

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 .



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 138 for more information about the **host-name** keyword.

Additional keywords that can be entered with the protocol are as follows:

- autocommand**—Specifies an EXEC command for an outgoing connection. The command executes upon connection to a host. You can issue any EXEC command and any switch or host name as an argument to the **autocommand** command. If the string following **autocommand** has one or more spaces as part of the string, you must place quotation marks (“ ”) around the string.

If you want to enable AppleTalk Remote Access (ARA) on an outgoing connection, specify the **autocommand arap** keywords. These keywords are necessary for ARA because ARA does not use addressing, and this option permits you to invoke the ARA string.

- **virtual-template**—Associates a virtual template with a virtual access interface. See the **translate lat (virtual access interfaces)** command description for more information.

<i>outgoing-options</i>	(Optional) Outgoing connection request options. Choices depend upon the protocol or command entered. See Table 136, Table 137, Table 138, and Table 139 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 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.

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 136, Table 137, Table 138, and Table 139 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 L SVC ?
```

```
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 L SVC tcp ?
```

```
Hostname or A.B.C.D IP address
```

```
Router(config)# translate lat L SVC 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 136 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.

Table 136 LAT-to-PPP Outgoing Translation Options (continued)**Outgoing PPP Connection Request Options**

Add any of the following keywords to configure PPP connection requests:

- **authentication {pap | chap}**—Sets Challenge Handshake Authentication Protocol (CHAP) or Password Authentication Protocol (PAP) authentication for PPP on virtual asynchronous interfaces. If you specify both keywords, order is significant; the system will try to use the first authentication type, then the second.
- **header-compression**—Implements header compression on IP packets only.
- **ipx loopback number**—Specifies the loopback interface to be created and permits clients running IPX-PPP to connect through virtual terminal lines on the router. A loopback interface must have been created and configured with a Novell IPX network number before IPX-PPP can work on the virtual terminal line. The virtual terminal line is assigned to the loopback interface.
- **keepalive number-of-seconds**—Specifies the interval at which keepalive packets are sent on Serial Line Internet Protocol (SLIP) and PPP virtual asynchronous interfaces. By default, keepalive packets are enabled and sent every 10 seconds. To shut off keepalive packets, use a value of 0. The active keepalive interval is 1 through 32,767 seconds. When you do not change from the default of 10, the keepalive interval does not appear in **more system:running-config** or **show translate** command output.
- **mtu bytes**—Sets the interface maximum transmission unit (MTU) of packets that the virtual asynchronous interface supports. The default MTU is 1500 bytes on a virtual asynchronous interface. The acceptable range is from 64 to 1,000,000 bytes.
- **routing**—Permits routing updates between connections. This keyword is required if the destination device is not on a subnet connected to one of the interfaces on the router.
- **use-tacacs**—Uses TACACS to verify PPP authentications for CHAP or PAP on virtual asynchronous interfaces.

Table 137 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 137 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 138 LAT-to-TCP Outgoing Options**Outgoing TCP Translation**

tcp ip-address

Translates LAT to TCP/IP Telnet. Supply an IP address as a standard, four-part dotted decimal IP address, the name of an IP host that can be resolved by the DNS, or explicit specification in an **ip host** command (refer to the description for the **host-name** keyword in the “Outgoing TCP Connection Request Options” section).

Outgoing TCP Connection Request Options

Any of the following optional keywords can be used to configure TCP connection requests:

- **binary**—Negotiates Telnet binary mode on the connection.
- **host-name**—Stores the host name rather than its IP address, thereby allowing the host name to be resolved at connection time instead of configuration time. There is also a **rotor** keyword suboption that you can use to modify the behavior of the **host-name** keyword by allowing one of the IP addresses defined by the **ip host** configuration command to be chosen randomly. If one address fails, another one will be tried, and so on until all address choices are exhausted. You can use the **rotor** keyword, therefore, to provide basic load sharing of the IP destinations.
- **multibyte-IAC**—Always treat multiple Interpret as Command (IAC) escape character codes as a Telnet command.
- **port number**—For outgoing connections, enter the number of the port to match. The default is port 23 (Telnet).

Table 138 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 139 LAT-to-X.25 Outgoing Translation Options**Outgoing X.25 Translation****x25** *x.121-address*

Translates LAT to the X.25 protocol. Supply an X.121 address that conforms to the specifications provided in the *CCITT 1984 Red Book*, or the name of an X.25 host that can be resolved by the DNS, or explicit specification in an **x25 host** command.

The address number generally consists of a portion that is administered by the public data network (PDN) and a portion that is locally assigned. You must be sure that the numbers that you assign agree with the addresses assigned to you by the X.25 service provider. The X.121 addresses generally will be subaddresses of the X.121 address for the X.25 network interface.

Outgoing X.25 Connection Request Options

Any of the following optional keywords can be used to configure X.25 connection requests:

- **cud** *c-u-data*—Sends the specified X.25 Call User Data (CUD) text as part of an outgoing call request after the protocol identification bytes.
- **no-reverse**—Specifies that outgoing calls not request the X.25 reverse charge facility, when the interface default is that all outgoing calls are reverse charged.
- **profile** *profile*—Sets the X.3 packet assembler/disassembler (PAD) parameters as defined in the profile created by the **x29 profile** command.
- **pvc** *number* [**interface serial number** | **packetsize** *in-size out-size* | **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 139 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 translation** 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.

The command syntax that follows shows how to apply a virtual interface template in place of outgoing **translate** options. If you are using virtual templates for protocol translation, all outgoing options are defined in the virtual interface template. Table 140 lists all outgoing options and their corresponding interface configuration commands.

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

Release	Modification
10.0	This command was introduced.

Usage Guidelines You define the protocol translation connections by choosing a protocol keyword and supplying the appropriate address, host name, or service name. The protocol connection information is followed by optional features for that connection, as appropriate. For example, the **binary** keyword is only appropriate with TCP/IP connections. The global options, in general, apply to all the connection types, but there are exceptions.

Rather than specifying outgoing translation options in the **translate** command, configure these options as interface configuration commands under the virtual interface template, then apply the virtual interface template to the **translate** command. Table 140 maps outgoing **translate** command options to interface commands you can configure in the virtual interface template.

Table 140 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
```

■ translate lat (virtual access interfaces)

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 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.
<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 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.

<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>	Additional keywords that can be entered with the protocol are as follows: <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 141, Table 142, Table 143, and Table 144 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 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 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

Release	Modification
10.0	This command was introduced.
12.1	The no-reset permanent virtual circuits (PVCs) 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 TCP protocol translation command can be as simple as the following example:

```
Router(config)# translate tcp 10.1.1.1 X.25 1236672
```

However, the Cisco IOS software provides a broad range of options that support protocol translations in many networking environments. Table 141, Table 142, Table 143, and Table 144 list the **translate tcp** translation options by protocol.

You can also use the Cisco IOS command-line interface to help you understand how these keywords are entered. In global configuration mode, begin entering the **translate** command and add a question mark at each portion of the command to display the options available. Some examples follow:

```
Router(config)# translate tcp ?
```

```
  Hostname or A.B.C.D  IP address
```

```
Router(config)# translate tcp 1.1.1.1 ?
```

```
  autocommand      Associate a command with a translation on this connections
  binary           Negotiate Telnet binary mode on the connection
  lat              DEC LAT protocol
  port             Port Number
  ppp              Virtual async PPP
  printer          Enable non-interactive (implies global quiet)
  slip             Virtual async SLIP
  stream           Enable stream processing
  tcp              TCP/IP Telnet
  virtual-template Associate a virtual template with virtual access interface
  x25              X.25
```

```
Router(config)# translate tcp 1.1.1.1 lat LAT-1 ?
```

```
  access-class    Allow access list parameters to be used by source hosts
  local           Allow Telnet protocol negotiations not to be translated
  login           Require that the user log in before the outgoing connection is
                  made
  max-users       Limit the number of simultaneous users of the translation
  node            LAT node name
  port            LAT port name
  quiet           Suppress printing of user-information messages
  unadvertised    Prevent service advertisements from being broadcast to the
                  network
```

**Note**

If you plan to translate to X.25 on a permanent virtual circuit (PVC), see the description for the **translate x25** command for important configuration notes.

Table 141 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 142 TCP-to-PPP Outgoing Options**Outgoing PPP Translation****ppp** {*ip-address* / **ip-pool** [**scope-name** *name*]}

Translates from TCP to virtual asynchronous PPP. Supply an IP address as a standard, four-part dotted decimal IP address.

The **ip-pool** keyword obtains an IP address from a Dynamic Host Configuration Protocol (DHCP) proxy client or a local pool. If the **scope-name** keyword is not specified, the address is obtained from a DHCP proxy client. If the **scope-name** keyword is specified, the IP address is obtained from the specified local pool. The **scope-name** keyword can specify a range of IP addresses.

Outgoing PPP Connection Request Options

Any of the following optional keywords can be used to configure PPP connection requests:

- **authentication** {**pap** | **chap**}—Sets Challenge Handshake Authentication Protocol (CHAP) or Password Authentication Protocol (PAP) authentication for PPP on virtual asynchronous interfaces. If you specify both keywords, order is significant; the system will try to use the first authentication type, then the second.
- **header-compression** [**passive**]—Implements header compression on IP packets only. The **passive** keyword permits compression on outgoing packets only if incoming TCP packets on the same virtual asynchronous interface are compressed. The default (without the **passive** keyword) permits compression on all traffic.
- **ipx loopback** *number*—Specifies the loopback interface to be created and permits clients running IPX-PPP to connect through virtual terminal lines on the router. A loopback interface must have been created and configured with a Novell IPX network number before IPX-PPP can work on the virtual terminal line. The virtual terminal line is assigned to the loopback interface.
- **keepalive** *number-of-seconds*—Specifies the interval at which keepalive packets are sent on Serial Line Internet Protocol (SLIP) and PPP virtual asynchronous interfaces. By default, keepalive packets are enabled and sent every 10 seconds. To shut off keepalive packets, use a value of 0. The active keepalive interval is 1 through 32,767 seconds. When you do not change from the default of 10, the keepalive interval does not appear in **more system:running-config** or **show translate** command output.

Table 142 TCP-to-PPP Outgoing Options (continued)

- **mtu bytes**—Sets the interface maximum transmission unit (MTU) of packets that the virtual asynchronous interface supports. The default MTU is 1500 bytes on a virtual asynchronous interface. The acceptable range is from 64 to 1,000,000 bytes.
- **routing**—Permits routing updates between connections. This keyword is required if the destination device is not on a subnet connected to one of the interfaces on the router.
- **use-tacacs**—Uses TACACS to verify PPP authentications for CHAP or PAP on virtual asynchronous interfaces.

Table 143 TCP-to-SLIP Outgoing Options**Outgoing SLIP Translation**

slip {*ip-address* | **ip-pool** [**scope-name** *name*]}

Translates from TCP to virtual asynchronous SLIP. Supply an IP address as a standard, four-part dotted decimal IP address.

The **ip-pool** keyword obtains an IP address from a DHCP proxy client or a local pool. If the optional **scope-name** keyword is not specified, the address is obtained from a DHCP proxy client. If the **scope-name** keyword is specified, the IP address is obtained from the specified local pool. The **scope-name** keyword can specify a range of IP addresses.



Note The **slip** keyword applies only to outgoing connections; SLIP is not supported on incoming protocol translation connections.

Outgoing SLIP Connection Request Options

Any of the following optional keywords can be used to configure SLIP connection requests:

- **header-compression** [**passive**]—Implements header compression on IP packets only. The **passive** keyword permits compression on outgoing packets only if incoming TCP packets on the same virtual asynchronous interface are compressed. The default (without the **passive** keyword) permits compression on all traffic.
- **ipx loopback** *number*—Specifies the loopback interface to be created and permits clients running IPX-PPP over X.25 to connect through virtual terminal lines on the router. A loopback interface must have been created and configured with a Novell IPX network number before IPX-PPP can work on the virtual terminal line. The virtual terminal line is assigned to the loopback interface.
- **keepalive** *number-of-seconds*—Specifies the interval at which keepalive packets are sent on SLIP and PPP virtual asynchronous interfaces. By default, keepalive packets are enabled and sent every 10 seconds. To shut off keepalive packets, use a value of 0. The active keepalive interval is 1 through 32,767 seconds. When you do not change from the default of 10, the keepalive interval does not appear in **more system:running-config** or **show translate** command output.
- **mtu bytes**—Sets the interface MTU of packets that the virtual asynchronous interface supports. The default MTU is 1500 bytes on a virtual asynchronous interface. The acceptable range is from 64 to 1,000,000 bytes.
- **routing**—Permits routing updates between connections. This keyword is required if the destination device is not on a subnet connected to one of the interfaces on the router.

Table 144 TCP-to-X.25 Outgoing Options**Outgoing X.25 Translation****x25** *x.121-address*

Translates TCP to the X.25 protocol. Supply an X.121 address that conforms to the specifications provided in the *CCITT 1984 Red Book*, or the name of an X.25 host that can be resolved by the DNS, or explicit specification in an **x25 host** command.

The address number generally consists of a portion that is administered by the public data network (PDN) and a portion that is locally assigned. You must be sure that the numbers that you assign agree with the addresses assigned to you by the X.25 service provider. The X.121 addresses generally will be subaddresses of the X.121 address for the X.25 network interface.

Outgoing X.25 Connection Request Options

Any of the following optional keywords can be used to configure X.25 connection requests:

- **cud** *c-u-data*—Sends the specified X.25 Call User Data (CUD) text as part of an outgoing call request after the protocol identification bytes.
- **no-reverse**—Specifies that outgoing calls not request the X.25 reverse charge facility, when the interface default is that all outgoing calls are reverse charged.
- **profile** *profile*—Sets the X.3 packet assembler/disassembler (PAD) parameters as defined in the profile created by the **x29 profile** command.
- **pvc** *number* [**interface** *serial number* | **packetsize** *in-size out-size* | **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 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 *in-size* and *out-size*.
 - **no-reset**—Causes the Cisco router to send a no Reset packet request at startup of a TCP or LAT to PVC translation session.
- **reverse**—Provides reverse charging for X.25 on a per-call rather than a per-interface basis. Requests reverse charges on a specified X.121 address, even if the serial interface is not configured to request reverse charge calls.
- **use-map**—Applies **x25 map pad** command entry options (such as CUD and idle) and facilities (such as packet in, packet out, win in, and win out) to the outgoing protocol translation call. When the **use-map** keyword is specified on the **translate** command, the Destination address and optional PAD Protocol Identification (PID), CUD, and facilities are checked against a configured list of **x25 map pad** command entries. If a match is found, the map entry PID, CUD, and facilities are applied to the outgoing protocol translation call. The X.25 map facilities applied to the outgoing translation can be displayed with the **show translation** command throughout the duration of the translation session.

Examples

The following example illustrates the use of the TCP incoming protocol **printer** keyword for an incoming TCP connection:

```
translate tcp 172.19.32.250 printer x25 5678
```

The following example permits clients running IPX-PPP to connect through the device virtual terminal lines to a server running PPP:

```
interface loopback0
  no ip address
  ipx network 544
  ipx sap-interval 2000
!
translate tcp 172.21.14.67 port 1234 ppp 10.0.0.2 ipx loopback0
```

Related Commands

Command	Description
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.

The command syntax that follows shows how to apply a virtual interface template in place of outgoing **translate** options. If you are using virtual templates for protocol translation, all outgoing options are defined in the virtual interface template.

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

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

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 <i>number</i>	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	Release	Modification
	10.0	This command was introduced.

Usage Guidelines You define the protocol translation connections by choosing a protocol keyword and supplying the appropriate address, host name, or service name. The protocol connection information is followed by optional features for that connection, as appropriate. For example, the **binary** keyword is only appropriate with TCP/IP connections. The global options, in general, apply to all the connection types, but there are exceptions.

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 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 conform 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 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 not configured, the CUD in the Incoming Call packet must be blank. • idle minutes—Specifies the number of minutes the virtual circuit is idle. This keyword enables the protocol translation function to clear a switched virtual circuit after a set period of inactivity, where <i>minutes</i> is the number of minutes in the period. Calls either originated or terminated are cleared. The maximum value of <i>minutes</i> is 255. The default value of <i>minutes</i> is zero. • printer—Supports local-area transport (LAT) and TCP printing over an X.25 network among multiple sites. Provides an “interlock mechanism” between the acceptance of an incoming X.25 connection and the opening of an outgoing LAT or TCP connection. This keyword causes the Cisco IOS software to delay the call confirmation of an incoming X.25 call request until after the outgoing protocol connection (to TCP or LAT) has been successfully established. An unsuccessful outgoing connection attempt to the router results in the incoming X.25 connection being refused, rather than being accepted and then closed, which is the default behavior. Note that using this keyword will force the global quiet keyword to be applied to the translation.

- **profile profile**—Sets the X.3 packet assembler/disassembler (PAD) parameters as defined in the profile created by the **x29 profile** command.
- **pvc number [interface serial number | packetsize in-size out-size | window-size in-size out-size]**—Specifies that the outgoing connection is actually a PVC. The *number* argument specifies the virtual circuit channel number of the connection, which must be less than the virtual circuits assigned to the switched virtual circuit (SVC). Only one session is allowed per PVC. Use the following optional keywords to further define the connection:
 - **interface serial number**—Specifies a PVC interface on which to set up the PVC connection.
 - **packetsize in-size out-size**—Specifies the input packet size (*in-size*) and output packet size (*out-size*) for the PVC. Valid packet size values are as follows: 16, 32, 64, 128, 256, 512, 1024, 2048, and 4096.
 - **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*.

protocol
outgoing-address

A protocol name followed by an address or host name. Protocol translation choices are **lat**, **ppp**, **slip**, and **tcp**.



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 148 for more information about the **host-name** keyword.

Additional keywords that can be entered with the protocol are as follows:

- **autocommand**—Specifies an EXEC command for an outgoing connection. The command executes upon connection to a host. You can issue any EXEC command and any switch or host name as an argument to the **autocommand** keyword. If the string following **autocommand** has one or more spaces as part of the string, you must place quotation marks (“ ”) around the string. If you want to enable AppleTalk Remote Access (ARA) on an outgoing connection, specify the **autocommand arap** keywords. These keywords are necessary for ARA because ARA does not use addressing, and this option permits you to invoke the ARA string.
- **virtual-template**—Associates a virtual template with a virtual access interface. See the **translate x25 (virtual access interfaces)** command description for more information.

<i>outgoing-options</i>	(Optional) Outgoing connection request option. Choices depend upon the protocol or command entered. See Table 145, Table 146, Table 147, and Table 148 for a list of outgoing protocol translation options.
<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. • 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.

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 1.1.1.1
```

However, the Cisco IOS software provides a broad range of options that support protocol translations in many networking environments. Table 145, Table 146, Table 147, and Table 148 lists the **translate x25** translation options by protocol.

You can also use the Cisco IOS command-line interface to help you understand how these keywords are entered. In global configuration mode, begin entering the **translate** command and add a question mark at each portion of the command to display the options available. Some examples follow:

```
Router(config)# translate x25 ?

WORD X.121 Address pattern

Router(config)# translate x25 66666 ?

accept-reverse  Accept reverse charge on a per-call basis
autocommand     Associate a command with a translation on this connections
cud             Specify the Call User Data (CUD)
idle           Specify VC idle timer
lat            DEC LAT protocol
ppp           Virtual async PPP
printer        Enable non-interactive (implies global quiet)
profile        Use a defined X.3 profile
pvc           An incoming connection is actually a PVC
slip          Virtual async SLIP
tcp           TCP/IP Telnet
virtual-template Associate a virtual template with virtual access interface
x25           X.25

Router(config)# translate x25 66666 tcp 1.1.1.1 ?

access-class    Allow access list parameters to be used by source hosts
binary         Negotiate Telnet binary mode on the connection
host-name      Store the host name rather than its IP address
local         Allow Telnet protocol negotiations not to be translated
login         Require that the user log in before the outgoing connection
              is made
max-users      Limit the number of simultaneous users of the translation
multibyte-IAC  Always treat multiple IACs as telnet command
port          Port Number
quiet         Suppress printing of user-information messages
source-interface Specify source interface
stream        Treat telnet escape characters as data
swap         Allow X.3 parameters to be set on the protocol translator
              by the host originating the X.25 call
```

Table 145 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 146 X.25-to-PPP Outgoing Options**Outgoing PPP Translation**

ppp {*ip-address* / **ip-pool** [**scope-name** *name*]}

Translates from X.25 to virtual asynchronous PPP. Supply an IP address as a standard, four-part dotted decimal IP address.

The **ip-pool** keyword obtains an IP address from a Dynamic Host Configuration Protocol (DHCP) proxy client or a local pool. If the optional **scope-name** keyword is not specified, the address is obtained from a DHCP proxy client. If the **scope-name** keyword is specified, the IP address is obtained from the specified local pool. The **scope-name** keyword can specify a range of IP addresses.

Outgoing PPP Connection Request Options

Any of the following optional keywords can be used to configure PPP connection requests:

- **authentication** {**pap** | **chap**}—Sets Challenge Handshake Authentication Protocol (CHAP) or Password Authentication Protocol (PAP) authentication for PPP on virtual asynchronous interfaces. If you specify both options, order is significant; the system will try to use the first authentication type, then the second.
- **header-compression**—Configures header compression on IP packets only.
- **ipx loopback** *number*—Specifies the loopback interface to be created and permits clients running IPX-PPP over X.25 to connect through virtual terminal lines on the router. A loopback interface must have been created and configured with a Novell IPX network number before IPX-PPP can work on the virtual terminal line. The virtual terminal line is assigned to the loopback interface.
- **keepalive** *number-of-seconds*—Specifies the interval at which keepalive packets are sent on Serial Line Internet Protocol (SLIP) and PPP virtual asynchronous interfaces. By default, keepalive packets are enabled and sent every 10 seconds. To shut off keepalive packets, use a value of 0. The active keepalive interval is 1 through 32,767 seconds. When you do not change from the default of 10, the keepalive interval does not appear in **more system:running-config** or **show translate** command output.
- **mtu** *bytes*—Sets the interface MTU of packets that the virtual asynchronous interface supports. The default MTU is 1500 bytes on a virtual asynchronous interface. The acceptable range is from 64 to 1,000,000 bytes.
- **routing**—Permits routing updates between connections. This option is required if the destination device is not on a subnet connected to one of the interfaces on the router.
- **use-tacacs**—Uses TACACS to verify PPP authentications for CHAP or PAP on virtual asynchronous interfaces.

Table 147 X.25-to-SLIP Outgoing Options**Outgoing SLIP Translation****slip** {*ip-address* / **ip-pool** [**scope-name** *name*]}

Translates from X.25 to virtual asynchronous SLIP. Supply an IP address as a standard, four-part dotted decimal IP address.

The **ip-pool** keyword obtains an IP address from a DHCP proxy client or a local pool. If the optional **scope-name** keyword is not specified, the address is obtained from a DHCP proxy client. If the **scope-name** keyword is specified, the IP address is obtained from the specified local pool. The **scope-name** keyword can specify a range of IP addresses.



Note The **slip** argument applies only to outgoing connections; SLIP is not supported on incoming protocol translation connections.

Outgoing SLIP Connection Request Options

Any of the following optional keywords can be used to configure SLIP connection requests:

- **header-compression** [**passive**]—Implements header compression on IP packets only. The **passive** keyword permits compression on outgoing packets only if incoming TCP packets on the same virtual asynchronous interface are compressed. The default (without the **passive** keyword) permits compression on all traffic.
- **ipx loopback** *number*—Specifies the loopback interface to be created and permits clients running IPX-PPP over X.25 to connect through virtual terminal lines on the router. A loopback interface must have been created and configured with a Novell IPX network number before IPX-PPP can work on the virtual terminal line. The virtual terminal line is assigned to the loopback interface.
- **keepalive** *number-of-seconds*—Specifies the interval at which keepalive packets are sent on SLIP and PPP virtual asynchronous interfaces. By default, keepalive packets are enabled and sent every 10 seconds. To shut off keepalive packets, use a value of 0. The active keepalive interval is 1 through 32,767 seconds. When you do not change from the default of 10, the keepalive interval does not appear in **more system:running-config** or **show translate** command output.
- **mtu** *bytes*—Sets the interface MTU of packets that the virtual asynchronous interface supports. The default MTU is 1500 bytes on a virtual asynchronous interface. The acceptable range is from 64 to 1,000,000 bytes.
- **routing**—Permits routing updates between connections. This keyword is required if the destination device is not on a subnet connected to one of the interfaces on the router.

Table 148 X.25-to-TCP Outgoing Options**Outgoing TCP Translation****tcp** *ip-address*

Translates X.25 to TCP/IP Telnet. Supply an IP address as a standard, four-part dotted decimal IP address, or the name of an IP host that can be resolved by the DNS, or explicit specification in an **ip host** command (refer to the description for the **host-name** keyword in the “Outgoing TCP Connection Request Options” section).

Table 148 X.25-to-TCP Outgoing Options (continued)**Outgoing TCP Connection Request Options**

Any of the following optional keywords can be used to configure TCP connection requests:

- **binary**—Negotiates Telnet binary mode on the connection.
- **host-name**—Stores the host name rather than its IP address, thereby allowing the host name to be resolved at connection time instead of configuration time. There is also a **rotor** keyword suboption that you can use to modify the behavior of the **host-name** keyword by allowing one of the IP addresses defined by the **ip host** configuration command to be chosen randomly. If one address fails, another one will be tried, and so on until all address choices are exhausted. You can use the **rotor** keyword, therefore, to provide basic load sharing of the IP destinations.
- **multibyte-IAC**—Always treat multiple Interpret as Command (IAC) escape character codes as a Telnet command.
- **port number**—For incoming connections, enter the number of the port to match. The default is port 23 (Telnet).
- **source-interface**—Specifies the source address used for Telnet connections initiated by the router.
- **stream**—Performs stream processing, which enables a raw TCP stream with no Telnet control sequences. A stream connection does not process or generate any Telnet options, and also prevents Telnet processing of the data stream. This option might be useful for connections to ports running UNIX-to-UNIX Copy Program (UUCP) or other non-Telnet protocols, or to ports connected to printers. For ports connected to printers using Telnet, the **stream** keyword prevents some of the problems associated with using Telnet for printers, such as unusual events happening to carriage returns or line feeds and echoing of data back to VMS systems.

Protocol Translation and X.25 PVCs Functional Description

This section describes how the protocol translator works with X.25 PVCs. It will help you understand the overall behavior of incoming and outgoing X.25 PVCs associated with a **translate** command, enabling you to correctly configure protocol translator PVCs for your application.

Configuring X.25 PVCs

When the **translate x25** command is configured with a PVC, an attempt is made to create the PVC. The following conditions can cause this attempt to fail:

- The PVCs assignment of the X.25 interface does not include the PVC number in the **translate x25** command.
- The PVC number in the **translate x25** command is already in use.
- An X.25 destination in a **translate x25** command is routed to X.25 over TCP/IP (XOT), Connection Mode Network Service (CMNS), or Annex G, which do not support translated PVCs.

PVC numbers must be unique across an X.25 connection; however, PVC 1 on Serial 1/0 is different (and therefore unique) from PVC 1 on Serial 2/1.

If, once the **translate x25** command is accepted, the X.25 interface on which the PVC is created goes down, the PVC enters an inactive state, the TCP or LAT connection is terminated, but the existing PAD context remains inactive.

An incoming TCP or LAT connection associated with a down outgoing PVC (displaying a “P/Inactive message”) will be rejected by the protocol translator.

If any X.25 traffic is received while the corresponding TCP or LAT connection is terminated, and if a Data packet is received in state D1, a Reset with a diagnostic message will be displayed, similar to the following:

```
20:17:11.809: Serial2: X.25 O D1 Reset (5) 8 lci 4
20:17:11.809: Cause 29, Diag 113 (Network out of order (PVC)/Remote network problem)
```

The number of outgoing and incoming protocol translation PVCs is limited only by the number of virtual terminal lines supported on the Cisco router. Remember that each protocol translation session uses a virtual terminal line, which lowers the number of virtual terminal lines available for Telnet sessions.

By default, the Cisco router sends a Reset packet with the cause “PVC Network Operational” and diagnostic “Maintenance action” messages at the start of a TCP or LAT to PVC translation session, to announce that the connection is established and that the PVC is able to handle data traffic. To suppress the PVC Reset packet at TCP or LAT session startup, configure the **no-reset** outgoing PVC keyword as shown in the following example:

```
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 Serial12/0
```

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 Serial11/5
translate tcp 10.0.155.61 port 2502 x25 32785223344502 pvc 1
```

You can simply enter one **translate** command that links the IP port number with the X.121 address and specifies the interface on which to establish the PVC, as follows:

```
translate tcp 10.0.155.61 port 2502 x25 2502 pvc 1 interface Serial 1/5
```

This is the recommended approach and should be adopted in place of **translate** commands that cause the destination address to be looked up in the route table.

Examples

The following example shows how to use the **translate** global configuration command to translate from an X.25 PAD to a LAT device on Network A. It is applied to Router-A. The configuration example includes an access list that limits remote LAT access through Router-A to connections from PAD-C. This example typifies the use of access lists in the Cisco IOS software. The first two lines define the scope of access-list 1. The first line specifies that access list 1 will permit all calls from X.121 address 44444. The caret symbol (^) specifies that the first number 4 is the beginning of the address number. The second line of the definition explicitly denies calls from any other number. (Refer to the appendix “Regular Expressions” in the *Cisco IOS Terminal Services Configuration Guide*, Release 12.2, for details concerning the use of special characters in defining X.121 addresses.)

```
! 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 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
```

Related Commands

Command	Description
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.

The command syntax that follows shows how to apply a virtual interface template in place of outgoing **translate x25** options. If you are using virtual templates for protocol translation, all outgoing options are defined in the virtual interface template. Table 149 lists all outgoing options and their corresponding interface configuration commands.

```
translate x25 incoming-address [incoming-options [pvc number [pvc-options]]] protocol
    outgoing-address [outgoing-options] virtual-template number [global-options]
```

```
no translate x25 incoming-address [incoming-options [pvc number [pvc-options]]] protocol
    outgoing-address [outgoing-options] virtual-template number [global-options]
```

Syntax Description

<i>incoming-address</i>	<p>An X.25 and X.121 address that conform 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 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) Incoming connection request keywords and arguments, 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. This is an incoming option only. • cud c-u-data—Sends the specified Call User Data (CUD) text (<i>c-u-data</i>) as part of an outgoing call request after the protocol identification bytes. • 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 [interface serial number | packetsize in-size out-size | window-size in-size out-size]**—Specifies that the outgoing connection is actually a PVC. The *number* argument specifies the virtual circuit channel number of the connection, which must be less than the virtual circuits assigned to the switched virtual circuit (SVC). Only one session is allowed per PVC. Use the following optional keywords and arguments to further define the connection:
 - **interface serial number**—Specifies a PVC interface on which to set up the PVC connection.
 - **packetsize in-size out-size**—Specifies the input packet size (*in-size*) and output packet size (*out-size*) for the PVC. Valid packet size values are as follows: 16, 32, 64, 128, 256, 512, 1024, 2048, and 4096.
 - **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*.

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 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 149 maps outgoing **translate** command options to interface commands you can configure in the virtual interface template.

Table 149 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.

transport input { **all** | **lat** | **mop** | **nasi** | **none** | **pad** | **rlogin** | **ssh** | **telnet** | **v120** }

Syntax Description

all	Selects all protocols.
lat	Selects the Digital 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 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	Select the V.120 protocol for incoming async over ISDN connections.

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 .

Usage Guidelines

Cisco routers do not accept incoming network connections to asynchronous ports (TTY lines) by default. You have to 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 restores pre-Cisco IOS software Release 11.0 defaults. If you are upgrading to Cisco IOS software version 11.1(1) or later from Cisco IOS software Release 11.0 or earlier, you must add 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 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 vty 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.

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	Select 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 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	Select the V.120 protocol for outgoing asynchronous over ISDN connections.

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.

transport preferred { **all** | **lat** | **mop** | **nasi** | **none** | **pad** | **rlogin** | **telnet** | **v120** }

Syntax Description	all	Specifies all recognized protocols.
	lat	Selects the Digital LAT protocol, which is the protocol used most often to connect routers to Digital hosts.
	mop	Specifies the Maintenance Operation Protocol (MOP).
	nasi	Specifies the 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 PAD, used most often to connect routers to X.25 hosts.
	rlogin	Selects the UNIX rlogin protocol for TCP connections. The rlogin setting is part of the Telnet protocol. If an rlogin attempt to a particular host fails, the failure is 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	Specifies 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. Use the **no** form of this command to delete any named ttycap entry from the configuration file.

ttycap *ttycap-name termcap-entry*

no ttycap *ttycap-name*

Syntax Description

<i>ttycap-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 150. For the definitions of supported string-sequence *ttycap* capabilities, see Table 151. For the definitions of supported number-sequence *ttycap* capabilities, see Table 152. For the definitions of supported color-sequence *ttycap* capabilities, see Table 153.

Table 150 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 151 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

Table 151 Definitions of *ttycap* Capabilities: String Sequences (continued)

String Sequence	Description
cm	Move cursor to row # and col #
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 152 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 153 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 chapter “Configuring Dial-In Terminal Services” in the *Cisco IOS Dial Services Configuration Guide: Terminal Services* 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.

tunnel

To set up a network layer connection to a router, use the **tunnel** user EXEC command.

tunnel *host*

Syntax Description	<i>host</i>	Name or IP address of a specific host on a network that can be reached by the router.
---------------------------	-------------	---

Command Modes	User EXEC
----------------------	-----------

Command History	Release	Modification
	10.0	This command was introduced.

Usage Guidelines

If you are a mobile user, it is often impractical to dial in to your “home” router from a remote site. The asynchronous mobility feature allows you to dial in to different routers elsewhere on the internetwork while experiencing the same server environment that you would if you were connecting directly to your home router.

This asynchronous host mobility is accomplished by packet tunneling, a technique by which raw data from the dial-in user is encapsulated and transported directly to the host site where your home router performs the actual protocol processing.

You enable asynchronous mobility by entering the **tunnel** command to set up a network layer connection to a specified host. From a router other than a Cisco router, however, you need to use the Telnet protocol.

After a connection is established, you receive an authentication dialog or prompt from your home router and can proceed as if you are connected directly to it. When communications are complete, the network connection can be closed and terminated from either end of the connection.

Examples

The following example establishes a network layer connection with an IBM host named mktg:

```
router> tunnel mktg
```

txspeed

To set the terminal transmit speed (how fast the terminal sends information to the modem), use the **txspeed** command in line configuration mode.

txspeed *bps*

Syntax Description	<i>bps</i>	Baud rate in bits per second (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.

virtual-profile aaa

To enable virtual profiles by authentication, authorization, and accounting (AAA) configuration, use the **virtual-profile aaa** command in global configuration mode. Use the **no** form of this command to disable virtual profiles.

virtual-profile aaa

no virtual-profile aaa

Syntax Description This command has no arguments or keywords.

Defaults Disabled

Command Modes Global configuration

Command History	Release	Modification
	11.2F	This command was introduced.
	12.0(7)T	This command was enhanced to allow virtual profiles to be downloaded from an AAA server using the HDLC, LAPB-TA, X.25, and Frame Relay encapsulations, in addition to the originally supported PPP encapsulation.

Usage Guidelines Effect of this command for any specific user depends on the router being configured for AAA and the AAA server being configured for that user's specific configuration information.

Examples The following example configures virtual profiles by AAA configuration only:

```
virtual-profile aaa
```

Related Commands	Command	Description
	aaa authentication	Enables AAA authentication to determine if a user can access the privileged command level.
	virtual-profile if-needed	Enables virtual profiles by virtual interface template.

virtual-profile if-needed

To specify that a virtual profile be used to create a virtual access interface only if the inbound connection requires a virtual access interface, use the **virtual-profile if-needed** command in global configuration mode. Use the **no** form of this command to create virtual access interfaces for every inbound connection.

virtual-profile if-needed

no virtual-profile if-needed

Syntax Description This command has no arguments or keywords.

Defaults Disabled

Command Modes Global configuration

Command History	Release	Modification
	12.0(5)T	This command was introduced.

Usage Guidelines This command is intended to prevent the creating of virtual-access interfaces for inbound calls on physical interfaces that do not require virtual-access interfaces.

This command is compatible with local, RADIUS, and TACACS+ AAA.

Examples The following example enables selective virtual-access interface creation:

```
virtual-profile if-needed
```

Related Commands	Command	Description
	interface virtual-template	Creates a virtual template interface that can be configured and applied dynamically in creating virtual access interfaces.
	virtual-profile virtual-template	Enables virtual profiles by virtual interface template.
	virtual-profile aaa	Enables virtual profiles by AAA configuration.

virtual-profile virtual-template

To enable virtual profiles by virtual interface template, use the **virtual-profile virtual-template** command in global configuration mode.

virtual-profile virtual-template *number*

Syntax Description	<i>number</i> Number of the virtual template to apply, in the range 1 to 30.
---------------------------	--

Defaults	Disabled. No virtual template is defined, and no default virtual template number is used.
-----------------	---

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

Command History	Release	Modification
	11.2F	This command was introduced.

Usage Guidelines	<p>When virtual profiles are configured by virtual templates only, any interface-specific configuration information that is downloaded from the AAA server is ignored in configuring the virtual access interface for a user.</p> <p>The interface virtual-template command defines a virtual template to be used for virtual profiles. Because several virtual templates might be defined for different purposes on the router (such as MLP, PPP over ATM, and virtual profiles), it is important to be clear about the virtual template number to use in each case.</p>
-------------------------	--

Examples	<p>The following example configures virtual profiles by virtual templates only. The number 2 was chosen because virtual template 1 was previously defined for use by Multilink PPP.</p>
-----------------	---

```
virtual-profile virtual-template 2
```

Related Commands	Command	Description
	interface virtual-template	Creates a virtual template interface that can be configured and applied dynamically in creating virtual access interfaces.

virtual-template

To specify which virtual template will be used to clone virtual access interfaces, use the **virtual-template** command in accept-dialin configuration mode. To remove the virtual template from an accept-dialin Virtual Private Dialup Network (VPDN) subgroup, use the **no** form of this command.

virtual-template *template-number*

no virtual-template

Syntax Description	<i>template-number</i> Number of the virtual template that will be used to clone virtual-access interfaces.
---------------------------	---

Defaults	Disabled
-----------------	----------

Command Modes	Accept-dialin configuration
----------------------	-----------------------------

Command History	Release	Modification
	12.0(5)T	This command was introduced.

Usage Guidelines	<p>Each accept-dialin group can only clone virtual-access interfaces using one virtual template. If you enter a second virtual-template command on an accept-dialin subgroup, it will replace the first virtual-template command.</p> <p>You must first enable a tunneling protocol on the accept-dialin VPDN subgroup (using the protocol command) before you can enable the virtual-template command. Removing or modifying the protocol command will remove virtual-template command from the request-dialin subgroup.</p>
-------------------------	---

Examples	<p>The following example enables the LNS to accept an L2TP tunnel from a LAC named mugsy. A virtual-access interface will be cloned from virtual template 1:</p> <pre>vpdn-group 1 accept dialin protocol l2tp virtual-template 1 terminate-from hostname mugsy</pre>
-----------------	---

Related Commands	Command	Description
	accept dialin	Specifies the LNS to use for authenticating, and the virtual template to use for cloning, new virtual access interfaces when an incoming L2TP tunnel connection is requested from a specific peer.