

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

<i>incoming-service-name</i>	A LAT service name. When used on the incoming portion of the command, <i>incoming-service-name</i> 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.
<i>incoming-options</i>	(Optional) An incoming connection request option. For LAT, the only option currently supported is: <ul style="list-style-type: none"> 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.)
<i>protocol</i> <i>outgoing-address</i>	A protocol name followed by an address or host name. Protocol translation choices are: ppp , slip , tcp , and x25 . <p>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 48 for more information about the host-name keyword.</p> <p>Additional keywords that can be entered with the protocol are as follows:</p> <ul style="list-style-type: none"> 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. <p>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.</p> virtual-template—Associates a virtual template with a virtual access interface. See the translate lat (virtual access interfaces) command description for more information.

<i>outgoing-options</i>	(Optional) Outgoing connection request options. Choices depend upon the protocol or command entered. See Table 46 , Table 47 , Table 48 , and Table 49 for more information.
<i>global-options</i>	(Optional) One or more of the following translation options can be used by any connection type: <ul style="list-style-type: none"> • access-class <i>number</i>—Allows the incoming call to be used by source hosts that match the access list parameters. The argument <i>number</i> 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 <i>not</i> 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 <i>number</i>—Limits the number of simultaneous users of the translation to <i>number</i> (an integer you specify). • quiet—Suppresses printing of user-information messages.

Defaults No default translation parameters

Command Modes Global configuration

Command History	Release	Modification
	10.0	This command was introduced.
	12.1	The no-reset permanent virtual circuits (PVC) subkeyword was added to support outgoing PVCs.

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 46](#), [Table 47](#), [Table 48](#), and [Table 49](#) 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:

```
Router(config)# translate lat ?

WORD LAT service name
```

```

Router(config)# translate lat LSVC ?

autocommand      Associate a command with a translation on this connections
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 46 LAT-to-PPP Outgoing Translation Options

Outgoing PPP Translation

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.
-

Table 46 LAT-to-PPP Outgoing Translation Options (continued)

-
- **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 47 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.

Table 47 LAT-to-SLIP Outgoing Translation Options (continued)

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 48 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 Domain Name System (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).
-

Table 48 LAT-to-TCP Outgoing Options (continued)

-
- **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 49 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:

- **culd** *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* | **window****size** *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.
 - **window****size** *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.
-

Table 49 LAT-to-X.25 Outgoing Translation Options (continued)

- **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** 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 translate** command throughout the duration of the 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

Command	Description
show translate	Displays configured translation sessions.
translate tcp	Translates a TCP connection request automatically to another outgoing protocol connection.
translate x25	Translates an X.25 connection request automatically to another outgoing protocol connection.
x29 access-list	Limits access to the access server from certain X.25 hosts.
x29 profile	Creates a PAD profile script for use by the translate command.

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.

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	
<i>incoming-service-name</i>	A LAT service name. When used on the incoming portion of the translate lat command, <i>service-name</i> 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.
<i>incoming-options</i>	(Optional) An incoming connection request option. For LAT, the only keyword currently supported is: <ul style="list-style-type: none"> • 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 <i>number</i>	Applies the virtual interface template specified by the <i>number</i> argument in place of outgoing options.
<i>global-options</i>	(Optional) Translation options that can be used by any connection type and can be one or more of the following: <ul style="list-style-type: none"> • access-class <i>number</i>—Allows the incoming call to be used by source hosts that match the access list parameters. The argument <i>number</i> 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 <i>number</i>—Limits the number of simultaneous users of the translation to <i>number</i> (an integer you specify). • local—Allows Telnet protocol negotiations to <i>not</i> 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.

Defaults No default translation parameters

Command Modes Global configuration

Command History	Release	Modification
	10.0	This command was introduced.

Usage Guidelines The command syntax documented here 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 50](#) lists all outgoing options and their corresponding interface configuration commands.

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 50](#) maps outgoing **translate** command options to interface commands you can configure in the virtual interface template.

Table 50 Mapping Outgoing translate lat Options to Interface Commands

translate lat Command Options	Corresponding Interface Configuration Command
ip-pool	peer default ip address { ip-address dhcp pool [poolname] }
header-compression	ip tcp header compression [on off passive]
routing	ip routing or ipx routing
mtu	mtu
keepalive	keepalive
authentication { chap pap }	ppp authentication { chap pap }
ppp use-tacacs	ppp use-tacacs
ipx loopback	ipx ppp-client loopback number

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-Template1
 ip unnumbered Ethernet0
 peer default ip address 10.12.118.12
 ppp authentication chap
!
translate lat pt-printer1 unadvertised virtual-template 1
```

Related Commands	Command	Description
	show translate	Displays configured translation sessions.
	translate tcp	Translates a TCP connection request automatically to another outgoing protocol connection.
	translate x25	Translates an X.25 connection request automatically to another outgoing protocol connection.
	x29 access-list	Limits access to the access server from certain X.25 hosts.
	x29 profile	Creates a PAD profile script for use by the translate command.

translate ruleset

To define a unique name for a translation ruleset, specify the translated protocols, and enable translate ruleset configuration mode, use the **translate ruleset** command in global configuration mode. To remove the ruleset, use the **no** form of this command.

translate ruleset *name* **from** *incoming-protocol* **to** *outgoing-protocol*

no translate ruleset *name* **from** *incoming-protocol* **to** *outgoing-protocol*

Syntax Description

<i>name</i>	Unique name for the protocol translation ruleset.
from <i>incoming-protocol</i>	Specifies the incoming protocol that will be translated, and can be one of the following: <ul style="list-style-type: none"> • pad—X.25 PAD • telnet—TCP/IP Telnet
to <i>outgoing-protocol</i>	Specifies the outgoing protocol or option to translate the incoming protocol to, and can be one of the following: <ul style="list-style-type: none"> • pad—X.25 PAD • telnet—TCP/IP Telnet

Defaults

No default behavior or values

Command Modes

Global configuration

Command History

Release	Modification
12.3(8)T	This command was introduced.

Usage Guidelines

This command places the shell into the translate ruleset configuration mode to create the ruleset template of protocol translation criteria.

The translation ruleset compares an incoming connection attempt to the set of criteria to determine whether it is eligible for protocol translation processing, then uses additional information to complete the transaction.

Existing one-line translate command protocol translation configurations (using the **translate tcp** and **translate x25** commands, for example) take precedence over a protocol translation ruleset. This rule is reinforced when the configuration is generated by placing all one-line **translate** command protocol translation configurations *before* protocol translation ruleset configurations.

Some protocol translation options available with the one-line protocol translation commands are not available with this command.

An incoming PAD call with a standard protocol identification (PID) value that cannot be identified as PAD (for example, its first four bytes indicate IP encapsulation) will not be eligible for protocol translation.

Incoming TCP connections must be made on standard Telnet port 23; otherwise, the connection is not eligible for protocol translation.

If a ruleset is deleted, it will not affect an established protocol translation session, but a new, incoming connection attempt may terminate, depending upon how far along it is in the connection process.

Examples

The following example shows how to start the translate ruleset configuration mode:

```
Router(config)# translate ruleset Pad-Telnet from pad to telnet
```

```
Router(cfg-pt-ruleset)# ?
```

Protocol Translation ruleset configuration commands:

```
description  User-specified description of the ruleset
exit         Exit from PT ruleset configuration mode
match        Define a pattern of input parameters for a connection to process
no           Negate or set default values of a command
options      Define the ruleset options
set          Define parameter values
skip         Define a pattern of input parameters for a connection to skip
substitute   Define a parameter pattern match and substitute into another
test         Define parameter pattern(s) to test for conditionally setting parameters
```

Related Commands

Command	Description
description (ruleset)	Adds a description about a translation ruleset.
match (ruleset)	Identifies a connection for processing by the translation ruleset.
options (ruleset)	Specifies protocol translation options in a translation ruleset.
set (ruleset)	Unconditionally sets one or more connection parameters to a fixed value for a translation ruleset.
show translate ruleset	Displays a summary of a specific or of all configured translation rulesets, behavioral parameters, and usage statistic.
skip (ruleset)	Identifies a connection for omission by the translation ruleset.
substitute (ruleset)	Matches an available protocol and substitutes another in a translation ruleset.
test (ruleset)	Tests parameter values in a translation ruleset using regular expressions.
test translate	Displays a trace of protocol translation behavior for a connection attempt.
x25 pvc translate ruleset	Configures PVCs that are valid for protocol translation ruleset handling.

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

<i>incoming-address</i>	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.
<i>incoming-options</i>	(Optional) An incoming connection request option. Choices are as follows: <ul style="list-style-type: none"> • 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 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 VPN/Security Management Solution (VMS) systems.

<i>protocol</i>	A protocol name followed by an address or host name. Protocol translation choices are: lat , ppp , slip , and x25 .
<i>outgoing-address</i>	<p>Additional keywords that can be entered instead of a protocol name are as follows:</p> <ul style="list-style-type: none">• 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.
<i>outgoing-options</i>	(Optional) Outgoing connection request options. Choices depend upon the protocol or command entered. See Table 51 , Table 52 , Table 53 , and Table 54 for more information.

<i>global-options</i>	<p>(Optional) One or more of the following translation options can be used by any connection type:</p> <ul style="list-style-type: none"> • access-class <i>number</i>—Allows the incoming call to be used by source hosts that match the access list parameters. The <i>number</i> argument 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 from 1300 to 1999. • authorize <i>method-list tag</i>—Enables authorization for protocol translation sessions. The <i>method-list</i> argument is the list of authorization methods defined by the aaa authorization command with the translate keyword. The <i>method-list</i> argument may have the value of the <i>list-name</i> argument or the default keyword. The <i>tag</i> argument is an alphanumeric string of up to 64 characters. The <i>tag</i> argument need not be unique; more than one instance of the translate command can specify identical values for the <i>tag</i> argument. The authorize option is available only if the outgoing protocol keyword is x25 or autocommand. • eor marker [insert]—Defines the End-of-Record (EOR) marker for the translation session. The <i>marker</i> argument may be any set of characters from 1 to 4 in length. Nonprintable characters must be entered in hexadecimal format. Printable characters may be typed in. The insert keyword allows the EOR marker to be inserted into the TCP stream after each X.25 packet without a More-bit (M-bit) set is received. • local—Allows Telnet protocol negotiations to <i>not</i> 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 <i>number</i>—Limits the number of simultaneous users of the translation to <i>number</i> (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 packet assembler/disassembler (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.
-----------------------	--

Defaults	No default translation parameters
-----------------	-----------------------------------

Command Modes	Global configuration
----------------------	----------------------

Command History

Release	Modification
10.0	This command was introduced.
12.1	The no-reset permanent virtual circuits (PVCs) subkeyword was added to support outgoing PVCs.
12.2(13)T	The dynamic keyword was added to support a backup interface.
12.3(2)T	The authorize <i>method-list tag</i> keyword and arguments were added to support authorization of protocol translation sessions. The eor marker [insert] keywords and argument were added to support the EOR function for DCNs.

Usage Guidelines

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 x25 1236672
```

However, the Cisco IOS software provides a broad range of options that support protocol translations in many networking environments. [Table 51](#), [Table 52](#), [Table 53](#), and [Table 54](#) list the **translate tcp** translation options by protocol.

You can also use the Cisco IOS command-line interface (CLI) 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 10.1.1.1 ?
```

```
  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 10.1.1.1 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 PVC, see the description for the [translate x25](#) command for important configuration notes.

Table 51 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 52 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 vtys 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 vty. The vty is assigned to the loopback interface.
-

Table 52 TCP-to-PPP Outgoing Options (continued)

-
- **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 53 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 vtys 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 vty. The vty 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.
-

Table 53 TCP-to-SLIP Outgoing Options (continued)

-
- **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 54 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:

- **cuu 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 PAD parameters as defined in the profile created by the **x29 profile** command.
 - **pvc number [interface serial number | packetsize in-size out-size | window size in-size out-size | no-reset | dynamic]**—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.
 - **window size 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 the *in-size* and *out-size* arguments.
 - **no-reset**—Causes the Cisco router to send a no reset packet request at startup of a TCP or LAT-to-PVC translation session.
 - **dynamic**—Causes the TCP-to-X25 PVC session to terminate when the interface goes down. The default behavior is to keep the PVC in existence as long as it is configured.
-

Table 54 TCP-to-X.25 Outgoing Options (continued)

- **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

Dynamic PVCs to Support Primary and Secondary Interfaces Example

The following partial example shows how to configure the PVC dynamic option. The **dynamic** keyword allows an X.25 PVC to be created on a backup interface when the primary interface goes down. The example configures a primary serial interface (serial 0/0) for X.25 encapsulation and assigns a secondary serial interface (serial 1/1) as backup for the primary. Comments explain the configuration.

```
interface serial 0/0
  encapsulation x25
  backup active interface Serial1/1
!
interface serial 1/1
  encapsulation x25
!
! Configure an X.25 route specifying the primary interface on which the router
! should create the dynamic PVCs.
x25 route ^10 interface serial 0/0
!
! Configure a second X.25 route specifying the secondary backup interface on which
! the router can recreate the PVCs should the primary go down.
x25 route ^10 interface serial 1/1
!
! Configure the translate commands with the PVC dynamic option.
translate tcp 10.60.150.128 port 1031 x25 10 pvc 1 dynamic
.
.
.
translate tcp 10.60.150.128 port 1036 x25 10 pvc 6 dynamic
!
```

When the primary interface serial 0/0 is the active link and it is operational (up), the PVCs will be created on that interface.

If the primary interface goes down, the following will occur:

- The TCP sessions associated with the X.25 PVCs on serial interface 0/0 will be terminated.
- The X25 PVCs created on serial interface 0/0 will be deleted.
- The primary link on serial interface 0/0 will go into the TESTING state.
- The backup link on serial interface 1/1 will change from the X.25 TESTING state to the UP state.
- This change of the interface state (from TESTING to UP) will cause the secondary link on serial interface 1/1 to become active.

- The X.25 PVCs will be recreated on the secondary link on serial interface 1/1 based on the configuration of the second **x25 route** command.
- Incoming TCP connections will be reestablished with the X.25 PVCs on serial interface 1/1.

TCP-to-X.25 PVC connections will remain on the backup secondary interface (serial 1/1) even when the primary interface (serial 0/0) becomes operational. Only when the backup interface (serial 1/1) fails does the process described repeat and move the TCP-to-X.25 PVC connections back to the primary interface (serial 0/0).

Incoming TCP Connection to a Printer Example

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
```

IPX-PPP Client Connects to a Server Running PPP Example

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
```

Configuring the EOR Marker for a TCP-to-X.25 Protocol Translation Session Example

The following example configures a TCP-to-X.25 protocol translation session to insert an EOR marker in a TCP packet after each received X.25 packet that does not contain the M-bit set. The EOR marker in this example consists of nonprintable characters and is entered in hexadecimal format.

```
translate tcp 10.60.155.63 x25 12345678 pvc 3 dynamic eor 0x19 insert
```

Configuring Translation Authorization for a TCP-to-X.25 Protocol Translation Session Example

The following example uses an authorization method list named mygroup. Serial interfaces 2/0 and 2/1 connect to X.25 hosts, each of which provides multiple services at different X.25 subaddresses. Some of the translate statements specify unique authorization tags so the services can be individually controlled; others specify generic tags (perhaps because they are less critical, such as a monitoring service rather than one which permits configuration changes).

```
aaa authorization network mygroup group tacacs+
x25 routing
!
interface Ethernet0/0
  ip address 10.60.155.30 255.255.255.0
!
interface Serial2/0
  encapsulation x25 dce
  x25 ltc 30
!
interface Serial2/1
  encapsulation x25 dce
  x25 ltc 30
!
x25 route ^13033 interface Serial2/0
x25 route ^13133 interface Serial2/1
!
```

```

translate tcp 10.60.155.36 port 2001 x25 1303301 login authorize mygroup a-port01
translate tcp 10.60.155.36 port 2002 x25 1303302 login authorize mygroup a-port02
translate tcp 10.60.155.36 port 2003 x25 1303303 login authorize mygroup monitor
translate tcp 10.60.155.36 port 2004 x25 1303304 login authorize mygroup monitor
translate tcp 10.60.155.36 port 2005 x25 13033 pvc 1 login authorize mygroup a-admin01
!
translate tcp 10.60.155.36 port 2101 x25 1313301 login authorize mygroup b-port01
translate tcp 10.60.155.36 port 2102 x25 1313302 login authorize mygroup b-port02
translate tcp 10.60.155.36 port 2103 x25 1313303 login authorize mygroup monitor
translate tcp 10.60.155.36 port 2104 x25 1313304 login authorize mygroup monitor

```

With this configuration, the router accepts Telnet requests to 10.60.155.36 at any of the TCP ports listed. The user is required to log in, then the router sends an authorization request specifying “translate” as the value of the “service” AV pair, and the authorization tag from the corresponding **translate** command as the value of the “azn-tag” AV pair. The user id and remote address of the Telnet session are also included in the authorization request. If the authorization server approves the request, the connection to the specified X.25 address is attempted; if the request is denied, the Telnet connection is closed.

The authorization server would not be able to distinguish between connections to 10.60.155.36 port 2003 and 10.60.155.36 port 2104, because they specify the same authorization tag.

Related Commands

Command	Description
aaa authorization	Sets parameters that restrict user access to a network.
show translate	Displays configured translation sessions.
translate lat	Translates a LAT connection request automatically to another outgoing protocol connection.
translate x25	Translates an X.25 connection request automatically to another outgoing protocol connection.
x29 access-list	Limits access to the access server from certain X.25 hosts.
x29 profile	Creates a PAD profile script for use by the translate command.

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.

translate tcp *incoming-address* [*incoming-options*] **virtual-template** *number* [*global-options*]

no translate tcp *incoming-address* [*incoming-options*] **virtual-template** *number* [*global-options*]

Syntax Description

<i>incoming-address</i>	TCP/IP Telnet and a standard IP address or host name. The <i>ip-address</i> 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.
<i>incoming-options</i>	(Optional) Incoming connection request options. These arguments can have the following values: <ul style="list-style-type: none"> • 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.) • 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. • 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. • 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.
virtual-template number	Applies the virtual interface template specified by the <i>number</i> argument in place of outgoing options.

<i>global-options</i>	<p>(Optional) One or more of the following translation options can be used by any connection type:</p> <ul style="list-style-type: none"> • access-class <i>number</i>—Allows the incoming call to be used by source hosts that match the access list parameters. The argument <i>number</i> 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 <i>not</i> 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 <i>number</i>—Maximum number of simultaneous users of the translation. • 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.
-----------------------	---

Defaults	No default translation parameters
-----------------	-----------------------------------

Command Modes	Global configuration
----------------------	----------------------

Command History	<table border="1"> <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>	Release	Modification	10.0	This command was introduced.
Release	Modification				
10.0	This command was introduced.				

Usage Guidelines

The command syntax documented here 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.

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:

```
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

Command	Description
show translate	Displays configured translation sessions.
translate tcp	Translates a TCP connection request automatically to another outgoing protocol connection.
translate x25	Translates an X.25 connection request automatically to another outgoing protocol connection.
x29 access-list	Limits access to the access server from certain X.25 hosts.
x29 profile	Creates a PAD profile script for use by the translate command.

translate use telnet

To specify the required IP address in a Telnet-to-PAD protocol translation ruleset, use the **translate use telnet** command in global configuration mode. To disable the address, use the **no** form of this command.

translate use telnet *ip-address*

no translate use telnet *ip-address*

Syntax Description	<i>ip-address</i>	Incoming Telnet IP address used by translation rulesets.
---------------------------	-------------------	--

Defaults	No default behavior or values
-----------------	-------------------------------

Command Modes	Global configuration
----------------------	----------------------

Command History	Release	Modification
	12.3(8)T	This command was introduced.

Usage Guidelines	An IP address must be specified for the protocol translator to respond to Address Resolution Protocol (ARP) attempts for that address. The IP address that the protocol translation software listens for must be on a connected subnet; it cannot be used by another interface unless you also specify a TCP port number, and there cannot be another host that responds to ARPs for that address.
-------------------------	--

Examples	The following example sets the IP address for a translation ruleset:
-----------------	--

```
translate use telnet 10.10.1.5
!
translate ruleset T_to_P from telnet to pad
description Site1 10.10.1.5 Area Code 555 exchg 900
match dest-addr ^10.10.1.5$ dest-port ^[1-5]...$
substitute telnet dest-port ^1(...) into pad dest-addr 555900\1
.
.
.
```

Related Commands	Command	Description
	description (ruleset)	Adds a description about a translation rule set.
	match (ruleset)	Identifies a connection for processing by the translation rule set.
	options (ruleset)	Specifies protocol translation options in a translation rule set.
	set (ruleset)	Unconditionally sets one or more connection parameters to a fixed value for a translation rule set.

Command	Description
show translate ruleset	Displays a summary of a specific or of all configured translation rule sets, behavioral parameters, and usage statistic.
skip (ruleset)	Identifies a connection for omission by the translation rule set.
substitute (ruleset)	Matches an available protocol and substitutes another in a translation rule set.
test (ruleset)	Tests parameter values in a translation rule set using regular expressions.
test translate	Displays a trace of protocol translation behavior for a connection attempt.
translate ruleset	Defines a unique name for a translation rule set, specifies translated protocols, and enters translate ruleset configuration mode.
x25 pvc translate ruleset	Configures PVCs that are valid for protocol translation rule set handling.

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

<i>incoming-address</i>	<p>An X.25 and X.121 address that conforms to specifications provided in the <i>CCITT 1984 Red Book</i>.</p> <p>This address 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. 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.</p> <p>Do not use the same address on the interface and for translation.</p>
<i>incoming-options</i>	<p>(Optional) An incoming connection request option. Choices are as follows:</p> <ul style="list-style-type: none"> • 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 the cud keyword is 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 (SVC) after a set period of inactivity, where <i>minutes</i> is the number of minutes in the period. Calls either originated or terminated are cleared. The maximum value of the <i>minutes</i> argument is 255. The default value of the <i>minutes</i> argument 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 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 [pvc-options]	<ul style="list-style-type: none"> • pvc number [interface serial number packetsize in-size out-size window size in-size out-size]—Specifies that the outgoing connection is actually a PVC. The <i>number</i> argument specifies the virtual circuit channel number of the connection, which must be less than the virtual circuits assigned to the SVC. Only one session is allowed per PVC. Use the following optional keywords to further define the connection: <ul style="list-style-type: none"> – 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 (<i>in-size</i>) and output packet size (<i>out-size</i>) for the PVC. Valid packet size values are as follows: 16, 32, 64, 128, 256, 512, 1024, 2048, and 4096. – window size in-size out-size—Specifies the packet count for input windows (<i>in-size</i>) and output windows (<i>out-size</i>) for the outgoing translation. Values of <i>in-size</i> and <i>out-size</i> 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 the <i>in-size</i> and <i>out-size</i> arguments. <p>Note When the incoming connection is a PVC, the login and authorize global options and the autocommand option for the outbound protocol are not available.</p>
<i>protocol</i> <i>outgoing-address</i>	<p>A protocol name followed by an address or host name. Protocol translation choices are lat, ppp, slip, and tcp.</p> <p>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 58 for more information about the host-name keyword.</p> <p>Additional keywords that can be entered instead of a protocol name are as follows:</p> <ul style="list-style-type: none"> • 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. <p>Note The autocommand option is not available when the incoming connection is a PVC.</p> <ul style="list-style-type: none"> • virtual-template—Associates a virtual template with a virtual access interface. See the translate x25 (virtual access interfaces) command description for more information.
<i>outgoing-options</i>	<p>(Optional) Outgoing connection request option. Choices depend upon the protocol or command entered. See Table 55, Table 56, Table 57, and Table 58 for a list of outgoing protocol translation options.</p>

<i>global-options</i>	<p>(Optional) One or more of the following translation options can be used by any connection type:</p> <ul style="list-style-type: none"> • access-class number—Allows the incoming call to be used by source hosts that match the access list parameters. The <i>number</i> argument 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 from 1300 to 1999. • authorize method-list tag—Enables authorization for protocol translation sessions. The <i>method-list</i> argument is the list of authorization methods defined by the aaa authorization command with the translate keyword. The <i>method-list</i> argument may have the value of the <i>list-name</i> argument or the default keyword. The <i>tag</i> argument is an alphanumeric string of up to 64 characters. The <i>tag</i> argument need not be unique; more than one instance of the translate command can specify identical values for the <i>tag</i> argument. <p>Note The authorize option is not available when the incoming connection is a PVC.</p> <ul style="list-style-type: none"> • eor marker [insert]—Defines the End-of-Record (EOR) marker for the translation session. The <i>marker</i> argument may be any set of characters from 1 to 4 in length. Nonprintable characters must be entered in hexadecimal format. Printable characters may be typed in. The insert keyword allows the EOR marker to be inserted into the TCP stream after each X.25 packet without a More-bit (M-bit) set is received. • local—Allows Telnet protocol negotiations to <i>not</i> 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. <p>Note The login option is not available when the incoming connection is a PVC.</p> <ul style="list-style-type: none"> • max-users number—Limits the number of simultaneous users of the translation to <i>number</i> (an integer you specify). • quiet—Suppresses printing of user-information messages. • swap—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 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.
-----------------------	--

Defaults No default translation parameters.

Command Modes Global configuration

Command History

Release	Modification
10.0	This command was introduced.
12.3(2)T	The authorize <i>method-list tag</i> keyword and arguments were added to support authorization of protocol translation sessions. The eor <i>marker</i> [insert] keywords and argument were added to support the EOR function for DCNs.

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 1236672 tcp 10.1.1.1
```

However, the Cisco IOS software provides a broad range of options that support protocol translations in many networking environments. [Table 55](#), [Table 56](#), [Table 57](#), and [Table 58](#) 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 10.1.1.1 ?
```

```
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 55 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 56 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 vtys 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 vty. The vty 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 56 X.25-to-PPP Outgoing Options (continued)

- **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 57 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 vty's 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 vty. The vty 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 58 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 (see 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 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 VPN/Security Management Solution (VMS) systems.

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 PVC number in the **translate x25** command is not within the range of logical channels defined for use by PVCs in the interface configuration.
- 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 interface 1/0 is different (and therefore unique) from PVC 1 on serial interface 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.

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 vtys supported on the Cisco router. Remember that each protocol translation session uses a vty, which lowers the number of vtys 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:

```
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:

```
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:

```
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:

```
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:

```
Router# clear line vty 4
```

- Step 3** Enter configuration mode, delete the **translate** command, and reassign PVC 5 to an interface:

```
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:

```
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:

```
x25 route ^32785223344502 interface Serial1/5
translate tcp 10.0.155.61 port 2502 x25 32785223344502 pvc 1
```

You can 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:

```
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* for details concerning the use of special characters in defining X.121 addresses.)

```
! 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.

```
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.

```
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 AppleTalk Remote Access (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 noquest
  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
```

The following example configures an X.25-to-TCP protocol translation session to insert an EOR marker in a TCP packet after each received X.25 packet that does not contain the M-bit set. The EOR marker in this example consists of printable characters.

```
translate x25 12345678 pvc 3 tcp 10.60.155.63 eor AAA insert
```

The following example uses the default authorization method list. Incoming PAD calls to the router on serial interface 1/1 are translated to Telnet calls to various destinations based on the X.25 subaddress. Use of the first two translate statements is restricted to users that are approved by the authorization server for access to group1; the third translate statement will complete the connection only if the authorization server grants access to group2.

```
aaa authorization network default group tacacs+
!
interface Serial1/1
 encapsulation x25
 x25 address 5551088
!
translate x25 555108801 tcp 10.60.155.1 login authorize default group1
translate x25 555108802 tcp 10.60.155.2 login authorize default group1
translate x25 555108803 tcp 10.60.155.3 login authorize default group2
```

Related Commands

Command	Description
aaa authorization	Sets parameters that restrict user access to a network.
show translate	Displays configured translation sessions.
translate lat	Translates a LAT connection request automatically to another outgoing protocol connection.
translate tcp	Translates a TCP connection request automatically to another outgoing protocol connection.
x29 access-list	Limits access to the access server from certain X.25 hosts.
x29 profile	Creates a PAD profile script for use by the translate command.

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.

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 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. 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.
- **profile profile**—Sets the X.3 PAD parameters as defined in the profile created by the **x29 profile** command.

pvc number [<i>pvc-options</i>]	<ul style="list-style-type: none"> • pvc number [interface serial number packetsize in-size out-size window size in-size out-size]—Specifies that the outgoing connection is actually a PVC. The <i>number</i> 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: <ul style="list-style-type: none"> – 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 (<i>in-size</i>) and output packet size (<i>out-size</i>) for the PVC. Valid packet size values are as follows: 16, 32, 64, 128, 256, 512, 1024, 2048, and 4096. – window size in-size out-size—Specifies the packet count for input windows (<i>in-size</i>) and output windows (<i>out-size</i>) for the outgoing translation. Values of <i>in-size</i> and <i>out-size</i> 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 <i>in-size</i> and <i>out-size</i>.
virtual-template number	Applies the virtual interface template specified by the <i>number</i> argument in place of outgoing options.
<i>global-options</i>	<p>(Optional) Translation options that can be used by any connection type and can be one or more of the following:</p> <ul style="list-style-type: none"> • access-class number—Allows the incoming call to be used by source hosts that match the access list parameters. The argument <i>number</i> 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 <i>not</i> 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 <i>number</i> (an integer you specify). • quiet—Suppresses printing of user-information messages. • swap—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 swap keyword allows connections from an X.25 host that wants to connect to the router, and then treats it like a PAD.

Defaults No default translation parameters.

Command Modes Global configuration

Release	Modification
10.0	This command was introduced.

Usage Guidelines The command syntax documented here 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 59](#) lists all outgoing options and their corresponding interface configuration commands.

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 59](#) maps outgoing **translate** command options to interface commands you can configure in the virtual interface template.

Table 59 Mapping Outgoing translate x25 Options to Interface Commands

translate x25 Command Options	Corresponding Interface Configuration Command
ip-pool	peer default ip address { ip-address dhcp pool [poolname] }
header-compression	ip tcp header compression [on off passive]
routing	ip routing or ipx routing
mtu	mtu
keepalive	keepalive
authentication { chap pap }	ppp authentication { chap pap }
ppp use-tacacs	ppp use-tacacs
ipx loopback	ipx ppp-client loopback number

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

Command	Description
interface virtual-template	Creates a virtual template interface that can be configured and applied dynamically in creating virtual access interfaces.
show translate	Displays configured translation sessions.
translate lat	Translates a LAT connection request automatically to another outgoing protocol connection type.
translate tcp	Translates a TCP connection request automatically to another outgoing protocol connection type.
x29 access-list	Limits access to the access server from certain X.25 hosts.
x29 profile	Creates a PAD profile script for use by the translate command.

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 | ssh | telnet | v120}

no transport input {all | lat | mop | nasi | none | pad | rlogin | ssh | telnet | v120}

Syntax Description

all	Selects all protocols.
lat	Selects the digital (Local-Area Transport) LAT protocol and specifies both incoming reverse LAT and host-initiated connections.
mop	Selects Maintenance Operation Protocol (MOP).
nasi	Select NetWare Access Servers Interface (NASI) as the input transport protocol.
none	Prevents any protocol selection on the line. This makes the port unusable by incoming connections.
pad	Selects X.3 packet assembler/disassembler (PAD) incoming connections.
rlogin	Selects the UNIX rlogin protocol.
ssh	Selects the Secure Shell (SSH) protocol.
telnet	Specifies all types of incoming TCP/IP connections.
v120	Selects the V.120 protocol for incoming asynchronous connections over ISDN.

Defaults

No protocols allowed on the line (**none**).

Command Modes

Line configuration

Command History

Release	Modification
10.0	This command was introduced.
11.1	The none keyword was added and became the default. Before Cisco IOS Release 11.1, the default keyword was all .
12.3(4)T	The ssh keyword was added.

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:

```
line vty 0 32
 transport input telnet
```

Related Commands

Command	Description
transport output	Determines the protocols that can be used for outgoing connections from a line.
transport preferred	Specifies the transport protocol that the Cisco IOS software uses if the user does not specify one when initiating a connection.

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		
all	Selects all protocols.	
lat	Selects the Digital LAT protocol, which is the protocol used most often to connect routers to Digital hosts.	
mop	Selects Maintenance Operation Protocol (MOP).	
nasi	Selects NetWare Access Server Interface (NASI) as the output transport protocol.	
none	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.	
pad	Selects X.3 packet assembler/disassembler (PAD), used most often to connect routers to X.25 hosts.	
rlogin	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.	
telnet	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.	
v120	Selects the V.120 protocol for outgoing asynchronous connections over ISDN.	

Defaults Telnet

Command Modes Line configuration

Command History	Release	Modification
	10.0	This command was introduced.
	11.1	The following keywords were added: <ul style="list-style-type: none"> • all • lat • pad • rlogin • v120

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

Command	Description
transport input	Defines which protocols to use to connect to a specific line of the router.
transport preferred	Specifies the transport protocol that the Cisco IOS software uses if the user does not specify one when initiating a connection.

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** | **ssh** | **telnet** | **v120** }

no transport preferred { **all** | **lat** | **mop** | **nasi** | **none** | **pad** | **rlogin** | **ssh** | **telnet** | **v120** }

Syntax Description

all	Selects all recognized protocols.
lat	Selects the Digital LAT protocol, which is the protocol used most often to connect routers to Digital hosts.
mop	Selects Maintenance Operation Protocol (MOP).
nasi	Selects NetWare Access Server Interface (NASI) protocol.
none	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 is attempted if the command is not recognized.
pad	Selects X.3 packet assembler/disassembler (PAD), used most often to connect routers to X.25 hosts.
rlogin	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.
ssh	Selects the Secure Shell (SSH) protocol.
telnet	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.
v120	Selects the asynchronous protocols over ISDN.

Defaults

Telnet

Command Modes

Line configuration

Command History

Release	Modification
10.0	This command was introduced.
11.1	The following keywords were added: <ul style="list-style-type: none"> • lat • pad • rlogin

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

Command	Description
terminal transport preferred	Specifies the preferred protocol to use for the current session when a command does not specify one.
transport input	Defines which protocols to use to connect to a specific line of the router.
transport output	Determines the protocols that can be used for outgoing connections from a line.

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, the **no** form of this command.

ttycap *tycap-name termcap-entry*

no ttycap *tycap-name*

Syntax Description

<i>tycap-name</i>	Name of a file. It can be up to 32 characters long and must be unique.
<i>termcap-entry</i>	Commands that define the ttycap. Consists of two parts. (See the “Usage Guidelines” section for details.)

Defaults

VT100 terminal emulation

Command Modes

Global configuration

Command History

Release	Modification
10.3	This command was introduced.

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.

```
co#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 60](#). For the definitions of supported string-sequence ttycap capabilities, see [Table 61](#). For the definitions of supported number-sequence ttycap capabilities, see [Table 62](#). For the definitions of supported color-sequence ttycap capabilities, see [Table 63](#).

Table 60 *Definitions of ttycap Capabilities: Boolean Flags*

Boolean Flag	Description
am	Automatic margin
bs	Terminal can backspace with bs
ms	Safe to move in standout modes
nc	No currently working carriage return
xn	NEWLINE ignored after 80 columns (Concept)
xs	Standout not erased by overwriting (Hewlett-Packard)

Table 61 *Definitions of ttycap Capabilities: String Sequences*

String Sequence	Description
AL	Add line below with cursor sequence
bc	Backspace if not ^h
bt	Backtab sequence
ce	Clear to end of line
cl	Clear screen, cursor to upper left
cm	Move cursor to row number and column number

Table 61 *Definitions of ttypcap Capabilities: String Sequences (continued)*

String Sequence	Description
cr	Carriage return sequence
cs	Change scrolling region
DL	Delete the line the cursor is on
ei	End insert mode
ho	Home, move cursor to upper left
ic	Character insert
im	Begin insert mode
is	Initialization string (typically tab stop initialization)
ll	Move cursor to lower left corner
md	Turn on bold (extra bright) character attribute
me	Turn off all character attributes
nd	Nondestructive space
nl	Newline sequence
pc	Pad character if not NULL
rc	Restore cursor position
rs	Resets terminal to known starting state
sc	Save cursor position
se	End standout mode (highlight)
so	Start standout mode (highlight)
ta	Tab
te	End programs that use cursor motion
ti	Initialization for programs that use cursor motion
uc	Underline character at cursor
ue	End underline mode
up	Move cursor up
us	Begin underline mode
vb	Visual bell
vs	Visual cursor
ve	Normal cursor

Table 62 Definitions of ttycap Capabilities: Number Sequences

Number Sequence	Description
li	Lines on the screen
co	Columns on the screen
sg	Standout glitch, number of spaces printed when entering or leaving standout display mode
ug	Underline glitch, number of spaces printed when entering or leaving underline mode

Table 63 Definitions of ttycap Capabilities: Color Sequences

Color Sequence	Description
x0	Black
x1	Blue
x2	Red or orange
x3	Pink or purple
x4	Green, which is the default color
x5	Turquoise
x6	Yellow
x7	Gray or white

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 “Configuring Dial-In Terminal Services” chapter 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\
d0|vt100|vt100-am|vt100am|dec vt100:do=^J:co#80:li#24:\
cl=50^[[;H^[[2J:bs:am:cm=5^[[%i%d;%dH:nd=2^[[C:up=2^[[A:\
ce=3^[[K:so=2^[[7m:se=2^[[m:us=2^[[4m:ue=2^[[m:md=2^[[1m:\
me=2^[[m:ho=^[[H:xn:sc=^[7:rc=^[8:cs=^[[%i%d;%dr:

```

Related Commands

Command	Description
keymap-type	Specifies the keyboard map for a terminal connected to the line.
terminal-type	Specifies the type of terminal connected to a line.

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	<i>bps</i>	Baud rate, in bits per second (bps). The default value is 9600 bps.
--------------------	------------	---

Defaults	9600 bps
----------	----------

Command Modes	Line configuration
---------------	--------------------

Command History	Release	Modification
	10.0	This command was introduced.

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	Command	Description
	rotary-group	Sets the terminal receive speed (how fast the terminal receives information from the modem).
	source template	Sets the flow control start character.
	terminal txspeed	Sets the terminal transmit speed (how fast the terminal can send information) on the current line and session.

where

To list the open sessions, use the **where** command in EXEC mode.

where

Syntax Description This command has no arguments or keywords.

Command Modes EXEC

Command History	Release	Modification
	10.0	This command was introduced in a release prior to Cisco IOS Release 10.0.

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:

```
Router# where

Conn Host                Address                Byte   Idle  Conn Name
  1 MATHOM                192.168.7.21           0      0    MATHOM
* 2 CHAFF                172.18.12.19           0      0    CHAFF
```

The asterisk (*) indicates the current terminal session.

[Table 64](#) describes the significant fields shown in the display.

Table 64 *where* Field Descriptions

Field	Description
Conn	Name or address of the remote host to which the connection is made.
Host	Remote host to which the router is connected through a Telnet session.
Address	IP address of the remote host.
Byte	Number of unread bytes for the user to see on the connection.
Idle	Interval, in minutes, since data was last sent on the line.
Conn Name	Assigned name of the connection.

Related Commands	Command	Description
	show sessions	Displays information about open LAT, Telnet, or rlogin connections.

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, use the **no** form of this command.

```
x25 subaddress {line | number} [no-zero-pad]
```

```
no x25 subaddress {line | number}
```

Syntax Description	line	Physical port number for the indicated line to be appended to the X.121 address as the subaddress.
	<i>number</i>	Numeric variable assigned to a specific line.
	no-zero-pad	(Optional) Specifies that a leading zero should not be appended to subaddresses with a value of nine or lower (0-9).

Defaults No default behavior or values

Command Modes Line configuration

Command History	Release	Modification
	11.2 F	This command was introduced.
	12.3(2)T	The no-zero-pad keyword was added.

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.

Use the **no-zero-pad** option to eliminate the inclusion of a leading zero for subaddresses with a value of nine or lower (0-9).

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
```

The following example configures a subaddress of 6 for a set of vty lines, and specifies that a leading zero should not be appended to the subaddress value:

```
line vty 0 9
x25 subaddress 6 no-zero-pad
```

Related Commands

Command	Description
line	Identifies a specific line for configuration and starts the line configuration command collection mode.

x25 pvc translate ruleset

To configure the permanent virtual circuits (PVCs) that are valid for protocol translation ruleset handling, use the **x25 pvc translate ruleset** command in interface configuration mode. To remove or change the configuration, use the **no** form of this command.

```
x25 pvc number translate ruleset [name | x121-address | packet-size in-size out-size | window-size in-size out-size]
```

```
no x25 pvc number translate ruleset [name | x121-address | packet-size in-size out-size | window-size in-size out-size]
```

Syntax Description

<i>number</i>	PVC number to reserve for protocol translation ruleset handling.
<i>name</i>	(Optional) Ruleset name specified by the translate ruleset command that governs the PVC.
<i>x121-address</i>	(Optional) Destination host address.
packet-size <i>in-size out-size</i>	(Optional) Specifies the input packet size (<i>in-size</i>) and output packet size (<i>out-size</i>) for the PVC. Valid packet size values are: 16, 32, 64, 128, 256, 512, 1024, 2048, and 4096.
window-size <i>in-size out-size</i>	(Optional) Specifies the packet count for input windows (<i>in-size</i>) and output windows (<i>out-size</i>) for the outgoing translation. Values of the <i>in-size</i> and <i>out-size</i> arguments range from 1 to 127 and must not be greater than the value set for the x25 modulo command.
Note	You must specify the same value for the <i>in-size</i> and <i>out-size</i> arguments.

Defaults

No default behavior or values

Command Modes

Interface configuration

Command History

Release	Modification
12.3(8)T	This command was introduced.

Usage Guidelines

If the optional name or X.121 address is given, PVC usage is restricted to the translation ruleset of that name.

Examples

The following example shows how to reserve a PVC for protocol translation ruleset handling, and select the outbound X.25 serial interface and PVC number based on the IP port number:

```
interface serial 2/0
x25 pvc 4 translate ruleset port_to_pvc
!
translate use telnet 10.10.1.6
```

```

!
translate ruleset port_to_pvc from telnet to pad
match dest-addr ^10.10.1.6$ dest-port ^[12]00[0-7][1-3]$
substitute telnet dest-port ^..0([0-7]) into pad interface serial 0/\1
substitute telnet dest-port ^....(.) into pad pvc \1
test telnet dest-port ^.0... set pad profile TEMS
test telnet dest-port ^.1... set pad profile SQAS
test telnet dest-port ^.2... set pad profile NMA
substitute telnet dest-port (.)$ into pad dest-addr 876543\1

x.29 profile TEMS 2:0 3:128 4:0
x.29 profile SQAS 2:0 3:128 4:0
x.29 profile NMA 2:0 3:128 4:0

```

Related Commands

Command	Description
description (ruleset)	Adds a description about a translation ruleset.
match (ruleset)	Identifies a connection for processing by the translation ruleset.
options (ruleset)	Specifies protocol translation options in a translation ruleset.
set (ruleset)	Unconditionally sets one or more connection parameters to a fixed value for a translation ruleset.
show translate ruleset	Displays a summary of a specific or of all configured translation rulesets, behavioral parameters, and usage statistic.
skip (ruleset)	Identifies a connection for omission by the translation ruleset.
substitute (ruleset)	Matches an available protocol and substitutes another in a translation ruleset.
test (ruleset)	Tests parameter values in a translation ruleset using regular expressions.
test translate	Displays a trace of protocol translation behavior for a connection attempt.
translate ruleset	Defines a unique name for a translation ruleset, specifies translated protocols, and enters translate ruleset configuration mode.

x28

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 <i>character-string</i>	(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 <i>file-name</i>	(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	Release	Modification
	11.2 F	This command was introduced.

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 65 lists the commands available in both standard and extended command syntax.

Table 65 Available PAD Command Signals

Standard Syntax	Extended Syntax	Description
break		Simulate an asynchronous break.
call		Place a virtual call to a remote device.
clr	clear	Clear a virtual call.
<i>command-signal¹</i>		Specifies a call request without using a standard X.28 command, which is entered with the following syntax: <i>facilities-x121-addressDcall-user-data.</i>
help		Display help information. (See Table 67.)
iclr	iclear	Request the remote device to clear the call.
int	interrupt	Send an Interrupt Packet.
par? par	parameter read	Display the current values of local parameters. (See Table 66.)
prof	profile file-name	Load a standard or a named profile.
reset		Reset the call.
rpar?	rread	Display the current values of remote parameters.
rset?	rsetread	Set and then read the values of remote parameters.
set		Change the values of local parameters. (See Table 66.)
set?	setread	Change and then read the values of parameters.
stat	status	Request the status of a connection.
selection pad		Set up a virtual call.

1. This is an example of issuing a call request command: the **R,G23,P2-234234Duser1** command.

Table 66 lists the different types of parameters you can set using the **set parameter-number:new-value** PAD command signal from X.28 mode. Refer to the “X.3 PAD Parameters” appendix in the *Cisco IOS Terminal Services Configuration Guide* for more complete information about these parameters. See Table 34 in this publication for a list of ASCII characters.

Table 66 Supported X.3 PAD Parameters

Parameter Number	ITU-T Parameter Name	ITU-T X.3 and Cisco Values
Parameter Number	ITU-T Parameter Name	ITU-T X.3 and Cisco Values
1	PAD recall using a character	Minimum value: 0; maximum value: 126; X.28 PAD user emulation mode default: 1. Note Not supported by PAD EXEC user interface.
2	Echo	Minimum value: 0; maximum value: 1; PAD EXEC mode and X.28 PAD user emulation mode default: 1.

Table 66 Supported X.3 PAD Parameters (continued)

Parameter Number	ITU-T Parameter Name	ITU-T X.3 and Cisco Values
3	Selection of data forwarding character	Minimum value: 0; maximum value: 255; PAD EXEC mode default: 2 (CR); X.28 PAD user emulation mode default: 126 (~).
4	Selection of idle timer delay	Minimum value: 0; maximum value: 255; PAD EXEC mode default: 1; X.28 PAD user emulation mode default: 0.
5	Ancillary device control	Minimum value: 0; maximum value: 2; PAD EXEC mode default: 0; X.28 PAD user emulation mode default: 1.
6	Control of PAD service signals	Minimum value: 0; maximum value: 255; PAD EXEC mode default: 0; X.28 PAD user emulation mode default: 2. Note Not supported by PAD EXEC user interface.
7	Action upon receipt of a BREAK signal	Minimum value: 0; maximum value: 31; PAD EXEC mode default: 4; X.28 PAD user emulation mode default: 2.
8	Discard output	Minimum value: 0; maximum value: 1; PAD EXEC mode and X.28 PAD user emulation mode default: 0.
9	Padding after Return	Minimum value: 0; maximum value: 255; PAD EXEC mode and X.28 PAD user emulation mode default: 0.
10	Line folding	Not supported.
11	DTE speed (binary speed of start-stop mode DTE)	Minimum value: 0; maximum value: 18; PAD EXEC mode and X.28 PAD user emulation mode default: 14.
12	Flow control of the PAD by the start-stop DTE	Minimum value: 0; maximum value: 1; PAD EXEC mode default: 0; X.28 PAD user emulation mode default: 1.
13	Line feed insertion (after a Return)	Minimum value: 0; maximum value: 7; PAD EXEC mode and X.28 PAD user emulation mode default: 0.
14	Line feed padding	Minimum value: 0; maximum value: 255; PAD EXEC mode and X.28 PAD user emulation mode default: 0.
15	Editing	Minimum value: 0; maximum value: 1; PAD EXEC mode and X.28 PAD user emulation mode default: 0.
16	Character delete	Minimum value: 0; maximum value: 127; PAD EXEC mode and X.28 PAD user emulation mode default: 127 (DEL).
17	Line delete	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).
18	Line display	Minimum value: 0; maximum value: 127; PAD EXEC mode and X.28 PAD user emulation mode default: 18 (DC2 or Ctrl-R).
19	Editing PAD service signals	Minimum value: 0; maximum value: 126; PAD EXEC mode default: 0; X.28 PAD user emulation mode default: 2. Note Not supported by PAD EXEC user interface.

Table 66 Supported X.3 PAD Parameters (continued)

Parameter Number	ITU-T Parameter Name	ITU-T X.3 and Cisco Values
20	Echo mask	Minimum value: 0; maximum value: 255; PAD EXEC mode and X.28 PAD user emulation mode default: 0. Note Not supported by PAD EXEC user interface.
21	Parity treatment	Minimum value: 0; maximum value: 4; PAD EXEC mode and X.28 PAD user emulation mode default: 0. Note For additional values that can be selected for parameter 21, including parity treatment to conform 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 <i>Cisco IOS Terminal Services Configuration Guide</i> .

**Note**

Abbreviated X.121 addresses are not supported. Such addresses start with a period, are alphanumeric, and are mapped to a full X.121 address by the PAD.

Table 67 lists the options for the X.28 **help** command.

Table 67 X.28 help Options

Command	Description
help	Describes the help PAD command.
help command	Displays the list of available PAD command signals.
help parameter	Displays the list of available X.3 PAD parameters.
help parameter number	Displays the specified X.3 PAD parameter and its current value.
help list	Lists the available help subjects.
help profiles	Lists available profiles.
help profile name	Displays the specified parameter name and current value.
help any-PAD-command	Describes the specified PAD command signal.

You can issue call requests from X.28 mode without using standard X.28 commands by using the following command syntax:

```
facilities-x121-addressDcall-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 68 shows examples of parsed call requests.

Table 68 Example X.28 Call Requests

Command	Description
123456789	Calls this X.121 address.
123456789*userdata	Calls this X.121 address, with specified data.
123456789Puserdata	Calls this X.121 address, with specified data.
123456789Duserdata	Calls this X.121 address, with specified data.
Nabcd-123456789	Calls this X.121 address, with NUI set to abcd.
Nabcd,R-123456789	Calls 123456789 with NUI of abcd, and with reverse charging.

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

Command	Description
pad	Logs in to a PAD.

x3

To set X.3 packet assembler/disassembler (PAD) parameters, use the **x3** command in EXEC mode.

x3 *parameter:value*

Syntax Description	<i>parameter:value</i> Sets the PAD parameters. (See Table 66 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

Release	Modification
11.2	This command was introduced.

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

Command	Description
resume (X.3 PAD)	Sets X.3 parameters for PAD connections.

xremote



Note

Effective with release 12.2(15)T, the **xremote** command is no longer available in Cisco IOS software.

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

Release	Modification
11.1	This command was introduced.
12.2(15)T	This command was removed and is no longer available in Cisco IOS software.

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 dialup: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 server
Trying SERVER.CISCO.COM (722.18.1.55)... Open

SunOS UNIX (server)

login: deal
Password:

Last login: Fri Apr 1 17:17:46 from dialup.cisco.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 SERVER closed by foreign host]

dialup> xremote
Entering XRemote
```

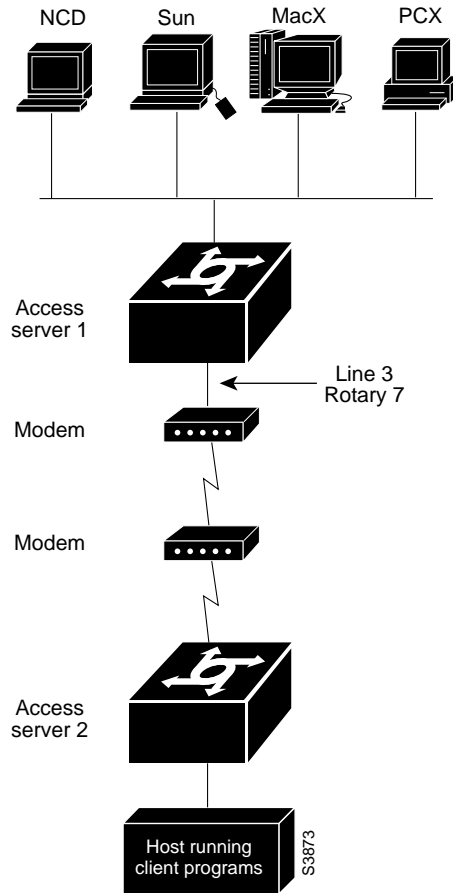
The following procedure shows how an XRemote connection is established for a configuration like the one shown in [Figure 1](#). This example assumes that the administrator has set the display environment variable for the user to identify the X display terminal.

-
- Step 1** From the PCX, MacX, or UNIX machine in [Figure 1](#), 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 1 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 server 9016

Trying 192.168.7.84 ...
Connected to server.cisco.com.
Escape character is '^]'.

User Access Verification

Password:
Password OK
```

```

--- Outbound XRemote service ---
Enter X server name or IP address: router
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: user1
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 server2
Trying SERVER2.CISCO.COM (172.18.1.55)... Open

SunOS UNIX (sparks)

login: user1
Password:
Last login: Fri Apr 1 17:17:46 from dialup.cisco.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 SERVER2 closed by foreign host]

dialup> xremote

```

Related Commands

Command	Description
xremote lat	Initiates a DECwindow session over a LAT connection.
xremote xdm	Activates automatic session startup for an XRemote connection.

xremote lat



Note

Effective with release 12.2(15)T, the **xremote lat** command is no longer available in Cisco IOS software.

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

Release	Modification
11.2	This command was introduced.
12.2(15)T	This command was removed and is no longer available in Cisco IOS software.

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

Command	Description
xremote	Prepares the router for manual startup and initiates an XRemote connection.
xremote xdm	Activates automatic session startup for an XRemote connection.

xremote tftp buffersize



Note

Effective with release 12.2(15)T, the **xremote tftp buffersize** command is no longer available in Cisco IOS software.

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

<i>buffersize</i>	Buffer size in bytes. This is a decimal number in the range from 4096 to 70000 bytes. The default is 70000.
-------------------	---

Defaults

70000 bytes

Command Modes

Global configuration

Command History

Release	Modification
10.0	This command was introduced.
12.2(15)T	This command was removed and is no longer available in Cisco IOS software.

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



Note

Effective with release 12.2(15)T, the **xremote tftp host** command is no longer available in Cisco IOS software.

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

Release	Modification
10.0	This command was introduced.
12.2(15)T	This command was removed and is no longer available in Cisco IOS software.

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



Note

Effective with release 12.2(15)T, the **xremote tftp retries** command is no longer available in Cisco IOS software.

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	<i>retries</i> (Optional) Number of retries. Acceptable values are decimal numbers in the range from 1 to 15.						
Defaults	3 retries						
Command Modes	Global configuration						
Command History	<table border="1"> <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>12.2(15)T</td> <td>This command was removed and is no longer available in Cisco IOS software.</td> </tr> </tbody> </table>	Release	Modification	10.0	This command was introduced.	12.2(15)T	This command was removed and is no longer available in Cisco IOS software.
Release	Modification						
10.0	This command was introduced.						
12.2(15)T	This command was removed and is no longer available in Cisco IOS software.						
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	<p>The following example sets the number of font loader retries to 5:</p> <pre>xremote tftp retries 5</pre>						

xremote xdm



Note

Effective with release 12.2(15)T, the **xremote xdm** command is no longer available in Cisco IOS software.

To activate automatic session startup for an XRemote connection, use the **xremote xdm** command in EXEC mode.

```
xremote xdm [host-name]
```

Syntax Description

host-name (Optional) Host computer name or IP address.

Command Modes

EXEC

Command History

Release	Modification
11.2	This command was introduced.
12.2(15)T	This command was removed and is no longer available in Cisco IOS software.

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:

```
Router# xremote xdm host1
```

Related Commands	Command	Description
	xremote	Prepares the router for manual startup and initiates an XRemote connection.
	xremote lat	Initiates a DECwindow session over a LAT connection.

