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>interval</code></td>
<td>Number of seconds between service announcements. Note that the granularity offered by this command is 10-second intervals, and the <code>interval</code> value is rounded up.</td>
</tr>
</tbody>
</table>

**Defaults**

20 seconds

**Command Modes**

Global configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>11.1</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

This command adjusts the time, in seconds, between LAT service announcements for services offered by the router. This function is useful in large networks with many LAT services and limited bandwidth.

**Examples**

The following example sets the interval between LAT service advertisements to 11, and illustrates the rough granularity of the `lat service-timer` command:

```
! The time between LAT service advertisements is set to 11. Because the
! granularity is in ten-second intervals, the actual time between advertisement
! is 20 seconds.
l lat service-timer 11
! 20 seconds between updates.
l lat service-timer 19
! 120 seconds between updates.
l lat service-timer 120
```
lat vc-sessions

To set the maximum number of sessions to be multiplexed onto a single local-area transport (LAT) virtual circuit, use the `lat vc-sessions` command in global configuration mode. To remove the definition of a prior session, use the `no` form of this command.

```
lat vc-sessions maximum-number

no lat vc-sessions maximum-number
```

**Syntax Description**

- `maximum-number` Specifies the number of sessions that will be multiplexed onto a single LAT virtual circuit. This number cannot be greater than 255.

**Defaults**

255 sessions per virtual circuit

**Command Modes**

Global configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Setting the number of sessions to a lower number can increase throughput if many sessions are running on one host, especially with routers with many physical ports. It can also increase overhead if the same host has little traffic but a large number of sessions.

**Examples**

The following example sets the maximum number of sessions to be multiplexed onto a single LAT virtual circuit at 100:

```
lat vc-sessions 100
```
**lat vc-timer**

To set the interval of time local-area transport (LAT) waits before sending any traffic, use the `lat vc-timer` command in global configuration mode. To remove a timer definition, use the `no` form of this command.

```
lat vc-timer milliseconds

no lat vc-timer milliseconds
```

**Syntax Description**

| milliseconds | Timer value. Specifies the amount of time LAT will wait before sending traffic. Acceptable values are from 10 to 1000 milliseconds. |

**Defaults**

80 milliseconds

**Command Modes**

Global configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Smaller timer values increase the overhead on both the router and the host. However, you can use smaller values to correct buffer overflows, which happen when the router receives more data than it can buffer during a virtual circuit timer interval.

Larger values increase the need for buffering and can cause noticeable echoing delay. However, increased values can reduce traffic. In environments with slow bridging, retransmissions can be reduced if you increase the value to at least three times the worst-case, round-trip interval.

**Examples**

The following example sets the time between sending messages to 500 milliseconds:

```
lat vc-timer 500
```
To identify a specific line for configuration and enter line configuration collection mode, use the `line` command in global configuration mode.

```
line [aux | console | tty | vty] line-number [ending-line-number]
```

### Syntax Description
- **aux**: (Optional) Auxiliary EIA/TIA-232 DTE port. Must be addressed as relative line 0. The auxiliary port can be used for modem support and asynchronous connections.
- **console**: (Optional) Console terminal line. The console port is DCE.
- **tty**: (Optional) Standard asynchronous line.
- **vty**: (Optional) Virtual terminal line for remote console access.
- **line-number**: Relative number of the terminal line (or the first line in a contiguous group) that you want to configure when the line type is specified. Numbering begins with zero.
- **ending-line-number**: (Optional) Relative number of the last line in a contiguous group that you want to configure. If you omit any keyword, then `line-number` and `ending-line-number` are absolute rather than relative line numbers.

### Defaults
There is no default line.

### Command Modes
Global configuration

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

### Usage Guidelines
You can address a single line or a consecutive range of lines with the `line` command. A line number is necessary, though, and you will receive an error message if you forget to include it.

Entering the `line` command with the optional line type (`aux`, `console`, `tty`, or `vty`) designates the line number as a relative line number. For example, to configure line parameters for line 7 (a TTY line), you could enter the following:

```
line tty 7
```

You also can use the `line` command without specifying a line type. In this case, the line number is treated as an absolute line number. For example, to configure line parameters for line 5, which can be of any type, you could enter the following:

```
line 5
```
Absolute line numbers increment consecutively and can be difficult to manage on large systems. Relative line numbers are a shorthand notation used in configuration. Internally, the Cisco IOS software uses absolute line numbers. You cannot use relative line numbers everywhere, but you can use absolute line numbers everywhere.

The absolute line number of the auxiliary port is 1. The relative line number of the auxiliary port is 0. See the `modem` line configuration command to set up modem support on the auxiliary port.

The software keeps a table of absolute and relative line numbers that you can display with the `show users all` EXEC command. A sample display follows:

```
Router> show users all

Line  User     Host(s)               Idle   Location
0 con 0                                        con2 console
1 tty 1                                        Engineering printer
2 tty 2
3 tty 3           HOST1                1:07   Employee1 x1111
4 tty 4                                        Console E3-D
5 tty 5                                        Mkt. demo area
6 tty 6
7 tty 7           HOST1                  14   Employee2 x1112
10 tty 10

.
.
.
135 tty 135
136 tty 136
137 tty 137           rp4-printer
140 tty 140                                        Braille printer
141 aux 0
142 vty 0   User1     idle                    ROUTER-MAC.CISCO.COM
143 vty 1   User2     idle                    0 HOST1.CISCO.COM
144 vty 2
145 vty 3
146 vty 4
147 vty 5
```

The absolute line numbers are listed at the far left, followed by the line type, and then the relative line number. Relative line numbers always begin numbering at zero and define the type of line. Addressing the second virtual terminal line as line VTY 1, for example, is easier than remembering it as line 143—it’s absolute line number.

The line types are ranked as follows in the line table:

1. Console 0 (con 0)
2. Standard asynchronous line (TTY)
3. Auxiliary port (aux)
4. Virtual terminal line (VTY)
5. Printer

The terminal from which you locally configure the router is attached to the console port. To configure line parameters for the console port, enter the following:

```
line console 0
```

The console relative line number must be 0.
Virtual terminal lines are used to allow remote access to the router. A virtual terminal line is not associated with either the auxiliary or console port. The router has five virtual terminal lines by default. However, you can create additional virtual terminal lines as described in the chapter “Configuring Protocol Translation and Virtual Asynchronous Devices” in the Cisco IOS Terminal Services Configuration Guide.

Configuring the console port or virtual terminal lines allows you to perform such tasks as setting communication parameters, specifying autobaud connections, and configuring terminal operating parameters for the terminal you are using.

**Examples**

The following example starts configuration for virtual terminal lines 0 to 4:

```plaintext
line vty 0 4
```

In the following example, the user creates and configures the maximum 100 virtual terminal lines with the `no login` command:

```plaintext
line vty 0 99
no login
```

In the following example, the user eliminates virtual terminal line number 5 and all higher-numbered virtual terminal lines. Only virtual terminal lines 0 to 4 will remain.

```plaintext
no line vty 5
```

In the following example, the user configures console line 0, auxiliary line 0, and virtual terminal lines 0 to 4:

```plaintext
line vty 0 4
login
line console 0
    password secretWord
line aux 0
    password Mypassword
no exec
    access-class 1 in
    speed 19200
line vty 0
    exec-timeout 0 0
    password Mypassword
line vty 1
    exec-timeout 0 0
    password Mypassword
line vty 2
    exec-timeout 0 0
    password Mypassword
line vty 3
    password Mypassword
line vty 4
    password Mypassword
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>show line</td>
<td>Displays the parameters of a terminal line.</td>
</tr>
<tr>
<td>show users</td>
<td>Displays information about the active lines on the router.</td>
</tr>
</tbody>
</table>
**login (EXEC)**

To change a login username, use the `login` command in EXEC mode.

```plaintext
login
```

**Syntax Description**

This command has no arguments or keywords.

**Command Modes**

EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

You can change a login username if you must match outgoing access list requirements or other login prompt requirements.

When you enter this command, the Cisco IOS software prompts you for a username and password. Enter the new username and the original password. If the username does not match, but the password does, the Cisco IOS software updates the session to the new username with which the `login` command attempt was made.

If no username and password prompts appear when you enter this command, the network administrator did not specify that a username and password be required at login time. If both the username and password are entered correctly, the session becomes associated with the specified username.

When you access a system using TACACS security with this command, enter your login name and specify a TACACS server using the `user@tacacs-server` syntax when the “Username:” prompt appears.

The TACACS server must be one of those defined in a Cisco IOS software configuration file. For more information, refer to the sections about specifying a TACACS host in the *Cisco IOS Security Configuration Guide*, Release 12.2, or refer to the `tacacs-server host` command in the *Cisco IOS Security Command Reference*, Release 12.2.

If you do not specify a host, the Cisco IOS software tries each of the TACACS servers in the list until it receives a response.

If you do specify a host that does not respond, no other TACACS server is queried. The Cisco IOS software will deny access or function according to the action specified by the `tacacs-server last-resort` command, if one is configured.

If you specified a TACACS server host with the `user@tacacs-server` command, the TACACS server specified will be used for all subsequent authentication or notification queries, with the possible exception of Serial Line Internet Protocol (SLIP) address queries.

**Examples**

The following example shows how login usernames and passwords can be changed. In this example, a user currently logged in under the username user1 attempts to change that login name to user2. After entering the `login` command, the user enters the new username, but enters an incorrect password. Because the password does not match the original password, the system rejects the attempt to change the username.
Router> **login**
Username: **user2**
Password: 
% Access denied
Still logged in as "user1"

Next, the user attempts the login change again, with the username user2, but enters the correct (original) password. This time the password matches the current login information, the login username is changed to user2, and the user is allowed access to the EXEC at the user level.

Router> **login**
Username: **user2**
Password: 
Router>

### Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>line-power</td>
<td>Sets up a temporary password on a line.</td>
</tr>
<tr>
<td>lockable</td>
<td>Enables the lock EXEC command.</td>
</tr>
</tbody>
</table>
**login (line)**

This command cannot be used with AAA/TACACS+. Cisco recommends that you use the `login authentication` command instead of the `login` line configuration command. Refer to the *Cisco IOS Security Command Reference* for a description of the `login authentication` command.

To enable password checking at login, use the `login` command in line configuration mode. To disable password checking and allow connections without a password, use the `no` form of this command.

```
login [local | tacacs]
no login
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>local</code></td>
<td>(Optional) Selects local password checking. Authentication is based on the username specified with the <code>username</code> global configuration command.</td>
</tr>
<tr>
<td><code>tacacs</code></td>
<td>(Optional) Selects the TACACS-style user ID and password-checking mechanism.</td>
</tr>
</tbody>
</table>

**Defaults**

Virtual terminals require a password. If you do not set a password for a virtual terminal, it responds to attempted connections by displaying an error message and closing the connection.

**Command Modes**

Line configuration

**Command History**

```
Release  Modification
10.0      This command was introduced.
```

**Usage Guidelines**

If you specify the `login` command without the `local` or `tacacs` option, authentication is based on the password specified with the `password` line configuration command.

**Examples**

The following example sets the password letmein on vty 4:

```
line vty 4
  password letmein
  login
```

The following example enables the TACACS-style user ID and password-checking mechanism:

```
line 0
  password mypassword
  login tacacs
```
<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>enable password</td>
<td>Sets a local password to control access to various privilege levels.</td>
</tr>
<tr>
<td>peer default ip address</td>
<td>Specifies an IP address, an address from a specific IP address pool, or an address from the DHCP mechanism to be returned to a remote peer connecting to this interface.</td>
</tr>
<tr>
<td>virtual-profile aaa</td>
<td>Enables virtual profiles by AAA configuration.</td>
</tr>
</tbody>
</table>
login-string

To define a string of characters that is sent to a host after a successful Telnet connection, use the **login-string** command in global configuration mode. To remove the login string, use the **no** form of this command.

```
login-string host-name d message [ %secp ] [ %secw ] [ %b ] [ %m ] d
no login-string host-name
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>host-name</strong></td>
<td>Specifies the name of the host.</td>
</tr>
<tr>
<td><strong>d</strong></td>
<td>Sets a delimiting character of your choice—a pound sign (#), for example. You cannot use the delimiting character in the busy message.</td>
</tr>
<tr>
<td><strong>message</strong></td>
<td>Specifies the login string.</td>
</tr>
<tr>
<td><strong>%secp</strong></td>
<td>(Optional) Sets a pause in seconds. To insert pauses into the login string, embed a percent sign (%) followed by the number of seconds to pause and the letter “p.”</td>
</tr>
<tr>
<td><strong>%secw</strong></td>
<td>(Optional) Prevents users from issuing commands or keystrokes during a pause.</td>
</tr>
<tr>
<td><strong>%b</strong></td>
<td>(Optional) Sends a Break character.</td>
</tr>
<tr>
<td><strong>%m</strong></td>
<td>(Optional) Supports TN3270 terminals. Sends only CR and no LINE FEED.</td>
</tr>
</tbody>
</table>

**Defaults**

No login strings are defined.

**Command Modes**

Global configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Follow this command with one or more blank spaces and a delimiting character of your choice. Then enter one or more lines of text, terminating the message with the second occurrence of the delimiting character. To use a percent sign in the login string, precede it with another percent sign; that is, type the characters “%%.” The options can be used anywhere within the message string.

This command applies only to rlogin and Telnet sessions.

**Examples**

In the following example, the value %5p causes a 5-second pause:

```
login-string office #ATDT 555-1234
%5p hello
#`
```
To log in to a packet assembler/disassembler (PAD), use the `pad` command in EXEC mode.

```
pad {x121-address \ host-name} [/cud text] [/debug] [/profile name] [/quiet message] [/reverse] [/use-map]
```

**Syntax Description**

- **x121-address**: Specifies the X.121 address of the X.25 host.
- **host-name**: Specifies the X.25 host name if the host-to-address mapping has been set with the `X.25 host` command.
- **/cud text**: (Optional) Includes the specified `text` in the Call User Data (CUD) field of the outgoing Call Request Packet.
- **/debug**: (Optional) Displays the informational level of logging messages whenever the remote host changes an X.3 parameter setting or sends any other X.29 control packet.
- **/profile name**: (Optional) Sets X.3 PAD parameters for the `name` script. Using this keyword and profile name argument is the same as issuing the `x29 profile` global configuration command when translating X.25. If the X.29 profile is set to `default`, the profile is applied to all incoming X.25 PAD calls, including the calls used for protocol translation.
- **/quiet message**: (Optional) Suppresses information messages. Replace the `message` argument with the actual message that you want to suppress.
- **/reverse**: (Optional) Causes reverse-charge calls to be accepted on a per-call (rather than a per-interface) basis.
- **/use-map**: (Optional) Applies `x25 map pad` command entry options (such as CUD and idle) and facilities (such as packet in, packet out, win in, and win out) to the outgoing PAD call. This function occurs only if a matching X.121 destination address exists in an `x25 map pad` command entry.

**Command Modes**

User EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>11.2</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

The `pad` command supports one-word connections. You need not enter the `pad` command; entering the address is enough to start the connection. A PAD can also be accessed and X.3 parameters configured with the `x28 EXEC` command, which uses the standard X.28 user interface.

You can have several PAD connections open at the same time and switch between them. You also can exit a connection and return to the user EXEC prompt at any point. To open a new connection, first exit the current connection by entering the escape sequence (`Ctrl-Shift-6` then `x` [`Ctrl^x`] by default) to return to the EXEC prompt, then open the new connection. If the `/use-map` option is selected on the outgoing `pad` command, the `x25 map pad` command entries are searched for a matching X.121
destination address. If a match is found, the x25 map pad command entry options (such as CUD and idle) and facilities (such as packet in, packet out, win in, and win out) are applied to the outgoing PAD call.

To display information about packet transmission and X.3 PAD parameter settings, use the show x25 pad command. To exit a session, simply log out of the remote system. Then, terminate the active session by entering the exit command.

**Examples**

Use the ? command to display pad command options, as shown in the following example:

```
Router# pad / ?
/cud      Call user data
/debug    Debugging option
/profile  Use a defined X.3 profile
/quiet    Suppress informational messages
/reverse  X25 Address reverse
/use-map  Use x25 map pad command facilities for outgoing Calls
```

The following example starts a PAD session:

```
Router> pad 123456789
Trying 123456789...Open
Router>
```

You can also access a PAD using standard X.28 commands. The following example enters X.28 mode with the x28 EXEC command and configures a PAD with the set X.3 parameter command. The set command sets the idle time delay to 40 seconds.

```
Router# x28
* set 4:40
```

The following example uses the /use-map option to configure a larger window and packet size than the default specified on the interface, and it sets the virtual circuit idle time to 2 seconds. Notice that the map values are used rather than the interface default values.

```
Router-A(config-if)# x25 map pad 2194441 cud gmcmilla windowsize 7 7 packetsize 1024 1024 idle 2
Router-A(config-if)# end
Router-A# %SYS-5-CONFIG_I: Configured from console by console.
Router-A# pad 2194441 /cud gmcmilla /use-map
Trying 2194441....Open
```

<table>
<thead>
<tr>
<th>Related Commands</th>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>translate x25</td>
<td>Automatically translates the request to another outgoing protocol connection type when an X.25 connection request to a particular destination address is received.</td>
</tr>
</tbody>
</table>
resume (setting X.3 PAD parameters)

To set X.3 parameters, use the `resume` command in EXEC mode.

```
resume [connection] [/set parameter:value]
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>connection</code></td>
<td>(Optional) The name or number of the connection; the default is the most recent connection.</td>
</tr>
<tr>
<td><code>/set parameter:value</code></td>
<td>(Optional) Sets the X.3 connection options and packet assembler/disassembler (PAD) parameters for the Cisco IOS software. See Table 9 in the Usage Guidelines for the PAD parameter numbers. Refer to the chapter “Configuring the Cisco PAD Facility for X.25 Connections” of the <em>Cisco IOS Terminal Services Configuration Guide</em>, Release 12.2, for a list of these connection options.</td>
</tr>
</tbody>
</table>

**Defaults**

For outgoing connections, the X.3 parameters default to the following:

2:1, 3:2, 4:1, 7:4, 16:127, 17:21, 18:19

All other parameters default to zero, but can be changed using the `/set` switch option with either the `resume` command or the `x3` command.

For incoming PAD connections, the software sends an X.29 SET PARAMETER packet to set only the following parameters:

2:0, 4:1, 7:21, 15:0

**Command Modes**

EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>9.1</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>
### Usage Guidelines

Table 9 summarizes the X.3 PAD Parameters supported on Cisco devices. Refer to the “X.3 PAD Parameters” appendix in the Cisco IOS Terminal Services Configuration Guide, Release 12.2 for more complete information about these parameters. See Table 36 in this publication for a list of ASCII characters.

#### Table 9  Supported X.3 PAD Parameters

<table>
<thead>
<tr>
<th>Parameter Number</th>
<th>ITU-T Parameter Name</th>
<th>ITU-T X.3 and Cisco Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>PAD recall using a character</td>
<td>Minimum value: 0; maximum value: 126; X.28 PAD user emulation mode default: 1. Not supported by PAD EXEC user interface.</td>
</tr>
<tr>
<td>2</td>
<td>Echo</td>
<td>Minimum value: 0; maximum value: 1; PAD EXEC mode and X.28 PAD user emulation mode default: 1.</td>
</tr>
<tr>
<td>3</td>
<td>Selection of data forwarding character</td>
<td>Minimum value: 0; maximum value: 255; PAD EXEC mode default: 2 (CR); X.28 PAD user emulation mode default: 126 (~).</td>
</tr>
<tr>
<td>4</td>
<td>Selection of idle timer delay</td>
<td>Minimum value: 0; maximum value: 255; PAD EXEC mode default: 1; X.28 PAD user emulation mode default: 0.</td>
</tr>
<tr>
<td>5</td>
<td>Ancillary device control</td>
<td>Minimum value: 0; maximum value: 2; PAD EXEC mode default: 0; X.28 PAD user emulation mode default: 1.</td>
</tr>
<tr>
<td>6</td>
<td>Control of PAD service signals</td>
<td>Minimum value: 0; maximum value: 255; PAD EXEC mode default: 0; X.28 PAD user emulation mode default: 2. Not supported by PAD EXEC user interface.</td>
</tr>
<tr>
<td>7</td>
<td>Action upon receipt of a BREAK signal</td>
<td>Minimum value: 0; maximum value: 31; PAD EXEC mode default: 4; X.28 PAD user emulation mode default: 2.</td>
</tr>
<tr>
<td>8</td>
<td>Discard output</td>
<td>Minimum value: 0; maximum value: 1; PAD EXEC mode and X.28 PAD user emulation mode default: 0.</td>
</tr>
<tr>
<td>9</td>
<td>Padding after Return</td>
<td>Minimum value: 0; maximum value: 255; PAD EXEC mode and X.28 PAD user emulation mode default: 0.</td>
</tr>
<tr>
<td>10</td>
<td>Line folding</td>
<td>Not supported.</td>
</tr>
<tr>
<td>11</td>
<td>DTE speed (binary speed of start-stop mode DTE)</td>
<td>Minimum value: 0; maximum value: 18; PAD EXEC mode and X.28 PAD user emulation mode default: 14.</td>
</tr>
<tr>
<td>12</td>
<td>Flow control of the PAD by the start-stop DTE</td>
<td>Minimum value: 0; maximum value: 1; PAD EXEC mode default: 0; X.28 PAD user emulation mode default: 1.</td>
</tr>
<tr>
<td>13</td>
<td>Line feed insertion (after a Return)</td>
<td>Minimum value: 0; maximum value: 7; PAD EXEC mode and X.28 PAD user emulation mode default: 0.</td>
</tr>
<tr>
<td>14</td>
<td>Line feed padding</td>
<td>Minimum value: 0; maximum value: 255; PAD EXEC mode and X.28 PAD user emulation mode default: 0.</td>
</tr>
<tr>
<td>15</td>
<td>Editing</td>
<td>Minimum value: 0; maximum value: 1; PAD EXEC mode and X.28 PAD user emulation mode default: 0.</td>
</tr>
</tbody>
</table>
The /set switch sets the X.3 parameters defined by parameter number and value, separated by a colon. You set one or more X.3 PAD parameters, as follows:

**Step 1**  Escape out of the current session by pressing the escape sequence (Ctrl-Shift-6 then x [Ctrl^x] by default) and return to the EXEC prompt.

**Step 2**  Issue the where command, to list the open sessions. All open sessions associated with the current terminal line are displayed.

**Step 3**  Enter the resume command, followed by the parameter, a colon, and then the value to be set.

---

**Table 9  Supported X.3 PAD Parameters (continued)**

<table>
<thead>
<tr>
<th>Parameter Number</th>
<th>ITU-T Parameter Name</th>
<th>ITU-T X.3 and Cisco Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>16</td>
<td>Character delete</td>
<td>Minimum value: 0; maximum value: 127; PAD EXEC mode and X.28 PAD user emulation mode default: 127 (DEL).</td>
</tr>
<tr>
<td>17</td>
<td>Line delete</td>
<td>Minimum value: 0; maximum value: 127; PAD EXEC mode default: 21 (NAK or Ctrl-U); X.28 PAD user emulation mode default: 24 (CAN or Ctrl-X).</td>
</tr>
<tr>
<td>18</td>
<td>Line display</td>
<td>Minimum value: 0; maximum value: 127; PAD EXEC mode and X.28 PAD user emulation mode default: 18 (DC2 or Ctrl-R).</td>
</tr>
<tr>
<td>19</td>
<td>Editing PAD service signals</td>
<td>Minimum value: 0; maximum value: 126; PAD EXEC mode default: 0; X.28 PAD user emulation mode default: 2.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Note Not supported by PAD EXEC user interface.</td>
</tr>
<tr>
<td>20</td>
<td>Echo mask</td>
<td>Minimum value: 0; maximum value: 255; PAD EXEC mode and X.28 PAD user emulation mode default: 0.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Note Not supported by PAD EXEC user interface.</td>
</tr>
<tr>
<td>21</td>
<td>Parity treatment</td>
<td>Minimum value: 0; maximum value: 4; PAD EXEC mode and X.28 PAD user emulation mode default: 0.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Note For additional values that can be selected for parameter 21, including parity treatment to conforms to the French Transpac public switched data network and its technical specification and utilization of networks standards (STUR), see Appendix A, “X.3 PAD Parameters,” in the Cisco IOS Terminal Services Configuration Guide, Release 12.2.</td>
</tr>
<tr>
<td>22</td>
<td>Page wait</td>
<td>Not supported.</td>
</tr>
</tbody>
</table>

---

The /set switch sets the X.3 parameters defined by parameter number and value, separated by a colon. You set one or more X.3 PAD parameters, as follows:

**Step 1**  Escape out of the current session by pressing the escape sequence (Ctrl-Shift-6 then x [Ctrl^x] by default) and return to the EXEC prompt.

**Step 2**  Issue the where command, to list the open sessions. All open sessions associated with the current terminal line are displayed.

**Step 3**  Enter the resume command, followed by the parameter, a colon, and then the value to be set.
**Examples**

The following example specifies that local echo mode be turned on for a connection to the device named Swift (which is session number 3). As shown in Table 7, “local echo on” uses the parameter 2 and the value 1 (represented as 2:1 in this example):

```
Swift% ^X
Router> resume 3 /set 2:1
Swift%
```
resume (switching sessions)

To switch to another open Telnet, rlogin, local-area transport (LAT), or packet assembler/disassembler (PAD) session, use the `resume` command in EXEC mode.

```
resume [connection] [keyword] [/set parameter:value]
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>connection</td>
<td>(Optional) The name or number of the connection; the default is the most recent connection.</td>
</tr>
<tr>
<td>keyword</td>
<td>(Optional) One of the options listed in Table 8.</td>
</tr>
<tr>
<td>/set parameter:value</td>
<td>(Optional) Sets PAD parameters for the Cisco IOS software (see Table 7).</td>
</tr>
</tbody>
</table>

**Command Modes**

EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>9.1</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Table 10 lists Telnet and rlogin resume options.

**Table 10  Telnet and rlogin resume Options**

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/debug</td>
<td>Displays parameter changes and messages. In the Cisco IOS software, this option displays informational messages whenever the remote host changes an X.3 parameter, or sends an X.29 control packet.</td>
</tr>
<tr>
<td>/echo</td>
<td>Performs local echo.</td>
</tr>
<tr>
<td>/line</td>
<td>Enables line-mode editing.</td>
</tr>
<tr>
<td>/nodebug</td>
<td>Cancels printing of parameter changes and messages.</td>
</tr>
<tr>
<td>/noecho</td>
<td>Disables local echo.</td>
</tr>
<tr>
<td>/noline(^1)</td>
<td>Disables line mode and enables character-at-a-time mode, which is the default.</td>
</tr>
<tr>
<td>/nostream</td>
<td>Disables stream processing.</td>
</tr>
<tr>
<td>/set parameter:value</td>
<td>Sets X.3 connection options. Refer to the chapter “Configuring the Cisco PAD Facility for X.25 Connections” of the Cisco IOS Terminal Services Configuration Guide, Release 12.2, for a list of these connection options.</td>
</tr>
<tr>
<td>/stream</td>
<td>Enables stream processing.</td>
</tr>
</tbody>
</table>

\(^1\) /noline is the default keyword.
You can have several concurrent sessions open and switch between them. The number of sessions that can be open is defined by the `session-limit` command.

You can switch between sessions by escaping one session and resuming a previously opened session, as follows:

---

**Step 1**

Escape out of the current session by pressing the escape sequence (`Ctrl-Shift-6` then `x` [`Ctrl^x`] by default) and return to the EXEC prompt.

**Step 2**

Enter the `where` command, to list the open sessions. All open sessions associated with the current terminal line are displayed.

**Step 3**

Enter the `resume` command and the session number to make the connection.

You also can resume the previous session by pressing the `Return` key.

---

The `Ctrl^x`, `where`, and `resume` commands are available with all supported connection protocols.

**Examples**

The following example shows how to escape out of a connection and to resume connection 2:

```
Swift% ^x
Router> resume 2
```

You can omit the command name and simply enter the connection number to resume that connection. The following example illustrates how to resume connection 3:

```
Router> 3
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>show tn3270 ascii-hexval</code></td>
<td>Displays ASCII-hexadecimal character mappings.</td>
</tr>
<tr>
<td><code>where</code></td>
<td>Lists open sessions associated with the current terminal line.</td>
</tr>
</tbody>
</table>
To log in to a UNIX host using rlogin, use the rlogin command in EXEC mode.

```
 rlogin host [-l username] [/user username] [/quiet] [debug]
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>host</td>
<td>Specifies the host name or IP address.</td>
</tr>
<tr>
<td>-l username</td>
<td>(Optional) The Berkeley Standard Distribution (BSD) UNIX syntax that specifies a username for the remote login. If you do not use this option, the remote username is your local username.</td>
</tr>
<tr>
<td>/user username</td>
<td>(Optional) The EXEC command syntax that specifies a remote username in the initial exchange with the remote host. The rlogin protocol will not present you with the login prompt.</td>
</tr>
<tr>
<td>/quiet</td>
<td>(Optional) Prevents onscreen display of all messages from the Cisco IOS software.</td>
</tr>
<tr>
<td>debug</td>
<td>(Optional) Enables debugging output from the rlogin protocol.</td>
</tr>
</tbody>
</table>

**Command Modes**

EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.1</td>
<td>The /quiet keyword was added.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

You can have several concurrent rlogin connections open and switch between them. To open a new connection, first suspend the current connection by pressing the escape sequence (Ctrl-Shift-6 then x [Ctrl^x] by default) to return to the EXEC prompt. Then open a new connection. A user cannot automatically log in to a UNIX system from the router, but must provide a user ID and a password for each connection.

If your preferred transport is set to rlogin, you can use the connect command in place of the rlogin command. Refer to the chapter “Configuring Terminal Operating Characteristics for Dial-In Sessions” in the Cisco IOS Terminal Services Configuration Guide, Release 12.2, for more information about configuring a preferred transport type. When your preferred transport is set to none or to another protocol, you must use the rlogin command to connect to a host.

To terminate an active rlogin session, enter one of the following commands at the UNIX prompt of the device to which you are connecting:

- close
- disconnect
- exit
- logout
- quit
Examples

The following example illustrates how a user with the login name jsmith can use the `rlogin ?` help command and the `debug` command mode to establish and troubleshoot a remote connection to the host named Alviso:

```
Router> rlogin ?
WORD IP address or hostname of a remote system
Router> rlogin Alviso ?
-l Specify remote username
/quiet Suppress login/logout messages
/user Specify remote username
debug Enable rlogin debugging output
<cr>
Router> rlogin Alviso -l ?
WORD Remote user name
Router> rlogin Alviso -l jsmith?
debug Enable rlogin debugging output
<cr>
Router> rlogin Alviso -l jsmith debug
```

The following example illustrates debug return on the host named zipper by the user named staff:

```
Router# rlogin zipper.cisco.com -l staff debug
Trying zipper.cisco.com (171.69.63.31)... Open
RLOGIN: local username is: ciscoTS
RLOGIN: remote username is: staff
Password:
Last login: Wed Jun 24 06:15:36 from itech-view3.cisc
1 zipper> uptime
  1:40pm  up 42 day(s), 20:53,  80 users,  load average: 1.44, 2.67, 3.39
2 zipper> logout
[Connection to zipper.cisco.com closed by foreign host]
Router#
```

The following example makes an rlogin connection to a host at address 10.30.21.2 for a user named supervisor and enables the message mode for debugging:

```
Router> rlogin 10.30.21.2 -l supervisor debug
```

The following example makes an rlogin connection to a host named headquarters for the user named admin:

```
Router> rlogin headquarters -l admin
```

The following example suppresses all onscreen messages from the Cisco IOS software during login and logout:

```
Router> rlogin host2 /quiet
```

Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>connect</td>
<td>Logs in to a host that supports Telnet, rlogin, or LAT.</td>
</tr>
<tr>
<td>telnet</td>
<td>Logs in to a host that supports Telnet.</td>
</tr>
</tbody>
</table>
To choose an authentication method for determining the local username to send to the remote rlogin server, use the `rlogin trusted-localuser-source` command in global configuration mode. To restore the default rlogin behavior, use the `no` form of this command.

```
rlogin trusted-localuser-source [local | radius | tacacs]
no rlogin trusted-localuser-source [local | radius | tacacs]
```

### Syntax Description

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>local</td>
<td>(Optional) Uses local username from any authentication method.</td>
</tr>
<tr>
<td>radius</td>
<td>(Optional) Uses local username from RADIUS authentication.</td>
</tr>
<tr>
<td>tacacs</td>
<td>(Optional) Uses local username from TACACS authentication.</td>
</tr>
</tbody>
</table>

### Defaults

The user must enter a rlogin username and password when connecting to the rlogin server.

### Command Modes

Global configuration

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>11.1</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

### Usage Guidelines

Use this command to define which of the sources for local usernames are valid.

The rlogin protocol passes three types of information: the remote username, the local username, and the local host name of the router. The `rlogin trusted-localuser-source` command allows you to configure one of three behaviors for making connections to the rlogin server, as follows:

- The user must enter a login username and password to connect (default).
- The Cisco IOS-authenticated username can be passed to the rlogin server so the user need only enter a password to connect.
- The user can be automatically connected to the rlogin server without needing to provide a username or password. This configuration is made by using both the `rlogin trusted-localuser-source` and `rlogin trusted-remoteuser-source local` commands where both the Cisco IOS authenticated username and the rlogin server username are the same.
**Examples**

The following example uses the local username from RADIUS authentication:

Router# `configure terminal`

Router(config)# `rlogin trusted-localuser-source ?`
- `local` Use local username from any authentication method
- `radius` Use local username from radius authentication
- `tacacs` Use local username from tacacs authentication

Router(config)# `rlogin trusted-localuser-source radius`

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>ip alias</code></td>
<td>Assigns an IP address to the service provided on a TCP port.</td>
</tr>
<tr>
<td><code>retry keepalive</code></td>
<td>Logs in to a UNIX host using rlogin.</td>
</tr>
<tr>
<td><code>rlogin trusted-remoteuser-source local</code></td>
<td>Determines the remote username to send to the remote rlogin server.</td>
</tr>
<tr>
<td><code>template</code></td>
<td>Temporarily defines the list of services to which you or another user can connect.</td>
</tr>
</tbody>
</table>
rlogin trusted-remoteuser-source local

To determine the remote username to send to the remote rlogin server, use the `rlogin trusted-remoteuser-source local` command in global configuration mode. To restore the default rlogin behavior, which is to prompt the user for the remote username, use the `no` form of this command.

```
rlogin trusted-remoteuser-source local
no rlogin trusted-remoteuser-source local
```

**Syntax Description**
This command has no arguments or keywords.

**Defaults**
The user must enter an rlogin username and password when connecting to the rlogin server.

**Command Modes**
Global configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>11.1</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**
The current username is used only if the `rlogin host/user username` command is not enabled. If the current username is not known, rlogin falls back to providing the “login:” prompt to discover a remote username.

After you issue the `rlogin trusted-remoteuser-source local` command, you will not be prompted for a username. The first response you see is the password prompt from the remote system. For example, when this command is not enabled, you must enter your username twice (once at initial system login and once for the `rlogin` command).

⚠️ **Caution**
Configuring the remote host to consider the Cisco router a “trusted” host should be considered a security hole.

**Examples**
The following sample output shows the two prompts a user must reply to when the `rlogin trusted-remoteuser-source local` command is not set:

```
User Access Verification
Username: gmcvilla
Password: xxxxxx

Router> rlogin puli
Trying puli.cisco.com (172.16.3.154)... Open
login: gmcvilla
Password: xxxxxx
```
The following example shows that after you issue the `rlogin trusted-remoteuser-source local` command, you no longer need to specify the username after the `rlogin` command. The username is automatically copied from the user ID of the router:

```
Router# enable
Password: xxxxx
Router# configure terminal
Router(config)# rlogin ?
  trusted-localuser-source   Allowed authentication types for local username
  trusted-remoteuser-source  Method used to get remote username
Router(config)# rlogin trusted-remoteuser-source local
Router(config)# ^Z
```

```
Router# rlogin puli
Trying puli.cisco.com (172.16.3.154)... Open
Password: xxxxx
login:
```

The following example uses the `/user root` keyword option as an override:

```
Router# rlogin puli /user root
Trying puli.cisco.com (172.16.3.154)... Open
Password: xxxxx
login:
```

---

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ip alias</td>
<td>Assigns an IP address to the service provided on a TCP port.</td>
</tr>
<tr>
<td>retry keepalive</td>
<td>Logs in to a UNIX host using rlogin.</td>
</tr>
<tr>
<td>rlogin</td>
<td>Chooses an authentication method for determining the local username to send to the remote rlogin server.</td>
</tr>
<tr>
<td>trusted-localuser-source</td>
<td></td>
</tr>
<tr>
<td>template</td>
<td>Temporarily defines the list of services to which you or another user can connect.</td>
</tr>
</tbody>
</table>
rxspeed

To set the terminal receive speed (how fast the terminal receives information from the modem), use the `rxspeed` command in line configuration mode. To reset the default value, use the `no` form of this command.

```
rxspeed bps

no rxspeed
```

### Syntax Description

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>bps</code></td>
<td>Baud rate in bits per second (bps).</td>
</tr>
</tbody>
</table>

### Defaults

9600 bps

### Command Modes

Line configuration

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

### Usage Guidelines

Set the speed to match the baud rate of whatever device you have connected to the port. Some baud rates available on devices connected to the port might not be supported on the system. The system will indicate if the speed you select is not supported.

**Note**

If the line was previously configured for automatic baud rate detection (autobaud), disable autobaud by entering the `no autobaud` command before entering the `rxspeed` command to fix the speed of the port.

### Examples

The following example sets the line 5 receive rate to 2400 bps:

```
line 5
rxspeed 2400
```

### Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>source template</code></td>
<td>Sets the flow control start character.</td>
</tr>
<tr>
<td><code>terminal rxspeed</code></td>
<td>Sets the terminal receive speed (how fast information is sent to the terminal) for the current line and session.</td>
</tr>
<tr>
<td><code>txspeed</code></td>
<td>Sets the terminal transmit speed (how fast the terminal sends information to the modem).</td>
</tr>
</tbody>
</table>
service exec-callback

To enable call back to clients who request a callback from the EXEC level, use the service exec-callback command in global configuration mode.

```
service exec-callback
```

### Syntax Description
This command has no arguments or keywords.

### Defaults
Callback is not enabled.

### Command Modes
Global configuration

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>11.1</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

### Usage Guidelines
This command enables the Cisco IOS software to return a call to a device that dials in, connects to the EXEC, and requests callback.

### Examples
The following example enables EXEC level callback:

```
service exec-callback
```

### Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>arap callback</td>
<td>Enables an ARA client to request a callback from an ARA client.</td>
</tr>
<tr>
<td>debug callback</td>
<td>Displays callback events when the router is using a modem and a chat script to call back on a terminal line.</td>
</tr>
<tr>
<td>debug confmodem</td>
<td>Displays information associated with the discovery and configuration of the modem attached to the router.</td>
</tr>
<tr>
<td>ppp callback (PPP client)</td>
<td>Enables a dialer interface that is not a DTR interface to function either as a callback client that requests callback or as a callback server that accepts callback requests.</td>
</tr>
<tr>
<td>script arap-callback</td>
<td>Specifies that a chat script start on a line when an ARA client requests a callback.</td>
</tr>
<tr>
<td>username</td>
<td>Establishes a username-based authentication system, such as PPP CHAP and PAP.</td>
</tr>
</tbody>
</table>
service old-slip-prompts

To provide backward compatibility for client software scripts expecting Serial Line Internet Protocol (SLIP) and PPP dialogs to be formatted with Cisco IOS software Release 9.1 or earlier releases, use the service old-slip-prompts command in global configuration mode. To disable this function, use the no form of this command:

```
  service old-slip-prompts

  no service old-slip-prompts
```

**Syntax Description**
This command has no arguments or keywords.

**Defaults**
The prompts and information sent by SLIP and PPP are formatted with the current release of Cisco IOS software.

**Command Modes**
Global configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>11.1</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**
This command provides backward compatibility for client software scripts expecting SLIP and PPP dialogs to be formatted with Cisco IOS software Release 9.1 or earlier releases.

**Examples**
The following example shows the output of a SLIP command after the `service old-slip-prompts` command is enabled:

```
Router# configure terminal
Router(config)# service old-slip-prompts
Router(config)# exit
Router# slip
IP address or hostname: 10.2.2.2
Entering SLIP mode.
Your IP address is 10.2.2.2. MTU is 1500 bytes
```
service pt-vty-logging

To log the X.121 calling address, Call User Data (CUD), and IP address assigned to a vty asynchronous connection, use the `service pt-vty-logging` command in global configuration mode. To disable this function, use the `no` form of this command.

```
   service pt-vty-logging

   no service pt-vty-logging
```

**Syntax Description**

This command has no arguments or keywords.

**Defaults**

This feature is disabled.

**Command Modes**

Global configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>11.2</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

This command permits you to log the X.121 calling address, CUD, and IP address assigned to a vty asynchronous connection and direct this information to the console, an internal buffer, or a UNIX syslog server, depending on the logging configuration command you use. This authentication information can be used to associate an incoming packet assembler/disassembler (PAD) vty-asynchronous connection with an IP address.

By default, the Cisco IOS software displays all messages to the console terminal.

**Examples**

The following example enables you to log the X.121 calling address, CUD, and IP address assigned to a vty asynchronous connection and save this information to a syslog server:

```
   service pt-vty-logging
```

The following is sample output from the `service pt-vty-logging` command:

```
01:24:31: PAD18: call from 00011890 on LCI 10 PID 1 0 0 0 CUD "xyz"
```

**Table 11**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>01:24:31:</td>
<td>Time stamp.</td>
</tr>
<tr>
<td>PAD18:</td>
<td>Active vty line number using the PAD connection.</td>
</tr>
</tbody>
</table>
### Table 11  service pt-vty-logging Field Descriptions (continued)

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>00011890</td>
<td>The source or calling address.</td>
</tr>
<tr>
<td>on LCI 10</td>
<td>Incoming call is initiated on Logical Channel 10.</td>
</tr>
<tr>
<td>PID 1 0 0 0</td>
<td>The PAD Protocol Identifier is “01000000.”</td>
</tr>
<tr>
<td>CUD “xyz”</td>
<td>CUD “xyz.” If no CUD is available, this field will appear as follows: CUD “ ”</td>
</tr>
</tbody>
</table>

### Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>logging</td>
<td>Logs messages to a syslog server host.</td>
</tr>
<tr>
<td>logging buffered</td>
<td>Logs messages to an internal buffer.</td>
</tr>
</tbody>
</table>
**session-limit**

To set the maximum number of terminal sessions per line, use the `session-limit` command in line configuration mode. To remove any specified session limit, use the `no` form of this command.

```
session-limit session-number
no session-limit
```

**Syntax Description**

| session-number | Specifies the maximum number of sessions. |

**Defaults**

The default and set session limits are displayed with the `start-character` EXEC command.

**Command Modes**

Line configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Examples**

The following example limits the number of sessions to eight on a ten-line range:

```
line 2 12
session-limit 8
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>line vty</td>
<td>Specifies a virtual terminal for remote console access.</td>
</tr>
</tbody>
</table>
session-timeout

To set the interval for closing the connection when there is no input or output traffic, use the session-timeout command in line configuration mode. To remove the timeout definition, use the no form of this command.

```
no session-timeout
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>minutes</td>
<td>Specifies the timeout interval in minutes.</td>
</tr>
<tr>
<td>output</td>
<td>(Optional) Specifies that when traffic is sent to an asynchronous line from the router (within the specified interval), the connection is retained.</td>
</tr>
</tbody>
</table>

**Defaults**

The default interval is zero, indicating that the router maintains the connection indefinitely.

**Command Modes**

Line configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

This command sets the interval that the Cisco IOS software waits for traffic before closing the connection to a remote computer and returning the terminal to an idle state.

If only the session timeout command is specified, the session timeout interval is based solely on detected input from the user.

If the session timeout command is specified with the output keyword, the interval is based on both input and output traffic. You can specify a session timeout on each port.

The session-timeout command behaves slightly differently on virtual (vty) terminals than on physical console, auxiliary (aux), and terminal (tty) lines. When a timeout occurs on a vty, the user session returns to the EXEC prompt. When a timeout occurs on physical lines, the user session is logged out and the line returned to the idle state.

You can use a combination of the exec-timeout and session-timeout line configuration commands, set to approximately the same values, to get the same behavior from virtual lines that the session-timeout command causes on physical lines.

**Examples**

The following example sets an interval of 20 minutes and specifies that the timeout is subject to traffic detected from the user (input only):

```
line 5
  session-timeout 20
```
The following example sets an interval of 10 minutes, subject to traffic on the line in either direction:

```
line 5
  session-timeout 10 output
```

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>absolute-timeout</td>
<td>Sets the interval for closing the connection on a virtual terminal line.</td>
</tr>
<tr>
<td>exec-timeout</td>
<td>Sets the interval that the EXEC command interpreter waits until user input is detected.</td>
</tr>
</tbody>
</table>
show arap

To display information about a running AppleTalk Remote Access Protocol (ARAP) connection, use the `show arap` command in EXEC mode.

```
show arap [line-number]
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>line-number</code></td>
<td>(Optional) Number of the line on which an ARAP connection is established and active.</td>
</tr>
</tbody>
</table>

**Command Modes**

EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `show arap` command with no arguments to display a summary of the ARAP traffic since the router was last booted.

**Examples**

The following is sample output from the `show arap` command:

```
Router# show arap

Statistics are cumulative since last reboot
Total ARAP connections: 2
Total Appletalk packets output: 157824
Total Appletalk packets input: 12465

These fields refer to the sum of all of the ARAP connections since the box was last reloaded.

The following sample output results in a display of information about ARAP activity on a specific line (line 3):

Router# show arap 3

Active for 23 minutes
"Unlimited time left" or "22 minutes left"
"Doing smartbuffering" or "Smartbuffering disabled"
Appletalk packets output: 157824
Appletalk packets input: 12465
Appletalk packets overflowed: 1642
Appletalk packets dropped: 586
V42bis compression efficiency (incoming/outgoing): (percentage/percentage)
MNP4 packets received: 864
MNP4 packets sent: 1068
MNP4 garbled packets received: 4
MNP4 out of order packets received: 0
MNP4 packets resent: 0
MNP4 nobuffers: 0
```

Table 12 describes the significant fields shown in the display.
**Table 12  show arap Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Active for <code>integer</code> minutes</td>
<td>Number of minutes since ARAP started on the line.</td>
</tr>
<tr>
<td>Unlimited time left or <code>integer</code> minutes left</td>
<td>Remaining time limit on the line, if applicable on the line.</td>
</tr>
<tr>
<td>Doing smartbuffering or Smartbuffering disabled</td>
<td>Obsolete. Always says “Doing smartbuffering.”</td>
</tr>
<tr>
<td>Appletalk packets output:</td>
<td>Number of AppleTalk packets that have been received from the Apple Macintosh and out to the network during this connection.</td>
</tr>
<tr>
<td>Appletalk packets input:</td>
<td>Number of AppleTalk packets that have been received from the network and sent to the Apple Macintosh during this connection.</td>
</tr>
<tr>
<td>Appletalk packets overflowed:</td>
<td>Number of packets from the network that have been dropped because the link to the Apple Macintosh was congested.</td>
</tr>
<tr>
<td>Appletalk packets dropped:</td>
<td>Number of packets from the network that have been dropped because it was unnecessary to pass them (frequently RTMP).</td>
</tr>
<tr>
<td>V42bis compression efficiency (incoming/outgoing):</td>
<td>Performance of the v42bis protocol underneath ARA, expressed as a percentage of incoming/percentage outgoing. If the efficiency is low, a network user is probably copying already compressed files across the link. Generally, low efficiency means slow performance.</td>
</tr>
<tr>
<td>MNP4 packets received:</td>
<td>Number of link-level packets that have been received from the Apple Macintosh.</td>
</tr>
<tr>
<td>MNP4 packets sent:</td>
<td>Number of link-level packets that have been sent to the Apple Macintosh.</td>
</tr>
<tr>
<td>MNP4 garbled packets received:</td>
<td>Number of garbled packets that have been received from the Apple Macintosh.</td>
</tr>
<tr>
<td>MNP4 out of order packets received:</td>
<td>Number of out-of-order packets that have been received from the Apple Macintosh.</td>
</tr>
<tr>
<td>MNP4 packets resent:</td>
<td>Number of times packets have been re-sent.</td>
</tr>
<tr>
<td>MNP4 nobuffers:</td>
<td>Number of times MNP4 has run out of buffers. This field should be zero.</td>
</tr>
</tbody>
</table>

1. Each of these fields indicates line noise. The higher the value, the higher the noise.
show entry

To display the list of queued host-initiated connections to a router, use the show entry command in EXEC mode.

show entry

**Syntax Description**

This command has no arguments or keywords.

**Command Modes**

EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>11.1</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

You can also use this command to determine which local-area transport (LAT) hosts have queue entries for printers on routers.

**Examples**

The following is sample output from the show entry command. The display shows that two LAT connections are waiting for access to port 5. The list is ordered so that the lower-numbered entry has been waiting longer, and will use the line next.

```
Router# show entry
1 waiting 0:02:22 for port 5 from LAT node BLUE
2 waiting 0:00:32 for port 5 from LAT node STELLA
```

*Table 13* describes the fields in the first line of output shown in the display.

**Table 13  show entry Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Number assigned to the queued connection attempt.</td>
</tr>
<tr>
<td>waiting 0:02:22</td>
<td>Interval (hours:minutes:seconds) during which the connection attempt has been waiting.</td>
</tr>
<tr>
<td>for port 5</td>
<td>Port for which the connection attempt is waiting.</td>
</tr>
<tr>
<td>from LAT node BLUE</td>
<td>Name of the user (BLUE) attempting to make the connection.</td>
</tr>
</tbody>
</table>
show keymap

To test for the availability of a keymap after a connection on a router takes place, use the show keymap command in EXEC mode.

```
show keymap [keymap-name | all]
```

**Syntax Description**

- `keymap-name` (Optional) Name of the keymap.
- `all` (Optional) Lists the names of all defined keymaps. The name of the default keymap is not listed.

**Command Modes**

EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

The Cisco IOS software searches for the specified keymap in its active configuration image and lists the complete entry if found. If the keymap is not found, an appropriate “not found” message appears.

If you do not use any arguments with the show keymap command, then the keymap currently used for the terminal is displayed.

**Examples**

The following is sample output from the show keymap command:

```
Router# show keymap

ciscodefault { clear = ‘^z’; flinp = ‘^x’; enter = ‘^m’;
delete = ‘^d’ | ‘^?’;
synch = ‘^r’; reshow = ‘^v’; eof = ‘^e’; tab = ‘^i’;
btab = ‘^b’; nl = ‘^a’; left = ‘^h’; right = ‘^j’;
up = ‘^k’; down = ‘^j’; einp = ‘^w’; reset = ‘^t’;
xoff = ‘^s’; xon = ‘^g’; escape = ‘^c’; ferase = ‘^u’;
insert = ‘\E’;
}
pa1 = ‘^p1’; pa2 = ‘^p2’; pa3 = ‘^p3’;
pfk1 = ‘\E1’; pfk2 = ‘\E2’; pfk3 = ‘\E3’; pfk4 = ‘\E4’;
pfk5 = ‘\E5’; pfk6 = ‘\E6’; pfk7 = ‘\E7’; pfk8 = ‘\E8’;
pfk9 = ‘\E9’; pfk10 = ‘\E0’; pfk11 = ‘\E-’; pfk12 = ‘\E=’;
pfk13 = ‘\E!’; pfk14 = ‘\E@’; pfk15 = ‘\E#’; pfk16 = ‘\ES’;
pfk17 = ‘\E$’; pfk18 = ‘\E^’; pfk19 = ‘\E&’; pfk20 = ‘\E*’;
pfk21 = ‘\E(’; pfk22 = ‘\E)’; pfk23 = ‘\E_’; pfk24 = ‘\E+’;
}
```
show lat advertised

To display the local-area transport (LAT) services a router offers to other systems running LAT on the network, use the **show lat advertised** command in EXEC mode.

```
show lat advertised
```

**Syntax Description**

This command has no arguments or keywords.

**Command Modes**

EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Advertised services are created with the **lat service** commands. The display includes the service rating, rotary group if present, and whether the service is enabled for incoming connections.

**Examples**

The following is sample output from the **show lat advertised** command:

```
Router# show lat advertised

Service Name   Rating  Rotary  Flags  Autocommand: telnet bear-cat
BEAR-CAT       4(Dynamic) None  Enabled
MODEMS         0(Dynamic) 12   Enabled
    Ident: SpaceBlazer modem services
RECLUSE        4(Dynamic) None  Enabled
    Ident: white recluse...
```

The display shows output from a router named sloth that has three services defined: BEAR-CAT, MODEMS, and RECLUSE.

**Table 14** describes the significant fields shown in the display.

**Table 14  show lat advertised Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Service Name</td>
<td>Lists the LAT service name.</td>
</tr>
<tr>
<td>Rating</td>
<td>Lists the static service rating set, if any.</td>
</tr>
<tr>
<td>Rotary</td>
<td>Lists the associated rotary service.</td>
</tr>
<tr>
<td>Flags</td>
<td>Lists whether a service is enabled.</td>
</tr>
<tr>
<td>Autocommand</td>
<td>Defines the autocommand associated with the service.</td>
</tr>
<tr>
<td>Ident</td>
<td>Lists the advertised identification for the service.</td>
</tr>
</tbody>
</table>
show lat groups

To display the groups that were defined in the Cisco IOS software with the `lat group-list` command, use the `show lat groups` command in EXEC mode.

```
show lat groups
```

**Syntax Description**
This command has no arguments or keywords.

**Command Modes**
EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Examples**
The following is sample output from the `show lat groups` command:

```
Router# show lat groups

Group Name          Len   Groups
cafeteria           3      13  15  23
engineering         7      55
manufacturing       10     70  71  72
```

`Table 15` describes only the significant fields shown in the display.

**Table 15 show lat groups Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Group Name</td>
<td>Assigned group name.</td>
</tr>
<tr>
<td>Len</td>
<td>Size of internal data structure used to contain the group code map.</td>
</tr>
<tr>
<td>Groups</td>
<td>Group codes associated with the learned group.</td>
</tr>
</tbody>
</table>

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>lat group-list</td>
<td>Allows a name to be assigned to the group list, which is any combination of group names, numbers, or ranges.</td>
</tr>
</tbody>
</table>
show lat nodes

To display information about all known local-area network (LAT) nodes, use the `show lat nodes` command in EXEC mode.

```
show lat nodes
```

**Syntax Description**

This command has no arguments or keywords.

**Command Modes**

EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Examples**

The following is sample output from the `show lat nodes` command:

```
Router# show lat nodes

Node "CHAOS", usage -1, Interface Ethernet0, Address 0000.0c01.0509
  Timer 89, sequence 188, changes 131, flags 0x0, protocol 5.1
  Facility 0, Product code 0, Product version 0
  Recv 0/0/0, Xmit 0/0/0, 0 Dups, 0 ReXmit
  Bad messages: 0, Bad slots: 0, Solicits accepted: 0
  Solicits rejected: 0, Multiple nodes: 0
  Groups: 0
  Service classes: 1

Node "CONFUSED", usage -1, Local
  Timer 99, sequence 4, changes 151, flags 0x0, protocol 5.2
  Facility 0, Product code 0, Product version 0
  Recv 0/0/0, Xmit 0/0/0, 0 Dups, 0 ReXmit
  Bad messages: 0, Bad slots: 0, Solicits accepted: 0
  Solicits rejected: 0, Multiple nodes: 0
  Groups: 0
  Service classes: 1

Node "EMAN-cs", usage -1, Interface Ethernet0, Address 0000.0cff.c9ed
  Timer 99, sequence 9, changes 159, flags 0x0, protocol 5.1
  Facility 0, Product code 0, Product version 0
  Recv 0/0/0, Xmit 0/0/0, 0 Dups, 0 ReXmit
  Bad messages: 0, Bad slots: 0, Solicits accepted: 0
  Solicits rejected: 0, Multiple nodes: 0
  Groups: 0
  Service classes: 1

Node "TARMAC", usage -1, Interface Ethernet0, Address 0000.0c02.c7c1
  Timer -10351, sequence 1, changes 131, flags 0x40, protocol 5.2
  Facility 0, Product code 0, Product version 0
  Recv 0/0/0, Xmit 0/0/0, 0 Dups, 0 ReXmit
  Bad messages: 0, Bad slots: 0, Solicits accepted: 0
  Solicits rejected: 0, Multiple nodes: 0
  Groups: 0
  Service classes: 1
```

Table 16 describes the significant fields shown in the display.
### show lat nodes Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Node</td>
<td>The node name as reported by the host computer.</td>
</tr>
<tr>
<td>usage</td>
<td>The number of virtual circuits currently active to this node.</td>
</tr>
<tr>
<td>Interface</td>
<td>Node interface type and number.</td>
</tr>
<tr>
<td>Address</td>
<td>The MAC address of the Ethernet interface for the node.</td>
</tr>
<tr>
<td>Timer</td>
<td>The number of seconds remaining until the service advertisement message for this node will time out; this value is set to three times the nodes multicast timer value whenever a new service advertisement message is received.</td>
</tr>
<tr>
<td>sequence</td>
<td>The sequence number received in the last service advertisement message received. Nodes increment their sequence number when the contents of the service advertisement change.</td>
</tr>
<tr>
<td>changes</td>
<td>The internal representation of what changed in the multicast message the last time the sequence number changed.</td>
</tr>
<tr>
<td>flags</td>
<td>The internal representation of various state information about the node.</td>
</tr>
<tr>
<td>protocol</td>
<td>The LAT protocol version used by the node.</td>
</tr>
<tr>
<td>Facility</td>
<td>The remote facility number.</td>
</tr>
<tr>
<td>Product code</td>
<td>The remote product code.</td>
</tr>
<tr>
<td>Product version</td>
<td>The remote product version.</td>
</tr>
<tr>
<td>Recv and Xmit</td>
<td>The number of messages, slots, and bytes received or sent to the node. The number of messages is the number of LAT virtual circuit messages. Each virtual circuit message contains some number of slots, which contain actual terminal data or control information. Bytes is the number of data bytes (input or output characters) exchanged.</td>
</tr>
<tr>
<td>Dups</td>
<td>The number of duplicate virtual circuit messages received.</td>
</tr>
<tr>
<td>ReXmit</td>
<td>The number of virtual circuit messages resent.</td>
</tr>
<tr>
<td>Bad messages</td>
<td>The number of bad messages received.</td>
</tr>
<tr>
<td>Bad slots</td>
<td>The number of bad slots received.</td>
</tr>
<tr>
<td>Solicits accepted</td>
<td>The number of solicit-information requests accepted.</td>
</tr>
<tr>
<td>Solicits rejected</td>
<td>The number of solicit-information requests rejected.</td>
</tr>
<tr>
<td>Multiple nodes</td>
<td>The total of multiple nodes seen.</td>
</tr>
<tr>
<td>Groups</td>
<td>The list of group codes advertised by the service-advertisement message of the node.</td>
</tr>
<tr>
<td>Service classes</td>
<td>The number of service classes.</td>
</tr>
</tbody>
</table>
show lat services

To display information about learned local-area transport (LAT) services in the Cisco IOS software, use the `show lat services` command in EXEC mode.

```
show lat services [service-name]
```

### Syntax Description

- **service-name** (Optional) Name of a specific LAT service.

### Command Modes

EXEC

### Command History

Release | Modification
---|---
11.2 | This command was introduced.

### Examples

The following is sample output from the `show lat services` command:

```
Router# show lat services
Service Name     Rating   Interface  Node (Address)
ABCDEFGHIJ            5   Ethernet0  CONFUSED (0000.0c00.391f)
GLAD                 84   Ethernet0  BLUE (aa00.0400.9205)
Ident: Welcome to Big Blue Gateway
WHEEL                83   Ethernet0  WHEEL (aa00.0400.9005)
ZXYW                  5   Ethernet0  CONFUSED (0000.0c00.391f)
```

Table 17 describes the significant fields shown in the display.

### Table 17  show lat services Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Service Name</td>
<td>LAT service name.</td>
</tr>
<tr>
<td>Rating</td>
<td>Rating of the service. If a single service is provided by more than one host, the Cisco IOS software connects to the one with the highest rating.</td>
</tr>
<tr>
<td>Interface</td>
<td>Interface type.</td>
</tr>
<tr>
<td>Node</td>
<td>Connection address.</td>
</tr>
<tr>
<td>(Address)</td>
<td>Advertised identification for the service.</td>
</tr>
</tbody>
</table>

### Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>show lat sessions</td>
<td>Displays active LAT sessions.</td>
</tr>
<tr>
<td>show resource-pool call</td>
<td>Displays specific LAT learned services.</td>
</tr>
</tbody>
</table>
show lat sessions

To display active local-area transport (LAT) sessions, use the `show lat sessions` command in EXEC mode.

```
show lat sessions [line-number]
```

**Syntax Description**

- `line-number` (Optional) Displays an active LAT session on a specific line.

**Command Modes**

- EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>11.2</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Examples**

The following is sample output from the `show lat sessions` command. In this example, information about all active LAT sessions is displayed. The output is divided into three sections for each session (in this case two sessions): TTY data, session data, and remote node data.

```
Router> show lat sessions

tty0, connection 1 to service TERMI
TTY data:
  Name "0", Local usage 1/0, Remote usage disabled
  Flags: Local Connects, Enabled
  Type flags: none
  Config flags: -FlowOut, -FlowIn, Parameter Info
  Flow control ^S/^Q in ^S/^Q out, Mode Normal, Parity None, databits 8
  Groups: 0
Session data:
  Name TERMI, Remote Id 1, Local Id 1
  Remote credits 2, Local credits 0, Advertised Credits 2
  Flags: none
  Max Data Slot 255, Max Attn Slot 255, Stop Reason 0
Remote Node data:
  Node "TERMI", Address 0000.0C00.291F, usage 1
  Timer 59, sequence 159, changes 159, flags 0x0, protocol 5.1
  Recv 56/22/83, Xmit 41/23/14, 0 Dups, 0 ReXmit
  Groups: 0

tty10, connection 1 to service ENG2
TTY data:
  Name "10", Local usage 1/0, Remote usage disabled
  Flags: Local Connects, Enabled
  Type flags: none
  Config flags: -FlowOut, +FlowIn, Set Parameters, 0x40000000
  Flow control ^S/^Q in ^S/^Q out, Mode Normal, Parity None, databits 8
  Groups: 0
Session data:
  Name ENG2, Remote Id 1, Local Id 1
  Remote credits 1, Local credits 0, Advertised Credits 2
  Flags: none
  Max Data Slot 255, Max Attn Slot 255, Stop Reason 0
Remote Node data:
```
The following sample output displays information about active LAT sessions on one line (line 10). The output is divided into three sections: TTY data, session data, and remote node data.

Router> show lat sessions 10

tty10, connection 1 to service ENG2
TTY data:
Name "10", Local usage 1/0, Remote usage disabled
Flags: Local Connects, Enabled
Type flags: none
Config flags: -FlowOut, +FlowIn, Set Parameters, 0x40000000
Flow control ^S/^Q in ^S/^Q out, Mode Normal, Parity None, databits 8
Groups: 0
Session data:
Name ENG2, Remote Id 1, Local Id 1
Remote credits 1, Local credits 0, Advertised Credits 2
Flags: none
Max Data Slot 255, Max Attn Slot 255, Stop Reason 0
Remote Node data:
Node "ENG2", Address AA00.0400.34DC, usage 1
Timer 179, sequence 60, changes 255, flags 0x0, protocol 5.1
Recv 58/29/186, Xmit 50/36/21, 0 Dups, 0 ReXmit
Groups: 0

Table 18 describes the screen output for the preceding two examples.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TTY data</td>
<td>Summary of the LAT-oriented terminal-line-specific data.</td>
</tr>
<tr>
<td>Name</td>
<td>Name used for this port as a port identification string. The name is reported to remote systems, which can display it in some operating-system dependent manner. This value is also used for targets of host-initiated connections. Currently, this value is hard-wired to be the line number of the associated terminal line.</td>
</tr>
<tr>
<td>Local/Remote usage</td>
<td>Current status of the terminal. The number is reported as current/maximum, where current is the current number of sessions of a given type, and maximum is the maximum number of sessions allowed (or zero if there is no maximum). If a terminal is being used for outgoing sessions, the local usage is equal to the number of current LAT sessions. If the terminal is being used for incoming sessions, local usage is disabled, and the remote count and maximum is one.</td>
</tr>
<tr>
<td>Flags</td>
<td>Current state of the line, and whether there are any queued host-initiated connections.</td>
</tr>
<tr>
<td>Type flags</td>
<td>Report flags not used in the current software release.</td>
</tr>
<tr>
<td>Config flags</td>
<td>Current port state as reflected by the most recent configuration message exchange.</td>
</tr>
<tr>
<td>Flow control</td>
<td>Lists set flow control characters.</td>
</tr>
<tr>
<td>Groups</td>
<td>Group code list currently in use for the line.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>----------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Session data</td>
<td>Reports various parameters about the connection.</td>
</tr>
<tr>
<td>Name</td>
<td>For the outbound connection, the name of the remote service to which it is connected. For inbound connections, this field is currently unused.</td>
</tr>
<tr>
<td>Remote/Local Id</td>
<td>Slot IDs being used to uniquely identify the session multiplexed over the underlying LAT virtual circuit.</td>
</tr>
<tr>
<td>Remote/Local/Advertised Credits</td>
<td>Number of flow control credits that the Cisco IOS software will be sending to the host as soon as possible. The advertised credits are the number of credits that have already been sent.</td>
</tr>
<tr>
<td>Flags</td>
<td>Transient conditions in the LAT-state machine dealing with the current connection status.</td>
</tr>
<tr>
<td>Max Data Slot</td>
<td>Maximum number of characters that can be sent in a single data slot.</td>
</tr>
<tr>
<td>Max Attn Slot</td>
<td>Maximum amount of data that can be sent in an attention message. Current LAT implementations only send 1-byte attention messages (attention messages are used to flush buffered output). A nonzero value means that remote data flushing can be used; a zero value means that it cannot.</td>
</tr>
<tr>
<td>Stop Reason</td>
<td>Reason the session was stopped, if it was stopped but not deleted. This value is usually zero, indicating that the session has not yet been stopped. If a session persists for a long time with a nonzero stop reason, there is probably a problem in the local LAT software.</td>
</tr>
<tr>
<td>Remote Node data</td>
<td>Reports information about the remote node. The data includes the same fields as those from the show lat nodes output.</td>
</tr>
<tr>
<td>Node</td>
<td>Node name as reported by the host computer.</td>
</tr>
<tr>
<td>Address</td>
<td>MAC address of the Ethernet interface for the node.</td>
</tr>
<tr>
<td>usage</td>
<td>Number of virtual circuits currently active to the node.</td>
</tr>
<tr>
<td>Timer</td>
<td>Number of seconds remaining until the service advertisement message for the node will time out; this value is set to three times greater than the node multicast timer value whenever a new service-advertisement message is received.</td>
</tr>
<tr>
<td>sequence</td>
<td>Sequence number received in the last service-advertisement message. Nodes increment their sequence number when the contents of the service-advertisement change.</td>
</tr>
<tr>
<td>changes</td>
<td>Internal representation of what changed in the multicast message the last time the sequence number changed.</td>
</tr>
<tr>
<td>flags</td>
<td>Internal representation of various state information about the node.</td>
</tr>
<tr>
<td>protocol</td>
<td>LAT protocol version used by the node.</td>
</tr>
<tr>
<td>Recv and Xmit</td>
<td>Number of messages, slots, and bytes received or sent to the node. The number of messages is the number of LAT virtual circuit messages. Each virtual circuit message contains some number of slots, which contain actual terminal data or control information.</td>
</tr>
<tr>
<td>Dups</td>
<td>Number of duplicate virtual circuit messages received.</td>
</tr>
<tr>
<td>ReXmit</td>
<td>Number of virtual circuit messages resent.</td>
</tr>
<tr>
<td>Groups</td>
<td>Group codes advertised by the service-advertisement message of the node.</td>
</tr>
</tbody>
</table>
### Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>show lat services</code></td>
<td>Displays information about learned LAT services in the Cisco IOS software.</td>
</tr>
<tr>
<td><code>show resource-pool call</code></td>
<td>Displays specific LAT learned services.</td>
</tr>
</tbody>
</table>
show lat traffic

To display information about traffic and resource utilization statistics on all active lines, use the **show lat traffic** command in EXEC mode.

```
show lat traffic
```

**Syntax Description**

This command has no arguments or keywords.

**Command Modes**

EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Examples**

The following is sample output from the **show lat traffic** command:

```
Router# show lat traffic

Local host statistics:
  0/100 circuits, 0/500 sessions, 1/500 services
  100 sessions/circuit, circuit timer 80, keep-alive timer 5
Recv:  335535 messages (2478 duplicates),  161722 slots,  1950146 bytes
  0 bad circuit messages,  3458 service messages (52 used)
Xmit:  182376 messages (2761 retransmit),  146490 slots,  36085 bytes
  1 circuit timeouts
Total:  23 circuits created,  38 sessions
```

Table 19 describes the significant fields shown in the display.

**Table 19  show lat traffic Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Local host statistics</td>
<td>Information about the router.</td>
</tr>
<tr>
<td>circuits</td>
<td>Current number and maximum support number of virtual circuits.</td>
</tr>
<tr>
<td>sessions</td>
<td>Current and maximum number of sessions.</td>
</tr>
<tr>
<td>services</td>
<td>Current number of known remote services, and the maximum supported.</td>
</tr>
<tr>
<td>sessions/circuit</td>
<td>Number of sessions per virtual circuit supported by the software.</td>
</tr>
<tr>
<td>circuit timer</td>
<td>Value of the virtual circuit timer parameter defined by the <strong>lat vc-timer</strong> global configuration command.</td>
</tr>
<tr>
<td>keep-alive timer</td>
<td>Value defined by the <strong>lat ka-timer</strong> global configuration command.</td>
</tr>
<tr>
<td>Recv messages</td>
<td>Statistics about local node receive totals.</td>
</tr>
<tr>
<td>duplicates</td>
<td>Number of duplicate virtual circuit messages received.</td>
</tr>
<tr>
<td>slots</td>
<td>Number of slots received.</td>
</tr>
</tbody>
</table>
### Table 19  
**show lat traffic Field Descriptions (continued)**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>bytes</td>
<td>Number of data bytes received.</td>
</tr>
<tr>
<td>bad circuit messages</td>
<td>Count of invalid messages received.</td>
</tr>
<tr>
<td>service messages</td>
<td>Number of service advertisement multicast messages received.</td>
</tr>
<tr>
<td>used</td>
<td>Number of multicast messages that caused the local node information to be updated.</td>
</tr>
<tr>
<td>Xmit messages</td>
<td>Various transmission totals.</td>
</tr>
<tr>
<td>retransmit</td>
<td>Total number of virtual circuit messages sent.</td>
</tr>
<tr>
<td>slots</td>
<td>Number of data and control slots sent.</td>
</tr>
<tr>
<td>bytes</td>
<td>Count of user data bytes sent.</td>
</tr>
<tr>
<td>circuit timeouts</td>
<td>Count of times that a virtual circuit timed out because the remote node stopped responding (due to a node failure or communications failure).</td>
</tr>
<tr>
<td>Total</td>
<td>Count of virtual circuits and sessions that have existed since the router booted or rebooted.</td>
</tr>
</tbody>
</table>
**show line**

To display parameters of a terminal line, use the `show line` command in EXEC mode.

```
show line [line-number | aux | console | summary]
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>line-number</code></td>
<td>(Optional) Absolute line number of the line for which you want to list parameters.</td>
</tr>
<tr>
<td><code>aux</code></td>
<td>(Optional) Auxiliary line.</td>
</tr>
<tr>
<td><code>console</code></td>
<td>(Optional) Primary terminal line.</td>
</tr>
<tr>
<td><code>summary</code></td>
<td>(Optional) Line status summary.</td>
</tr>
</tbody>
</table>

**Command Modes**

EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>11.3(1)T</td>
<td>The <code>summary</code> keyword added.</td>
</tr>
<tr>
<td>12.1</td>
<td>Output from this command was modified to show the transport method configured.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

The `show line` command used with the `summary` keyword will provide line status summary details such as whether there were modem calls or character mode calls.

**Examples**

The following sample output from the `show line vty 4` command shows that virtual terminal line 4 has a send and receive rate of 9600 bits per second (bps). Also shown are the terminal screen width and length, modem state, preferred transport method, and other characteristics.

```
Router# show line vty 4

Tty Typ     Tx/Rx    A Modem  Roty AccO AccI   Uses   Noise  Overruns
22 VTY              -    -      -    -    -      0       0     0/0 -

Line 22, Location: "", Type: ""
Length: 24 lines, Width: 80 columns
Baud rate (TX/RX) is 9600/9600
Status: No Exit Banner
Capabilities: CUG Security Enabled
Modem state: Idle
Special Chars: Escape Hold Stop Start Disconnect Activation
   ^x    none   -      -       none
Timeouts:      Idle EXEC    Idle Session   Modem Answer  Session
               never         never         not set
Dispatch
   never            never         none         not set
Idle Session Disconnect Warning
   never
Login-sequence User Response
   00:00:30
Autoselect Initial Wait
   not set
```
Modem type is unknown.
Session limit is not set.
Time since activation: never
Editing is enabled.
History is enabled, history size is 10.
DNS resolution in show commands is enabled
Full user help is disabled
Allowed input transports are none.
Allowed output transports are pad v120 telnet rlogin udptn.
Preferred transport is telnet.
No output characters are padded
No special data dispatching characters

Table 20 describes the significant fields shown in the display.

Table 20  show line Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tty</td>
<td>Line number.</td>
</tr>
<tr>
<td>Typ</td>
<td>Type of line. In this case, a virtual terminal line, which is active, in asynchronous mode denoted by the preceding “A.” All possible values follow:</td>
</tr>
<tr>
<td></td>
<td>VTY—virtual terminal line</td>
</tr>
<tr>
<td></td>
<td>CTY—console</td>
</tr>
<tr>
<td></td>
<td>AUX—auxiliary port</td>
</tr>
<tr>
<td></td>
<td>TTY—asynchronous terminal port</td>
</tr>
<tr>
<td></td>
<td>lpt—parallel printer</td>
</tr>
<tr>
<td>Tx/Rx</td>
<td>Transmit rate/receive rate of the line.</td>
</tr>
<tr>
<td>A</td>
<td>Indicates whether autobaud has been configured for the line. A value of F indicates that autobaud has been configured; a hyphen indicates that it has not been configured.</td>
</tr>
<tr>
<td>Modem</td>
<td>Types of modem signals that have been configured for the line. Possible values follow:</td>
</tr>
<tr>
<td></td>
<td>callin</td>
</tr>
<tr>
<td></td>
<td>callout</td>
</tr>
<tr>
<td></td>
<td>cts-req</td>
</tr>
<tr>
<td></td>
<td>DTR-Act</td>
</tr>
<tr>
<td></td>
<td>inout</td>
</tr>
<tr>
<td></td>
<td>RIisCD</td>
</tr>
<tr>
<td>Roty</td>
<td>Rotary group configured for the line, if set.</td>
</tr>
<tr>
<td>AccO, AccI</td>
<td>Output or input access list number configured for the line.</td>
</tr>
<tr>
<td>Uses</td>
<td>Number of connections established to or from the line since the system was restarted.</td>
</tr>
<tr>
<td>Noise</td>
<td>Number of times noise has been detected on the line since the system restarted.</td>
</tr>
</tbody>
</table>
Overruns

Hardware Universal Asynchronous Receiver/Transmitter (UART) overruns or software buffer overflows, both defined as the number of overruns or overflows that have occurred on the specified line since the system was restarted. Hardware overruns are buffer overruns; the UART chip has received bits from the software faster than it can process them. A software overflow occurs when the software has received bits from the hardware faster than it can process them.

A (or 1 or *)

An A at the upper left of the display indicates that the user is running an asynchronous interface; an I indicates that the line has an asynchronous interface available; an asterisk (*) indicates that the line is otherwise active (in character mode).

Line

Definition of the specified protocol and address of the line.

Location

Location of the current line.

Type

Type of line, as specified by the line global configuration command.

Length

Length of the terminal or screen display, in rows.

Width

Width of the terminal or screen display, in columns.

Baud rate (TX/RX)

Transmit rate/receive rate of the line, in bps.

Status

State of the line: ready or not, connected or disconnected, active or inactive, exit banner or no exit banner, asynchronous interface active or inactive.

Capabilities

Current terminal capabilities.

Modem state

Modem control state. Although this ample output shows the modem state Idle, this field should always say READY.

Special Chars

Current settings of special characters that were input by the user (or taken by default) from the following global configuration commands:

- escape-character
- hold-character
- stop-character
- start-character
- disconnect-character
- activation-character

Timeouts

Current settings that were input by the user (or taken by default) from the following global configuration commands:

- exec-timeout
- session-timeout
- dispatch-timeout
- modem answer-timeout
- session-disconnect-warning
- timeout login response
- autoselect timeout

Session limit

Maximum number of sessions.

Time since activation

Last time start_process was run.

Table 20 show line Field Descriptions (continued)
Table 20 show line Field Descriptions (continued)

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Editing</td>
<td>Whether command-line editing is enabled.</td>
</tr>
<tr>
<td>History</td>
<td>Current history list size, set by the user (or taken by default) from the history configuration command.</td>
</tr>
<tr>
<td>DNS resolution in show commands is</td>
<td>Whether Open Shortest Path First (OSPF) is configured to look up Domain Name System (DNS) names for use in show EXEC command displays.</td>
</tr>
<tr>
<td>Full user help</td>
<td>Whether full user help has been set by the user with the terminal full-help EXEC command or by the administrator with the full-help line configuration command.</td>
</tr>
<tr>
<td>Allowed input transports are</td>
<td>Current set transport method, set by the user (or taken by default) from the transport input line configuration command.</td>
</tr>
<tr>
<td>Allowed output transports are</td>
<td>Current set transport method, set by the user (or taken by default) from the transport output line configuration command.</td>
</tr>
<tr>
<td>Preferred transport is</td>
<td>Current set transport method, set by the user (or taken by default) from the transport preferred line configuration command.</td>
</tr>
<tr>
<td>...characters are padded</td>
<td>Current set padding, set by the user (or taken by default) from the padding line configuration command.</td>
</tr>
<tr>
<td>...data dispatching characters</td>
<td>Current dispatch character set by the user (or taken by default) from the dispatch-character line configuration command.</td>
</tr>
</tbody>
</table>

The following sample output from the show line summary command shows line summary status for lines 1/3/36 through 1/3/102. Each row of output represents 36 lines, and the line status characters in groups of four for readability. For example, the first row represents information on rows 1/3/36 through 1/3/71, and the first line status character “U” corresponds to line 1/3/36.

Router# show line summary

1/3/36: U??? ---- ---- ---- ---- --u- ---- ???? ....
1/3/72: ---- ???? ---- ???? ???? MMMM MMMM MMMM MMMM

2 character mode users.           (U)
19 lines never used.              (?)
31 lines used, but currently idle. (-)
4 lines do not exist.              (.)
16 lines in use by modem management (M)

18 total lines in use, 1 not authenticated (lowercase)

Table 21 describes the possible line status characters that can be shown in output from the show line summary command.

Table 21 Line Status Character Descriptions

<table>
<thead>
<tr>
<th>Line Status Character</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>?</td>
<td>Line has never been used.</td>
</tr>
<tr>
<td>-</td>
<td>Lines has been used but is currently idle.</td>
</tr>
<tr>
<td>.</td>
<td>Line does not exist.</td>
</tr>
</tbody>
</table>
### Table 21  Line Status Character Descriptions (continued)

<table>
<thead>
<tr>
<th>Line Status Character</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Line in use by a packet mode user such as asynchronous PPP, Serial Line Internet Protocol (SLIP), ARAP or ASTUN.</td>
</tr>
<tr>
<td>D</td>
<td>Line in use by a digit mode user such as V.110 or V.120.</td>
</tr>
<tr>
<td>F</td>
<td>Line in use by a TCP fast-stream user.</td>
</tr>
<tr>
<td>M</td>
<td>Line in use by modem management.</td>
</tr>
<tr>
<td>U</td>
<td>Line in use by character mode user, such as an EXEC user making an outbound packet connection using Telnet, rlogin, local-area transport (LAT), packet assembler/disassembler (PAD), or normal (nonfast-stream) TCP clear.</td>
</tr>
<tr>
<td>V</td>
<td>Line in use by a voice mode user.</td>
</tr>
</tbody>
</table>
show node

To display information about local-area transport (LAT) nodes, use the **show node** command in EXEC mode.

```
show node [all | node-name] [counters | status | summary]
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>all</td>
<td>(Optional) Specifies all nodes.</td>
</tr>
<tr>
<td>node-name</td>
<td>(Optional) Indicates the name of the node for which status is required.</td>
</tr>
<tr>
<td>counters</td>
<td>(Optional) Specifies the various node counters.</td>
</tr>
<tr>
<td>status</td>
<td>(Optional) Specifies detailed node status. This is the default if a node name is specified.</td>
</tr>
<tr>
<td>summary</td>
<td>(Optional) Specifies a status summary for the node. This is the default if no node name is specified.</td>
</tr>
</tbody>
</table>

**Command Modes**

EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Entering the **show node** command with no arguments is the same as entering the **show node all summary** command and shows a one-line summary of all known nodes.

You can enter the **show node** command with either a specific node name or the **all** keyword, but not both.

The **show node** command displays three different sets of information about a node: the node counters, the node status, or a one-line summary of the node status.

You can enter the **show node** command with only one of the **counters**, **status**, or **summary** keywords. If you enter **show node** and two of these keywords without specifying a node name, the first keyword is treated as a node name, causing an error. If you enter the **show node node-name** command and two of these keywords, the second keyword will be treated as ambiguous.

The **show node** command with a **node-name** argument but no **counters**, **status**, or **summary** keyword defaults to **show node node-name status**.

**Examples**

The following is sample output from the **show node** command with no further keywords (the same as the **show node all summary** command):

```
Router> show node

Node Name     Status     Identification
CHAOS         Reachable  
MUD-DY-RIVER  Reachable  
TARMAC        Reachable  
WHEEL         Reachable  Welcome to VAX/VMS V5.4-2
```
Table 22 describes the significant fields shown in the display.

**Table 22  show node Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Node Name</td>
<td>Lists the names of the nodes.</td>
</tr>
<tr>
<td>Status</td>
<td>Indicates whether the node is reachable or not.</td>
</tr>
<tr>
<td>Identification</td>
<td>Identification string for the node.</td>
</tr>
</tbody>
</table>

The following is sample output from the `show node` output that defaults to `show node chaos status`. It results in a display of the detailed status of node chaos.

```
Router> show node chaos

Node: CHAOS     Address: 00-00-0C-01-05-09
LAT Protocol: V5.1   Data Link Frame Size: 1500
Identification:
Node Groups: 0
Service Name  Status Rating Identification
CHAOS   Available 80
```

Table 23 describes the significant fields shown in the display.

**Table 23  show node status Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Node</td>
<td>Lists the node name as reported by the host computer.</td>
</tr>
<tr>
<td>Address</td>
<td>Identifies the MAC address of the node Ethernet interface.</td>
</tr>
<tr>
<td>LAT protocol</td>
<td>Lists the version of the LAT protocol used by the node.</td>
</tr>
<tr>
<td>Data Link Frame Size</td>
<td>Lists the size of the largest packet that can be sent to the LAT host.</td>
</tr>
<tr>
<td>Identification</td>
<td>Lists the identification string for the node.</td>
</tr>
<tr>
<td>Node Groups</td>
<td>Lists the group code list that is advertised by the remote node, which comes from the service advertisement of the remote node.</td>
</tr>
<tr>
<td>Service Name</td>
<td>Lists the LAT service name.</td>
</tr>
<tr>
<td>Status</td>
<td>Indicates whether the node is currently available on the network.</td>
</tr>
<tr>
<td>Rating</td>
<td>Indicates the rating of the service: an integer from 0 to 255, with the highest number being the preferred service. Used for load balancing.</td>
</tr>
</tbody>
</table>

The following sample output displays the counter information for a specific node:

```
Router> show node tarmac counters

Node: tarmac
Seconds Since Zeroed: 100 Multiple Node Addresses: 0
Messages Received: 0 Duplicates Received: 0
Messages Transmitted: 0 Messages Re-transmitted: 0
Slots Received: 0 Illegal Messages Received: 0
Slots Transmitted: 0 Illegal Slots Received: 0
Bytes Received: 0 Solicitations Accepted: 0
Bytes Transmitted: 0 Solicitations Rejected: 0
```
In the following sample command and output displays, the **status** keyword is treated as the node name:

Router> **show node status counters**

Local -710- Node STATUS not known

In the following example, the second keyword **counters** is treated as ambiguous:

Router> **show node lager status counters**

Local -702- Keyword "COUNTERS" not known or ambiguous
**show service**

To display specific local-area transport (LAT) learned services, use the `show service` command in EXEC mode.

```
show service [service-name]
```

**Syntax Description**

- `service-name` (Optional) The name of a specific LAT service.

**Command Modes**

- EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

The `show service` command without a service name displays a list of known LAT learned services. When entered with the `service-name` argument, it displays a more-detailed status of the named service. If no LAT learned service by the specified name is known, then a lookup is done for an IP host of that name.

**Examples**

The following is sample output from the `show service` command:

```
Router> show service

Service Name       Status       Identification
BLUE              Available     Welcome to VAX/VMS V5.4
CHAOS             Available     
MRL12             Available     
MUDY-RIVER        Available     
STELLA-BLUE       Available     Welcome to VAX/VMS V5.4
```

The following is sample output of the `show service` command for a specific service:

```
Router> show service blue

Service BLUE - Available
Node Name     Status       Rating       Identification
BLUE          reachable 84   Welcome to VAX/VMS V5.4
```

Table 24 describes the significant fields shown in the two previous displays.

**Table 24  show service Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Service</td>
<td>Name of the service.</td>
</tr>
<tr>
<td>Node Name</td>
<td>Name of the nodes advertising the service.</td>
</tr>
</tbody>
</table>
### Table 24  show service Field Descriptions (continued)

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Status</td>
<td>Status of the service: Available or Unknown when a command is entered without a service name. Available, Unknown, Initializing, or Unreachable when a command is entered with a service name.</td>
</tr>
<tr>
<td>Rating</td>
<td>Rating of the service: An integer from 0 to 255, with the highest number being the preferred service. Used for load balancing.</td>
</tr>
<tr>
<td>Identification</td>
<td>Identification string.</td>
</tr>
</tbody>
</table>
show terminal

To obtain information about the terminal configuration parameter settings for the current terminal line, use the `show terminal` command in EXEC mode.

```
show terminal
```

Syntax Description

This command has no arguments or keywords.

Command Modes

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

Examples

The following is sample output from the `show terminal` command:

```
Router# show terminal
Line 2, Location: "", Type: ""
Length: 24 lines, Width: 80 columns
Baud rate (TX/RX) is 9600/9600
Status: Ready, Active, No Exit Banner
Capabilities: Enabled
Modem state: Ready
Special Chars: Escape Hold Stop Start Disconnect Activation
  ^"x none - - none
Timeouts: Idle EXEC Idle Session Modem Answer Session Dispatch
  never never 0:00:15 not imp not set
Session limit is not set.
Allowed transports are telnet rlogin. Preferred is telnet
No output characters are padded
```

Table 25 describes the fields in the first three lines of the `show terminal` output.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Line 2</td>
<td>Current terminal line.</td>
</tr>
<tr>
<td>Location: &quot;&quot;</td>
<td>Location of the current terminal line, as specified using the <code>location</code> line configuration command.</td>
</tr>
<tr>
<td>Type: &quot;&quot;</td>
<td>Type of the current terminal line, as specified using the <code>line</code> global configuration command.</td>
</tr>
<tr>
<td>Length: 24 lines</td>
<td>Length of the terminal display.</td>
</tr>
<tr>
<td>Width: 80 columns</td>
<td>Width of the terminal display, in character columns.</td>
</tr>
<tr>
<td>Baud rate (TX/RX) is 9600/9600</td>
<td>Transmit rate/receive rate of the line.</td>
</tr>
</tbody>
</table>
The following line of output indicates the status of the line:

Status: Ready, Active, No Exit Banner

Table 26 describes possible values for the Status field.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Active</td>
<td>A process is actively using the line.</td>
</tr>
<tr>
<td>Autobauding</td>
<td>The line is running the autobaud process.</td>
</tr>
<tr>
<td>Carrier Dropped</td>
<td>Some sense of “carrier” has been dropped, so the line process should be killed.</td>
</tr>
<tr>
<td>Connected</td>
<td>The line has at least one active connection.</td>
</tr>
<tr>
<td>Dialing Out</td>
<td>A dial-on-demand routing (DDR) asynchronous interface is dialing a remote site on this line.</td>
</tr>
<tr>
<td>Echo Off</td>
<td>The line is not echoing what the user types in (for example, because a password must be entered).</td>
</tr>
<tr>
<td>Escape Started</td>
<td>The first character of the escape sequence has been typed.</td>
</tr>
<tr>
<td>Escape Typed</td>
<td>Both characters of the escape sequence have been typed.</td>
</tr>
<tr>
<td>Hanging Up</td>
<td>The line state is “hanging up.”</td>
</tr>
<tr>
<td>Hardware XON/XOFF</td>
<td>The line uses a Universal Asynchronous Receiver/Transmitter (UART) that supports XON/XOFF flow control in hardware. (This does not mean that the line is currently using software flow control.)</td>
</tr>
<tr>
<td>Hold Typed</td>
<td>The user typed the “hold character” (and the line is paused).</td>
</tr>
<tr>
<td>Idle</td>
<td>The line modem state is “idle” (see modem state diagrams).</td>
</tr>
<tr>
<td>Idle Timeout</td>
<td>An idle timeout has occurred.</td>
</tr>
<tr>
<td>Input Stopped</td>
<td>The input has been turned off because of hardware flow control or overflow.</td>
</tr>
<tr>
<td>No Exit Banner</td>
<td>The normal exit banner will not be displayed on this line.</td>
</tr>
<tr>
<td>PSI Enabled</td>
<td>The line is paying attention to typed escape characters.</td>
</tr>
<tr>
<td>Rcvd BREAK</td>
<td>A BREAK sequence has been received on the line.</td>
</tr>
<tr>
<td>Rcvd Command</td>
<td>The line has received a special command sequence (for example, ^C for send break).</td>
</tr>
<tr>
<td>Rcvd CR</td>
<td>The last character received was a carriage return.</td>
</tr>
<tr>
<td>Ready</td>
<td>The line state is “ready.”</td>
</tr>
<tr>
<td>Ring Transition</td>
<td>A transition has occurred on the RING signal of the line.</td>
</tr>
<tr>
<td>Send Break Soon</td>
<td>Send a BREAK on the line soon.</td>
</tr>
<tr>
<td>Send XOFF Soon</td>
<td>Buffers are full and an XOFF should be sent soon.</td>
</tr>
<tr>
<td>Sending Break</td>
<td>A BREAK sequence is being sent on the line.</td>
</tr>
<tr>
<td>Sent XOFF</td>
<td>Buffers were full, so an XOFF was sent.</td>
</tr>
<tr>
<td>SLIP Mode</td>
<td>The line is running SLIP or PPP.</td>
</tr>
</tbody>
</table>
The following line of output indicates the status of the capabilities of the line. These capabilities correspond closely to configurable parameters that can be set using configuration commands.

Capabilities: Enabled

Table 27 describes possible values for the Capabilities field.

**Table 27  show terminal Field Descriptions—Capabilities Field Values**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Autobaud Full Range</td>
<td>Corresponds to the <code>autobaud</code> command.</td>
</tr>
<tr>
<td>Character Padding</td>
<td>At least one <code>pad c x</code> configuration command has been used.</td>
</tr>
<tr>
<td>Enabled</td>
<td>The user has “enabled” successfully.</td>
</tr>
<tr>
<td>EXEC Suppressed</td>
<td>Corresponds to the <code>no exec</code> command.</td>
</tr>
<tr>
<td>Hangup on Last Close</td>
<td>Corresponds to the <code>autohangup</code> command.</td>
</tr>
<tr>
<td>Hardware Flowcontrol In</td>
<td>Corresponds to the <code>flowcontrol hardware in</code> command.</td>
</tr>
<tr>
<td>Hardware Flowcontrol Out</td>
<td>Corresponds to the <code>flowcontrol hardware out</code> command.</td>
</tr>
<tr>
<td>Insecure</td>
<td>Corresponds to the <code>insecure</code> command.</td>
</tr>
<tr>
<td>Lockable</td>
<td>Corresponds to the <code>lockable</code> command.</td>
</tr>
<tr>
<td>Modem Callin</td>
<td>Corresponds to the <code>modem callin</code> command.</td>
</tr>
<tr>
<td>Modem Callout</td>
<td>Corresponds to the <code>modem callout</code> command.</td>
</tr>
<tr>
<td>Modem CTS-Required</td>
<td>Corresponds to the <code>modem cts-required</code> command.</td>
</tr>
<tr>
<td>Modem DTR-Active</td>
<td>Corresponds to the <code>modem dtr-active</code> command.</td>
</tr>
<tr>
<td>Modem RI is CD</td>
<td>Corresponds to the <code>modem ri-is-cd</code> command.</td>
</tr>
<tr>
<td>No Login Banner</td>
<td>Corresponds to the <code>no exec-banner</code> command.</td>
</tr>
<tr>
<td>Notification Set</td>
<td>Corresponds to the <code>notify</code> command.</td>
</tr>
<tr>
<td>Output Non-Idle</td>
<td>Corresponds to the <code>session-timeout</code> command.</td>
</tr>
<tr>
<td>Permanent SLIP</td>
<td>Corresponds to the <code>slip-dedicated</code> command.</td>
</tr>
<tr>
<td>Private Line</td>
<td>Corresponds to the <code>private</code> command.</td>
</tr>
<tr>
<td>Refuse Suppress-GA</td>
<td>Corresponds to the <code>telnet refuse</code> command.</td>
</tr>
<tr>
<td>Receives Logging Output</td>
<td>Corresponds to the <code>monitor</code> command.</td>
</tr>
<tr>
<td>Refuse Telnet Echo</td>
<td>Corresponds to the <code>telnet refuse</code> command.</td>
</tr>
<tr>
<td>Send BREAK on IP</td>
<td>Corresponds to the <code>telnet break-on-ip</code> command.</td>
</tr>
<tr>
<td>SLIP allowed</td>
<td>Corresponds to the <code>slip address</code> command.</td>
</tr>
<tr>
<td>Software Flowcontrol In</td>
<td>Corresponds to the <code>flowcontrol software in</code> command.</td>
</tr>
<tr>
<td>Software Flowcontrol Out</td>
<td>Corresponds to the <code>flowcontrol software out</code> command.</td>
</tr>
<tr>
<td>Telnet Transparent Mode</td>
<td>Corresponds to the <code>telnet transparent</code> command.</td>
</tr>
</tbody>
</table>

The following line of output indicates the modem state. Values include Autobauding, Carrier Dropped, Hanging Up, Idle, and Ready.

Modem state: Ready
The following lines of output indicate the special characters that can be entered to activate various terminal operations. The none or hyphen (-) values imply that no special characters are set.

Special Chars: Escape Hold Stop Start Disconnect Activation
  ^^x  none  -  -  none

The following lines of output indicate the timeout values that have been configured for the line:

Timeouts:   Idle EXEC  Idle Session  Modem Answer  Session  Dispatch
            never       never        0:00:15      not imp    not set

Table 28 describes the fields in the preceding lines of output.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Idle EXEC</td>
<td>Interval that the EXEC command interpreter waits for user input before resuming the current connection; or if no connections exist, returning the terminal to the idle state and disconnecting the incoming session. This interval is set using the <code>exec-timeout</code> command.</td>
</tr>
<tr>
<td>Idle Session</td>
<td>Interval that the Cisco IOS software waits for traffic before closing the connection to a remote computer and returning the terminal to an idle state. This interval is set using the <code>session-timeout</code> command.</td>
</tr>
<tr>
<td>Modem Answer</td>
<td>Interval during which the software raises DTR in response to RING and the modem response to CTS. This interval is set using the <code>modem answer-timeout</code> command.</td>
</tr>
<tr>
<td>Session</td>
<td>Not implemented in this release.</td>
</tr>
<tr>
<td>Dispatch</td>
<td>Number of milliseconds the software waits after putting the first character into a packet buffer before sending the packet. This interval is set using the <code>dispatch-timeout</code> command.</td>
</tr>
</tbody>
</table>

The following lines of output indicate how various options have been configured:

- Session limit is not set.
- Allowed transports are telnet rlogin. Preferred is telnet
- No output characters are padded
show tn3270 ascii-hexval

To determine ASCII-hexadecimal character mappings, use the show tn3270 ascii-hexval command in EXEC mode.

show tn3270 ascii-hexval

Syntax Description
This command has no arguments or keywords.

Command Modes
EXEC

Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.3</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

Usage Guidelines
Use the show tn3270 ascii-hexval command to display the hexadecimal value of a character on your keyboard. After you enter the show tn3270 ascii-hexval command, you will be prompted to press a key. The hexadecimal value of the ASCII character is displayed. This command is useful for users who do not know the ASCII codes associated with various keys or do not have manuals for their terminals.

Examples
The following is sample output from the show tn3270 ascii-hexval command:

Router> show tn3270 ascii-hexval
Press key> 7 - hexadecimal value is 0x37.

Router> show tn3270 ascii-hexval
Press key> f - hexadecimal value is 0x66.

Router> show tn3270 ascii-hexval
Press key> not printable - hexadecimal value is 0xD.

Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>show tn3270 character-map</td>
<td>Displays character mappings between ASCII and EBCDIC.</td>
</tr>
<tr>
<td>tn3270 character-map</td>
<td>Converts incoming EBCDIC characters into ASCII characters for TN3270 connections.</td>
</tr>
</tbody>
</table>
show tn3270 character-map

To display character mappings between ASCII and EBCDIC, use the `show tn3270 character-map` command in EXEC mode.

```
show tn3270 character-map { all | ebcdic-in-hex }
```

### Syntax Description

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>all</td>
<td>Displays all nonstandard character mappings.</td>
</tr>
<tr>
<td>ebcdic-in-hex</td>
<td>Displays the ASCII mapping for a specific EBCDIC character.</td>
</tr>
</tbody>
</table>

### Command Modes

EXEC

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>11.1</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

### Examples

The following is sample output from the `show tn3270 character-map` command:

```
Router# show tn3270 character-map all
EBCDIC 0x81 <=> 0x78 ASCII
EBCDIC 0x82 <=> 0x79 ASCII
EBCDIC 0x83 <=> 0x7A ASCII
```

### Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>show tn3270 ascii-hexval</td>
<td>Displays ASCII-hexadecimal character mappings.</td>
</tr>
<tr>
<td>tn3270 character-map</td>
<td>Converts incoming EBCDIC characters into ASCII characters for TN3270 connections.</td>
</tr>
</tbody>
</table>
show translate

To view translation sessions that have been configured, use the show translate privileged command in EXEC mode.

```
show translate
```

**Syntax Description**
This command has no arguments or keywords.

**Command Modes**
Privileged EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.3</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**
The display from this command shows each translation session set up on the router. It shows the incoming device and virtual terminal protocol, and the outgoing device and protocol.

**Examples**
The following show translate sample output is based on the following translate command configuration:

```
translate x25 3131415912345 ppp ip-pool scope-name cardinal keepalive 0
```

If the previous translate command is enabled, the following output is created by the show translate command:

```
Router# show translate
Translate From: x25 3131415912345
   To:   PPP ip-pool scope-name cardinal keepalive 0
       1/1 users active, 1 peak, 1 total, 0 failures
```

Table 29 describes the significant fields shown in the display.

**Table 29  show translate Field Descriptions—X.25 to IP Translation**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Translate From: x25 3131415912345</td>
<td>Protocol (X.25) and address (3131415912345) of the incoming device.</td>
</tr>
<tr>
<td>To: PPP</td>
<td>The virtual terminal protocol (PPP).</td>
</tr>
<tr>
<td>ip-pool</td>
<td>Obtain an IP address from a DHCP proxy client or a local pool.</td>
</tr>
<tr>
<td>scope-name cardinal</td>
<td>Specific local scope name (cardinal) from which to obtain an IP address.</td>
</tr>
<tr>
<td>keepalive 0</td>
<td>Indicates that keepalive updates have been disabled for the current translation session.</td>
</tr>
<tr>
<td>1/1 users active</td>
<td>Number of users active over the total number of users.</td>
</tr>
</tbody>
</table>
The following `show translate` sample output is based on the following `translate` command configuration:

```plaintext
translate x25 31301234 PPP 192.168.14.23 ipx-client Loopback0
```

If the previous `translate` command is enabled, the following output is created by the `show translate` command:

```plaintext
Router# show translate
Translate From: x25 31301234
    To:   PPP 192.168.14.23  ipx-client Loopback0
    1/1 users active, 1 peak, 1 total, 0 failures
```

Table 29 describes the significant fields shown in the display.

### Table 29  `show translate` Field Descriptions—X.25 to IP Translation (continued)

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 peak</td>
<td>Maximum number of translate sessions up at any given time.</td>
</tr>
<tr>
<td>1 total</td>
<td>Total number of translation sessions.</td>
</tr>
<tr>
<td>0 failures</td>
<td>Number of failed translation attempts resulting from this configuration.</td>
</tr>
</tbody>
</table>

The following `show translate` sample output is based on the following `translate` command configuration:

```plaintext
translate x25 31301234 PPP 192.168.14.23 ipx-client Loopback0
```

If the previous `translate` command is enabled, the following output is created by the `show translate` command:

```plaintext
Router# show translate
Translate From: x25 31301234
    To:   PPP 192.168.14.23  ipx-client Loopback0
    1/1 users active, 1 peak, 1 total, 0 failures
```

Table 30 describes the significant fields shown in the display.

### Table 30  `show translate` Field Descriptions—X.25 to IPX Translation

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Translate From: x25 31301234</td>
<td>Protocol (X.25) and address (31301234) of the incoming device.</td>
</tr>
<tr>
<td>To: PPP 192.168.14.23</td>
<td>The virtual terminal protocol (PPP) and IP address of the outgoing device.</td>
</tr>
<tr>
<td>ipx-client loopback0</td>
<td>Indicates that loopback interface 0 has been configured in client mode.</td>
</tr>
<tr>
<td>1/1 users active</td>
<td>Number of users active over the total number of users.</td>
</tr>
<tr>
<td>1 peak</td>
<td>Maximum number of translate sessions up at any given time.</td>
</tr>
<tr>
<td>1 total</td>
<td>Total number of translation sessions.</td>
</tr>
<tr>
<td>0 failures</td>
<td>Number of failed translation attempts resulting from this configuration.</td>
</tr>
</tbody>
</table>
show ttycap

To test for the availability of a ttycap after a connection on a router takes place, use the show ttycap command in EXEC mode.

```
show ttycap [ttycap-name | all]
```

**Syntax Description**

- **ttycap-name** (Optional) Name of a ttycap.
- **all** (Optional) Lists the names of all defined ttycaps. The name of the default ttycap is not listed.

**Command Modes**

EXEC

**Command History**

- **Release** 10.0 This command was introduced.

**Usage Guidelines**

The Cisco IOS software searches for the specified ttycap in its active configuration image, and lists the complete entry if found. If it is not found, an appropriate “not found” message appears.

If you do not include any arguments with the show ttycap command, then the current keymap used for the terminal is displayed.

**Examples**

The following is sample output from the show ttycap command:

```
Router# show ttycap
d0|vt100|vt100-am|vt100am|dec vt100:do="J:co#0:li#24:\n cl=50"[H][J:bs:am:cm=5"[I%d;dh:nd=2"[[C:up=2"[[A:ce=3["k:so=2"[7m:me=2"[4m:ue=2"[m:md=2"[1m:\me=2"[m:ho=8[H:;H:sc=8;cm="[%%m;\dr:

Router# show ttycap all
ttycap3   d0|vt100|vt100-am|vt100am|dec vt100
          ttycap2   d1|vt200|vt220|vt200-js|vt220-js|dec vt200 series with jump scroll
          ttycap1   ku|h19-u|h19u|heathkit with underscore cursor

Router# show ttycap ttycap1
          ttycap1   ku|h19-u|h19u|heathkit with underscore cursor:
```

Cisco IOS Terminal Services Command Reference
show users

To display information about the active lines on the router, use the **show users** command in EXEC mode.

```
show users [all]
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>all</td>
<td>(Optional) Specifies that all lines be displayed, regardless of whether anyone is using them.</td>
</tr>
</tbody>
</table>

**Command Modes**

EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

This command displays the line number, connection name, idle time, hosts (including virtual access interfaces), and terminal location. An asterisk (*) indicates the current terminal session.

**Examples**

The following is sample output from the **show users** command:

```
Router# show users
Line    User    Host(s)    Idle   Location
0 con 0  idle    0 BASHFUL.CISCO.COM
* 2 vty 0 rose    idle    0 BASHFUL.CISCO.COM
```

The following is sample output identifying an active virtual access interface:

```
Router# show users
Line    User    Host(s)    Idle   Location
* 0 con 0  idle    01:58
10 vty 0 Virtual-Access2 0 1212321
```

The following is sample output from the **show users all** command:

```
Router# show users all
Line    User    Host(s)    Idle   Location
* 0 vty 0 rose    idle    0 BASHFUL.CISCO.COM
1 vty 1
2 con 0
3 aux 0
4 vty 2
```

**Table 31** describes the significant fields shown in the displays.
Table 31  show users Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
</table>
| Line  | Contains three subfields:  
|       | - The first subfield (0 in the sample output) is the absolute line number.  
|       | - The second subfield (vty) indicates the type of line. Possible values follow:  
|       | con—console  
|       | aux—auxiliary port  
|       | tty—asynchronous terminal port  
|       | vty—virtual terminal  
|       | - The third subfield (0 in the * sample output) indicates the relative line number within the type.  
| User  | User using the line. If no user is listed in this field, no one is using the line.  
| Host(s)| Host to which the user is connected (outgoing connection). A value of idle means that there is no outgoing connection to a host.  
| Idle  | Interval (in minutes) since the user has entered something.  
| Location | Either the hard-wired location for the line or, if there is an incoming connection, the host from which incoming connection came.  

Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
</table>
| line    | Identifies a specific line for configuration and starts the line configuration command collection mode.  
| show line | Displays the parameters of a terminal line.  

show x25 pad

To display information about current open connections, including packet transmissions, X.3 parameter settings, and the current status of virtual circuits, use the **show x25 pad** command in EXEC mode.

```
show x25 pad
```

**Syntax Description**

This command has no arguments or keywords.

**Command Modes**

EXEC

**Command History**

Release      Modification
11.2          This command was introduced.

**Examples**

The following is sample output from the **show x25 pad** command:

```
Router# show x25 pad
tty2, Incoming PAD connection
Total input: 61, control 6, bytes 129. Queued: 0 of 7 (0 bytes).
Total output: 65, control 6, bytes 696.
Flags: 1, State: 3, Last error: 1
  ParamsIn:  1:1, 2:0, 3:2, 4:1, 5:1, 6:0, 7:21,
             8:0, 9:0, 10:0, 11:14, 12:0, 13:0, 14:0, 15:1,
             16:127, 17:21, 18:18, 19:0, 20:0, 21:0, 22:0,
  ParamsOut:  1:1, 2:1, 3:2, 4:1, 5:0, 6:0, 7:4,
             8:0, 9:0, 10:0, 11:14, 12:0, 13:0, 14:0, 15:0,
             16:127, 17:21, 18:18, 19:0, 20:0, 21:0, 22:0,
    LCI: 1, State: D1, Interface: Serial0
Started 0:11:10, last input 0:00:16, output 0:00:16
Connected to 313700540651
Window size input: 7, output: 7
Packet size input: 512, output: 512
PS: 1 PR: 5 ACK: 5 Remote PR: 1 RCNT: 0 ENR: FALSE
Retransmits: 0 Timer (secs): 0 Reassembly (bytes): 0
Held Fragments/Packets: 0/0
Bytes 696/129 Packets 65/61 Resets 0/0 RNRs 0/0 REJs 0/0 INTs 0/0
```

**Table 32** describes fields seen in the display from the **show x25 pad** command.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total input/output</td>
<td>Number of packets received or sent for the connection.</td>
</tr>
<tr>
<td>control</td>
<td>Number of packets with Qbit set (X.29 control packets).</td>
</tr>
<tr>
<td>bytes</td>
<td>Number of bytes in each direction.</td>
</tr>
<tr>
<td>Queued</td>
<td>Number of unread packets waiting for the connection.</td>
</tr>
<tr>
<td>Waiting to send</td>
<td>Local data packet bit not sent (part of a line).</td>
</tr>
</tbody>
</table>
### Table 32  show x25 pad Field Descriptions (continued)

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Flags, State, Last error</td>
<td>Displays data for detecting errors and tracing initialization status. Only useful to your Cisco-certified technical support personnel.</td>
</tr>
<tr>
<td>ParamsIn</td>
<td>Parameters read from the PAD at the start of the connection.</td>
</tr>
<tr>
<td>ParamsOut</td>
<td>Active X.3 parameters.</td>
</tr>
<tr>
<td>LCI, State, Interface</td>
<td>Status of the X.25 virtual circuit associated with the PAD connection. This is the same display that the <code>show x25 vc</code> command shows.</td>
</tr>
</tbody>
</table>
show xremote

To display XRemote connections and monitor XRemote traffic through the router, use the `show xremote` command in EXEC mode.

```
show xremote
```

**Syntax Description**

This command has no arguments or keywords.

**Command Modes**

EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

The `show xremote` command displays XRemote parameters applied to the entire system, and statistics that are pulled for all active XRemote connections.

**Examples**

The following is sample output from the `show xremote` command when XRemote is enabled and XRemote sessions are active:

```
Router# show xremote

XRemote server-wide parameters:
  Font buffersize:     72000  Font retries: 3
  Font memory errors:  0

TFTP font load statistics for host 172.16.1.111:
  Bytes read:          2697239  Files read: 258
  Network errors:      4  File errors: 0

LAT font load statistics for service WHEEL, incarnation 5:
  Bytes read 182401  Files read: 14
  Protocol errors: 0  Insufficient memory: 0

XRemote statistics for tty2:
  Current clients:     9  Total clients: 17
  Requesting client:   5  Current request size: 0
  Replying client:     6  Current reply size: 0
  XDM state:           10  Next timeout: 172460
  Retransmit counter:  0  Local UDP port: 53616
  Keepalive dormancy:  180  Session id: 94
  Unread input:        0  Unwritten output: 0
  Input buffer size:   1024  Output buffer size: 108
  Protocol version:    2  Line state: Connected
  Transmit packets:   50768  Receive packets: 49444
  Transmit errors:    0  Receive errors: 37
  Retransmissions:    403  Receive out of sequence: 76
  Round trip time:    383  Retransmit interval: 766
  Transmit window:    7  Transmit next: 6
  Transmit unacked:   6
```

Cisco IOS Terminal Services Command Reference
Connection 0 - TCP connection from 172.16.1.55  [Display Manager]
Client state: CS_ACTIVE  Byte order: MSBfirst
Unread input: 0  Unwritten output: 0
Input buffer size: 1024  Output buffer size: 1024

Connection 1 - LAT connection from WHEEL
Client state: CS_ACTIVE  Byte order: LSBfirst
Unread input: 0  Unwritten output: 0
Input buffer size: 1024  Output buffer size: 1024

Table 33 describes the significant fields shown in the display.

### Table 33  show xremote Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>XRemote server-wide parameters</td>
<td>This section displays XRemote parameters that apply to the protocol translator.</td>
</tr>
<tr>
<td>Font buffersize</td>
<td>XRemote font buffer size that was specified with the <code>xremote tftp buffersize</code> global configuration command.</td>
</tr>
<tr>
<td>Font retries</td>
<td>Number of retries the font loader (host) will attempt before declaring an error condition.</td>
</tr>
<tr>
<td>Font memory errors</td>
<td>Number of font memory error conditions that have been declared for the protocol translator.</td>
</tr>
<tr>
<td>TFTP font load statistics for host 172.16.1.111</td>
<td>This section displays XRemote statistics for fonts that have been loaded from a TFTP font server at the IP address shown.</td>
</tr>
<tr>
<td>Bytes read</td>
<td>Number of bytes the host read in order to load the fonts.</td>
</tr>
<tr>
<td>Files read</td>
<td>Number of files the host read in order to load the fonts.</td>
</tr>
<tr>
<td>XRemote statistics for tty2</td>
<td>This section displays XRemote for the specified line.</td>
</tr>
<tr>
<td>Current clients</td>
<td>Number of clients using this line for active XRemote sessions.</td>
</tr>
<tr>
<td>Total clients</td>
<td>Includes the number of clients using this line for active XRemote sessions.</td>
</tr>
<tr>
<td>Requesting client</td>
<td>Number of clients requesting XRemote service.</td>
</tr>
<tr>
<td>Retransmit counter</td>
<td>Number of times that XRemote connection requests were resent.</td>
</tr>
<tr>
<td>Local UDP port</td>
<td>Number assigned to the local UDP port.</td>
</tr>
<tr>
<td>Keepalive dormancy</td>
<td>Amount of time between keepalive messages.</td>
</tr>
</tbody>
</table>
show xremote line

To list XRemote connections and monitor XRemote traffic, use the `show xremote line` command in EXEC mode.

```
show xremote line number
```

**Syntax Description**

```
number
```

A decimal value representing the number of virtual terminal lines about which to display information.

**Command Modes**

EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Examples**

The following is sample output from the `show xremote line` command (line 3 is specified) when XRemote is enabled and XRemote sessions are active. Only information specific to an individual terminal line is provided. See Table 33 for output field descriptions.

```
Router# show xremote line 3
Xremote statistics for tty3:
  Current clients: 11                 Total clients: 19
  Requesting client: 10                Current request size: 0
  Replying client: 10                  Current reply size: 0
  XDM state: 10                        Next timeout: 173304
  Retransmit counter: 0                Local UDP port: 28384
  Keepalive dormancy: 180              Session id: 29
  Unread input: 0                      Unwritten output: 0
  Input buffer size: 1024               Output buffer size: 108
  Protocol version: 2                  Line state: Connected
  Transmit packets: 28875              Receive packets: 18644
  Transmit errors: 0                   Receive errors: 13
  Retransmissions: 53                  Receive out of sequence: 41
  Round trip time: 184                 Retransmit interval: 768
  Transmit window: 7                   Receive window: 7
  Transmit next: 2                     Receive next: 7
  Transmit unacked: 2                  Receive unacked: 0

  Connection 0 - TCP connection from 172.16.1.27 [Display Manager]
    Client state: CS_ACTIVE             Byte order: MSBfirst
    Unread input: 0                     Unwritten output: 0
    Input buffer size: 1024             Output buffer size: 1024

  Connection 1 - TCP connection from 172.16.1.27
    Client state: CS_ACTIVE             Byte order: MSBfirst
    Unread input: 0                     Unwritten output: 0
    Input buffer size: 1024             Output buffer size: 1024

  Connection 2 - TCP connection from 172.16.1.27
    Client state: CS_ACTIVE             Byte order: MSBfirst
    Unread input: 0                     Unwritten output: 0
    Input buffer size: 1024             Output buffer size: 1024
```
slip

To start a serial connection to a remote host by using Serial Line Internet Protocol (SLIP), use the `slip` command in EXEC mode.

```
slip [/default] {remote-ip-address | remote-name} [@tacacs-server] [/routing] [/compressed]
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>/default</code></td>
<td>(Optional) Makes a SLIP connection when a default address has been configured.</td>
</tr>
<tr>
<td><code>remote-ip-address</code></td>
<td>IP address of the client workstation or PC.</td>
</tr>
<tr>
<td><code>remote-name</code></td>
<td>Name of the client workstation or PC.</td>
</tr>
<tr>
<td><code>@tacacs-server</code></td>
<td>(Optional) IP address or IP host name of the TACACS server to which your TACACS authentication request is sent.</td>
</tr>
<tr>
<td><code>/routing</code></td>
<td>(Optional) Indicates that the remote system is a router. Line must be configured for asynchronous routing using SLIP encapsulation.</td>
</tr>
<tr>
<td><code>/compressed</code></td>
<td>(Optional) Indicates that IP header compression should be negotiated.</td>
</tr>
</tbody>
</table>

**Command Modes**

EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced in a release prior to Cisco IOS Release 10.0.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

When you connect from a remote node computer to the EXEC facility on a router and want to connect from the router to a device on the network, issue the `slip` command.

If you specify an address for the TACACS server by using `/default` or `tacacs-server` arguments, the address must be the first parameter in the command after you enter `slip`. If you do not specify an address or enter `/default`, you are prompted for an IP address or host name. You can enter the `/default` keyword at this point.

If you do not use the `tacacs-server` argument to specify a TACACS server for SLIP address authentication, the TACACS server specified at login (if any) is used for the SLIP address query.

To optimize bandwidth on a line, SLIP enables compression of the SLIP packets using Van Jacobson TCP header compression as defined in RFC 1144.

Your system administrator must configure the system with the `ip tcp header-compression passive` command for the `/compressed` command option to be valid in EXEC mode. The `ip tcp header-compression` command forces header compression on or off. The default is to not compress the packets. The configuration file must have header compression on and the `slip /compressed` EXEC command must be entered for header compression to occur.

To terminate a session initiated with the `slip` command, disconnect from the device on the network using the command specific to that device. Then, exit from EXEC mode by using the `exit` command.
The following example makes a connection when a default IP address is assigned. Once a correct password is entered, you are placed in SLIP mode, and the IP address is displayed.

```
Router> slip
Password:
Entering SLIP mode.
Your IP address is 192.168.7.28, MTU is 1524 bytes
```

The following example illustrates the prompts displayed and the response required when you use dynamic addressing to assign the SLIP address:

```
Router> slip
IP address or hostname? 192.168.6.15
Password:
Entering SLIP mode
Your IP address is 192.168.6.15, MTU is 1524 bytes
```

In the preceding example, the address 172.31.6.15 has been assigned as the default. Password verification is still required before SLIP mode can be enabled.

```
Router> slip /default
Password:
Entering SLIP mode
Your IP address is 192.168.6.15, MTU is 1524 bytes
```

The following example illustrates the implementation of header compression on the interface with the IP address 172.24.2.1:

```
Router> slip 172.24.2.1 /compressed
Password:
Entering SLIP mode.
Interface IP address is 172.24.2.1, MTU is 1500 bytes.
Header compression will match your system
```

In the preceding example, the interface is configured for the `ip tcp header-compression passive` command, which permits the user to enter the `/compressed` keyword at the EXEC mode prompt. The message “Header compression will match your system” indicates that the user specified compression. If the line was configured for the `ip tcp header-compression on` command, this line would read “Header compression is On.”

The following example specifies a TACACS server named server1 for address authentication:

```
Router> slip 10.0.0.1@server1
Password:
Entering SLIP mode.
Interface IP address is 10.0.0.1, MTU is 1500 bytes
Header compression will match your system
```
To log in to a host that supports Telnet, use the **telnet** command in EXEC mode.

```
telnet host [port] [keyword]
```

### Syntax Description

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>host</code></td>
<td>A host name or an IP address.</td>
</tr>
<tr>
<td><code>port</code></td>
<td>(Optional) A decimal TCP port number; the default is the Telnet router port (decimal 23) on the host.</td>
</tr>
<tr>
<td><code>keyword</code></td>
<td>(Optional) One of the keywords listed in Table 34.</td>
</tr>
</tbody>
</table>

### Command Modes

EXEC

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.1</td>
<td>The <code>/quiet</code> keyword was added.</td>
</tr>
</tbody>
</table>

### Usage Guidelines

Table 34 lists the optional telnet command keywords.

#### Table 34 telnet Keyword Options

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>/debug</code></td>
<td>Enables Telnet debugging mode.</td>
</tr>
</tbody>
</table>
| `/encrypt kerberos`| Enables an encrypted Telnet session. This keyword is available only if you have the Kerberized Telnet subsystem.  
If you authenticate using Kerberos Credentials, the use of this keyword initiates an encryption negotiation with the remote server. If the encryption negotiation fails, the Telnet connection will be reset. If the encryption negotiation is successful, the Telnet connection will be established, and the Telnet session will continue in encrypted mode (all Telnet traffic for the session will be encrypted). |
| `/line`            | Enables Telnet line mode. In this mode, the Cisco IOS software sends no data to the host until you press the Enter key. You can edit the line using the standard Cisco IOS software command-editing characters. The `/line` keyword is a local switch; the remote router is not notified of the mode change. |
| `/noecho`          | Disables local echo.                                                        |
| `/quiet`           | Prevents onscreen display of all messages from the Cisco IOS software.      |
| `/route path`      | Specifies loose source routing. The `path` argument is a list of host names or IP addresses that specify network nodes and ends with the final destination. |
| `/source-interface`| Specifies the source interface.                                              |
**Table 34  telnet Keyword Options (continued)**

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/stream</td>
<td>Turns on stream processing, which enables a raw TCP stream with no Telnet control sequences. A stream connection does not process Telnet options and can be appropriate for connections to ports running UNIX-to-UNIX Copy Program (UUCP) and other non-Telnet protocols.</td>
</tr>
<tr>
<td>port-number</td>
<td>Port number.</td>
</tr>
<tr>
<td>bgp</td>
<td>Border Gateway Protocol.</td>
</tr>
<tr>
<td>chargen</td>
<td>Character generator.</td>
</tr>
<tr>
<td>cmd rcmd</td>
<td>Remote commands.</td>
</tr>
<tr>
<td>daytime</td>
<td>Daytime.</td>
</tr>
<tr>
<td>discard</td>
<td>Discard.</td>
</tr>
<tr>
<td>domain</td>
<td>Domain Name Service.</td>
</tr>
<tr>
<td>echo</td>
<td>Echo.</td>
</tr>
<tr>
<td>exec</td>
<td>EXEC.</td>
</tr>
<tr>
<td>finger</td>
<td>Finger.</td>
</tr>
<tr>
<td>ftp</td>
<td>File Transfer Protocol.</td>
</tr>
<tr>
<td>ftp-data</td>
<td>FTP data connections (used infrequently).</td>
</tr>
<tr>
<td>gopher</td>
<td>Gopher.</td>
</tr>
<tr>
<td>hostname</td>
<td>Host name server.</td>
</tr>
<tr>
<td>ident</td>
<td>Ident Protocol.</td>
</tr>
<tr>
<td>irc</td>
<td>Internet Relay Chat.</td>
</tr>
<tr>
<td>klogin</td>
<td>Kerberos login.</td>
</tr>
<tr>
<td>kshell</td>
<td>Kerberos shell.</td>
</tr>
<tr>
<td>login</td>
<td>Login (rlogin).</td>
</tr>
<tr>
<td>lpd</td>
<td>Printer service.</td>
</tr>
<tr>
<td>nntp</td>
<td>Network News Transport Protocol.</td>
</tr>
<tr>
<td>node</td>
<td>Connect to a specific LAT node</td>
</tr>
<tr>
<td>pop2</td>
<td>Post Office Protocol v2.</td>
</tr>
<tr>
<td>pop3</td>
<td>Post Office Protocol v3.</td>
</tr>
<tr>
<td>port</td>
<td>Destination LAT port name.</td>
</tr>
<tr>
<td>smtp</td>
<td>Simple Mail Transport Protocol.</td>
</tr>
<tr>
<td>sunrpc</td>
<td>Sun Remote Procedure Call.</td>
</tr>
<tr>
<td>syslog</td>
<td>Syslog.</td>
</tr>
<tr>
<td>tacacs</td>
<td>Specify TACACS security.</td>
</tr>
<tr>
<td>talk</td>
<td>Talk.</td>
</tr>
<tr>
<td>telnet</td>
<td>Telnet.</td>
</tr>
<tr>
<td>time</td>
<td>Time.</td>
</tr>
<tr>
<td>uucp</td>
<td>UNIX-to-UNIX Copy Program.</td>
</tr>
</tbody>
</table>
With the Cisco IOS implementation of TCP/IP, you are not required to enter the `connect` or `telnet` commands to establish a terminal connection. You can enter only the learned host name—as long as the following conditions are met:

- The host name is different from a command word for the router.
- The preferred transport protocol is set to `telnet`.

To display a list of the available hosts, use the `show hosts` command. To display the status of all TCP connections, use the `show tcp` command.

The Cisco IOS software assigns a logical name to each connection, and several commands use these names to identify connections. The logical name is the same as the host name, unless that name is already in use, or you change the connection name with the `name-connection` EXEC command. If the name is already in use, the Cisco IOS software assigns a null name to the connection.

The Telnet software supports special Telnet commands in the form of Telnet sequences that map generic terminal control functions to operating system-specific functions. To issue a special Telnet command, enter the escape sequence and then a command character. The default escape sequence is Ctrl-\` (press and hold the Ctrl-Shift-6). You can enter the command character as you hold down Ctrl or with Ctrl released; you can use either uppercase or lowercase letters. Table 35 lists the special Telnet escape sequences.

### Table 34  telnet Keyword Options (continued)

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>whois</td>
<td>Nickname.</td>
</tr>
</tbody>
</table>

### Table 35  Special Telnet Escape Sequences

<table>
<thead>
<tr>
<th>Escape Sequence¹</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ctrl-` b</td>
<td>Break</td>
</tr>
<tr>
<td>Ctrl-` c</td>
<td>Interrupt Process (IP)</td>
</tr>
<tr>
<td>Ctrl-` h</td>
<td>Erase Character (EC)</td>
</tr>
<tr>
<td>Ctrl-` o</td>
<td>Abort Output (AO)</td>
</tr>
<tr>
<td>Ctrl-` t</td>
<td>Are You There? (AYT)</td>
</tr>
<tr>
<td>Ctrl-` u</td>
<td>Erase Line (EL)</td>
</tr>
</tbody>
</table>

¹. The caret (\`) symbol refers to Shift-6 on your keyboard.

At any time during an active Telnet session, you can list the Telnet commands by pressing the escape sequence keys followed by a question mark at the system prompt:

`Ctrl-\` ?`

A sample of this list follows.
In this sample output, the first caret (^) symbol represents the Ctrl key, and the second caret represents Shift-6 on your keyboard:

```
Router> ^?  
[Special telnet escape help]
^^B  sends telnet BREAK
^^C  sends telnet IP
^^H  sends telnet EC
^^O  sends telnet AO
^^T  sends telnet AYT
^^U  sends telnet EL
```

You can have several concurrent Telnet sessions open and switch between them. To open a subsequent session, first suspend the current connection by pressing the escape sequence (Ctrl-Shift-6 then x [Ctrl^x] by default) to return to the system command prompt. Then open a new connection with the `telnet` command.

To terminate an active Telnet session, enter any of the following commands at the prompt of the device to which you are connecting:

- `close`
- `disconnect`
- `exit`
- `logout`
- `quit`

**Examples**

The following example establishes an encrypted Telnet session from a router to a remote host named `host1`:

```
Router> telnet host1 /encrypt kerberos
```

The following example routes packets from the source system `host1` to `kl.sri.com`, then to `10.1.0.11`, and finally back to `host1`:

```
Router> telnet host1 /route:kl.sri.com 10.1.0.11 host1
```

The following example connects to a host with logical name `host1`:

```
Router> host1
```

The following example suppresses all onscreen messages from the Cisco IOS software during login and logout:

```
Router> telnet host2 /quiet
```

The following example shows the limited number of messages displayed when connection is done using the optional /quiet keyword:

```text
login:User2
Password:
Welcome to OpenVMS VAX version V6.1 on node CRAW
Last interactive login on Tuesday, 15-DEC-1998 11:01
Last non-interactive login on Sunday, 3-JAN-1999 22:32
Server3)logout
User2 logged out at 16-FEB-2000 09:38:27.85
```
<table>
<thead>
<tr>
<th>Related Commands</th>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>connect</td>
<td>Logs in to a host that supports Telnet, rlogin, or LAT.</td>
</tr>
<tr>
<td></td>
<td>kerberos clients mandatory</td>
<td>Causes the rsh, rcp, rlogin, and telnet commands to fail if they cannot negotiate the Kerberos Protocol with the remote server.</td>
</tr>
<tr>
<td></td>
<td>rlogin</td>
<td>Logs in to a UNIX host using rlogin.</td>
</tr>
</tbody>
</table>
**telnet break-on-ip**

To cause the system to generate a hardware BREAK signal on the EIA/TIA-232 line that is associated with a reverse Telnet connection when a Telnet Interrupt-Process command is received on that connection, use the **telnet break-on-ip** command in line configuration mode.

**Syntax Description**

This command has no arguments or keywords.

**Defaults**

No hardware Break signal is generated when an Interrupt-Process command is received.

**Command Modes**

Line configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

This command causes the system to generate a hardware BREAK signal on the EIA/TIA-232 line that is associated with a reverse Telnet connection. It is useful when a Telnet Interrupt-Process command is received on that connection because it can control the translation of Telnet Interrupt-Process commands into X.25 BREAK indications. It is also a useful workaround in the following situations:

- Several user Telnet programs send an Interrupt-Process command, but cannot send a Telnet BREAK signal.
- Some Telnet programs implement a BREAK signal that sends an Interrupt-Process command.
- Some EIA/TIA-232 hardware devices use a hardware BREAK signal for various purposes.

A hardware BREAK signal is generated when a Telnet BREAK command is received.

**Examples**

In the following example, line 5 is configured with the **telnet break-on-ip** command. The location text notes that this line is the location of the high-speed modem. The **telnet transparent** command sets end-of-line handling.

```
line 5
  location high-speed modem
  telnet transparent
  telnet break-on-ip
```
### Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>connect</strong></td>
<td>Logs in to a host that supports Telnet, rlogin, or LAT.</td>
</tr>
<tr>
<td><strong>telnet</strong></td>
<td>Logs in to a host that supports Telnet.</td>
</tr>
<tr>
<td><strong>telnet transparent</strong></td>
<td>Configures the Cisco IOS software to send a CARRIAGE RETURN (CR) as a CR followed by a NULL instead of a CR followed by a LINE FEED (LF).</td>
</tr>
<tr>
<td><strong>terminal telnet break-on-ip</strong></td>
<td>Causes the access server to generate a hardware Break signal on the EIA/TIA-232 line, which is associated with a reverse Telnet connection, for the current line and sessions.</td>
</tr>
</tbody>
</table>
### telnet refuse-negotiations

To set a line using Telnet to refuse to negotiate full-duplex, remote echo requests on incoming connections, use the `telnet refuse-negotiations` command in line configuration mode. To disable this function, use the `no` form of this command.

```
telnet refuse-negotiations

no telnet refuse-negotiations
```

#### Syntax Description
This command has no arguments or keywords.

#### Defaults
No default behavior or values.

#### Command Modes
Line configuration

#### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

#### Usage Guidelines
Use this command on reverse Telnet connections to allow the Cisco IOS software to refuse full-duplex, remote echo option connection requests from the other end. This command suppresses negotiation of the Telnet Remote Echo and Suppress Go Ahead options.

This command does not apply to protocol translation configurations. It is intended for applications in which the router is functioning as a terminal server to allow terminal connections to remote devices through the asynchronous terminal ports of the router. Terminal server connections are those where the user types a command similar to the following to access network resources:

```
telnet access-server 2005
```

where `access-server` is the host name of the Cisco router functioning as a terminal server, and 2005 is the port number on the router to which the remote terminal is connected.

#### Examples
The following example shows how to set line 5 to refuse full-duplex, remote echo requests:

```
line 5
telnet refuse-negotiations
```

#### Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>connect</code></td>
<td>Logs in to a host that supports Telnet, rlogin, or LAT.</td>
</tr>
<tr>
<td><code>telnet</code></td>
<td>Logs in to a host that supports Telnet.</td>
</tr>
<tr>
<td><code>terminal telnet</code></td>
<td>Sets the current line to refuse to negotiate full-duplex, remote echo options on incoming connections for current sessions.</td>
</tr>
<tr>
<td><code>refuse-negotiations</code></td>
<td>Sets the current line to refuse to negotiate full-duplex, remote echo options on incoming connections for current sessions.</td>
</tr>
</tbody>
</table>
telnet speed

To allow negotiation of the transmission speed of the line to a connected device, use the `telnet speed` command in line configuration mode. To disable this function, use the `no` form of this command.

```
telnet speed default-speed maximum-speed
no telnet speed
```

**Syntax Description**

<table>
<thead>
<tr>
<th>default-speed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Line speed, in bits per second, that the Cisco IOS software will use if the device on the other end of the connection has not specified a speed.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>maximum-speed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximum speed, in bits per second, that the device on the port will use.</td>
</tr>
</tbody>
</table>

**Defaults**

No default behavior or values.

**Command Modes**

Line configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Negotiates speeds on reverse Telnet lines. You can match line speeds on remote systems in reverse Telnet, on host machines hooked up to a router used to access the network, or on a group of console lines hooked up to the router, when disparate line speeds are in use at the local and remote ends of the connection. Line speed negotiation adheres to the Remote Flow Control option defined in RFC 1080.

**Examples**

The following example allows a router to negotiate a bit rate on the line using the Telnet option. If no speed is negotiated, the line will run at 2400 bits per second (bps). If the remote host requests a speed of greater than 9600 bps, then 9600 will be used.

```
line 5
telnet speed 2400 9600
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>connect</td>
<td>Logs in to a host that supports Telnet, rlogin, or LAT.</td>
</tr>
<tr>
<td>telnet</td>
<td>Logs in to a host that supports Telnet.</td>
</tr>
<tr>
<td>terminal telnet speed</td>
<td>Allows the access server to negotiate transmission speed for the current line and session.</td>
</tr>
</tbody>
</table>
**telnet sync-on-break**

To configure the Cisco IOS software to cause an incoming connection to send a Telnet Synchronize signal when it receives a Telnet BREAK signal, use the `telnet sync-on-break` command in line configuration mode. To disable this function, use the `no` form of this command.

```
telnet sync-on-break

no telnet sync-on-break
```

This command has no arguments or keywords.

---

**Defaults**

No default behavior or values.

**Command Modes**

Line configuration

---

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

---

**Usage Guidelines**

This command causes a reverse Telnet line to send a Telnet Synchronize signal when it receives a Telnet BREAK signal. This option is used very rarely to ensure the ordering of BREAK reception with respect to data characters sent after the BREAK.

---

**Examples**

The following example configures the AUX line with the `telnet sync-on-break` command:

```
line aux 0
telnet sync-on-break
```

---

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>connect</strong></td>
<td>Logs in to a host that supports Telnet, rlogin, or LAT.</td>
</tr>
<tr>
<td><strong>telnet</strong></td>
<td>Logs in to a host that supports Telnet.</td>
</tr>
<tr>
<td><strong>terminal telnet</strong></td>
<td>Causes the access server to send a Telnet Synchronize signal when it receives a Telnet Break signal on the current line and session.</td>
</tr>
<tr>
<td><strong>sync-on-break</strong></td>
<td></td>
</tr>
</tbody>
</table>
**telnet transparent**

To configure the Cisco IOS software to send a CARRIAGE RETURN (CR) as a CR followed by a NULL instead of a CR followed by a LINE FEED (LF), use the `telnet transparent` command in line configuration mode. To return to the default setting, use the `no` form of this command.

```
telnet transparent
no telnet transparent
```

**Syntax Description**
This command has no arguments or keywords.

**Defaults**
CARRIAGE RETURN followed by a LINE FEED.

**Command Modes**
Line configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**
This command is useful for coping with different interpretations of end-of-line handling in the Telnet protocol specification.

**Examples**
The following example causes the Cisco IOS software, when sending a CR, to send a CR followed by a NULL character:

```
line 7
telnet transparent
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>connect</code></td>
<td>Logs in to a host that supports Telnet, rlogin, or LAT.</td>
</tr>
<tr>
<td><code>telnet</code></td>
<td>Logs in to a host that supports Telnet.</td>
</tr>
<tr>
<td><code>terminal telnet transparent</code></td>
<td>Causes the current terminal line to send a Return character (CR) as a CR followed by a NULL instead of a CR followed by a LINE FEED (LF) for the current session.</td>
</tr>
</tbody>
</table>
terminal lat out-group

To temporarily define the list of services to which you or another user can connect, use the `terminal lat out-group` command in EXEC mode.

```
terminal lat out-group group-number [start-end] [disabled | enabled]
```

**Syntax Description**
- **group-number**: Number of the group that has access to the system through the specified line. This number is identified by the system administrator. You also can specify a range of group numbers. Separate the beginning and end of the range with a hyphen.
- **[start-end]** (Optional) You can specify a range of group numbers for the `group-number` argument. Separate the beginning and end of the range with a hyphen.
- **disabled**: Incrementally removes specified groups from a list.
- **enabled**: Incrementally adds specified groups to a list.

**Command Modes**
- EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>11.2</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

To temporarily define the list of services to which you or another user can connect, you define the group code lists used for connections from specific lines. You limit the connection choices for an individual line by defining the group code lists for an outgoing connection. When a user initiates a connection with a local-area transport (LAT) host, the line must share a common group number with the remote LAT host before a connection can be made.

The group code range entered in this command must fall within the group code range already configured for the line.

**Examples**

The following example defines a group code list for the outgoing group 4:

```
terminal lat out-group 4, 6-189
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>connect</code></td>
<td>Logs in to a host that supports Telnet, rlogin, or LAT.</td>
</tr>
<tr>
<td><code>l2f ignore-mid-sequence</code></td>
<td>Specifies a connection to a particular LAT node that offers LAT services.</td>
</tr>
</tbody>
</table>
terminal lat remote-modification

To set a line running local-area transport (LAT) to be remotely modifiable, use the `terminal lat remote-modification` command in EXEC mode.

```
terminal lat remote-modification
```

**Syntax Description**
This command has no arguments or keywords.

**Command Modes**
EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>11.2</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Examples**
The following example sets line 6 to be remotely modifiable:

```
terminal lat remote-modification 6
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>connect</code></td>
<td>Logs in to a host that supports Telnet, rlogin, or LAT.</td>
</tr>
<tr>
<td><code>l2f ignore-mid-sequence</code></td>
<td>Specifies a connection to a particular LAT node that offers LAT services.</td>
</tr>
</tbody>
</table>
**terminal transport preferred**

To specify the preferred protocol to use for the current session when a command does not specify one, use the `terminal transport preferred` command in EXEC mode.

```plaintext
terminal transport preferred { all | lat | mop | nasi | none | pad | rlogin | telnet | v120 }
```

<table>
<thead>
<tr>
<th>Syntax Description</th>
<th>all</th>
<th>Specifies all recognized protocols.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>lat</td>
<td>Specifies the local-area transport (LAT) protocol.</td>
</tr>
<tr>
<td></td>
<td>mop</td>
<td>Specifies the Maintenance Operation Protocol (MOP).</td>
</tr>
<tr>
<td></td>
<td>nasi</td>
<td>Specifies the NetWare Asynchronous Services Interface (NASI) protocol.</td>
</tr>
<tr>
<td></td>
<td>none</td>
<td>Prevents any protocol selection on the line. The router default is that any unrecognized command is a host name. If the preferred protocol is set to none, the router will not attempt any connections if the command is not recognized.</td>
</tr>
<tr>
<td></td>
<td>pad</td>
<td>Specifies X.3 packet assembler/disassembler (PAD), which is used most often to connect a server product to X.25 hosts.</td>
</tr>
<tr>
<td></td>
<td>rlogin</td>
<td>Specifies UNIX rlogin.</td>
</tr>
<tr>
<td></td>
<td>telnet</td>
<td>Specifies the TCP/IP Telnet protocol.</td>
</tr>
<tr>
<td></td>
<td>v120</td>
<td>Selects the V.120 protocol for incoming asynchronous connections over ISDN.</td>
</tr>
</tbody>
</table>

**Defaults**

`lat` (if LAT is not supported, `telnet`)

**Command Modes**

EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command first appeared in a release prior to Cisco IOS Release 10.0.</td>
</tr>
<tr>
<td>11.2</td>
<td>The following keywords were added:</td>
</tr>
<tr>
<td></td>
<td>• all</td>
</tr>
<tr>
<td></td>
<td>• lat</td>
</tr>
<tr>
<td></td>
<td>• mop</td>
</tr>
<tr>
<td></td>
<td>• nasi</td>
</tr>
<tr>
<td></td>
<td>• pad</td>
</tr>
<tr>
<td></td>
<td>• pad</td>
</tr>
<tr>
<td></td>
<td>• preferred</td>
</tr>
<tr>
<td></td>
<td>• rlogin</td>
</tr>
<tr>
<td></td>
<td>• v120</td>
</tr>
</tbody>
</table>
Examples

The following example configures the console so that it does not connect when an unrecognized command is entered:

```
terminal transport preferred none
```

Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>transport preferred</td>
<td>Specifies the transport protocol that the Cisco IOS software uses if the user does not specify one when initiating a connection.</td>
</tr>
</tbody>
</table>
To begin a TN3270 session, use the `tn3270` command in EXEC mode.

```
  tn3270 host
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>host</code></td>
<td>Name or IP address of a specific host on a network that can be reached by the router. The default terminal emulation mode allows access using a VT100 emulation.</td>
</tr>
</tbody>
</table>

**Command Modes**

EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>11.1</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Unlike Telnet and local-area transport (LAT) connections, you must enter the `tn3270` command to make a connection to an IBM TN3278 host.

To terminate an active TN3270 session, enter the escape sequence (Ctrl-Shift-6 then x [Ctrl^x] by default) and enter the `disconnect` command at the EXEC prompt. Or log off the remote system by issuing the command specific to that system (such as `exit`, `logout`, `quit`, `close`, or `disconnect`).

**Examples**

The following example establishes a terminal session with an IBM TN3270 host named finance:

```
  tn3270 finance
```
tn3270 8bit display

To configure the Cisco IOS software to use the mask set by the data-character-bits \{7 | 8\} command in line configuration mode or the terminal data-character-bits \{7 | 8\} EXEC command, use the tn3270 8bit display command in line configuration mode. To restore the default 7-bit mask used for TN3270 connections, use the no form of this command.

`tn3270 8bit display`

`no tn3270 8bit display`

**Syntax Description**
This command has no arguments or keywords.

**Defaults**
Disabled

**Command Modes**
Line configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**
Use the tn3270-character-map command to map between extended EBCDIC or extended ASCII characters.

**Examples**
The following example configures the Cisco IOS software to use the mask set by the data-character-bits line configuration and EXEC commands on line 5:

```
line 5
  tn3270 8bit display
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>data-character-bits</td>
<td>Sets the number of data bits per character that are interpreted and generated by the Cisco IOS software.</td>
</tr>
<tr>
<td>terminal</td>
<td>Sets the number of data bits per character that are interpreted and generated by the Cisco IOS software for the current line and session.</td>
</tr>
</tbody>
</table>
tn3270 8bit transparent-mode

To configure the Cisco IOS software to use the mask set by the `data-character-bits {7 | 8}` command in line configuration mode or the `terminal data-character bits {7 | 8}` EXEC command, use the `tn3270 8bit transparent-mode` command in line configuration mode. To restore the default 7-bit mask used for TN3270 connections, use the `no` form of this command.

```
  tn3270 8bit transparent-mode
  no tn3270 8bit transparent-mode
```

**Syntax Description**

This command has no arguments or keywords.

**Defaults**

Disabled

**Command Modes**

Line configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

This command is needed if you are using a file transfer protocol such as Kermit in 8-bit mode or you are using 8-bit graphics, both of which rely on transparent mode.

**Examples**

The following example configures the software to use the mask set by the `data-character-bits` line configuration and EXEC commands on line 5:

```
line 5
  tn3270 8bit transparent-mode
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>data-character-bits</strong></td>
<td>Sets the number of data bits per character that are interpreted and generated by the Cisco IOS software.</td>
</tr>
<tr>
<td><strong>terminal data-character-bits</strong></td>
<td>Sets the number of data bits per character that are interpreted and generated by the Cisco IOS software for the current line and session.</td>
</tr>
</tbody>
</table>
To convert incoming EBCDIC characters into ASCII characters, use the `tn3270 character-map` command in global configuration mode. To restore default character mappings, use the `no` form of this command.

```
no tn3270 character-map { all | ebcdic-in-hex | ascii-in-hex }
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ebcdic-in-hex</td>
<td>Hexadecimal value of an EBCDIC character.</td>
</tr>
<tr>
<td>ascii-in-hex</td>
<td>Hexadecimal value of an ASCII character.</td>
</tr>
<tr>
<td>all</td>
<td>Indicates all character mappings.</td>
</tr>
</tbody>
</table>

**Defaults**

Disabled

**Command Modes**

Global configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use this command to print international characters that are EBCDIC characters not normally printed, including umlauts (¨) and tildes (~). The command first restores default mapping for both EBCDIC and ASCII characters. In the `no` form of the command, the `all` keyword resets all character mappings to Cisco defaults.

Table 36 shows the default character mappings between ASCII and EBCDIC in decimal and hexadecimal format.

To convert outgoing ASCII characters into EBCDIC characters, use the `keymap` command to modify the keymap structure with the tag `ebcdic_xx=string`, where `xx` is a hexadecimal value and `string` is the sequence of characters that send the EBCDIC character.

**Table 36  Default ASCII, EBCDIC Character Mappings**

<table>
<thead>
<tr>
<th>Character</th>
<th>ASCII Decimal</th>
<th>ASCII Hexadecimal</th>
<th>EBCDIC Decimal</th>
<th>EBCDIC Hexadecimal</th>
</tr>
</thead>
<tbody>
<tr>
<td>!</td>
<td>33</td>
<td>0x21</td>
<td>90</td>
<td>0x5a</td>
</tr>
<tr>
<td>&quot;</td>
<td>34</td>
<td>0x22</td>
<td>127</td>
<td>0x7f</td>
</tr>
<tr>
<td>#</td>
<td>35</td>
<td>0x23</td>
<td>123</td>
<td>0x7b</td>
</tr>
<tr>
<td>$</td>
<td>36</td>
<td>0x24</td>
<td>91</td>
<td>0x5b</td>
</tr>
<tr>
<td>%</td>
<td>37</td>
<td>0x25</td>
<td>108</td>
<td>0x6c</td>
</tr>
<tr>
<td>&amp;</td>
<td>38</td>
<td>0x26</td>
<td>80</td>
<td>0x50</td>
</tr>
</tbody>
</table>
### Table 36  Default ASCII, EBCDIC Character Mappings (continued)

<table>
<thead>
<tr>
<th>Character</th>
<th>ASCII Decimal</th>
<th>ASCII Hexadecimal</th>
<th>EBCDIC Decimal</th>
<th>EBCDIC Hexadecimal</th>
</tr>
</thead>
<tbody>
<tr>
<td>'</td>
<td>39</td>
<td>0x27</td>
<td>125</td>
<td>0x7d</td>
</tr>
<tr>
<td>(</td>
<td>40</td>
<td>0x28</td>
<td>77</td>
<td>0x4d</td>
</tr>
<tr>
<td>)</td>
<td>41</td>
<td>0x29</td>
<td>93</td>
<td>0x5d</td>
</tr>
<tr>
<td>*</td>
<td>42</td>
<td>0x2a</td>
<td>92</td>
<td>0x5c</td>
</tr>
<tr>
<td>+</td>
<td>43</td>
<td>0x2b</td>
<td>78</td>
<td>0x4e</td>
</tr>
<tr>
<td>.</td>
<td>44</td>
<td>0x2c</td>
<td>107</td>
<td>0x6b</td>
</tr>
<tr>
<td>-</td>
<td>45</td>
<td>0x2d</td>
<td>96</td>
<td>0x60</td>
</tr>
<tr>
<td>.</td>
<td>46</td>
<td>0x2e</td>
<td>75</td>
<td>0x4b</td>
</tr>
<tr>
<td>/</td>
<td>47</td>
<td>0x2f</td>
<td>97</td>
<td>0x61</td>
</tr>
<tr>
<td>0</td>
<td>48</td>
<td>0x30</td>
<td>240</td>
<td>0xf0</td>
</tr>
<tr>
<td>1</td>
<td>49</td>
<td>0x31</td>
<td>241</td>
<td>0xf1</td>
</tr>
<tr>
<td>2</td>
<td>50</td>
<td>0x32</td>
<td>242</td>
<td>0xf2</td>
</tr>
<tr>
<td>3</td>
<td>51</td>
<td>0x33</td>
<td>243</td>
<td>0xf3</td>
</tr>
<tr>
<td>4</td>
<td>52</td>
<td>0x34</td>
<td>244</td>
<td>0xf4</td>
</tr>
<tr>
<td>5</td>
<td>53</td>
<td>0x35</td>
<td>245</td>
<td>0xf5</td>
</tr>
<tr>
<td>6</td>
<td>54</td>
<td>0x36</td>
<td>246</td>
<td>0xf6</td>
</tr>
<tr>
<td>7</td>
<td>55</td>
<td>0x37</td>
<td>247</td>
<td>0xf7</td>
</tr>
<tr>
<td>8</td>
<td>56</td>
<td>0x38</td>
<td>248</td>
<td>0xf8</td>
</tr>
<tr>
<td>9</td>
<td>57</td>
<td>0x39</td>
<td>249</td>
<td>0xf9</td>
</tr>
<tr>
<td>:</td>
<td>58</td>
<td>0x3a</td>
<td>122</td>
<td>0x7a</td>
</tr>
<tr>
<td>;</td>
<td>59</td>
<td>0x3b</td>
<td>94</td>
<td>0x5e</td>
</tr>
<tr>
<td>&lt;</td>
<td>60</td>
<td>0x3c</td>
<td>76</td>
<td>0x4c</td>
</tr>
<tr>
<td>=</td>
<td>61</td>
<td>0x3d</td>
<td>126</td>
<td>0x7e</td>
</tr>
<tr>
<td>&gt;</td>
<td>62</td>
<td>0x3e</td>
<td>110</td>
<td>0x6e</td>
</tr>
<tr>
<td>?</td>
<td>63</td>
<td>0x3f</td>
<td>111</td>
<td>0x6f</td>
</tr>
<tr>
<td>@</td>
<td>64</td>
<td>0x40</td>
<td>124</td>
<td>0x7c</td>
</tr>
<tr>
<td>A</td>
<td>65</td>
<td>0x41</td>
<td>193</td>
<td>0xc1</td>
</tr>
<tr>
<td>B</td>
<td>66</td>
<td>0x42</td>
<td>194</td>
<td>0xc2</td>
</tr>
<tr>
<td>C</td>
<td>67</td>
<td>0x43</td>
<td>195</td>
<td>0xc3</td>
</tr>
<tr>
<td>D</td>
<td>68</td>
<td>0x44</td>
<td>196</td>
<td>0xc4</td>
</tr>
<tr>
<td>E</td>
<td>69</td>
<td>0x45</td>
<td>197</td>
<td>0xc5</td>
</tr>
<tr>
<td>F</td>
<td>70</td>
<td>0x46</td>
<td>198</td>
<td>0xc6</td>
</tr>
<tr>
<td>G</td>
<td>71</td>
<td>0x47</td>
<td>199</td>
<td>0xc7</td>
</tr>
<tr>
<td>H</td>
<td>72</td>
<td>0x48</td>
<td>200</td>
<td>0xc8</td>
</tr>
<tr>
<td>I</td>
<td>73</td>
<td>0x49</td>
<td>201</td>
<td>0xc9</td>
</tr>
<tr>
<td>J</td>
<td>74</td>
<td>0x4a</td>
<td>209</td>
<td>0xd1</td>
</tr>
</tbody>
</table>
### Table 36  Default ASCII, EBCDIC Character Mappings (continued)

<table>
<thead>
<tr>
<th>Character</th>
<th>ASCII Decimal</th>
<th>ASCII Hexadecimal</th>
<th>EBCDIC Decimal</th>
<th>EBCDIC Hexadecimal</th>
</tr>
</thead>
<tbody>
<tr>
<td>K</td>
<td>75</td>
<td>0x4b</td>
<td>210</td>
<td>0xd2</td>
</tr>
<tr>
<td>L</td>
<td>76</td>
<td>0x4c</td>
<td>211</td>
<td>0xd3</td>
</tr>
<tr>
<td>M</td>
<td>77</td>
<td>0x4d</td>
<td>212</td>
<td>0xd4</td>
</tr>
<tr>
<td>N</td>
<td>78</td>
<td>0x4e</td>
<td>213</td>
<td>0xd5</td>
</tr>
<tr>
<td>O</td>
<td>79</td>
<td>0x4f</td>
<td>214</td>
<td>0xd6</td>
</tr>
<tr>
<td>P</td>
<td>80</td>
<td>0x50</td>
<td>215</td>
<td>0xd7</td>
</tr>
<tr>
<td>Q</td>
<td>81</td>
<td>0x51</td>
<td>216</td>
<td>0xd8</td>
</tr>
<tr>
<td>R</td>
<td>82</td>
<td>0x52</td>
<td>217</td>
<td>0xd9</td>
</tr>
<tr>
<td>S</td>
<td>83</td>
<td>0x53</td>
<td>226</td>
<td>0xe2</td>
</tr>
<tr>
<td>T</td>
<td>84</td>
<td>0x54</td>
<td>227</td>
<td>0xe3</td>
</tr>
<tr>
<td>U</td>
<td>85</td>
<td>0x55</td>
<td>228</td>
<td>0xe4</td>
</tr>
<tr>
<td>V</td>
<td>86</td>
<td>0x56</td>
<td>229</td>
<td>0xe5</td>
</tr>
<tr>
<td>W</td>
<td>87</td>
<td>0x57</td>
<td>230</td>
<td>0xe6</td>
</tr>
<tr>
<td>X</td>
<td>88</td>
<td>0x58</td>
<td>231</td>
<td>0xe7</td>
</tr>
<tr>
<td>Y</td>
<td>89</td>
<td>0x59</td>
<td>232</td>
<td>0xe8</td>
</tr>
<tr>
<td>Z</td>
<td>90</td>
<td>0x5a</td>
<td>233</td>
<td>0xe9</td>
</tr>
<tr>
<td>[</td>
<td>91</td>
<td>0x5b</td>
<td>173</td>
<td>0xad</td>
</tr>
<tr>
<td>\</td>
<td>92</td>
<td>0x5c</td>
<td>224</td>
<td>0xe0</td>
</tr>
<tr>
<td>]</td>
<td>93</td>
<td>0x5d</td>
<td>189</td>
<td>0xbd</td>
</tr>
<tr>
<td>^</td>
<td>94</td>
<td>0x5e</td>
<td>95</td>
<td>0x5f</td>
</tr>
<tr>
<td>_</td>
<td>95</td>
<td>0x5f</td>
<td>109</td>
<td>0x6d</td>
</tr>
<tr>
<td>`</td>
<td>96</td>
<td>0x60</td>
<td>121</td>
<td>0x79</td>
</tr>
<tr>
<td>a</td>
<td>97</td>
<td>0x61</td>
<td>129</td>
<td>0x81</td>
</tr>
<tr>
<td>b</td>
<td>98</td>
<td>0x62</td>
<td>130</td>
<td>0x82</td>
</tr>
<tr>
<td>c</td>
<td>99</td>
<td>0x63</td>
<td>131</td>
<td>0x83</td>
</tr>
<tr>
<td>d</td>
<td>100</td>
<td>0x64</td>
<td>132</td>
<td>0x84</td>
</tr>
<tr>
<td>e</td>
<td>101</td>
<td>0x65</td>
<td>133</td>
<td>0x85</td>
</tr>
<tr>
<td>f</td>
<td>102</td>
<td>0x66</td>
<td>134</td>
<td>0x86</td>
</tr>
<tr>
<td>g</td>
<td>103</td>
<td>0x67</td>
<td>135</td>
<td>0x87</td>
</tr>
<tr>
<td>h</td>
<td>104</td>
<td>0x68</td>
<td>136</td>
<td>0x88</td>
</tr>
<tr>
<td>i</td>
<td>105</td>
<td>0x69</td>
<td>137</td>
<td>0x89</td>
</tr>
<tr>
<td>j</td>
<td>106</td>
<td>0x6a</td>
<td>145</td>
<td>0x91</td>
</tr>
<tr>
<td>k</td>
<td>107</td>
<td>0x6b</td>
<td>146</td>
<td>0x92</td>
</tr>
<tr>
<td>l</td>
<td>108</td>
<td>0x6c</td>
<td>147</td>
<td>0x93</td>
</tr>
<tr>
<td>m</td>
<td>109</td>
<td>0x6d</td>
<td>148</td>
<td>0x94</td>
</tr>
<tr>
<td>n</td>
<td>110</td>
<td>0x6e</td>
<td>149</td>
<td>0x95</td>
</tr>
</tbody>
</table>
### Examples

The following example creates a two-way binding between an EBCDIC character and an ASCII character:

```plaintext
tn3270 character-map 0x81 0x78
```

### Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>show tn3270 asci-hexval</code></td>
<td>Displays ASCII-hexadecimal character mappings.</td>
</tr>
<tr>
<td><code>show tn3270 character-map</code></td>
<td>Displays character mappings between ASCII and EBCDIC.</td>
</tr>
</tbody>
</table>
### tn3270 datastream

To enable the TN3270 extended datastream, use the `tn3270 datastream` command in global configuration mode. To return to the normal TN3270 datastream, use the `no` form of this command.

```
  tn3270 datastream {extended | normal}
  no tn3270 datastream
```

#### Syntax Description

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>extended</code></td>
<td>Extended datastream.</td>
</tr>
<tr>
<td><code>normal</code></td>
<td>Normal datastream.</td>
</tr>
</tbody>
</table>

#### Defaults

Normal datastream

#### Command Modes

Global configuration

#### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.3</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

#### Usage Guidelines

This command causes an “-E” to be appended to the terminal type string sent to the IBM host, which allows you to use the extended TN3270 features.

#### Examples

The following example shows the supported TN3270 datastream options:

```
  tn3270 datastream ?
  extended  Use extended TN3270 datastream
  normal    Use normal TN3270 datastream
```
**tn3270 null-processing**

To specify how NULL signals are handled, use the `tn3270 null-processing` command in global configuration mode. To return to 7171 NULL processing, use the `no` form of this command.

```
 tn3270 null-processing [3270 | 7171]
 no tn3270 null-processing [3270 | 7171]
```

### Syntax Description

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>3270</td>
<td>(Optional) NULLs are compressed out of the string, as on a 3278-x terminal.</td>
</tr>
<tr>
<td>7171</td>
<td>(Optional) NULLs are converted to spaces, as on a 7171 controller.</td>
</tr>
</tbody>
</table>

### Defaults

7171 NULL processing

### Command Modes

Global configuration

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.3</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

### Usage Guidelines

If a user enters data, uses an arrow key to move the cursor to the right on the screen, and then enters more data, the intervening spaces are filled with NULLs. To specify how NULLs are handled, enter the `tn3270 null-processing` command either with the `3270` argument, where NULLs are compressed out of the string (as on a real 3278-x terminal) or the `7171` argument, where NULLs are converted to spaces as on a 7171 controller. Enter this command in global configuration.

### Examples

The following example shows the two available null processing methods:

```
 tn3270 null-processing ?
 3270  Use 3270-style null processing
 7171  Use 7171-style null processing
```
**tn3270 optimize-cursor-move**

To increase performance between a remote user and a TN3270 host by limiting cursor movement information that is sent to user terminals, use the `tn3270 optimize-cursor-move` command in global configuration mode. To ensure that all cursor movement information is sent between the terminal and the TN3270 host, use the `no` form of this command.

```
  tn3270 optimize-cursor-move

  no tn3270 optimize-cursor-move
```

**Syntax Description**
This command has no arguments or keywords.

**Defaults**
Cursor movement escape strings are sent to the terminal.

**Command Modes**
Global configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>11.2</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**
Issuing this command increases the speed of information transfer between users and TN3270 hosts through an access server.

If you do not issue this command, virtually every byte of information between the terminal and the TN3270 host is prepended and trailed by cursor-movement strings.

**Examples**
The following example disables status messages to users connected to 3278 terminals:

```
  tn3270 optimize-cursor-move
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>tn3270 status-message</code></td>
<td>Reenables the display of status messages after they have been disabled.</td>
</tr>
</tbody>
</table>
tn3270 reset-required

To lock a terminal after input error until the user resets the terminal, use the `tn3270 reset-required` command in global configuration mode. To return to the default of no reset required, use the `no` form of this command.

```
  tn3270 reset-required

  no tn3270 reset-required
```

**Syntax Description**: This command has no arguments or keywords.

**Defaults**: No reset is required.

**Command Modes**: Global configuration

**Command History**: On a 3278-x terminal, the keyboard is locked and further input is not permitted after input error (due to field overflow, invalid entry, and so on) until the user presses the RESET key. Most TN3270 implementations leave the keyboard unlocked and remove any error message on the next key input after the error. Use this command to lock the keyboard until the user performs a reset.
tn3270 status-message

To reenable the display of status messages after they have been disabled, use the **tn3270 status-message** command in global configuration mode. To save bandwidth on asynchronous lines by not displaying status messages, use the **no** form of this command.

```bash
tn3270 status-message

no tn3270 status-message
```

**Syntax Description**
This command has no arguments or keywords.

**Defaults**
Status messages appear.

**Command Modes**
Global configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>11.2</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**
Status messages appear on the console by default. These messages include “System Locked,” “Field error,” and “System UnLocked” messages. These messages are sent back to the terminal via the TTY line on the access server.

Disabling status messages saves bandwidth on asynchronous lines, which have very low bandwidth.

**Examples**
The following example disables status messages to users connected to 3270 terminals:

```bash
no tn3270 status-message
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>tn3270</strong></td>
<td>Increases performance between a remote user and a TN3270 host by limiting cursor movement information that is sent to user terminals.</td>
</tr>
<tr>
<td><strong>optimize-cursor-move</strong></td>
<td></td>
</tr>
</tbody>
</table>


tn3270 typeahead

To buffer keyboard data when a 3278 server is in locked mode, use the tn3270 typeahead command in global configuration mode. To disable the typeahead function, use the no form of this command.

```
   tn3270 typeahead
   no tn3270 typeahead
```

Syntax Description

This command has no arguments or keywords.

Defaults

No typeahead

Command Modes

Global configuration

Command History

Release       Modification
11.2           This command was introduced.

Usage Guidelines

When typeahead is enabled, the TN3270 client implementation in the Cisco IOS software permits you to continue typing while the system is trying to obtain a response from the TN3270 server. Information you type while a “System Locked” message appears on the terminal is stored in a buffer. After the “System Locked” message disappears, the information is then used as though it were just typed.

Examples

The following example saves user information when “System Locked” messages appear on the screen:

```
   tn3270 typeahead
```

Related Commands

```
Command                  Description
------------------------------------------------
  tn3270 reset-required   Locks a terminal after input error until the user resets the terminal.
```
translate lat

To translate a connection request to another protocol connection type when receiving a local-area transport (LAT) request, use the `translate lat` command in global configuration mode. To remove or change the translation request, use the `no` form of this command.

```
translate lat incoming-service-name [incoming-options] protocol outgoing-address
    [outgoing-options] [global-options]

no translate lat incoming-service-name [incoming-options] protocol outgoing-address
    [outgoing-options] [global-options]
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>incoming-service-name</code></td>
<td>A LAT service name. When used on the incoming portion of the command, <code>incoming-service-name</code> is the name of the service that users specify when trying to make a translated connection. This name can match the name of the final destination resource, but is not required to. This argument is useful when making remote translated connections.</td>
</tr>
<tr>
<td><code>incoming-options</code></td>
<td>(Optional) An incoming connection request option. For LAT, the only option currently supported is:</td>
</tr>
<tr>
<td></td>
<td>• <code>unadvertised</code>—Prevents service advertisements from being broadcast to the network. This keyword can be useful, for example, when you define translations for many printers, and you do not want these services advertised to other LAT terminal servers. (VMS systems will be able to connect to the service even though it is not advertised.)</td>
</tr>
<tr>
<td><code>protocol outgoing-address</code></td>
<td>A protocol name followed by an address or host name. Protocol translation choices are: <code>ppp</code>, <code>slip</code>, <code>tcp</code>, and <code>x25</code>.</td>
</tr>
<tr>
<td><code>global-options</code></td>
<td></td>
</tr>
</tbody>
</table>

**Note**

The host name is resolved to an address during configuration, unless you are translating to TCP and use the `host-name` keyword, which allows the host name to be resolved at connection time instead of configuration time. See Table 39 for more information about the `host-name` keyword.

Additional keywords that can be entered with the protocol are as follows:

- `autocommand`—Specifies an EXEC command for an outgoing connection. The command executes upon connection to a host. You can issue any EXEC command and any switch or host name as an argument to the `autocommand` command. If the string following `autocommand` has one or more spaces as part of the string, you must place quotation marks (" ") around the string.
If you want to enable AppleTalk Remote Access (ARA) on an outgoing connection, specify the **autocommand arap** keywords. These keywords are necessary for ARA because ARA does not use addressing, and this option permits you to invoke the ARA string.

- **virtual-template**—Associates a virtual template with a virtual access interface. See the *translate lat (virtual access interfaces)* command description for more information.

### outgoing-options

(Optional) Outgoing connection request options. Choices depend upon the protocol or command entered. See Table 37, Table 38, Table 39, and Table 40 for more information.

### global-options

(Optional) One or more of the following translation options can be used by any connection type:

- **access-class number**—Allows the incoming call to be used by source hosts that match the access list parameters. The argument *number* is an integer previously assigned to an access list. Standard access list numbers are in the range from 1 to 99; expanded standard access lists numbers are in the range 1300 to 1999.
- **local**—Allows Telnet protocol negotiations to *not* be translated.
- **login**—Requires that the user log in before the outgoing connection is made. This type of login is specified on the virtual terminal lines with the *login* command.
- **max-users number**—Limits the number of simultaneous users of the translation to *number* (an integer you specify).
- **quiet**— Suppresses printing of user-information messages.

### Defaults

No default translation parameters

### Command Modes

Global configuration

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>12.1</td>
<td>The <em>no-reset</em> permanent virtual circuits (PVC) subkeyword was added to support outgoing PVCs.</td>
</tr>
</tbody>
</table>

### Usage Guidelines

You define protocol translation connections by supplying a protocol keyword and the address, host name, or service name. A LAT protocol translation command can be as simple as the following example:

```
Router(config)# translate lat LAT-1 X.25 1236672
```

However, the Cisco IOS software provides a broad range of options that support protocol translations in many networking environments. Table 37, Table 38, Table 39, and Table 40 list the *translate lat* translation options by protocol.
You can also use the Cisco IOS command-line interface to help you understand how these keywords are entered. In global configuration mode, begin entering the `translate` command and add a question mark at each portion of the command to display the options available. Some examples follow:

```plaintext
Router(config)# translate lat ?
WORD  LAT service name

Router(config)# translate lat LSVC ?

- autocmd: Associate a command with a translation on this connection
- lat: DEC LAT protocol
- ppp: Virtual async PPP
- slip: Virtual async SLIP
- tcp: TCP/IP Telnet
- unadvertised: Prevent service advertisements from being broadcast to the network
- virtual-template: Associate a virtual template with virtual access interface
- x25: X.25

Router(config)# translate lat LSVC tcp ?

- Hostname or A.B.C.D  IP address

Router(config)# translate lat LSVC tcp 1.1.1.1 ?

- access-class: Allow access list parameters to be used by source hosts
- binary: Negotiate Telnet binary mode on the connection
- host-name: Store the host name rather than its IP address
- local: Allow Telnet protocol negotiations not to be translated
- login: Require that the user log in before the outgoing connection is made
- max-users: Limit the number of simultaneous users of the translation
- multibyte-IAC: Always treat multiple IACs as telnet command
- port: Port Number
- quiet: Suppress printing of user-information messages
- source-interface: Specify source interface
- stream: Treat telnet escape characters as data
```

*Note*

If you plan to translate to X.25 on a permanent virtual circuit (PVC), see the description for the `translate x25` command for important configuration notes.

**Table 37 LAT-to-PPP Outgoing Translation Options**

**Outgoing PPP Translation**

```plaintext
ppp \{ip-address \ ip-pool [scope-name name]\}
```

Translates from LAT to virtual asynchronous PPP. Supply an IP address as a standard, four-part dotted decimal IP address.

The `ip-pool` keyword obtains an IP address from a Dynamic Host Configuration Protocol (DHCP) proxy client or a local pool. If the optional `scope-name` keyword is not specified, the address is obtained from a DHCP proxy client. If the `scope-name` keyword is specified, the IP address is obtained from the specified local pool. The `scope-name` keyword can specify a range of IP addresses.
Outgoing PPP Connection Request Options
Add any of the following keywords to configure PPP connection requests:

- **authentication** [pap | chap]—Sets Challenge Handshake Authentication Protocol (CHAP) or Password Authentication Protocol (PAP) authentication for PPP on virtual asynchronous interfaces. If you specify both keywords, order is significant; the system will try to use the first authentication type, then the second.

- **header-compression**—Implements header compression on IP packets only.

- **ipx loopback number**—Specifies the loopback interface to be created and permits clients running IPX-PPP to connect through virtual terminal lines on the router. A loopback interface must have been created and configured with a Novell IPX network number before IPX-PPP can work on the virtual terminal line. The virtual terminal line is assigned to the loopback interface.

- **keepalive number-of-seconds**—Specifies the interval at which keepalive packets are sent on Serial Line Internet Protocol (SLIP) and PPP virtual asynchronous interfaces. By default, keepalive packets are enabled and sent every 10 seconds. To shut off keepalive packets, use a value of 0. The active keepalive interval is 1 through 32,767 seconds. When you do not change from the default of 10, the keepalive interval does not appear in `more system:running-config` or `show translate` command output.

- **mtu bytes**—Sets the interface maximum transmission unit (MTU) of packets that the virtual asynchronous interface supports. The default MTU is 1500 bytes on a virtual asynchronous interface. The acceptable range is from 64 to 1,000,000 bytes.

- **routing**—Permits routing updates between connections. This keyword is required if the destination device is not on a subnet connected to one of the interfaces on the router.

- **use-tacacs**—Uses TACACS to verify PPP authentications for CHAP or PAP on virtual asynchronous interfaces.

### Table 38 LAT-to-PPP Outgoing Translation Options (continued)

#### Table 37 LAT-to-PPP Outgoing Translation Options (continued)

#### Outgoing PPP Connection Request Options
Add any of the following keywords to configure PPP connection requests:

- **authentication** [pap | chap]—Sets Challenge Handshake Authentication Protocol (CHAP) or Password Authentication Protocol (PAP) authentication for PPP on virtual asynchronous interfaces. If you specify both keywords, order is significant; the system will try to use the first authentication type, then the second.

- **header-compression**—Implements header compression on IP packets only.

- **ipx loopback number**—Specifies the loopback interface to be created and permits clients running IPX-PPP to connect through virtual terminal lines on the router. A loopback interface must have been created and configured with a Novell IPX network number before IPX-PPP can work on the virtual terminal line. The virtual terminal line is assigned to the loopback interface.

- **keepalive number-of-seconds**—Specifies the interval at which keepalive packets are sent on Serial Line Internet Protocol (SLIP) and PPP virtual asynchronous interfaces. By default, keepalive packets are enabled and sent every 10 seconds. To shut off keepalive packets, use a value of 0. The active keepalive interval is 1 through 32,767 seconds. When you do not change from the default of 10, the keepalive interval does not appear in `more system:running-config` or `show translate` command output.

- **mtu bytes**—Sets the interface maximum transmission unit (MTU) of packets that the virtual asynchronous interface supports. The default MTU is 1500 bytes on a virtual asynchronous interface. The acceptable range is from 64 to 1,000,000 bytes.

- **routing**—Permits routing updates between connections. This keyword is required if the destination device is not on a subnet connected to one of the interfaces on the router.

- **use-tacacs**—Uses TACACS to verify PPP authentications for CHAP or PAP on virtual asynchronous interfaces.

#### Table 38 LAT-to-SLIP Outgoing Translation Options

#### Outgoing SLIP Translation

```
slip {ip-address|ip-pool [scope-name name]}
```

Translates from LAT to virtual asynchronous SLIP. Supply an IP address as a standard, four-part dotted decimal IP address.

The **ip-pool** keyword obtains an IP address from a DHCP proxy client or a local pool. If the optional **scope-name** keyword is not specified, the address is obtained from a DHCP proxy client. If the **scope-name** keyword is specified, the IP address is obtained from the specified local pool. The **scope-name** keyword can specify a range of IP addresses.

**Note** The **slip** argument applies only to outgoing connections; SLIP is not supported on incoming protocol translation connections.
Outgoing SLIP Connection Request Options
Add any of the following keywords to configure SLIP connection requests:

- **header-compression [passive]**—Implements header compression on IP packets only. The passive keyword permits compression on outgoing packets only if incoming TCP packets on the same virtual asynchronous interface are compressed. The default (without the passive keyword) permits compression on all traffic.

- **ipx loopback number**—Specifies the loopback interface to be created and permits clients running IPX-PPP to connect through virtual terminal lines on the router. A loopback interface must have been created and configured with a Novell IPX network number before IPX-PPP can work on the virtual terminal line. The virtual terminal line is assigned to the loopback interface.

- **keepalive number-of-seconds**—Specifies the interval at which keepalive packets are sent on SLIP and PPP virtual asynchronous interfaces. By default, keepalive packets are enabled and sent every 10 seconds. To shut off keepalive packets, use a value of 0. The active keepalive interval is 1 through 32,767 seconds. When you do not change from the default of 10, the keepalive interval does not appear in `more system:running-config` or `show translate` command output.

- **mtu bytes**—Sets the interface MTU of packets that the virtual asynchronous interface supports. The default MTU is 1500 bytes on a virtual asynchronous interface. The acceptable range is from 64 to 1,000,000 bytes.

- **routing**—Permits routing updates between connections. This keyword is required if the destination device is not on a subnet connected to one of the interfaces on the router.

Table 39  LAT-to-TCP Outgoing Options

Outgoing TCP Translation

**tcp ip-address**

Translates LAT to TCP/IP Telnet. Supply an IP address as a standard, four-part dotted decimal IP address, the name of an IP host that can be resolved by the DNS, or explicit specification in an `ip host` command (refer to the description for the `host-name` keyword in the “Outgoing TCP Connection Request Options” section).

Outgoing TCP Connection Request Options
Any of the following optional keywords can be used to configure TCP connection requests:

- **binary**—Negotiates Telnet binary mode on the connection.

- **host-name**—Stores the host name rather than its IP address, thereby allowing the host name to be resolved at connection time instead of configuration time. There is also a `rotor` keyword suboption that you can use to modify the behavior of the `host-name` keyword by allowing one of the IP addresses defined by the `ip host` configuration command to be chosen randomly. If one address fails, another one will be tried, and so on until all address choices are exhausted. You can use the `rotor` keyword, therefore, to provide basic load sharing of the IP destinations.

- **multibyte-IAC**—Always treat multiple Interpret as Command (IAC) escape character codes as a Telnet command.

- **port number**—For outgoing connections, enter the number of the port to match. The default is port 23 (Telnet).
source-interface—Specifies the source address used for Telnet connections initiated by the router.

stream—Performs stream processing, which enables a raw TCP stream with no Telnet control sequences. A stream connection does not process or generate any Telnet options, and also prevents Telnet processing of the data stream. This keyword might be useful for connections to ports running the UNIX-to-UNIX Copy Program (UUCP) or other non-Telnet protocols, or to ports connected to printers. For ports connected to printers using Telnet, the stream keyword prevents some of the problems associated with using Telnet for printers, such as unusual events happening to carriage returns or line feeds and echoing of data back to VMS systems.

### Table 40 LAT-to-X.25 Outgoing Translation Options

#### Outgoing X.25 Translation

**x25 x.121-address**

Translates LAT to the X.25 protocol. Supply an X.121 address that conforms to the specifications provided in the *CCITT 1984 Red Book*, or the name of an X.25 host that can be resolved by the DNS, or explicit specification in an **x25 host** command.

The address number generally consists of a portion that is administered by the public data network (PDN) and a portion that is locally assigned. You must be sure that the numbers that you assign agree with the addresses assigned to you by the X.25 service provider. The X.121 addresses generally will be subaddresses of the X.121 address for the X.25 network interface.

#### Outgoing X.25 Connection Request Options

Any of the following optional keywords can be used to configure X.25 connection requests:

- **cud c-u-data**—Sends the specified X.25 Call User Data (CUD) text as part of an outgoing call request after the protocol identification bytes.

- **no-reverse**—Specifies that outgoing calls not request the X.25 reverse charge facility, when the interface default is that all outgoing calls are reverse charged.

- **profile profile**—Sets the X.3 packet assembler/disassembler (PAD) parameters as defined in the profile created by the **x29 profile** command.

- **pvc number [interface serial number | packetsize in-size out-size | windowsize in-size out-size | no-reset]**—Specifies that the outgoing connection is actually a PVC. The **number** argument specifies the virtual circuit channel number of the connection, which must be less than the virtual circuits assigned to the switched virtual circuit (SVC). Only one session is allowed per PVC. Use the following optional keywords to further define the connection:
  - **interface serial number**—Specifies a PVC interface on which to set up the PVC connection.
  - **packetsize in-size out-size**—Specifies the input packet size (in-size) and output packet size (out-size) for the PVC. Valid packet size values are: 16, 32, 64, 128, 256, 512, 1024, 2048, and 4096.
  - **windowsize in-size out-size**—Specifies the packet count for input windows (in-size) and output windows (out-size) for the outgoing translation. Values of in-size and out-size range from 1 to 127 and must not be greater than the value set for the **x25 modulo** command. You must specify the same value for in-size and out-size.
  - **no-reset**—Causes the Cisco router to send a no Reset packet request at startup of a TCP or LAT to permanent virtual circuit (PVC) translation session.
Examples

The following example illustrates incoming LAT to outgoing TCP translations. The **unadvertised** keyword prevents broadcast of service advertisements to other servers in the network. Outgoing translated packets are sent to IP host Host1, TCP port 4005.

```
translate lat pt-printer1 unadvertised tcp Host1 port 4005
```

The following example translates LAT on an incoming line to SLIP on an outgoing line. It uses header compression only if incoming TCP packets on the same interface are compressed.

```
translate lat Service1 slip 10.0.0.4 header-compression
```

The following example first shows how to disable keepalive packets on a PPP line using the `translate lat` command, then shows translated session output from the `show translate` EXEC command indicating keepalive packets have been turned off.

```
translate lat Service2 ppp 172.21.2.2 keepalive 0
.
.
.
Router# show translate
```

```
Translate From: LAT Service2
  To:   PPP 172.21.2.2 keepalive 0
       0/0 users active, 0 peak, 0 total, 0 failures
```

Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>show translate</strong></td>
<td>Displays configured translation sessions.</td>
</tr>
<tr>
<td><strong>translate tcp</strong></td>
<td>Translates a TCP connection request automatically to another outgoing protocol connection.</td>
</tr>
<tr>
<td><strong>translate x25</strong></td>
<td>Translates an X.25 connection request automatically to another outgoing protocol connection.</td>
</tr>
<tr>
<td><strong>x29 access-list</strong></td>
<td>Limits access to the access server from certain X.25 hosts.</td>
</tr>
<tr>
<td><strong>x29 profile</strong></td>
<td>Creates a PAD profile script for use by the <code>translate</code> command.</td>
</tr>
</tbody>
</table>
translate lat (virtual access interfaces)

When receiving a local-area transport (LAT) connection request to a service name, to set up the Cisco router to automatically translate the request to another outgoing protocol connection type, use the `translate lat` command in global configuration mode. To remove or change the translation request, use the `no` form of this command.

The command syntax that follows shows how to apply a virtual interface template in place of outgoing `translate` options. If you are using virtual templates for protocol translation, all outgoing options are defined in the virtual interface template. Table 41 lists all outgoing options and their corresponding interface configuration commands.

```plaintext
translate lat incoming-service-name [incoming-options] virtual-template number
[global-options]

no translate lat incoming-service-name [incoming-options] virtual-template number
[global-options]
```

### Syntax Description

- **incoming-service-name**: A LAT service name. When used on the incoming portion of the `translate lat` command, `service-name` is the name of the service that users specify when trying to make a translated connection. This name can match the name of the final destination resource, but this match is not required. Such matches can be useful when making remote translated connections.

- **incoming-options**: (Optional) An incoming connection request option. For LAT, the only keyword currently supported is:
  - `unadvertised`: Prevents service advertisements from being broadcast to the network. This keyword can be useful, for example, when you define translations for many printers, and you do not want these services advertised to other LAT terminal servers. (VMS systems will be able to connect to the service even though it is not advertised.)

- **virtual-template number**: Applies the virtual interface template specified by the `number` argument in place of outgoing options.

- **global-options**: (Optional) Translation options that can be used by any connection type and can be one or more of the following:
  - `access-class number`: Allows the incoming call to be used by source hosts that match the access list parameters. The argument `number` is an integer previously assigned to an access list. Standard access list numbers are in the range from 1 to 99; expanded standard access lists numbers are in the range 1300 to 1999.
  - `max-users number`: Limits the number of simultaneous users of the translation to `number` (an integer you specify).
  - `local`: Allows Telnet protocol negotiations to not be translated.
  - `login`: Requires that the user log in before the outgoing connection is made. This type of login is specified on the virtual terminal lines with the `login` command.
  - `quiet`: Suppresses printing of user-information messages.
Terminal Services Commands

translate lat (virtual access interfaces)

Defaults
No default translation parameters

Command Modes
Global configuration

Command History
Release   Modification
10.0   This command was introduced.

Usage Guidelines
You define the protocol translation connections by choosing a protocol keyword and supplying the appropriate address, host name, or service name. The protocol connection information is followed by optional features for that connection, as appropriate. For example, the binary keyword is only appropriate with TCP/IP connections. The global options, in general, apply to all the connection types, but there are exceptions.

Rather than specifying outgoing translation options in the translate command, configure these options as interface configuration commands under the virtual interface template, then apply the virtual interface template to the translate command. Table 41 maps outgoing translate command options to interface commands you can configure in the virtual interface template.

Table 41 Mapping Outgoing translate lat Options to Interface Commands

<table>
<thead>
<tr>
<th>translate lat Command Options</th>
<th>Corresponding Interface Configuration Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>ip-pool</td>
<td>peer default ip address {ip-address</td>
</tr>
<tr>
<td>header-compression</td>
<td>ip tcp header compression [on</td>
</tr>
<tr>
<td>routing</td>
<td>ip routing or ipx routing</td>
</tr>
<tr>
<td>mtu</td>
<td>mtu</td>
</tr>
<tr>
<td>keepalive</td>
<td>keepalive</td>
</tr>
<tr>
<td>authentication {chap</td>
<td>pap}</td>
</tr>
<tr>
<td>ppp use-tacacs</td>
<td>ppp use-tacacs</td>
</tr>
<tr>
<td>ipx loopback</td>
<td>ipx ppp-client loopback number</td>
</tr>
</tbody>
</table>

Examples
The following example configures PPP tunneling from a PC across a LAT network. The remote PC is given the IP address 10.12.118.12 when it dials in. The unadvertised keyword prevents broadcast of service advertisements to other servers.

interface Virtual-Templatel
ip unnumbered Ethernet0
peer default ip address 10.12.118.12
ppp authentication chap
!
translate lat pt-printer1 unadvertised virtual-template 1
<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>show translate</code></td>
<td>Displays configured translation sessions.</td>
</tr>
<tr>
<td><code>translate tcp</code></td>
<td>Translates a TCP connection request automatically to another outgoing protocol connection.</td>
</tr>
<tr>
<td><code>translate x25</code></td>
<td>Translates an X.25 connection request automatically to another outgoing protocol connection.</td>
</tr>
<tr>
<td><code>x29 access-list</code></td>
<td>Limits access to the access server from certain X.25 hosts.</td>
</tr>
<tr>
<td><code>x29 profile</code></td>
<td>Creates a PAD profile script for use by the <code>translate</code> command.</td>
</tr>
</tbody>
</table>
translate tcp

To translate a connection request to another protocol connection type when receiving a TCP connection request to a particular destination address or host name, use the `translate tcp` command in global configuration mode. To remove or change the translation request, use the `no` form of this command.

```
translate tcp incoming-address [incoming-options] protocol outgoing-address [outgoing-options] [global-options]
no translate tcp incoming-address [incoming-options] protocol outgoing-address [outgoing-options] [global-options]
```

**Syntax Description**

- **incoming-address**: Standard IP address in standard, four-part dotted decimal notation. The IP address cannot be in use by other routers, and it should be on a connected subnet.
- **incoming-options**: (Optional) An incoming connection request option. Choices are as follows:
  - **binary**: Negotiates Telnet binary mode on the Telnet connection. (This was the default in previous versions of the protocol translation software and is set automatically when you enter a `translate` command in the previous format.)
  - **port number**: The number of the port to match for incoming connections. The default is port 23 (Telnet). For outgoing connections, enter the number of the port to use. The default is port 23.
  - **printer**: Supports local-area transport (LAT) and X.25 printing over a TCP network among multiple sites. This keyword causes the protocol translation software to delay the completion of an incoming Telnet connection until after the outgoing protocol connection (to LAT or X.25) has been successfully established. An unsuccessful outgoing connection attempt results in the TCP connection to the router being refused, rather than being accepted and then closed, which is the default behavior. Note that using this keyword will force the global `quiet` keyword to be applied to the translation.
  - **stream**: Performs stream processing, which enables a raw TCP stream with no Telnet control sequences. A stream connection does not process or generate any Telnet options, and also prevents Telnet processing of the data stream. This keyword might be useful for connections to ports running the UNIX-to-UNIX Copy Program (UUCP) or other non-Telnet protocols, or to ports connected to printers. For ports connected to printers using Telnet, the `stream` keyword prevents some of the problems associated with using Telnet for printers, such as unusual events happening to carriage returns or line feeds and echoing of data back to VMS systems.
translate tcp

Defaults
No default translation parameters

Command Modes
Global configuration

**protocol**
A protocol name followed by an address or host name. Protocol translation choices are: *lat, ppp, slip, and x25.*

**outgoing-address**
Additional keywords that can be entered with the protocol are as follows:

- **autocommand**—Specifies an EXEC command for an outgoing connection. The command executes upon connection to a host. You can issue any EXEC command and any switch or host name as an argument to the autocommand keyword. If the string following autocommand has one or more spaces as part of the string, you must place quotation marks (" ") around the string. If you want to enable AppleTalk Remote Access (ARA) on an outgoing connection, specify the autocommand arap keywords. These keywords are necessary for ARA because ARA does not use addressing, and this option permits you to invoke the ARA string.

- **virtual-template**—Associates a virtual template with a virtual access interface. See the translate tcp (virtual access interfaces) command description for more information.

Optional) Outgoing connection request options. Choices depend upon the protocol or command entered. See Table 42, Table 43, Table 44, and Table 45 for more information.

Optional) One or more of the following translation options can be used by any connection type:

- **access-class number**—Allows the incoming call to be used by source hosts that match the access list parameters. The argument number is an integer previously assigned to an access list. Standard access list numbers are in the range from 1 to 99; expanded standard access lists numbers are in the range 1300 to 1999.

- **local**—Allows Telnet protocol negotiations to not be translated.

- **login**—Requires that the user log in before the outgoing connection is made. This type of login is specified on the virtual terminal lines with the login command.

- **max-users number**—Limits the number of simultaneous users of the translation to number (an integer you specify).

- **quiet**—Suppresses printing of user-information messages.

- **swap**—Valid for TCP-to-X.25 translations only, and allows X.3 parameters to be set on the router by the host originating the X.25 call, or by an X.29 profile. This configuration enables incoming and outgoing X.25 connections to be swapped so that the device is treated like a PAD when it accepts a call. By default, the router functions like a PAD for calls that it initiates, and like an X.25 host for calls it accepts. The swap keyword allows connections from an X.25 host that wants to connect to the router, and then treats it like a PAD.
You define protocol translation connections by supplying a protocol keyword and the address, host name, or service name. A TCP protocol translation command can be as simple as the following example:

```
Router(config)# translate tcp 10.1.1.1 X.25 1236672
```

However, the Cisco IOS software provides a broad range of options that support protocol translations in many networking environments. Table 42, Table 43, Table 44, and Table 45 list the `translate tcp` translation options by protocol.

You can also use the Cisco IOS command-line interface to help you understand how these keywords are entered. In global configuration mode, begin entering the `translate` command and add a question mark at each portion of the command to display the options available. Some examples follow:

```
Router(config)# translate tcp ?

Hostname or A.B.C.D IP address
```

```
Router(config)# translate tcp 1.1.1.1 ?

autocommand       Associate a command with a translation on this connections
binary            Negotiate Telnet binary mode on the connection
lat               DEC LAT protocol
port              Port Number
ppp               Virtual async PPP
printer           Enable non-interactive (implies global quiet)
slip              Virtual async SLIP
stream            Enable stream processing
tcp               TCP/IP Telnet
virtual-template  Associate a virtual template with virtual access interface
x25               X.25
```

```
Router(config)# translate tcp 1.1.1.1 lat LAT-1 ?

access-class  Allow access list parameters to be used by source hosts
local         Allow Telnet protocol negotiations not to be translated
login         Require that the user log in before the outgoing connection is made
max-users     Limit the number of simultaneous users of the translation
node          LAT node name
port          LAT port name
quiet         Suppress printing of user-information messages
unadvertised  Prevent service advertisements from being broadcast to the network
```

**Note**

If you plan to translate to X.25 on a permanent virtual circuit (PVC), see the description for the `translate x25` command for important configuration notes.
Table 42  TCP-to-LAT Outgoing Options

Outgoing LAT Translation

lat service-name

Translates TCP to the LAT protocol. The software must learn the service name through LAT service advertisements before it can use the service.

Outgoing LAT Connection Request Options

Any of the following optional keywords can be used to configure LAT connection requests:

- **node name**—Connects to the specified node that offers a LAT service. By default, the connection is made to the highest-rated node that offers the service.
- **port name**—Destination LAT port name in the format of the remote system. This parameter is usually ignored in most time-sharing systems, but is used by terminal servers that offer reverse-LAT services.
- **unadvertised**—Prevents LAT service advertisements from being broadcast to the network.

Table 43  TCP-to-PPP Outgoing Options

Outgoing PPP Translation

ppp {ip-address | ip-pool [scope-name name]}

Translates from TCP to virtual asynchronous PPP. Supply an IP address as a standard, four-part dotted decimal IP address.

The **ip-pool** keyword obtains an IP address from a Dynamic Host Configuration Protocol (DHCP) proxy client or a local pool. If the **scope-name** keyword is not specified, the address is obtained from a DHCP proxy client. If the **scope-name** keyword is specified, the IP address is obtained from the specified local pool. The **scope-name** keyword can specify a range of IP addresses.

Outgoing PPP Connection Request Options

Any of the following optional keywords can be used to configure PPP connection requests:

- **authentication {pap | chap}**—Sets Challenge Handshake Authentication Protocol (CHAP) or Password Authentication Protocol (PAP) authentication for PPP on virtual asynchronous interfaces. If you specify both keywords, order is significant; the system will try to use the first authentication type, then the second.
- **header-compression [passive]**—Implements header compression on IP packets only. The **passive** keyword permits compression on outgoing packets only if incoming TCP packets on the same virtual asynchronous interface are compressed. The default (without the **passive** keyword) permits compression on all traffic.
- **ipx loopback number**—Specifies the loopback interface to be created and permits clients running IPX-PPP to connect through virtual terminal lines on the router. A loopback interface must have been created and configured with a Novell IPX network number before IPX-PPP can work on the virtual terminal line. The virtual terminal line is assigned to the loopback interface.
- **keepalive number-of-seconds**—Specifies the interval at which keepalive packets are sent on Serial Line Internet Protocol (SLIP) and PPP virtual asynchronous interfaces. By default, keepalive packets are enabled and sent every 10 seconds. To shut off keepalive packets, use a value of 0. The active keepalive interval is 1 through 32,767 seconds. When you do not change from the default of 10, the keepalive interval does not appear in **more system:running-config** or **show translate** command output.
Table 43  TCP-to-PPP Outgoing Options (continued)

- **mtu bytes**—Sets the interface maximum transmission unit (MTU) of packets that the virtual asynchronous interface supports. The default MTU is 1500 bytes on a virtual asynchronous interface. The acceptable range is from 64 to 1,000,000 bytes.
- **routing**—Permits routing updates between connections. This keyword is required if the destination device is not on a subnet connected to one of the interfaces on the router.
- **use-tacacs**—Uses TACACS to verify PPP authentications for CHAP or PAP on virtual asynchronous interfaces.

Table 44  TCP-to-SLIP Outgoing Options

**Outgoing SLIP Translation**

```
slip {ip-address | ip-pool [scope-name name]}
```

Translates from TCP to virtual asynchronous SLIP. Supply an IP address as a standard, four-part dotted decimal IP address.

The `ip-pool` keyword obtains an IP address from a DHCP proxy client or a local pool. If the optional `scope-name` keyword is not specified, the address is obtained from a DHCP proxy client. If the `scope-name` keyword is specified, the IP address is obtained from the specified local pool. The `scope-name` keyword can specify a range of IP addresses.

**Note**  
The `slip` keyword applies only to outgoing connections; SLIP is not supported on incoming protocol translation connections.

**Outgoing SLIP Connection Request Options**

Any of the following optional keywords can be used to configure SLIP connection requests:

- **header-compression [passive]**—Implements header compression on IP packets only. The `passive` keyword permits compression on outgoing packets only if incoming TCP packets on the same virtual asynchronous interface are compressed. The default (without the `passive` keyword) permits compression on all traffic.
- **ipx loopback number**—Specifies the loopback interface to be created and permits clients running IPX-PPP over X.25 to connect through virtual terminal lines on the router. A loopback interface must have been created and configured with a Novell IPX network number before IPX-PPP can work on the virtual terminal line. The virtual terminal line is assigned to the loopback interface.
- **keepalive number-of-seconds**—Specifies the interval at which keepalive packets are sent on SLIP and PPP virtual asynchronous interfaces. By default, keepalive packets are enabled and sent every 10 seconds. To shut off keepalive packets, use a value of 0. The active keepalive interval is 1 through 32,767 seconds. When you do not change from the default of 10, the keepalive interval does not appear in `more system:running-config` or `show translate` command output.
- **mtu bytes**—Sets the interface MTU of packets that the virtual asynchronous interface supports. The default MTU is 1500 bytes on a virtual asynchronous interface. The acceptable range is from 64 to 1,000,000 bytes.
- **routing**—Permits routing updates between connections. This keyword is required if the destination device is not on a subnet connected to one of the interfaces on the router.
Table 45  

TCP-to-X.25 Outgoing Options

Outgoing X.25 Translation

x25 x.121-address

Translates TCP to the X.25 protocol. Supply an X.121 address that conforms to the specifications provided in the CCITT 1984 Red Book, or the name of an X.25 host that can be resolved by the DNS, or explicit specification in an x25 host command.

The address number generally consists of a portion that is administered by the public data network (PDN) and a portion that is locally assigned. You must be sure that the numbers that you assign agree with the addresses assigned to you by the X.25 service provider. The X.121 addresses generally will be subaddresses of the X.121 address for the X.25 network interface.

Outgoing X.25 Connection Request Options

Any of the following optional keywords can be used to configure X.25 connection requests:

- **cud** c-u-data—Sends the specified X.25 Call User Data (CUD) text as part of an outgoing call request after the protocol identification bytes.
- **no-reverse**—Specifies that outgoing calls not request the X.25 reverse charge facility, when the interface default is that all outgoing calls are reverse charged.
- **profile** profile—Sets the X.3 packet assembler/disassembler (PAD) parameters as defined in the profile created by the x29 profile command.
- **pvc number** [interface serial number | packetsize in-size out-size | windowsize in-size out-size | no-reset]—Specifies that the outgoing connection is actually a PVC. The number argument specifies the virtual circuit channel number of the incoming connection, which must be less than the virtual circuits assigned to the switched virtual circuit (SVC). Only one session is allowed per PVC. Use the following optional keywords to further define the connection:
  - **interface serial number**—Specifies a PVC interface on which to set up the PVC connection.
  - **packetsize in-size out-size**—Specifies the input packet size (in-size) and output packet size (out-size) for the PVC. Valid packet size values are: 16, 32, 64, 128, 256, 512, 1024, 2048, and 4096.
  - **windowsize in-size out-size**—Specifies the packet count for input windows (in-size) and output windows (out-size) for the outgoing translation. Values of in-size and out-size range from 1 to 127 and must not be greater than the value set for the x25 modulo command. You must specify the same value for in-size and out-size.
  - **no-reset**—Causes the Cisco router to send a no Reset packet request at startup of a TCP or LAT to PVC translation session.
- **reverse**—Provides reverse charging for X.25 on a per-call rather than a per-interface basis. Requests reverse charges on a specified X.121 address, even if the serial interface is not configured to request reverse charge calls.
- **use-map**—Applies x25 map pad command entry options (such as CUD and idle) and facilities (such as packet in, packet out, win in, and win out) to the outgoing protocol translation call. When the use-map keyword is specified on the translate command, the Destination address and optional PAD Protocol Identification (PID), CUD, and facilities are checked against a configured list of x25 map pad command entries. If a match is found, the map entry PID, CUD, and facilities are applied to the outgoing protocol translation call. The X.25 map facilities applied to the outgoing translation can be displayed with the show translation command throughout the duration of the translation session.
Examples

The following example illustrates the use of the TCP incoming protocol `printer` keyword for an incoming TCP connection:

```
translate tcp 172.19.32.250 printer x25 5678
```

The following example permits clients running IPX-PPP to connect through the device virtual terminal lines to a server running PPP:

```
interface loopback0
    no ip address
    ipx network 544
    ipx sap-interval 2000

translate tcp 172.21.14.67 port 1234 ppp 10.0.0.2 ipx loopback0
```

Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>show translate</code></td>
<td>Displays configured translation sessions.</td>
</tr>
<tr>
<td><code>translate lat</code></td>
<td>Translates a LAT connection request automatically to another outgoing protocol connection.</td>
</tr>
<tr>
<td><code>translate x25</code></td>
<td>Translates an X.25 connection request automatically to another outgoing protocol connection.</td>
</tr>
<tr>
<td><code>x29 access-list</code></td>
<td>Limits access to the access server from certain X.25 hosts.</td>
</tr>
<tr>
<td><code>x29 profile</code></td>
<td>Creates a PAD profile script for use by the <code>translate</code> command.</td>
</tr>
</tbody>
</table>
translate tcp (virtual access interfaces)

When receiving a TCP connection request to a particular destination address or host name, to set up the Cisco router to automatically translate the request to another outgoing protocol connection type, use the translate tcp command in global configuration mode. To remove or change the translation request, use the no form of this command.

The command syntax that follows shows how to apply a virtual interface template in place of outgoing translate options. If you are using virtual templates for protocol translation, all outgoing options are defined in the virtual interface template.

```
translate tcp incoming-address [incoming-options] virtual-template number [global-options]
no translate tcp incoming-address [incoming-options] virtual-template number [global-options]
```

<table>
<thead>
<tr>
<th>Syntax Description</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>incoming-address</td>
<td>TCP/IP Telnet and a standard IP address or host name. The ip-address argument is a standard, four-part dotted decimal IP address or the name of an IP host that can be resolved by the Domain Name System (DNS) or explicit specification in an ip host command.</td>
</tr>
<tr>
<td>incoming-options</td>
<td>(Optional) Incoming connection request options. These arguments can have the following values:</td>
</tr>
<tr>
<td></td>
<td>• binary—Negotiates Telnet binary mode on the Telnet connection. (This was the default in previous versions of the Cisco IOS software and is set automatically when you enter a translate command in the old format.)</td>
</tr>
<tr>
<td></td>
<td>• port number—For incoming connections, enter the number of the port to match. The default is port 23 (Telnet). For outgoing connections, enter the number of the port to use. The default is port 23.</td>
</tr>
<tr>
<td></td>
<td>• printer—Supports LAT and X.25 printing over a TCP network among multiple sites. This keyword causes the protocol translation software to delay the completion of an incoming Telnet connection until after the outgoing protocol connection (to LAT or X.25) has been successfully established. An unsuccessful outgoing connection attempt results in the TCP connection to the router being refused, rather than being accepted and then closed, which is the default behavior. Note that using this keyword will force the global quiet keyword to be applied to the translation.</td>
</tr>
<tr>
<td></td>
<td>• stream—Performs stream processing, which enables a raw TCP stream with no Telnet control sequences. A stream connection does not process or generate any Telnet options, and also prevents Telnet processing of the data stream. This keyword might be useful for connections to ports running the UNIX-to-UNIX Copy Program (UUCP) or other non-Telnet protocols, or to ports connected to printers. For ports connected to printers using Telnet, the stream keyword prevents some of the problems associated with using Telnet for printers, such as unusual events happening to carriage returns or line feeds and echoing of data back to VMS systems.</td>
</tr>
<tr>
<td>virtual-template</td>
<td>Applies the virtual interface template specified by the number argument in place of outgoing options.</td>
</tr>
<tr>
<td>number</td>
<td></td>
</tr>
</tbody>
</table>
Terminal Services Commands

translate tcp (virtual access interfaces)

**Defaults**

No default translation parameters

**Command Modes**

Global configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

You define the protocol translation connections by choosing a protocol keyword and supplying the appropriate address, host name, or service name. The protocol connection information is followed by optional features for that connection, as appropriate. For example, the `binary` keyword is only appropriate with TCP/IP connections. The global options, in general, apply to all the connection types, but there are exceptions.

**Examples**

The following example illustrates the use of the TCP incoming `printer` keyword for an incoming TCP connection:

```plaintext
interface Virtual-Template1
ip unnumbered Ethernet0
peer default ip address 10.12.108.1
ppp authentication chap

translate tcp 172.19.32.250 printer Virtual-Template1
```
## Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>show translate</strong></td>
<td>Displays configured translation sessions.</td>
</tr>
<tr>
<td><strong>translate tcp</strong></td>
<td>Translates a TCP connection request automatically to another outgoing protocol connection.</td>
</tr>
<tr>
<td><strong>translate x25</strong></td>
<td>Translates an X.25 connection request automatically to another outgoing protocol connection.</td>
</tr>
<tr>
<td><strong>x29 access-list</strong></td>
<td>Limits access to the access server from certain X.25 hosts.</td>
</tr>
<tr>
<td><strong>x29 profile</strong></td>
<td>Creates a PAD profile script for use by the <code>translate</code> command.</td>
</tr>
</tbody>
</table>
translate x25

To translate a connection request to another protocol connection type when receiving an X.25 connection request to a particular destination address or host name, use the `translate x25` command in global configuration mode. To remove or change the translation request, use the `no` form of this command.

```
translate x25 incoming-address [incoming-options [pvc number [pvc-options]]] protocol outgoing-address [outgoing-options] [global-options]
```

```
no translate x25 incoming-address [incoming-options [pvc number [pvc-options]]] protocol outgoing-address [outgoing-options] [global-options]
```

**Syntax Description**

- **incoming-address**: An X.25 and X.121 address that conform to specifications provided in the CCITT 1984 Red Book.

  This address generally consists of a portion that is administered by the PDN and a portion that is locally assigned. You must be sure that the numbers that you assign agree with the addresses assigned to you by the X.25 service provider. The X.121 addresses generally will be subaddresses of the X.121 address for the X.25 network interface. Typically, the interface address will be a 12-digit number. Any additional digits are interpreted as a subaddress. The PDN still routes these calls to the interface, and the Cisco IOS software is responsible for interpreting the extra digits.

  Do not use the same address on the interface and for translation.

- **incoming-options**: (Optional) An incoming connection request option. Choices are as follows:

  - **accept-reverse**: Accepts reverse charged calls on an X.121 address even if the serial interface is not configured to accept reverse charged calls.

  - **cud c-u-data**: Specifies the Call User Data (CUD) field to match in the X.25 Incoming Call packet. If not configured, the CUD in the Incoming Call packet must be blank.

  - **idle minutes**: Specifies the number of minutes the virtual circuit is idle. This keyword enables the protocol translation function to clear a switched virtual circuit after a set period of inactivity, where `minutes` is the number of minutes in the period. Calls either originated or terminated are cleared. The maximum value of `minutes` is 255. The default value of `minutes` is zero.

  - **printer**: Supports local-area transport (LAT) and TCP printing over an X.25 network among multiple sites. Provides an “interlock mechanism” between the acceptance of an incoming X.25 connection and the opening of an outgoing LAT or TCP connection. This keyword causes the Cisco IOS software to delay the call confirmation of an incoming X.25 call request until after the outgoing protocol connection (to TCP or LAT) has been successfully established. An unsuccessful outgoing connection attempt to the router results in the incoming X.25 connection being refused, rather than being accepted and then closed, which is the default behavior. Note that using this keyword will force the global `quiet` keyword to be applied to the translation.
- **profile** *profile*—Sets the X.3 packet assembler/disassembler (PAD) parameters as defined in the profile created by the *x29 profile* command.

- **pvc** *number* [*interface serial* *number* | *packetsize* *in-size* *out-size* | *windowsize* *in-size* *out-size* ]—Specifies that the outgoing connection is actually a PVC. The *number* argument specifies the virtual circuit channel number of the connection, which must be less than the virtual circuits assigned to the switched virtual circuit (SVC). Only one session is allowed per PVC. Use the following optional keywords to further define the connection:
  - **interface serial** *number*—Specifies a PVC interface on which to set up the PVC connection.
  - **packetsize** *in-size* *out-size*—Specifies the input packet size (*in-size*) and output packet size (*out-size*) for the PVC. Valid packet size values are as follows: 16, 32, 64, 128, 256, 512, 1024, 2048, and 4096.
  - **windowsize** *in-size* *out-size*—Specifies the packet count for input windows (*in-size*) and output windows (*out-size*) for the outgoing translation. Values of *in-size* and *out-size* range from 1 to 127 and must not be greater than the value set for the *x25 modulo* command. You must specify the same value for *in-size* and *out-size*.

### Table: Protocol Outgoing-Address

<table>
<thead>
<tr>
<th><strong>protocol outgoing-address</strong></th>
<th>A protocol name followed by an address or host name. Protocol translation choices are <strong>lat</strong>, <strong>ppp</strong>, <strong>slip</strong>, and <strong>tcp</strong>.</th>
</tr>
</thead>
</table>

**Note**  
The host name is translated to an address during configuration, unless you are translating to TCP and use the **host-name** keyword, which allows the host name to be resolved at connection time instead of configuration time. See Table 49 for more information about the **host-name** keyword.

Additional keywords that can be entered with the protocol are as follows:

- **autocommand**—Specifies an EXEC command for an outgoing connection. The command executes upon connection to a host. You can issue any EXEC command and any switch or host name as an argument to the **autocommand** keyword. If the string following **autocommand** has one or more spaces as part of the string, you must place quotation marks (" ") around the string. If you want to enable AppleTalk Remote Access (ARA) on an outgoing connection, specify the **autocommand arap** keywords. These keywords are necessary for ARA because ARA does not use addressing, and this option permits you to invoke the ARA string.

- **virtual-template**—Associates a virtual template with a virtual access interface. See the **translate x25 (virtual access interfaces)** command description for more information.
### Command Modes
Global configuration

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

### Usage Guidelines
You define protocol translation connections by supplying a protocol keyword and the address, host name, or service name. An X.25 protocol translation command can be as simple as the following example:

```
Router(config)# translate X.25 123672 tcp 1.1.1.1
```

However, the Cisco IOS software provides a broad range of options that support protocol translations in many networking environments. Table 46, Table 47, Table 48, and Table 49 lists the translate x25 translation options by protocol.
You can also use the Cisco IOS command-line interface to help you understand how these keywords are entered. In global configuration mode, begin entering the translate command and add a question mark at each portion of the command to display the options available. Some examples follow:

Router(config)# translate x25 ?

WORD X.121 Address pattern

Router(config)# translate x25 66666 ?

accept-reverse Accept reverse charge on a per-call basis
autocommand Associate a command with a translation on this connections
cud Specify the Call User Data (CUD)
idle Specify VC idle timer
lat DEC LAT protocol
ppp Virtual async PPP
printer Enable non-interactive (implies global quiet)
profile Use a defined X.3 profile
pvc An incoming connection is actually a PVC
slip Virtual async SLIP
tcp TCP/IP Telnet
virtual-template Associate a virtual template with virtual access interface
x25 X.25

Router(config)# translate x25 66666 tcp 1.1.1.1 ?

access-class Allow access list parameters to be used by source hosts
binary Negotiate Telnet binary mode on the connection
host-name Store the host name rather than its IP address
local Allow Telnet protocol negotiations not to be translated
login Require that the user log in before the outgoing connection is made
max-users Limit the number of simultaneous users of the translation
multibyte-IAC Always treat multiple IACs as telnet command
port Port Number
quiet Suppress printing of user-information messages
source-interface Specify source interface
stream Treat telnet escape characters as data
swap Allow X.3 parameters to be set on the protocol translator by the host originating the X.25 call

Table 46 X.25-to-LAT Outgoing Options

Outgoing LAT Translation

lat service-name

Translates X.25 to the LAT protocol. The software must learn the service name through LAT service advertisements before it can use the service.

Outgoing LAT Connection Request Options

Any of the following optional keywords can be used to configure LAT connection requests:

- **node name**—Connects to the specified node that offers a LAT service. By default, the connection is made to the highest-rated node that offers the service.
- **port name**—Destination LAT port name in the format of the remote system. This parameter is usually ignored in most time-sharing systems, but is used by terminal servers that offer reverse-LAT services.
- **unadvertised**—Prevents LAT service advertisements from being broadcast to the network.
Table 47  X.25-to-PPP Outgoing Options

Outgoing PPP Translation

```
ppp {ip-address | ip-pool [scope-name name]}
```

Translates from X.25 to virtual asynchronous PPP. Supply an IP address as a standard, four-part dotted decimal IP address.

The `ip-pool` keyword obtains an IP address from a Dynamic Host Configuration Protocol (DHCP) proxy client or a local pool. If the optional `scope-name` keyword is not specified, the address is obtained from a DHCP proxy client. If the `scope-name` keyword is specified, the IP address is obtained from the specified local pool. The `scope-name` keyword can specify a range of IP addresses.

Outgoing PPP Connection Request Options

Any of the following optional keywords can be used to configure PPP connection requests:

- **authentication {pap | chap}**—Sets Challenge Handshake Authentication Protocol (CHAP) or Password Authentication Protocol (PAP) authentication for PPP on virtual asynchronous interfaces. If you specify both options, order is significant; the system will try to use the first authentication type, then the second.
- **header-compression**—Configures header compression on IP packets only.
- **ipx loopback number**—Specifies the loopback interface to be created and permits clients running IPX-PPP over X.25 to connect through virtual terminal lines on the router. A loopback interface must have been created and configured with a Novell IPX network number before IPX-PPP can work on the virtual terminal line. The virtual terminal line is assigned to the loopback interface.
- **keepalive number-of-seconds**—Specifies the interval at which keepalive packets are sent on Serial Line Internet Protocol (SLIP) and PPP virtual asynchronous interfaces. By default, keepalive packets are enabled and sent every 10 seconds. To shut off keepalive packets, use a value of 0. The active keepalive interval is 1 through 32,767 seconds. When you do not change from the default of 10, the keepalive interval does not appear in `more system:running-config` or `show translate` command output.
- **mtu bytes**—Sets the interface MTU of packets that the virtual asynchronous interface supports. The default MTU is 1500 bytes on a virtual asynchronous interface. The acceptable range is from 64 to 1,000,000 bytes.
- **routing**—Permits routing updates between connections. This option is required if the destination device is not on a subnet connected to one of the interfaces on the router.
- **use-tacacs**—Uses TACACS to verify PPP authentications for CHAP or PAP on virtual asynchronous interfaces.
Table 48  X.25-to-SLIP Outgoing Options

Outgoing SLIP Translation

slip {ip-address | ip-pool [scope-name name]}

Translates from X.25 to virtual asynchronous SLIP. Supply an IP address as a standard, four-part dotted decimal IP address.

The ip-pool keyword obtains an IP address from a DHCP proxy client or a local pool. If the optional scope-name keyword is not specified, the address is obtained from a DHCP proxy client. If the scope-name keyword is specified, the IP address is obtained from the specified local pool. The scope-name keyword can specify a range of IP addresses.

Note  The slip argument applies only to outgoing connections; SLIP is not supported on incoming protocol translation connections.

Outgoing SLIP Connection Request Options

Any of the following optional keywords can be used to configure SLIP connection requests:

- header-compression [passive]—Implements header compression on IP packets only. The passive keyword permits compression on outgoing packets only if incoming TCP packets on the same virtual asynchronous interface are compressed. The default (without the passive keyword) permits compression on all traffic.

- ipx loopback number—Specifies the loopback interface to be created and permits clients running IPX-PPP over X.25 to connect through virtual terminal lines on the router. A loopback interface must have been created and configured with a Novell IPX network number before IPX-PPP can work on the virtual terminal line. The virtual terminal line is assigned to the loopback interface.

- keepalive number-of-seconds—Specifies the interval at which keepalive packets are sent on SLIP and PPP virtual asynchronous interfaces. By default, keepalive packets are enabled and sent every 10 seconds. To shut off keepalive packets, use a value of 0. The active keepalive interval is 1 through 32,767 seconds. When you do not change from the default of 10, the keepalive interval does not appear in more system:running-config or show translate command output.

- mtu bytes—Sets the interface MTU of packets that the virtual asynchronous interface supports. The default MTU is 1500 bytes on a virtual asynchronous interface. The acceptable range is from 64 to 1,000,000 bytes.

- routing—Permits routing updates between connections. This keyword is required if the destination device is not on a subnet connected to one of the interfaces on the router.

Table 49  X.25-to-TCP Outgoing Options

Outgoing TCP Translation

tcp ip-address

Translates X.25 to TCP/IP Telnet. Supply an IP address as a standard, four-part dotted decimal IP address, or the name of an IP host that can be resolved by the DNS, or explicit specification in an ip host command (refer to the description for the host-name keyword in the “Outgoing TCP Connection Request Options” section).
Protocol Translation and X.25 PVCs Functional Description

This section describes how the protocol translator works with X.25 PVCs. It will help you understand the overall behavior of incoming and outgoing X.25 PVCs associated with a translate command, enabling you to correctly configure protocol translator PVCs for your application.

Configuring X.25 PVCs

When the translate x25 command is configured with a PVC, an attempt is made to create the PVC. The following conditions can cause this attempt to fail:

- The PVCs assignment of the X.25 interface does not include the PVC number in the translate x25 command.
- The PVC number in the translate x25 command is already in use.
- An X.25 destination in a translate x25 command is routed to X.25 over TCP/IP (XOT), Connection Mode Network Service (CMNS), or Annex G, which do not support translated PVCs.

PVC numbers must be unique across an X.25 connection; however, PVC 1 on Serial 1/0 is different (and therefore unique) from PVC 1 on Serial 2/1.

If, once the translate x25 command is accepted, the X.25 interface on which the PVC is created goes down, the PVC enters an inactive state, the TCP or LAT connection is terminated, but the existing PAD context remains inactive.

An incoming TCP or LAT connection associated with a down outgoing PVC (displaying a “P/Inactive message”) will be rejected by the protocol translator.

Table 49  X.25-to-TCP Outgoing Options (continued)

Outgoing TCP Connection Request Options

Any of the following optional keywords can be used to configure TCP connection requests:

- **binary**—Negotiates Telnet binary mode on the connection.
- **host-name**—Stores the host name rather than its IP address, thereby allowing the host name to be resolved at connection time instead of configuration time. There is also a **rotor** keyword suboption that you can use to modify the behavior of the **host-name** keyword by allowing one of the IP addresses defined by the **ip host** configuration command to be chosen randomly. If one address fails, another one will be tried, and so on until all address choices are exhausted. You can use the **rotor** keyword, therefore, to provide basic load sharing of the IP destinations.
- **multibyte-IAC**—Always treat multiple Interpret as Command (IAC) escape character codes as a Telnet command.
- **port number**—For incoming connections, enter the number of the port to match. The default is port 23 (Telnet).
- **source-interface**—Specifies the source address used for Telnet connections initiated by the router.
- **stream**—Performs stream processing, which enables a raw TCP stream with no Telnet control sequences. A stream connection does not process or generate any Telnet options, and also prevents Telnet processing of the data stream. This option might be useful for connections to ports running UNIX-to-UNIX Copy Program (UUCP) or other non-Telnet protocols, or to ports connected to printers. For ports connected to printers using Telnet, the **stream** keyword prevents some of the problems associated with using Telnet for printers, such as unusual events happening to carriage returns or line feeds and echoing of data back to VMS systems.
If any X.25 traffic is received while the corresponding TCP or LAT connection is terminated, and if a Data packet is received in state D1, a Reset with a diagnostic message will be displayed, similar to the following:

20:17:11.809: Serial2: X.25 O D1 Reset (5) 8 lci 4
20:17:11.809: Cause 29, Diag 113 (Network out of order (PVC)/Remote network problem)

The number of outgoing and incoming protocol translation PVCs is limited only by the number of virtual terminal lines supported on the Cisco router. Remember that each protocol translation session uses a virtual terminal line, which lowers the number of virtual terminal lines available for Telnet sessions.

By default, the Cisco router sends a Reset packet with the cause “PVC Network Operational” and diagnostic “Maintenance action” messages at the start of a TCP or LAT to PVC translation session, to announce that the connection is established and that the PVC is able to handle data traffic. To suppress the PVC Reset packet at TCP or LAT session startup, configure the no-reset outgoing PVC keyword as shown in the following example:

```plaintext
translate tcp 192.168.22.102 port 5 x25 333 pvc 5 no-reset profile tcl
```

### Changing or Removing a translate Command PVC Configuration

Removing a `translate` command with an outgoing PVC specified is allowed only when there no active connection is associated with the outgoing PVC. An attempt to remove an active translation results in the following message:

Translate: Can’t delete/add entry - Connection(s) are currently active

For example, if PVC 5 is assigned to a `translate` command as shown in the following example:

```plaintext
translate tcp 10.0.155.61 port 5 x25 5 pvc 5 interface Serial2/0
```

And you want PVC 5 to be assigned under an X.25 interface instead of the `translate` command, as shown in the following example:

```plaintext
interface serial2/0
   x25 pvc 5 int s4/0 pvc 25
```

Perform the following steps to configure this reassignment:

**Step 1** Check whether a PVC is associated with a serial connection using the `show x25` EXEC command, as follows:

```plaintext
Router# show x25 vc 5

PVC 5, State:D1, Interface:Serial2/0
   Started ...
   Line:230 vty 4   Location:Host:nmos3m1
      connected to PAD <--> X25
```

**Step 2** If the PVC is associated with a TCP connection, terminate the connection by disconnecting the TCP session or by using the `clear line` EXEC command as shown in the following example:

```plaintext
Router# clear line vty 4
```

**Step 3** Enter configuration mode, delete the `translate` command, and reassign PVC 5 to an interface:

```plaintext
Router(config)# no translate tcp 10.0.155.61 port 5 x25 5 pvc 5 interface Serial2/0
Router(config)# interface serial2/0
Router(config-if)# x25 pvc 5 int s4/0 pvc 25
```
If you want to modify the `translate` command and change the PVC number from 5 to 12, follow steps 1 and 2, and modify the `translate` command with PVC 12, as follows:

```plaintext
Router(config)# translate tcp 10.0.155.61 port 12 x25 12 pvc 12 interface Serial2/0
```

### Understanding the X.25 Address and the PVC Interface Option on a `translate` Command

The protocol translator locates the X.121 destination address in the X.25 route table to determine the interface on which to establish the PVC. A more up-to-date, simpler approach uses the `translate` command with the `interface` keyword, which ignores the status of the interface by avoiding referencing the X.25 route table.

For example, instead of configuring an `x25 route` command for each translated PVC, and entering a long X.121 address on the `translate` command, as shown this example:

```plaintext
x25 route ^3278523344502 interface Serial1/5
translate tcp 10.0.155.61 port 2502 x25 3278523344502 pvc 1
```

You can simply enter one `translate` command that links the IP port number with the X.121 address and specifies the interface on which to establish the PVC, as follows:

```plaintext
translate tcp 10.0.155.61 port 2502 x25 2502 pvc 1 interface Serial 1/5
```

This is the recommended approach and should be adopted in place of `translate` commands that cause the destination address to be looked up in the route table.

### Examples

The following example shows how to use the `translate` global configuration command to translate from an X.25 PAD to a LAT device on Network A. It is applied to Router-A. The configuration example includes an access list that limits remote LAT access through Router-A to connections from PAD-C. This example typifies the use of access lists in the Cisco IOS software. The first two lines define the scope of access-list 1. The first line specifies that access list 1 will permit all calls from X.121 address 44444. The caret symbol (^) specifies that the first number 4 is the beginning of the address number. The second line of the definition explicitly denies calls from any other number. (Refer to the appendix “Regular Expressions” in the [Cisco IOS Terminal Services Configuration Guide](#), Release 12.2, for details concerning the use of special characters in defining X.121 addresses.)

```plaintext
! Define X25 access list to only allow pad-c.
x29 access-list 1 permit ^44444
x29 access-list 1 deny .*
!
! Set up translation.
translate x25 1111101 lat LAT-A access-class 1
```

The following example shows a simple X.25-to-TCP `translate x25` command. Packets coming in X.25 address 652365123 arrive via PVC 1 and are translated to TCP packets and sent out IP address 172.16.1.1.

```plaintext
translate x25 652365123 pvc 1 tcp 172.16.1.1
```

The following example shows a more complex configuration that calls an X.29 profile and swaps the default PAD operation of the router to that of an X.25 host. The name of the profile is fullpackets.

```plaintext
x29 profile fullpackets 2:0 3:0 4:100 7:21
translate x25 217536124 profile fullpackets tcp Host1 port 4006 swap
```
The following example shows the use of the X.25 incoming protocol **printer** keyword for an incoming X.25 connection:

```
translate x25 55555 printer tcp 172.16.1.1
```

The following examples causes the protocol translator to try connecting to IP address 172.16.1.1 and if that failed, to try IP address 172.16.2.1, and so on through all IP addresses listed in the **ip host** command:

```
ip host my-hosts 172.16.1.1 172.16.2.1 172.16.3.1
translate x25 55555 tcp my-hosts host-name
```

The following example uses the **rotor** keyword to cause the protocol translator to randomly choose one of the IP address listed in the **ip host** command and if it fails to connect, to try another IP address, until all are exhausted:

```
ip host my-hosts 172.16.1.1 172.16.2.1 172.16.3.1
translate x25 55555 tcp my-hosts host1 rotor
```

The following example translates X.25 packets to PPP. It enables routing updates between the two connections:

```
translate x25 12345678 ppp 10.0.0.2 routing
```

The following example permits clients running ARA to connect through the virtual terminal lines of the device to an AppleTalk network:

```
appletalk routing
translate x25 12345678 autocommand arap
    arap enable
    arap dedicated
    arap timelimit 45
    arap warningtime 5
    arap noguest
    arap require-manual-password
    arap net-access-list 614
```

The following example specifies IP pooling from a DHCP server named **D-Server1**. It then specifies that incoming TCP traffic be translated to SLIP. The DHCP server will dynamically assign IP addresses on the outgoing sessions.

```
ip address-pool dhcp-proxy-client
ip dhcp-server D-Server1
translate x25 5467835 ppp ip-pool scope-name D-Server1
```

The following example specifies a local IP pool named **Pool2** with IP addresses ranging from 172.18.10.10 to 172.18.10.110. It then specifies that incoming X.25 traffic be translated to PPP. The local IP pool **Pool2** will be used to dynamically assign IP addresses on the outgoing sessions.

```
ip-pool Pool2 172.18.10.10 172.18.10.110
translate x25 1234567 ppp ip-pool scope-name Pool2
```

The following example shows how to set the idle timer. X.25 calls are cleared if they are idle for the configured time.

```
translate x25 1234 idle 2 lat Service3
```
### Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>show translate</strong></td>
<td>Displays configured translation sessions.</td>
</tr>
<tr>
<td><strong>translate lat</strong></td>
<td>Translates a LAT connection request automatically to another outgoing protocol connection.</td>
</tr>
<tr>
<td><strong>translate tcp</strong></td>
<td>Translates a TCP connection request automatically to another outgoing protocol connection.</td>
</tr>
<tr>
<td><strong>x29 access-list</strong></td>
<td>Limits access to the access server from certain X.25 hosts.</td>
</tr>
<tr>
<td><strong>x29 profile</strong></td>
<td>Creates a PAD profile script for use by the <code>translate</code> command.</td>
</tr>
</tbody>
</table>
translate x25 (virtual access interfaces)

When receiving an X.25 connection request to a particular destination address, to set up the Cisco router to automatically translate the request to another outgoing protocol connection type, use the `translate x25` command in global configuration mode. To remove or change the translation request, use the `no` form of this command.

The command syntax that follows shows how to apply a virtual interface template in place of outgoing `translate x25` options. If you are using virtual templates for protocol translation, all outgoing options are defined in the virtual interface template. Table 50 lists all outgoing options and their corresponding interface configuration commands.

```
translate x25 incoming-address [incoming-options [pvc number [pvc-options]]] protocol outgoing-address [outgoing-options] virtual-template number [global-options]
no translate x25 incoming-address [incoming-options [pvc number [pvc-options]]] protocol outgoing-address [outgoing-options] virtual-template number [global-options]
```

**Syntax Description**

- **incoming-address**: An X.25 and X.121 address that conform to specifications provided in the CCITT 1984 Red Book.

  This address generally consists of a portion that is administered by the PDN and a portion that is locally assigned. You must be sure that the numbers that you assign agree with the addresses assigned to you by the X.25 service provider. The X.121 addresses generally will be subaddresses of the X.121 address for the X.25 network interface. Typically, the interface address will be a 12-digit number. Any additional digits are interpreted as a subaddress. The PDN still routes these calls to the interface, and the Cisco IOS software is responsible for interpreting the extra digits.

  Do not use the same address on the interface and for translation.

- **incoming-options**: (Optional) Incoming connection request keywords and arguments, as follows:
  
  - **accept-reverse**: Accepts reverse charged calls on an X.121 address even if the serial interface is not configured to accept reverse charged calls. This is an incoming option only.
  
  - **cud c-u-data**: Specifies the Call User Data (CUD) field to match in the X.25 Incoming Call packet. If not configured, the CUD in the Incoming Call packet must be blank.
  
  - **printer**: Supports LAT and TCP printing over an X.25 network among multiple sites. Provides an “interlock mechanism” between the acceptance of an incoming X.25 connection and the opening of an outgoing LAT or TCP connection. The `printer` keyword causes the protocol translation software to delay the call confirmation of an incoming X.25 call request until the outgoing protocol connection (to TCP or LAT) has been successfully established. An unsuccessful outgoing connection attempt to the router results in the incoming X.25 connection being refused, rather than being confirmed and then cleared, which is the default behavior. Note that using this keyword will force the global `quiet` keyword to be applied to the translation.
Terminal Services Commands

translate x25 (virtual access interfaces)

- **profile profile**—Sets the X.3 PAD parameters as defined in the profile created by the **x29 profile** command.

- **pvc number [interface serial number | packetsize in-size out-size | windowsize in-size out-size]**—Specifies that the outgoing connection is actually a PVC. The `number` argument specifies the virtual circuit channel number of the connection, which must be less than the virtual circuits assigned to the switched virtual circuit (SVC). Only one session is allowed per PVC. Use the following optional keywords and arguments to further define the connection:
  - **interface serial number**—Specifies a PVC interface on which to set up the PVC connection.
  - **packetsize in-size out-size**—Specifies the input packet size (`in-size`) and output packet size (`out-size`) for the PVC. Valid packet size values are as follows: 16, 32, 64, 128, 256, 512, 1024, 2048, and 4096.
  - **windowsize in-size out-size**—Specifies the packet count for input windows (`in-size`) and output windows (`out-size`) for the outgoing translation. Values of `in-size` and `out-size` range from 1 to 127 and must not be greater than the value set for the **x25 modulo** command. You must specify the same value for `in-size` and `out-size`.

<table>
<thead>
<tr>
<th>virtual-template number</th>
<th>Applies the virtual interface template specified by the <code>number</code> argument in place of outgoing options.</th>
</tr>
</thead>
<tbody>
<tr>
<td>global-options</td>
<td>(Optional) Translation options that can be used by any connection type and can be one or more of the following:</td>
</tr>
<tr>
<td></td>
<td>• <strong>access-class number</strong>—Allows the incoming call to be used by source hosts that match the access list parameters. The argument <code>number</code> is an integer previously assigned to an access list. Standard access list numbers are in the range from 1 to 99; expanded standard access lists numbers are in the range 1300 to 1999.</td>
</tr>
<tr>
<td></td>
<td>• <strong>local</strong>—Allows Telnet protocol negotiations to <strong>not</strong> be translated.</td>
</tr>
<tr>
<td></td>
<td>• <strong>login</strong>—Requires that the user log in before the outgoing connection is made. This type of login is specified on the virtual terminal lines with the <strong>login</strong> command.</td>
</tr>
<tr>
<td></td>
<td>• <strong>max-users number</strong>—Limits the number of simultaneous users of the translation to <code>number</code> (an integer you specify).</td>
</tr>
<tr>
<td></td>
<td>• <strong>quiet</strong>—Suppresses printing of user-information messages.</td>
</tr>
<tr>
<td></td>
<td>• <strong>swap</strong>—Valid for X.25-to-TCP translations only, and allows X.3 parameters to be set on the router by the host originating the X.25 call, or by an X.29 profile. This keyword allows incoming and outgoing X.25 connections to be swapped so that the device is treated like a PAD when it accepts a call. By default, the router functions like a PAD for calls that it initiates, and like an X.25 host for calls it accepts. The <strong>swap</strong> keyword allows connections from an X.25 host that wants to connect to the router, and then treats it like a PAD.</td>
</tr>
</tbody>
</table>
Terminal Services Commands

translate x25 (virtual access interfaces)

Defaults
No default translation parameters.

Command Modes
Global configuration

Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

Usage Guidelines

You define the protocol translation connections by choosing a protocol keyword and supplying the appropriate address or service name. The protocol connection information is followed by optional features for that connection, as appropriate. The global options, in general, apply to all the connection types, but there are exceptions. The swap keyword, for example, is for X.25- to-TCP translations only. See the example for more explanations on how to enter this command.

Rather than specifying outgoing translation options in the translate command, configure these options as interface configuration commands under the virtual interface template, then apply the virtual interface template to the translate command. Table 50 maps outgoing translate command options to interface commands you can configure in the virtual interface template.

Table 50  Mapping Outgoing translate x25 Options to Interface Commands

<table>
<thead>
<tr>
<th>translate x25 Command Options</th>
<th>Corresponding Interface Configuration Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>ip-pool</td>
<td>peer default ip address {ip-address</td>
</tr>
<tr>
<td>header-compression</td>
<td>ip tcp header compression [on</td>
</tr>
<tr>
<td>routing</td>
<td>ip routing or ipx routing</td>
</tr>
<tr>
<td>mtu</td>
<td>mtu</td>
</tr>
<tr>
<td>keepalive</td>
<td>keepalive</td>
</tr>
<tr>
<td>authentication {chap</td>
<td>pap}</td>
</tr>
<tr>
<td>ppp use-tacacs</td>
<td>ppp use-tacacs</td>
</tr>
<tr>
<td>ipx loopback</td>
<td>ipx ppp-client loopback number</td>
</tr>
</tbody>
</table>

Examples

The following example shows a virtual template with PPP encapsulation specified by default (not explicit). It also specifies CHAP authentication and an X.29 access list.

```
x29 access-list 1 permit ^5555
!
interface Virtual-Template1
  ip unnumbered Ethernet0
  peer default ip address 172.16.2.129
  ppp authentication chap
!
translate x25 5555667 virtual-template 1 access-class 1
```
### Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>interface virtual-template</td>
<td>Creates a virtual template interface that can be configured and applied</td>
</tr>
<tr>
<td></td>
<td>dynamically in creating virtual access interfaces.</td>
</tr>
<tr>
<td>show translate</td>
<td>Displays configured translation sessions.</td>
</tr>
<tr>
<td>translate lat</td>
<td>Translates a LAT connection request automatically to another outgoing</td>
</tr>
<tr>
<td></td>
<td>protocol connection type.</td>
</tr>
<tr>
<td>translate tcp</td>
<td>Translates a TCP connection request automatically to another outgoing</td>
</tr>
<tr>
<td></td>
<td>protocol connection type.</td>
</tr>
<tr>
<td>x29 access-list</td>
<td>Limits access to the access server from certain X.25 hosts.</td>
</tr>
<tr>
<td>x29 profile</td>
<td>Creates a PAD profile script for use by the translate command.</td>
</tr>
</tbody>
</table>
transport input

To define which protocols to use to connect to a specific line of the router, use the transport input command in line configuration mode. To change or remove the protocol, use the no form of this command.

```
transport input {all | lat | mop | nasi | none | pad | rlogin | telnet | v120}
```

```
no transport input {all | lat | mop | nasi | none | pad | rlogin | telnet | v120}
```

### Syntax Description

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>all</td>
<td>Selects all protocols.</td>
</tr>
<tr>
<td>lat</td>
<td>Selects the Digital LAT protocol and specifies both incoming reverse LAT and host-initiated connections.</td>
</tr>
<tr>
<td>mop</td>
<td>Selects Maintenance Operation Protocol (MOP).</td>
</tr>
<tr>
<td>nasi</td>
<td>Selects NetWare Access Servers Interface (NASI) as the input transport protocol.</td>
</tr>
<tr>
<td>none</td>
<td>Prevents any protocol selection on the line. This makes the port unusable by incoming connections.</td>
</tr>
<tr>
<td>pad</td>
<td>Selects X.3 packet assembler/disassembler (PAD) incoming connections.</td>
</tr>
<tr>
<td>rlogin</td>
<td>Selects the UNIX rlogin protocol.</td>
</tr>
<tr>
<td>telnet</td>
<td>Specifies all types of incoming TCP/IP connections.</td>
</tr>
<tr>
<td>v120</td>
<td>Selects the V.120 protocol for incoming asynchronous connections over ISDN.</td>
</tr>
</tbody>
</table>

### Defaults

No protocols allowed on the line (none).

### Command Modes

Line configuration

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>11.1</td>
<td>The none keyword was added and became the default. Before Cisco IOS Release 11.1, the default keyword was all.</td>
</tr>
</tbody>
</table>

### Usage Guidelines

Cisco routers do not accept incoming network connections to asynchronous ports (TTY lines) by default. You must specify an incoming transport protocol or specify the transport input all command before the line will accept incoming connections. For example, if you are using your router as a terminal server to make console-port connections to routers or other devices, you will not be able to use Telnet to connect to these devices. You will receive the message “Connection Refused.” This behavior is new as of Cisco IOS software Release 11.1. Previous to Release 11.1, the default was the transport input all command, and the all keyword restored pre-Cisco IOS software Release 11.0 defaults. If you are upgrading to Cisco IOS software Release 11.1(1) or later releases from earlier Cisco IOS software Releases, you must configure the transport input {protocol | all} command, or you will be locked out of your router.
You can specify one protocol, multiple protocols, all protocols, or no protocols. To specify multiple protocols, enter the keyword for each protocol, separated by a space.

This command can be useful in distributing resources among different types of users, or in making certain that only specific hosts can access a particular port. When using two-step protocol translation, the `transport input` command is useful in controlling exactly which protocols can be translated to other protocols.

Access lists for each individual protocol can be defined in addition to the allowances created by the `transport input` command. Any settings made with the `transport input` command override settings made with the `transport preferred` command.

**Examples**

The following example sets the incoming protocol to Telnet for virtual terminal lines 0 to 32:

```bash
line vty 0 32
transport input telnet
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>transport output</code></td>
<td>Determines the protocols that can be used for outgoing connections from a line.</td>
</tr>
<tr>
<td><code>transport preferred</code></td>
<td>Specifies the transport protocol that the Cisco IOS software uses if the user does not specify one when initiating a connection.</td>
</tr>
</tbody>
</table>
Transport output

To determine the protocols that can be used for outgoing connections from a line, use the transport output command in line configuration mode. To change or remove the protocol, use the no form of this command.

```
transport output { all | lat | mop | nasi | none | pad | rlogin | telnet | v120 }
no transport output { all | lat | mop | nasi | none | pad | rlogin | telnet | v120 }
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>all</td>
<td>Selects all protocols.</td>
</tr>
<tr>
<td>lat</td>
<td>Selects the Digital LAT protocol, which is the protocol used most often to connect routers to Digital hosts.</td>
</tr>
<tr>
<td>mop</td>
<td>Selects Maintenance Operation Protocol (MOP).</td>
</tr>
<tr>
<td>nasi</td>
<td>Selects NetWare Access Server Interface (NASI) as the output transport protocol.</td>
</tr>
<tr>
<td>none</td>
<td>Prevents any protocol selection on the line. The system normally assumes that any unrecognized command is a host name. If the protocol is set to none, the system no longer makes that assumption. No connection will be attempted if the command is not recognized.</td>
</tr>
<tr>
<td>pad</td>
<td>Selects X.3 packet assembler/disassembler (PAD), used most often to connect routers to X.25 hosts.</td>
</tr>
<tr>
<td>rlogin</td>
<td>Selects the UNIX rlogin protocol for TCP connections. The rlogin setting is a special case of Telnet. If an rlogin attempt to a particular host has failed, the failure will be tracked, and subsequent connection attempts will use Telnet instead.</td>
</tr>
<tr>
<td>telnet</td>
<td>Selects the TCP/IP Telnet protocol. It allows a user at one site to establish a TCP connection to a login server at another site.</td>
</tr>
<tr>
<td>v120</td>
<td>Selects the V.120 protocol for outgoing asynchronous connections over ISDN.</td>
</tr>
</tbody>
</table>

**Defaults**

Telnet

**Command Modes**

Line configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>11.1</td>
<td>The following keywords were added:</td>
</tr>
<tr>
<td></td>
<td>• all</td>
</tr>
<tr>
<td></td>
<td>• lat</td>
</tr>
<tr>
<td></td>
<td>• pad</td>
</tr>
<tr>
<td></td>
<td>• rlogin</td>
</tr>
<tr>
<td></td>
<td>• v120</td>
</tr>
</tbody>
</table>
**Usage Guidelines**

You can specify one protocol, multiple protocols, all protocols, or no protocols. To specify multiple protocols, enter the keyword for each protocol, separated by a space.

Any settings made with the `transport output` command override settings made with the `transport preferred` command.

**Examples**

The following example prevents any protocol selection:

```
transport output none
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>transport input</code></td>
<td>Defines which protocols to use to connect to a specific line of the router.</td>
</tr>
<tr>
<td><code>transport preferred</code></td>
<td>Specifies the transport protocol that the Cisco IOS software uses if the user does not specify one when initiating a connection.</td>
</tr>
</tbody>
</table>
transport preferred

To specify the transport protocol that the Cisco IOS software uses if the user does not specify one when initiating a connection, use the `transport preferred` command in line configuration mode. To change or remove the protocol, use the `no` form of this command.

```
transport preferred { all | lat | mop | nasi | none | pad | rlogin | telnet | v120 }
```

```
no transport preferred { all | lat | mop | nasi | none | pad | rlogin | telnet | v120 }
```

### Syntax Description

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>all</td>
<td>Selects all recognized protocols.</td>
</tr>
<tr>
<td>lat</td>
<td>Selects the Digital LAT protocol, which is the protocol used most often to connect routers to Digital hosts.</td>
</tr>
<tr>
<td>mop</td>
<td>Selects Maintenance Operation Protocol (MOP).</td>
</tr>
<tr>
<td>nasi</td>
<td>Selects NetWare Access Server Interface (NASI) protocol.</td>
</tr>
<tr>
<td>none</td>
<td>Prevents any protocol selection on the line. The system normally assumes that any unrecognized command is a host name. If the protocol is set to <code>none</code>, the system no longer makes that assumption. No connection is attempted if the command is not recognized.</td>
</tr>
<tr>
<td>pad</td>
<td>Selects X.3 packet assembler/disassembler (PAD), used most often to connect routers to X.25 hosts.</td>
</tr>
<tr>
<td>rlogin</td>
<td>Selects the UNIX rlogin protocol for TCP connections. The rlogin setting is a special case of Telnet. If an rlogin attempt to a particular host has failed, the failure will be tracked, and subsequent connection attempts will use Telnet instead.</td>
</tr>
<tr>
<td>telnet</td>
<td>Selects the TCP/IP Telnet protocol. It allows a user at one site to establish a TCP connection to a login server at another site.</td>
</tr>
<tr>
<td>v120</td>
<td>Selects the asynchronous protocols over ISDN.</td>
</tr>
</tbody>
</table>

### Defaults

Telnet

### Command Modes

Line configuration

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>11.1</td>
<td>The following keywords were added:</td>
</tr>
<tr>
<td></td>
<td>• lat</td>
</tr>
<tr>
<td></td>
<td>• pad</td>
</tr>
<tr>
<td></td>
<td>• rlogin</td>
</tr>
</tbody>
</table>
Usage Guidelines

Specify the `transport preferred none` command to prevent errant connection attempts.

Any settings made with the `transport input` or `transport output` commands override settings made with the `transport preferred` command.

Examples

The following example sets the preferred protocol to Telnet on physical terminal line 1:

```
line tty 1
transport preferred telnet
```

Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>terminal transport preferred</code></td>
<td>Specifies the preferred protocol to use for the current session when a command does not specify one.</td>
</tr>
<tr>
<td><code>transport input</code></td>
<td>Defines which protocols to use to connect to a specific line of the router.</td>
</tr>
<tr>
<td><code>transport output</code></td>
<td>Determines the protocols that can be used for outgoing connections from a line.</td>
</tr>
</tbody>
</table>
ttycap

To define characteristics of a terminal emulation file, use the `ttycap` command in global configuration mode. To delete any named ttycap entry from the configuration file, use the `no` form of this command.

```
ttycap ttycap-name termcap-entry

no ttycap ttycap-name
```

**Syntax Description**

<table>
<thead>
<tr>
<th><strong>ttycap-name</strong></th>
<th>Name of a file. It can be up to 32 characters long and must be unique.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>termcap-entry</strong></td>
<td>Commands that define the ttycap. Consists of two parts. (See the “Usage Guidelines” section for details.)</td>
</tr>
</tbody>
</table>

**Defaults**

VT100 terminal emulation

**Command Modes**

Global configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.3</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `show ttycap` EXEC command to test for the availability of a ttycap.

**Note**

Do not type a ttycap entry filename “default” or the Cisco IOS software will adopt the newly defined entry as the default.

The `termcap-entry` argument consists of two parts: a name portion and a capabilities portion.

The name portion is a series of names that can be used to refer to a specific terminal type. Generally, these names should represent commonly recognized terminal names (such as VT100 and VT200). Multiple names can be used. Each name is separated by a vertical bar symbol (|). The series is terminated by a colon symbol (:).

The following example illustrates a name specification for a VT100 termcap:

```
d0|vt100|vt100-am|vt100am|dec vt100:
```

The capabilities portion of the termcap entry consists of a sequence of termcap capabilities. These capabilities can include Boolean flags, string sequences, or numeric sequences. Each individual capability is terminated using a colon symbol (:). A Boolean flag can be set to true by including the two-character capability name in the termcap entry. The absence of any supported flag results in the flag being set to false.

The following is an example of a backspace Boolean flag:

```
bs:
```
A string sequence is a two-character capability name followed by an equal sign (=) and the character sequence.

The following example illustrates the capability for homing the cursor:

ho=\E[H:

The sequence \E represents the ESC character.

Control characters can be represented in string sequences by entering a two-character sequence starting with a caret symbol (^), followed by the character to be used as a control character.

The following example illustrates the definition of a control character.

bc=^h:

In this example, the backspace is entered into the termcap entry as the string sequence as the characters “^h.”

A numeric sequence is a two-character capability name followed by a number symbol (#) and the number.

The following example represents the number of columns on a screen.

c0=#80:

Use the backslash symbol (\) to extend the definition to multiple lines. The end of the ttycap termcap entry is specified by a colon terminating a line followed by an end-of-line character and no backslash.

For the definitions of supported Boolean-flag ttycap capabilities, see Table 51. For the definitions of supported string-sequence ttycap capabilities, see Table 52. For the definitions of supported number-sequence ttycap capabilities, see Table 53. For the definitions of supported color-sequence ttycap capabilities, see Table 54.

Table 51  Definitions of ttycap Capabilities: Boolean Flags

<table>
<thead>
<tr>
<th>Boolean Flag</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>am</td>
<td>Automatic margin</td>
</tr>
<tr>
<td>bs</td>
<td>Terminal can backspace with bs</td>
</tr>
<tr>
<td>ms</td>
<td>Safe to move in standout modes</td>
</tr>
<tr>
<td>nc</td>
<td>No currently working carriage return</td>
</tr>
<tr>
<td>xn</td>
<td>NEWLINE ignored after 80 columns (Concept)</td>
</tr>
<tr>
<td>xs</td>
<td>Standout not erased by overwriting (Hewlett-Packard)</td>
</tr>
</tbody>
</table>

Table 52  Definitions of ttycap Capabilities: String Sequences

<table>
<thead>
<tr>
<th>String Sequence</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AL</td>
<td>Add line below with cursor sequence</td>
</tr>
<tr>
<td>bc</td>
<td>Backspace if not ^h</td>
</tr>
<tr>
<td>bt</td>
<td>Backtab sequence</td>
</tr>
<tr>
<td>ce</td>
<td>Clear to end of line</td>
</tr>
<tr>
<td>cl</td>
<td>Clear screen, cursor to upper left</td>
</tr>
<tr>
<td>cm</td>
<td>Move cursor to row number and column number</td>
</tr>
</tbody>
</table>
### Table 52  Definitions of ttycap Capabilities: String Sequences (continued)

<table>
<thead>
<tr>
<th>String Sequence</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>cr</td>
<td>Carriage return sequence</td>
</tr>
<tr>
<td>cs</td>
<td>Change scrolling region</td>
</tr>
<tr>
<td>DL</td>
<td>Delete the line the cursor is on</td>
</tr>
<tr>
<td>ei</td>
<td>End insert mode</td>
</tr>
<tr>
<td>ho</td>
<td>Home, move cursor to upper left</td>
</tr>
<tr>
<td>ic</td>
<td>Character insert</td>
</tr>
<tr>
<td>im</td>
<td>Begin insert mode</td>
</tr>
<tr>
<td>is</td>
<td>Initialization string (typically tab stop initialization)</td>
</tr>
<tr>
<td>ll</td>
<td>Move cursor to lower left corner</td>
</tr>
<tr>
<td>md</td>
<td>Turn on bold (extra bright) character attribute</td>
</tr>
<tr>
<td>me</td>
<td>Turn off all character attributes</td>
</tr>
<tr>
<td>nd</td>
<td>Nondestructive space</td>
</tr>
<tr>
<td>nl</td>
<td>Newline sequence</td>
</tr>
<tr>
<td>pc</td>
<td>Pad character if not NULL</td>
</tr>
<tr>
<td>rc</td>
<td>Restore cursor position</td>
</tr>
<tr>
<td>rs</td>
<td>Resets terminal to known starting state</td>
</tr>
<tr>
<td>sc</td>
<td>Save cursor position</td>
</tr>
<tr>
<td>se</td>
<td>End standout mode (highlight)</td>
</tr>
<tr>
<td>so</td>
<td>Start standout mode (highlight)</td>
</tr>
<tr>
<td>ta</td>
<td>Tab</td>
</tr>
<tr>
<td>te</td>
<td>End programs that use cursor motion</td>
</tr>
<tr>
<td>ti</td>
<td>Initialization for programs that use cursor motion</td>
</tr>
<tr>
<td>uc</td>
<td>Underline character at cursor</td>
</tr>
<tr>
<td>ue</td>
<td>End underline mode</td>
</tr>
<tr>
<td>up</td>
<td>Move cursor up</td>
</tr>
<tr>
<td>us</td>
<td>Begin underline mode</td>
</tr>
<tr>
<td>vb</td>
<td>Visual bell</td>
</tr>
<tr>
<td>vs</td>
<td>Visual cursor</td>
</tr>
<tr>
<td>ve</td>
<td>Normal cursor</td>
</tr>
</tbody>
</table>
The ttycap database uses these color sequences to translate IBM directives into screen drawing commands. These color sequences control only foreground terminal colors. They do not control background color, which is configured to black by default.

### Examples

The following is an example of a ttycap file. Refer to the chapter “Configuring Dial-In Terminal Services” in the *Cisco IOS Terminal Services Configuration Guide* and the tn3270.examples file in the Cisco ftp@cisco.com directory for more examples.

```
ttycap ttycap1\ 
do|vt100|vt100-am|vt100am|dec vt100:do="J:co#80:li#24:\ 
c1=50\H[2J:bs:am:cm=5\i%d;%dH:nd=2\C:up=2\[A:\ 
se=3\K:so=2\7m:se=2\m:us=2\m:ue=2\m:md=2\1m:\me=2\m:ho=\H:xn:sc=\7rc=\8:cs=\i%d;dr:
```

### Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>keymap-type</td>
<td>Specifies the keyboard map for a terminal connected to the line.</td>
</tr>
<tr>
<td>terminal-type</td>
<td>Specifies the type of terminal connected to a line.</td>
</tr>
</tbody>
</table>

### Table 53  Definitions of ttycap Capabilities: Number Sequences

<table>
<thead>
<tr>
<th>Number Sequence</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>li</td>
<td>Lines on the screen</td>
</tr>
<tr>
<td>co</td>
<td>Columns on the screen</td>
</tr>
<tr>
<td>sg</td>
<td>Standout glitch, number of spaces printed when entering or leaving standout display mode</td>
</tr>
<tr>
<td>ug</td>
<td>Underline glitch, number of spaces printed when entering or leaving underline mode</td>
</tr>
</tbody>
</table>

### Table 54  Definitions of ttycap Capabilities: Color Sequences

<table>
<thead>
<tr>
<th>Color Sequence</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>x0</td>
<td>Black</td>
</tr>
<tr>
<td>x1</td>
<td>Blue</td>
</tr>
<tr>
<td>x2</td>
<td>Red or orange</td>
</tr>
<tr>
<td>x3</td>
<td>Pink or purple</td>
</tr>
<tr>
<td>x4</td>
<td>Green, which is the default color</td>
</tr>
<tr>
<td>x5</td>
<td>Turquoise</td>
</tr>
<tr>
<td>x6</td>
<td>Yellow</td>
</tr>
<tr>
<td>x7</td>
<td>Gray or white</td>
</tr>
</tbody>
</table>
txspeed

To set the terminal transmit speed (how fast the terminal sends information to the modem), use the `txspeed` command in line configuration mode. To return to the default setting, use the `no` form of this command.

```
   txspeed  bps
   no txspeed
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>bps</code></td>
<td>Baud rate, in bits per second (bps).</td>
</tr>
</tbody>
</table>

**Defaults**

9600 bps

**Command Modes**

Line configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Set the speed to match the baud rate of whatever device you have connected to the port. Some baud rates available on devices connected to the port might not be supported on the router. The Cisco IOS software will indicate if the speed you select is not supported.

**Note**

If the line was previously configured for automatic baud rate detection (autobaud), disable autobaud by entering the `no autobaud` command before entering the `txspeed` command to fix the speed of the port.

**Examples**

The following example sets the transmit speed for line 5 to 2400 bps:

```
line 5
   txspeed 2400
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>rotary-group</code></td>
<td>Sets the terminal receive speed (how fast the terminal receives information from the modem).</td>
</tr>
<tr>
<td><code>source template</code></td>
<td>Sets the flow control start character.</td>
</tr>
<tr>
<td><code>terminal txspeed</code></td>
<td>Sets the terminal transmit speed (how fast the terminal can send information) on the current line and session.</td>
</tr>
</tbody>
</table>
**where**

To list the open sessions, use the `where` command in EXEC mode.

```plaintext
where
```

**Syntax Description**

This command has no arguments or keywords.

**Command Modes**

EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced in a release prior to Cisco IOS Release 10.0.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

The `where` command displays all open sessions associated with the current terminal line.

The `Ctrl^x, where`, and `resume` commands are available with all supported connection protocols.

**Examples**

The following is sample output from the `where` command:

```plaintext
Router# where

<table>
<thead>
<tr>
<th>Conn</th>
<th>Host</th>
<th>Address</th>
<th>Byte</th>
<th>Idle</th>
<th>Conn Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>MATHOM</td>
<td>192.168.7.21</td>
<td>0</td>
<td>0</td>
<td>MATHOM</td>
</tr>
<tr>
<td>*</td>
<td>CHAFF</td>
<td>172.18.12.19</td>
<td>0</td>
<td>0</td>
<td>CHAFF</td>
</tr>
</tbody>
</table>
```

The asterisk (*) indicates the current terminal session.

**Table 55** describes the significant fields shown in the display.

**Table 55  where Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Conn</td>
<td>Name or address of the remote host to which the connection is made.</td>
</tr>
<tr>
<td>Host</td>
<td>Remote host to which the router is connected through a Telnet session.</td>
</tr>
<tr>
<td>Address</td>
<td>IP address of the remote host.</td>
</tr>
<tr>
<td>Byte</td>
<td>Number of unread bytes for the user to see on the connection.</td>
</tr>
<tr>
<td>Idle</td>
<td>Interval, in minutes, since data was last sent on the line.</td>
</tr>
<tr>
<td>Conn Name</td>
<td>Assigned name of the connection.</td>
</tr>
</tbody>
</table>

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>show sessions</code></td>
<td>Displays information about open LAT, Telnet, or rlogin connections.</td>
</tr>
</tbody>
</table>
x25 subaddress

To append either a physical port number or a value specified for a line as a subaddress to the X.121 calling address, use the x25 subaddress command in line configuration mode. To disable subaddressing, the no form of this command.

```
x25 subaddress {line | number}
no x25 subaddress {line | number}
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>line</td>
<td>Physical port number for the indicated line to be appended to the X.121 address as the subaddress.</td>
</tr>
<tr>
<td>number</td>
<td>Numeric variable assigned to a specific line.</td>
</tr>
</tbody>
</table>

**Defaults**

No default behavior or values.

**Command Modes**

Line configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>11.2 F</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the x25 subaddress line command to create a unique X.121 calling address by adding either a physical port number or a numeric value for a line as a subaddress to the X.121 calling address.

**Examples**

The following example shows how to configure subaddressing on virtual terminal lines 10 through 20 by appending the line number as a subaddress to the X.121 calling address:

```
line vty 10 20
x25 subaddress line
```

The following example shows how to configure subaddressing on the first five TTY lines by appending the value 09 as a subaddress to the X.121 calling address of an X.28 connection originating on these lines:

```
line 1 5
x25 subaddress 9
autocommand x28
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>line</td>
<td>Identifies a specific line for configuration and starts the line configuration command collection mode.</td>
</tr>
</tbody>
</table>
To enter X.28 mode and access an X.25 network or set X.3 packet assembler/disassembler (PAD) parameters, use the `x28` command in EXEC mode. To exit X.28 mode, use the `no` form of this command.

```
x28 [escape character-string] [noescape] [nuicud] [profile file-name] [reverse] [verbose]
```

```
no x28 [escape character-string] [noescape] [nuicud] [profile file-name] [reverse] [verbose]
```

### Syntax Description

- **escape character-string** *(Optional)* Specifies a character string to use to exit X.28 mode and return to EXEC mode. The character string can be any string of alphanumeric characters. The Ctrl key can be used in conjunction with the character string.

- **noescape** *(Optional)* Specifies that no escape character string is defined (user cannot return to EXEC mode). On the console line, the `noescape` option is ignored, and the default escape sequence is used (`exit` command).

- **nuicud** *(Optional)* Specifies the network user identification (NUI) data to not be placed in the NUI facility of the call request. Instead, the data is placed in the Call User Data (CUD) area of the call request packet.

- **profile file-name** *(Optional)* Specifies using a user-configured profile of X.3 parameters. A profile is created with the `x29 profile` EXEC command.

- **reverse** *(Optional)* Specifies reverse charges for outgoing calls made from the local router to the destination device.

- **verbose** *(Optional)* Displays optional service signals such as the called DTE address, facility block, and CUD.

### Defaults

Disabled. X.28 mode uses standard X.28 command syntax.

### Command Modes

EXEC

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>11.2 F</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

### Usage Guidelines

If both the `escape` and `noescape` options are not set, the default escape sequence is used (`exit` command).

X.28 mode is identified with an asterisk (*) router prompt. After you enter this mode, the standard X.28 user interface (with the exception of the escape sequence) is available. From this interface, you can configure a PAD device using X.3 parameters, or you can access an X.25 network.

In X.28 mode, you can set PAD command signals using standard or extended command syntax. For example, you can enter the `clr` command or `clear` command to clear a virtual call. A command specified with standard command syntax is merely an abbreviated version of the extended syntax version. Table 56 lists the commands available in both standard and extended command syntax.
Table 56  Available PAD Command Signals

<table>
<thead>
<tr>
<th>Standard Syntax</th>
<th>Extended Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>break</td>
<td></td>
<td>Simulate an asynchronous break.</td>
</tr>
<tr>
<td>call</td>
<td></td>
<td>Place a virtual call to a remote device.</td>
</tr>
<tr>
<td>clr</td>
<td>clear</td>
<td>Clear a virtual call.</td>
</tr>
<tr>
<td>command-signal</td>
<td></td>
<td>Specifies a call request without using a standard X.28 command, which is entered with the following syntax: facilities-x121-address:call-user-data.</td>
</tr>
<tr>
<td>help</td>
<td></td>
<td>Display help information. (See Table 58.)</td>
</tr>
<tr>
<td>iclr</td>
<td>iclear</td>
<td>Request the help interface to clear the call.</td>
</tr>
<tr>
<td>int</td>
<td>interrupt</td>
<td>Send an Interrupt Packet.</td>
</tr>
<tr>
<td>par? par</td>
<td>parameter read</td>
<td>Display the current values of local parameters. (See Table 57.)</td>
</tr>
<tr>
<td>prof</td>
<td>profile file-name</td>
<td>Load a standard or a named profile.</td>
</tr>
<tr>
<td>reset</td>
<td></td>
<td>Reset the call.</td>
</tr>
<tr>
<td>rpar? rpar?</td>
<td>rread</td>
<td>Display the current values of remote parameters.</td>
</tr>
<tr>
<td>rset? rset?</td>
<td>rsetread</td>
<td>Set and then read the values of remote parameters.</td>
</tr>
<tr>
<td>set</td>
<td>setread</td>
<td>Change the values of local parameters. (See Table 57.)</td>
</tr>
<tr>
<td>stat</td>
<td>status</td>
<td>Request the status of a connection.</td>
</tr>
<tr>
<td>selection pad</td>
<td></td>
<td>Set up a virtual call.</td>
</tr>
</tbody>
</table>

1. This is an example of issuing a call request command: the \texttt{R,G23,P2-234234Duser1} command.

Table 57 lists the different types of parameters you can set using the \texttt{set parameter-number: new-value} PAD command signal from X.28 mode. Refer to the “X.3 PAD Parameters” appendix in the \textit{Cisco IOS Terminal Services Configuration Guide}, Release 12.2 for more complete information about these parameters. See Table 36 in this publication for a list of ASCII characters.

Table 57  Supported X.3 PAD Parameters

<table>
<thead>
<tr>
<th>Parameter Number</th>
<th>ITU-T Parameter Name</th>
<th>ITU-T X.3 and Cisco Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>PAD recall using a character</td>
<td>Minimum value: 0; maximum value: 126; X.28 PAD user emulation mode default: 1. Note Not supported by PAD EXEC user interface.</td>
</tr>
<tr>
<td>2</td>
<td>Echo</td>
<td>Minimum value: 0; maximum value: 1; PAD EXEC mode and X.28 PAD user emulation mode defaults: 1.</td>
</tr>
<tr>
<td>3</td>
<td>Selection of data forwarding character</td>
<td>Minimum value: 0; maximum value: 255; PAD EXEC mode default: 2 (CR); X.28 PAD user emulation mode default: 126 (~).</td>
</tr>
<tr>
<td>Parameter Number</td>
<td>ITU-T Parameter Name</td>
<td>ITU-T X.3 and Cisco Values</td>
</tr>
<tr>
<td>------------------</td>
<td>--------------------------------------------------</td>
<td>------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>4</td>
<td>Selection of idle timer delay</td>
<td>Minimum value: 0; maximum value: 255; PAD EXEC mode default: 1; X.28 PAD user emulation mode default: 0.</td>
</tr>
<tr>
<td>5</td>
<td>Ancillary device control</td>
<td>Minimum value: 0; maximum value: 2; PAD EXEC mode default: 0; X.28 PAD user emulation mode default: 1.</td>
</tr>
<tr>
<td>6</td>
<td>Control of PAD service signals</td>
<td>Minimum value: 0; maximum value: 255; PAD EXEC mode default: 0; X.28 PAD user emulation mode default: 2.</td>
</tr>
<tr>
<td></td>
<td>Note</td>
<td>Not supported by PAD EXEC user interface.</td>
</tr>
<tr>
<td>7</td>
<td>Action upon receipt of a BREAK signal</td>
<td>Minimum value: 0; maximum value: 31; PAD EXEC mode default: 4; X.28 PAD user emulation mode default: 2.</td>
</tr>
<tr>
<td>8</td>
<td>Discard output</td>
<td>Minimum value: 0; maximum value: 1; PAD EXEC mode and X.28 PAD user emulation mode default: 0.</td>
</tr>
<tr>
<td>9</td>
<td>Padding after Return</td>
<td>Minimum value: 0; maximum value: 255; PAD EXEC mode and X.28 PAD user emulation mode default: 0.</td>
</tr>
<tr>
<td>10</td>
<td>Line folding</td>
<td>Not supported.</td>
</tr>
<tr>
<td>11</td>
<td>DTE speed (binary speed of start-stop mode DTE)</td>
<td>Minimum value: 0; maximum value: 18; PAD EXEC mode and X.28 PAD user emulation mode default: 14.</td>
</tr>
<tr>
<td>12</td>
<td>Flow control of the PAD by the start-stop DTE</td>
<td>Minimum value: 0; maximum value: 1; PAD EXEC mode default: 0; X.28 PAD user emulation mode default: 1.</td>
</tr>
<tr>
<td>13</td>
<td>Line feed insertion (after a Return)</td>
<td>Minimum value: 0; maximum value: 7; PAD EXEC mode and X.28 PAD user emulation mode default: 0.</td>
</tr>
<tr>
<td>14</td>
<td>Line feed padding</td>
<td>Minimum value: 0; maximum value: 255; PAD EXEC mode and X.28 PAD user emulation mode default: 0.</td>
</tr>
<tr>
<td>15</td>
<td>Editing</td>
<td>Minimum value: 0; maximum value: 1; PAD EXEC mode and X.28 PAD user emulation mode default: 0.</td>
</tr>
<tr>
<td>16</td>
<td>Character delete</td>
<td>Minimum value: 0; maximum value: 127; PAD EXEC mode and X.28 PAD user emulation mode default: 127 (DEL).</td>
</tr>
<tr>
<td>17</td>
<td>Line delete</td>
<td>Minimum value: 0; maximum value: 127; PAD EXEC mode default: 21 (Ctrl-U); X.28 PAD user emulation mode default: 24 (Ctrl-X).</td>
</tr>
<tr>
<td>18</td>
<td>Line display</td>
<td>Minimum value: 0; maximum value: 127; PAD EXEC mode and X.28 PAD user emulation mode default: 18 (Ctrl-R).</td>
</tr>
<tr>
<td>19</td>
<td>Editing PAD service signals</td>
<td>Minimum value: 0; maximum value: 126; PAD EXEC mode default: 0; X.28 PAD user emulation mode default: 2.</td>
</tr>
</tbody>
</table>

Not supported by PAD EXEC user interface.
Table 58 lists the options for the X.28 help command.

### Table 58 X.28 help Options

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>help</td>
<td>Describes the help PAD command.</td>
</tr>
<tr>
<td>help command</td>
<td>Displays the list of available PAD command signals.</td>
</tr>
<tr>
<td>help parameter</td>
<td>Displays the list of available X.3 PAD parameters.</td>
</tr>
<tr>
<td>help parameter number</td>
<td>Displays the specified X.3 PAD parameter and its current value.</td>
</tr>
<tr>
<td>help list</td>
<td>Lists the available help subjects.</td>
</tr>
<tr>
<td>help profiles</td>
<td>Lists available profiles.</td>
</tr>
<tr>
<td>help profile name</td>
<td>Displays the specified parameter name and current value.</td>
</tr>
<tr>
<td>help any-PAD-command</td>
<td>Describes the specified PAD command signal.</td>
</tr>
</tbody>
</table>

You can issue call requests from X.28 mode without using standard X.28 commands by using the following command syntax:

```
facilities-x121-address D call-user-data
```
where:

- **facilities** Applies X.25 facilities to the outgoing call. The hyphen is mandatory.
- **x121-address** Specifies the address of the remote X.25 device.
- **D** Facility request code that specifies CUD for the outgoing call.
- **call-user-data** Specifies the data that accompanies the call request packet sent to the remote X.25 device.

The following rules apply to all call requests parsed in X.28 mode:

- When an X.121 address specified using standard command syntax is followed by an optional CUD field, the call is placed to the X.121 address.
- When standard command syntax is used, one or more facility request codes can be entered, followed by the code value. Additional facility request codes and values also can be entered. Separate each entry with a comma, followed by a dash. An X.121 address and optional CUD can follow this entry.
- If an X.28 command is not entered, a call request is assumed.
- Ensure that the call request begins with a facility code letter, and that it contains a hyphen (-) followed by a string of digits (the X.121 address). The call request can be terminated by an asterisk (*), a “P,” or a “D,” followed by some data.
- When using extended command syntax is used, the call command uses the facility codes and X.121 address as its operand.
- If facility codes are entered without an X.121 address, remember the codes for the next call. When a call is completed, forget the facility codes until they are once again set.

Table 59 shows examples of parsed call requests.

**Table 59 Example X.28 Call Requests**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>123456789</td>
<td>Calls this X.121 address.</td>
</tr>
<tr>
<td>123456789*userdata</td>
<td>Calls this X.121 address, with specified data.</td>
</tr>
<tr>
<td>123456789Puserdata</td>
<td>Calls this X.121 address, with specified data.</td>
</tr>
<tr>
<td>123456789Ruserdata</td>
<td>Calls this X.121 address, with specified data.</td>
</tr>
<tr>
<td>Nabcd-123456789</td>
<td>Calls this X.121 address, with NUI set to abcd.</td>
</tr>
<tr>
<td>Nabcd.R-123456789</td>
<td>Calls 123456789 with NUI of abcd, and with reverse charging.</td>
</tr>
</tbody>
</table>

**Examples**

The following example uses the `?` command to display the optional X.28 keywords:

```
Router# x28 ?
```

- `debug` Turn on Debug Messages for X28 Mode
- `escape` Set the string to escape from X28 PAD mode
- `noescape` Never exit x28 mode (use with caution)
- `nuicud` All calls with NUI, are normal charge with the NUI placed in Call User Data
profile Use a defined X.3 Profile
reverse All calls default to reverse charge
verbose Turn on Verbose Messages for X28 Mode
<cr>

After you are in X.28 mode, use the call PAD signal command to place a virtual call:

Router# x28
* call 123456

The following example enters X.28 mode with the x28 EXEC command and configures a PAD with the set X.3 parameter command. The set command sets the idle time delay to 40 seconds.

Router# x28
* set 4:40

### Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>pad</td>
<td>Logs in to a PAD.</td>
</tr>
</tbody>
</table>
To set X.3 packet assembler/disassembler (PAD) parameters, use the x3 command in EXEC mode.

```
x3 parameter:value
```

### Syntax Description

| parameter:value | Sets the PAD parameters. (See Table 57 in the x28 command description.) |

### Defaults

For outgoing connections, the X.3 parameters default to the following:

- 2:1, 3:2, 4:1, 7:4, 16:127, 17:21, 18:19

All other parameters default to zero, but can be changed using the /set switch keyword with either the resume command or the x3 command.

For incoming PAD connections, the software sends an X.29 SET PARAMETER packet to set only the following parameters:

- 2:0, 4:1, 7:21, 15:0

For a complete description of the X.3 PAD parameters, see the appendix titled “X.3 PAD Parameters” in the *Cisco IOS Terminal Services Configuration Guide*.

### Command Modes

- EXEC

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>11.2</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

### Usage Guidelines

You can have several PAD connections open at the same time and switch between them. You can also exit a connection and return to the user EXEC prompt at any point.

To open a new connection, first suspend the current connection by pressing the escape sequence ([Ctrl]-[Shift]-6 then x [Ctrl^x]) by default) to return to the system command prompt, then open the new connection with the pad command.

You can have several concurrent sessions open and switch between them. The number of PAD sessions that can be open is defined by the session-limit command.

To switch between sessions you must escape one session and resume a previously opened session. Use the Ctrl^x sequence to escape out of a connection, use the where EXEC command to check the connection number, and then use the resume command with the connection number to resume the suspended connection. These commands are available with all supported connection protocols.

You can issue any of the following commands to terminate a terminal session:

- exit
- quit
- logout
To display information about packet transmission and X.3 PAD parameter settings, use the **show x25 pad** command.

### Examples

The following example shows how to change a local X.3 PAD parameter from a remote X.25 host using X.29 messages, which is a secure way to enable a remote host to gain control of local PAD. The local device is named Router-A. The remote host is named Router-B. The parameters listed in the ParamsIn field are incoming parameters, which are sent by the remote PAD. The parameters listed in the ParamsOut field are parameters sent by the local PAD.

```
Router-A# pad 123456
Trying 123456...Open

Router-B> x3 2:0
Router-B>
Router-A# show x25 pad

tty0, connection 1 to host 123456
Total input: 12, control 3, bytes 35. Queued: 0 of 7 (0 bytes).
Total output: 10, control 3, bytes 64.
Flags: 1, State: 3, Last error: 1
ParamsIn: 1:0, 2:0, 3:0, 4:0, 5:0, 6:0, 7:0,
          8:0, 9:0, 10:0, 11:0, 12:0, 13:0, 14:0, 15:0,
          16:0, 17:0, 18:0, 19:0, 20:0, 21:0, 22:0,
ParamsOut: 1:1, 2:0, 3:2, 4:1, 5:1, 6:0, 7:21,
          8:0, 9:1, 10:0, 11:14, 12:1, 13:0, 14:0, 15:0,
          16:127, 17:21, 18:18, 19:0, 20:0, 21:0, 22:0,
Router-A#
```

### Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>resume (X.3 PAD)</strong></td>
<td>Sets X.3 parameters for PAD connections.</td>
</tr>
</tbody>
</table>
xremote

To prepare the router for manual startup and initiate an XRemote connection, use the `xremote` command in EXEC mode. This command begins the instructions that prompt you through the connection.

```
xremote
```

**Syntax Description**
This command has no arguments or keywords.

**Defaults**
Disabled

**Command Modes**
EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>11.1</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**
If you do not use a host computer that supports XDMCP or LAT, you must use manual session startup. Manual session startup involves the following steps:

**Step 1**
Enable XRemote manually on the router port.

**Step 2**
Connect to the host computer by using a `telnet`, `lat`, or `rlogin` command, then log in as usual.

**Step 3**
Set the location of the X display.

**Step 4**
Start client applications.

**Step 5**
Return to the EXEC prompt.

**Step 6**
Enter the `xremote` command to enable XRemote manually again on the server port.

**Note**
In manual operation, the server and X terminal remain in XRemote mode until all clients disconnect or the access server receives a reset request from the X terminal. A session might terminate during startup because you invoked transient X clients that set some parameters (such as `xset` or `xmodmap`) and then disconnected. One session must always be open or the connection resets.

Refer to the *Cisco IOS Terminal Services Configuration Guide* for more information about how to establish XRemote sessions between servers.
Examples

The following example starts a manual XRemote session:

dialup> xremote
XRemote enabled; your display is dialup:2006
Start your clients and type XRemote again

The router replies with a message informing you of your X display location. Use this information to tell the XRemote host the location of your X display server. If no clients are found, you see the following message:

No X clients waiting - check that your display is darkstar:2006

The following example shows a connection from an X display terminal through a router to a host running client programs:

dialup> xremote
XRemote enabled; your display is dialup:2006
Start your clients and type XRemote again

dialup> telnet eureka
Trying EUREKA.NOWHERE.COM (722.18.1.55).... Open

SunOS UNIX (eureka)

login: deal
Password:

Last login: Fri Apr 1 17:17:46 from dialup.nowhere.com
SunOS Release (SERVER+FDDI+DBE.patched) #14: Fri Apr 8 10:37:29 PDT 1994

eureka% setenv DISPLAY dialup:2006
eureka% xterm &
[1] 15439

eureka% logout

[Connection to EUREKA closed by foreign host]

dialup> xremote
Entering XRemote

The following procedure shows how an XRemote connection is established for a configuration like the one shown in Figure 2. This example assumes that the administrator has set the display environment variable for the user to identify the X display terminal.

Step 1 From the PCX, MacX, or UNIX machine in Figure 2, the user connects to port 9003 on access server 1. If your administrator has configured a rotary number 7, the user connects to port 10007. For more information about rotary groups, refer to the Cisco IOS Dial Technologies Configuration Guide.

Step 2 Access server 1 connects the user to a modem.

Step 3 The modem calls access server 2.

Step 4 The user enters xremote at the access server 2 prompt.

Step 5 The user connects to the host from access server 2 using the telnet command.
Step 6
The user starts the X client program that will run on the host and display on the X display server (PCX, MacX, or UNIX host).

Step 7
The user escapes from the host back to the AccessServer2, or logs out if clients were run in the background, and enters the `xremote` command at the AccessServer2 prompt.

You can use the master indexes or search online to find documentation of related commands.

---

**Figure 2 XRemote Session Between Servers**

The following example shows how to make an XRemote connection between servers. The number 9016 in the first line of the display indicates a connection to individual line 16. If the administrator had configured a rotary connection, the user would enter 10000 plus the number of the rotary instead of 9016.

Router% `telnet golden-road 9016`

Trying 192.168.7.84 ... 
Connected to golden-road.cisco.com.
Escape character is `^]`.

User Access Verification

Password:
Password OK
--- Outbound XRemote service ---
Enter X server name or IP address: innerspace
Enter display number [0]:

Connecting to tty16... please start up XRemote on the remote system

atdt 13125554141
DIALING
RING
CONNECT 14400

User Access Verification
Username: deal
Password: Welcome to the cisco dial-up access server.
dialup> xremote
XRemote enabled; your display is dialup:2006
Start your clients and type XRemote again
dialup> telnet sparks
Trying SPARKS.NOWHERE.COM (172.18.1.55)... Open

SunOS UNIX (sparks)
login: deal
Password: Last login: Fri Apr 1 17:17:46 from dialup.nowhere.com
SunOS Release (SERVER+FDDI+DBE.patched) #14: Fri Apr 8 10:37:29 PDT 1994

sparks% setenv DISPLAY dialup:2006
sparks% xterm &
[1] 15439
sparks% logout
[Connection to SPARKS closed by foreign host]
dialup> xremote
Entering XRemote

Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>xremote lat</td>
<td>Initiates a DECwindow session over a LAT connection.</td>
</tr>
<tr>
<td>xremote xdm</td>
<td>Activates automatic session startup for an XRemote connection.</td>
</tr>
</tbody>
</table>
xremote lat

To initiate a DECwindow session over a local-area transport (LAT) connection, use the xremote lat command in EXEC mode.

```
xremote lat service
```

**Syntax Description**

| service | Name of the desired LAT service. |

**Command Modes**

EXEC

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>11.2</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

If your host computer supports DECwindows login sessions, you can use automatic session startup to make an XRemote session connection. Once the system administrator at the remote host configures support for DECwindows over LAT, use the xremote lat EXEC command to initiate the connection. After you issue this command, the following events occur:

- The XRemote font server down-line loads several initial fonts for the DECwindows login display.
- The terminal displays the DIGITAL logo and DECwindows login box.

Log in to the host. Upon completion of login, more fonts are loaded, and the remote session begins.

**Note**

Because of heavy font usage, DECwindows applications can take longer than expected to start when XRemote is used. Once the application starts, performance and access times should be as expected.

To exit XRemote sessions, you must quit all active X connections, usually with a command supported by your X client system. Usually when you quit the last connection (when all client processes are stopped), XRemote closes and you return to the EXEC prompt. However, your X client system determines how the session closes.

**Examples**

The following example begins connection with a LAT service named service1:

```
xremote lat service1
```

**Related Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>xremote</td>
<td>Prepares the router for manual startup and initiates an XRemote connection.</td>
</tr>
<tr>
<td>xremote xdm</td>
<td>Activates automatic session startup for an XRemote connection.</td>
</tr>
</tbody>
</table>
xremote tftp buffersize

To change the buffer size used for loading font files, use the `xremote tftp buffersize` command in global configuration mode. To restore the buffer size to the default value, use the `no` form of this command.

```
xremote tftp buffersize buffersize
no xremote tftp buffersize
```

**Syntax Description**

- `buffersize`  Buffer size in bytes. This is a decimal number in the range from 4096 to 70000 bytes.

**Defaults**

- 70000 bytes

**Command Modes**

- Global configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

When the X terminal requests that a font file be loaded, the Cisco IOS software must first load the font file into an internal buffer before passing it to the X terminal. The default value of 70000 bytes is adequate for most font files, but the size can be increased as necessary for nonstandard font files.

The buffer size can be set as low as 4096 bytes and as large as the available memory on the router will allow. If you are using local-area transport (LAT) font access, you should not lower the buffer size below the default, because the font directory for all of the LAT fonts (created internally) requires 70000 bytes.

This command applies to both TFTP and LAT font access.

**Examples**

The following example sets the buffer size to 20000 bytes:

```
xremote tftp buffersize 20000
```
**xremote tftp host**

To add a specific Trivial File Transfer Protocol (TFTP) font server as a source of fonts for the terminal, use the `xremote tftp host` command in global configuration mode. To remove a font server from the list, use the `no` form of this command.

```
xremote tftp host host-name
```

```
no xremote tftp host host-name
```

**Syntax Description**

- `host-name` IP address or name of the host containing fonts.

**Defaults**

No TFTP font server is specified.

**Command Modes**

Global configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Each time a new host name is entered, the list in the Cisco IOS software is updated. Font servers are queried in the order of their definition when the X terminal requests a font.

**Examples**

The following example sets the host named IBM-1 as an XRemote TFTP font server:
```
xremote tftp host IBM-1
```

The following example sets the host with IP address 10.0.0.7 as an XRemote TFTP font server:
```
xremote tftp host 10.0.0.7
```
xremote tftp retries

To specify the number of retries the font loader will attempt before declaring an error condition, use the `xremote tftp retries` command in global configuration mode. To restore the default retries number, use the `no` form of this command.

```
xremote tftp retries retries
no xremote tftp retries
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>retries</code></td>
<td>(Optional) Number of retries. Acceptable values are decimal numbers in the range from 1 to 15.</td>
</tr>
</tbody>
</table>

**Defaults**

3 retries

**Command Modes**

Global configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Under certain conditions, you might need to increase the number of retries, particularly if the font servers are known to be heavily loaded.

**Examples**

The following example sets the number of font loader retries to 5:

```
xremote tftp retries 5
```
xremote xdm

To activate automatic session startup for an XRemote connection, use the `xremote xdm` command in EXEC mode.

```
xremote xdm [host-name]
```

Syntax Description

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>host-name</td>
<td>(Optional) Host computer name or IP address.</td>
</tr>
</tbody>
</table>

Command Modes

EXEC

Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>11.2</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

Usage Guidelines

If your host computer supports a server running X Display Manager Control Protocol (XDMCP) (such as the xdm program included in X11R4 or later), you can use automatic session startup to make an XRemote session connection using the `xremote xdm` EXEC command.

This command sends an XDMCP session startup request to the host computer. If you do not specify a host name or IP address, a broadcast message is sent to all hosts. The first host to respond by starting up a session is used.

The XRemote (the host) server and X terminal stay in XRemote mode until either the display manager terminates the session or the XRemote server receives a reset request from the X terminal.

To exit XRemote sessions, you must quit all active X connections, usually with a command supported by your X client system. Usually when you quit the last connection (all client processes are stopped), XRemote closes and you return to the EXEC prompt. However, your remote X client system determines how the session closes.

To terminate a session, disconnect from the device on the network using the command specific to that device. Then exit from the EXEC by using the `exit` command.

Examples

The following example starts a session with a remote host named host1:

```
xremote xdm host1
```

Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>xremote</td>
<td>Prepares the router for manual startup and initiates an XRemote connection.</td>
</tr>
<tr>
<td>xremote lat</td>
<td>Initiates a DECwindow session over a LAT connection.</td>
</tr>
</tbody>
</table>
Symbols

<cr> xv
? command xiv

A
absolute-timeout command TR-2
access-class (LAT) command TR-3
access control
  ARA TR-7, TR-9, TR-14
  LAT
    access conditions TR-39
    connections TR-3
    group list TR-47
    group list name assignments TR-42
ARA (AppleTalk Remote Access)
  access control TR-7, TR-9, TR-14
  monitoring TR-92
arap dedicated command TR-5
arap enable command TR-6
arap net-access-list command TR-7
arap network command TR-9
arap noguest command TR-10
arap require-manual-password command TR-11
arap timelimit command TR-12
arap warningtime command TR-13
arap zonelist command TR-14
async default ip address command
  See peer default ip address command
  authentication translate option
    virtual interface template, mapping to TR-171, TR-196
  autocommand command TR-16

B
baud rates
  transmit line speed, configuring TR-208
buffers
  XRemote size TR-224
busy-message command TR-17

callback, enabling TR-85
carriage return (<cr>) xv
cautions, usage in text x
changed information in this release ix
character mapping
  configuring TR-153
  displaying TN3270 TR-121
chat scripts, SLIP and PPP backward compatibility TR-86
Cisco IOS configuration changes, saving xviii
clear entry command TR-18
command modes, understanding xiii to xiv
commands
  context-sensitive help for abbreviating xiv
default form, using xvii
  no form, using xvii
command syntax
  conventions ix
displaying (example) xv
configurations, saving xviii
connect command TR-19, TR-37, TR-137
classifications
  display open TR-209
  LAT TR-37, TR-38
NASI client TR-27
notification of pending output TR-76
queued host-initiated entries, clearing TR-18
refuse full duplex TR-142
rlogin TR-78, TR-80, TR-135 to TR-138
SLIP TR-133
switching between TR-74, TR-77
Telnet TR-135 to TR-138
   line, configuring TR-143
TN3270 TR-150
transparent, local TR-19, TR-78, TR-135
X.3 PAD TR-217
XRemote TR-219, TR-223, TR-227
console port
   configuration (example) TR-63

D
DEC LAT
   See LAT
disconnect warning time, configuring TR-13
dispatch-timeout command TR-120
documentation
   conventions ix
   feedback, providing xi
   modules v to vii
   online, accessing x
   ordering xi
Documentation CD-ROM x
documents and resources, supporting viii

E
escape character
   EXEC prompt TR-219
exec-timeout command TR-120

F
Feature Navigator
   See platforms, supported
filtering output, show and more commands xviii
flush-at-activation command TR-24

G
global configuration mode, summary of xiv
guests
   allowing TR-10
   disallowing TR-10

H
hardware platforms
   See platforms, supported
header compressions
   virtual interface template, translate map to TR-171, TR-196
help command xiv
host-failed message, configuring TR-17

I
indexes, master viii
interface configuration mode, summary of xiv
IP addresses
   TCP port assignment TR-26
ip alias command TR-26
ip-pool translate option
   map to virtual interface template TR-171, TR-196
IPX (Internet Packet Exchange)
   loopback interfaces TR-166, TR-167, TR-176, TR-177, TR-187, TR-188
   loopback translate option
   virtual interface template, mapping to TR-171, TR-196
ipx nasi-server enable command TR-27
K

keepalive translate option
  virtual interface template, mapping to TR-171, TR-196
keyboard mapping characteristics TR-29
keymap command TR-29
keymaps
  custom TR-31
  default TR-31
  special characters TR-30
  syntax TR-31
keymap-type command TR-36
keys, Break TR-69

L

LAT (local-area transport)
  access control
    access conditions TR-39
    group list TR-47
    group list name assignments TR-42
    on connections TR-3
  active sessions TR-101
  associate rotary group with a service TR-58
  available services list TR-37, TR-100
  connection queue entry, clearing TR-18
  connections TR-37 to TR-38
  description TR-37
  enabling TR-41
  group code mask, configuring TR-53
  inbound connections, enabling TR-52
  keepalive timer, configuring TR-45
  learned services TR-100, TR-115
  list of services, defining TR-146
  maximum number of sessions, configuring TR-60
  multiple connections TR-37
  node name TR-46
  password for a service, configuring TR-55
  pattern matching symbols (table) TR-39
  proxy node, configuring TR-57
  receive buffers
    host TR-44
    server TR-50
  remotely modifiable line, configuring TR-147
  service advertisement timer TR-59
  service announcements
    configuring TR-51
    reenabling TR-51
  service identification, configuring TR-54
  static service rating, configuring TR-56
  transmission interval, configuring TR-61
  lat access-list command TR-39
  lat command TR-37
  lat enabled command TR-41
  lat group-list command TR-42
  lat host-buffers command TR-44
  lat ka-timer command TR-45
  lat node command TR-46
  lat out-group command TR-47
  lat remote-modification command TR-48
  lat retransmit-limit command TR-49
  lat server-buffers command TR-50
  lat service-announcements command TR-51
  lat service enabled command TR-51
  lat service-group command TR-53
  lat service ident command TR-54
  lat service password command TR-55
  lat service rating command TR-56
  lat service-responder command TR-57, TR-59
  lat service rotary command TR-58
  lat service-timer command TR-59
  lat vc-sessions command TR-60
  lat vc-timer command TR-61
  line command TR-62
  lines
    ARA connections, configuring TR-5, TR-6
    auxiliary port absolute line number TR-63
    characteristics, configuring TR-48
console port  TR-63
dedicated ARA-only, configuring  TR-5
line configuration mode  TR-61, TR-62
relative line number  TR-62
session limits  TR-89
status  TR-126
types
  (table)  TR-63
  specifying  TR-62
virtual terminal  TR-64
logging in
  Break key  TR-69
  username, configuring  TR-65
login (EXEC) command  TR-65
login (line) command  TR-67
login-string command  TR-69

M
messages
  failed connection  TR-17
log in  TR-17, TR-69
  successful connection  TR-17, TR-69
MIB, descriptions online  viii
modem answer-timeout command  TR-120
modes
  See command modes
mtu translate option
  virtual interface template, mapping to  TR-171, TR-196

N
NASI (NetWare Access Server Interface)
  clients, enabling  TR-27
TTY lines, connecting through  TR-27
  virtual terminal lines, connecting through  TR-27
new information in this release  ix
node name
LAT, changing  TR-46
notes, usage in text  x

P
PAD (packet assembler/disassembler)
  log in  TR-70
  parameters
    (table)  TR-73
  parameters, configuring  TR-217
  parameters, setting  TR-72 to TR-74
  See also  X.3 PAD
pad command  TR-70
password command  TR-72
passwords
  ARAP  TR-11
  line assignment  TR-72
platforms, supported
  Feature Navigator, identify using  xix
  release notes, identify using  xix
ppp use-tacacs translate option
  virtual interface template, mapping to  TR-171, TR-196
privileged EXEC mode, summary of  xiv
prompts, system  xiv
protocols, terminal transport  TR-148
protocol translations
  ARA outgoing connection  TR-163, TR-174, TR-184
    (example)  TR-192
  global options
    LAT  TR-164, TR-170
    LAT virtual access interface  TR-170
    TCP  TR-181
    TCP virtual access interface  TR-181
    X.25  TR-185
    x.25 virtual access interface  TR-181
    LAT  TR-163
    LAT options, displaying  TR-164
    LAT virtual access interface  TR-170
    outgoing options
LAT-to-TCP  TR-167
LAT-to-X.25  TR-168
TCP-to-LAT  TR-176
TCP-to-PPP  TR-176
TCP-to-SLIP  TR-177
TCP-to-X.25  TR-178
X.25-to-LAT  TR-186
X.25-to-PPP  TR-187
X.25-to-SLIP  TR-188
X.25-to-TCP  TR-188
TCP  TR-173
TCP options, displaying  TR-175
TCP virtual access interface  TR-180
virtual interface template, mapping outgoing options to  TR-171, TR-196
X.25  TR-183
X.25 connections, swapping  TR-174, TR-185
X.25 interface assignments  TR-191
X.25 options, displaying  TR-185
X.25 PVCs
   configuring  TR-189
   removing  TR-190
X.25 virtual access interface  TR-194
proxy, configuring for LAT  TR-57

RFC
   full text, obtaining viii
RFC 1144, TCP/IP header compression  TR-133
rlogin
   connections, local transparent  TR-78
   login string  TR-69
   server connections  TR-82
   username authentication  TR-80
rlogin command  TR-78
rlogin trusted-localuser-source command  TR-80
rlogin trusted-remoteuser-source local command  TR-82
ROM monitor mode, summary of xiv
routing translate option
   virtual interface template, mapping to  TR-171, TR-196
rxspeed command  TR-84

S
server connections
   LAT  TR-37 to TR-38
   rlogin  TR-80, TR-138
   SLIP  TR-133 to TR-134
   Telnet  TR-135, TR-138
   TN3270  TR-150
   transparent, local  TR-135
   XRemote  TR-219, TR-223, TR-227

service advertisement timer, configuring  TR-59
service announcements, reenabling LAT  TR-51
service exec-callback command  TR-85
service old-slip-prompts command  TR-86
service pt-vty-logging command  TR-87
session-limit command  TR-89
session-timeout command  TR-90
setenv command  TR-220, TR-222
show arap command  TR-92
show entry command  TR-94
show keymap command  TR-95
show lat advertised command  TR-96
show lat groups command  TR-97

Q
question mark (?) command xiv

R
receive buffers
   LAT host  TR-44
   LAT server  TR-50
release notes
   See platforms, supported
resume (setting X.3 parameters) command  TR-72
resume (switching sessions) command  TR-76
retransmission limit, LAT  TR-49
show lat nodes command TR-98
show lat services command TR-100
show lat sessions command TR-101
show lat traffic command TR-105
show line command TR-107
show node command TR-112
show service command TR-115
show terminal command TR-117
show tn3270 ascii-hexval command TR-121
show tn3270 character-map command TR-122
show translate command TR-123
show ttycap command TR-125
show users command TR-126
show x25 pad command TR-128
show xremote command TR-130
show xremote line command TR-132
signals, Break generating TR-140
SLIP (Serial Line Internet Protocol), server connections TR-133, TR-134
slip command TR-133

special commands (table) TR-137
 Suppress Go Ahead option TR-142
telnet break-on-ip command TR-140
telnet command TR-135
telnet refuse-negotiations command TR-142
telnet speed command TR-143
telnet sync-on-break command TR-144
telnet transparent command TR-145
terminal lat out-group command TR-146
terminal lat remote-modification command TR-147
terminals
 baud rates, configuring TR-208
 session limits, setting TR-89
 session timeout interval, setting TR-90
 transport protocol TR-148
terminal transport preferred command TR-148
time limits
 sessions, configuring TR-12
 timeouts
 absolute TR-2
 session, setting TR-90
timers
 service advertisement, configuring TR-59
TN3270
 character mapping, configuring TR-153
 character mapping, monitoring TR-121, TR-122
data character bits, setting TR-151, TR-152
default keymaps TR-31
description TR-150
extended feature mode TR-157
international characters TR-153
 monitoring TR-121
null processing handling TR-158
require reset after user error TR-160
server connections TR-150
special characters TR-30
tn3270 8bit display command TR-151	nn3270 8bit transparent-mode command TR-152
tn3270 character-map command TR-153

Tab key, command completion xiv
TCP/IP
 rlogin connections TR-78, TR-80, TR-82
 Telnet connections TR-135
 transparent connections TR-78, TR-135
Telnet
 Break signal TR-140
 connections TR-135 to TR-138
 line, configuring TR-143
 transparent, local TR-135
description TR-137
 login string TR-69
 notification of pending output TR-76
 protocol translation TR-177, TR-188
 Remote Echo option TR-142

TN3270
 character mapping, configuring TR-153
 character mapping, monitoring TR-121, TR-122
data character bits, setting TR-151, TR-152
default keymaps TR-31
description TR-150
extended feature mode TR-157
international characters TR-153
 monitoring TR-121
null processing handling TR-158
require reset after user error TR-160
server connections TR-150
special characters TR-30
tn3270 8bit display command TR-151	nn3270 8bit transparent-mode command TR-152
tn3270 character-map command TR-153

Tab key, command completion xiv
TCP/IP
 rlogin connections TR-78, TR-80, TR-82
 Telnet connections TR-135
 transparent connections TR-78, TR-135
Telnet
 Break signal TR-140
 connections TR-135 to TR-138
 line, configuring TR-143
 transparent, local TR-135
description TR-137
 login string TR-69
 notification of pending output TR-76
 protocol translation TR-177, TR-188
 Remote Echo option TR-142
tn3270 command  TR-150
tn3270 datastream command  TR-157
tn3270 null-processing command  TR-158
tn3270 optimize-cursor-move command  TR-159
tn3270 reset-required command  TR-160
tn3270 status-message command  TR-161
tn3270 typeahead command  TR-162
translate lat (virtual access interfaces) command  TR-170
translate lat command  TR-163
translate tcp (virtual access interfaces) command  TR-180
translate tcp command  TR-173
translate x25 (virtual access interfaces) command  TR-194
translate x25 command  TR-183
translation options
LAT  TR-163, TR-170, TR-176, TR-186
LAT service advertisements, preventing  TR-176, TR-186
TCP  TR-173
TCP/IP
  binary mode  TR-180
  incoming address  TR-180
Telnet binary mode  TR-173
virtual interface template, mapping outgoing options to  TR-171, TR-196
X.25  TR-183
  outgoing connections, swapping  TR-195
  (example)  TR-191
transport input command  TR-198
transport output command  TR-200
transport preferred command  TR-202
ttycap command  TR-204
ttycap support  TR-205
txspeed command  TR-208

V

virtual interface template
  LAT protocol translation on  TR-170, TR-195
  TCP protocol translation on  TR-180
virtual terminal lines
  configuring  TR-64
  loopback interfaces for translation  TR-166, TR-167, TR-176, TR-177, TR-187, TR-188

W

where command  TR-209

X

X.25
  pad connections display  TR-128
X.29, SET PARAMETER packet  TR-72, TR-217
X.3 PAD
  connections  TR-217
  description  TR-217
X.3 PAD parameters
  change dynamically  TR-72
  default values  TR-72, TR-217
  setting  TR-217
  settings display  TR-128
  X.28 mode  TR-211
x25 subaddress command  TR-210
x28 command  TR-211
x3 command  TR-217
XDMCP (X Display Manager Control Protocol)
  XRemote, starting  TR-227
XRemote
  automatic session startup  TR-223
  buffer size, configuring  TR-224
  clients, starting  TR-219
  connections, monitoring  TR-132
  description  TR-219, TR-223, TR-227

U

user EXEC mode, summary of  xiv
EXEC prompt  TR-219
font server identification  TR-225
host computer connections  TR-219
manual session startup  TR-219
reenabling manually  TR-219
server connections  TR-219
server sessions  TR-221
terminal, configuring  TR-220
terminal line statistics  TR-132
terminate session  TR-219
X display location, configuring  TR-219
xremote command  TR-219
xremote lat command  TR-223
xremote tftp buffersize command  TR-224
xremote tftp host command  TR-225
xremote tftp retries command  TR-226
xremote xdm command  TR-227
X Window System
display location, configuring  TR-219