

Chapter 1

Protocol Translation

1

This chapter focuses on the configuration of Cisco's one-step method protocol translation facility. In addition, it provides configuration examples for typical protocol translation environments. You will find the following information in this chapter:

- Overview of one-step translation methods supported by Cisco's protocol translation software.
- Definition of the **translate** configuration command and its options.
- Examples illustrating command use.
- Extensive examples illustrating protocol translation applications.

Making two-step protocol translation connections is described in the "Terminal Server User Commands" chapter.

Cisco's Protocol Translation Options

Cisco's protocol translator (also called application gateway) function translates virtual terminal protocols to allow devices running dissimilar protocols to communicate. Cisco protocol translation software supports Telnet (TCP), LAT, and X.25. One-step protocol translation software performs bidirectional translation between any of the following protocols:

- Telnet <—> LAT
- Telnet <—> X.25
- LAT <—> X.25

Note: TN3270 and XRemote are also supported by the protocol translation software. However, to translate between these and other supported protocols, you must use the two-step method. Refer to the "Terminal Server User Commands" chapter for information concerning two-step translations in general. Refer to the chapters "TN3270 Configuration and Management" and "XRemote Configuration and Management," for information about TN3270 and XRemote, respectively.

The translation of virtual terminal protocols allows *connectivity* to devices running different protocol stacks. This option does not perform translation between other protocol services such as file transfer protocols.

Protocol translation is available in these Cisco products:

- Standalone protocol translator configurations—CPT and IPT.
- As an optional feature with the ASM and MSM chassis-based terminal server configurations with CSC/16 cards.
- As an option on the IGS router.

The chassis-based terminal server configurations can support up to 96 lines, which are divided up as either asynchronous terminal lines or protocol translation sessions. For example, the ASM terminal server offers 16 of its 96 lines for protocol translation sessions, and the MSM supports up to 64 protocol translation sessions.

This chapter assumes a working knowledge of Cisco protocol translation and software configuration conventions. It touches on specific configuration procedures for specifying system-wide facilities, as well as **translate** command examples. Before continuing with this chapter, be sure that you are familiar with the system configuration, system management, LAT, TCP, and X.25 descriptions provided in this publication. Refer to the “Terminal Server User Commands” chapter, for more information concerning one-step and two-step protocol translation procedures.

Configuring One-Step Translations

The ability to make single-step user connections to resources through Cisco terminal servers is made possible by the **translate** global configuration command. This section describes the **translate** command specifications and provides information about the use of protocol-specific incoming and outgoing addressing options. Examples of translation configuration commands and files are provided at the end of this section.

Using the Translate Command

The **translate** command causes incoming LAT, TCP, or X.25 requests for connections to a specified destination address/host name to result in the automatic conversion to the specified outgoing connection type. The connection request to the address specified in the **translate** command is executed.

Note: The **translate** command does not support TN3270 or XRemote one-step translations. To use the terminal server to convert these protocols, you must use the two-step method as described in the “Terminal Server User Commands” chapter. Also, refer to the other chapters in this part of the publication for details concerning connection services and options available for all protocols supported by the terminal server.

The **translate** command causes incoming connections to the specified input protocol and address to result in automatic formation of an outgoing connection to the specified protocol and address.

The **translate** global command has the following general syntax:

translate *protocol inaddress [inoptions]* *protocol outaddress [outoptions]* [*globaloptions*]

The first part of the command describes the expected incoming protocol, the protocol’s address or service name, followed by options that may be selected for that protocol, as needed or appropriate.

The next part of the command describes the outgoing protocol and its address or service name, also followed by options that may be selected for that protocol.

The last part of the command lists global options that apply to the entire command.

Table 1-1 illustrates the possible keywords, arguments, and options for the **translate** command. These keywords and arguments are described in the following sections.

Table 1-1 The Translate Command Options

	Incoming Protocol	Options	Outgoing Protocol	Options	Global Options
translate	<i>protocol inaddress</i>	[<i>inoptions</i>]	<i>protocol outaddress</i>	[<i>outoptions</i>]	[<i>globaloptions</i>]
	lat <i>servicename</i>	unadvertised	lat <i>servicename</i>	node <i>nodename</i>	access-class <i>number</i> max-users <i>number</i> local login quiet swap
				port <i>portname</i>	
	x25 <i>x121 address</i>	cud <i>c-u-data</i> profile <i>profile</i> reverse pvc <i>number</i>	x25 <i>x121 address</i>	cud <i>c-u-data</i> profile <i>profile</i> reverse	
	tcp <i>IPaddress</i>	port <i>portnumber</i> binary	tcp <i>IPaddress</i>	port <i>portnumber</i>	

Protocol Translation Options

Select one of the following keywords to supply the incoming protocol (*protocol inaddress*), and the outgoing protocol translation option (*protocol outaddress*). You must specify a protocol name (**tcp**, **lat**, or **x25**) followed by an address or service name:

- **lat** *servicename*—Specifies LAT. The argument *service-name* identifies a LAT service. The application of the service name can differ, depending on whether it is being used on the incoming portion (*inaddress*) or the outgoing (*outaddress*) portion of the command. When used on the *inaddress* portion, the service name is the name of the service that users specify when trying to make a translated connection. This name can, but does not need to, match the name of final destination resource. This can be useful when making remote translated connections.
- **x25** *X.121address*—Specifies X.25. The argument *X.121address* is an X.121 address, up to 14 decimal numbers in length. It must conform to specifications provided in the *CCITT 1984 Red Book*. This number 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 are in agreement with addresses assigned to you by the X.25 service provider.

The X.121 addresses used in the **translate** command will generally be subaddresses of the X.121 address of 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 terminal server itself is responsible for dealing with the extra digits in an appropriate way.

- **tcp** *IP-address*—Specifies TCP/IP Telnet. The argument *IPaddress* is a standard dotted decimal IP address or the name of an IP host that can be resolved via Domain Name Service, IEN116 name service, or by explicit specification in an **ip host** command.

Telnet/TCP Translation Options

You can select these incoming and outgoing options (*inoptions* and *outoptions*) for the Telnet/TCP translations, as appropriate:

- **port** *number*—Specifies outgoing TCP connections, or for matching only this port on incoming connections. The default port is port 23 (Telnet). For incoming connections, port 0 will match *any* port. The argument *number* specifies the desired port number.
- **binary**—Negotiates Telnet binary mode on the Telnet connection. (This was the default in previous versions of the terminal server software, and is set automatically when an old format **translate** subcommand is specified.)

LAT Translation Options

You can select these incoming and outgoing options (*inoptions* and *outoptions*) for the LAT translations, as appropriate:

- **node** *nodename*—Causes a connection to be made to the specified node (*nodename*) which offers a service. By default, the connection is made to the highest-rated node which offered the service.
- **port** *portname*—Specifies a destination LAT port name (argument *portname*) in the format of the remote system. This parameter is usually ignored in most timesharing systems, but is used by terminal servers which offer reverse-LAT services.
- **unadvertised**—Prevents service advertisements from being broadcast to the network. This 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.)

Note: The **node** and **port** keywords and arguments apply only to the destination. These options can also be combined.

X.25 Translation Options

You can select these incoming and outgoing options (*inoptions* and *outoptions*) for X.25 translation, as appropriate:

- **culd** *c-u-data*—Sends the specified Call User Data (CUD) text (*c-u-data*) as part of an outgoing call request after the protocol identification bytes.
- **profile** *profile*—Causes the X.3 PAD parameters to be set as defined in the profile created by the **x29 profile** subcommand. Creating these profiles is described in the “X.25 Configuration and Management” chapter.
- **reverse**—Provides reverse charging for X.25 on a per-call rather than a per-interface basis.
- **pvc** *number*—Specifies that the incoming connection (identified by the argument *number*) is actually a permanent virtual circuit (PVC).

Global Translation Options

Global options for the **translate** subcommand are as follows:

- **access-class** *number*—Allows the **translate** subcommand to be used by source hosts that match the access list parameters. This is currently supported only for incoming TCP and X.25 connections. The argument *number* is the number (integer) previously assigned to an access list.

- **local**—Allows Telnet protocol negotiations (such as ECHO and Line mode) to *not* be translated.
- **login**—Requires that the user log in before the outgoing connection is made. This type of login is specified on the VTY's using the **login** command.
- **quiet**—Suppresses printing of user-information messages.
- **max-users** *number*—Limits the number of simultaneous users of the translation to *number* (an integer you specify).
- **swap**—For X.25 to TCP translations only. Allows X.3 parameters to be set on the terminal server by the host originating the X.25 call, or by an X.29 profile. This allows incoming and outgoing X.25 connections to be swapped so that the terminal server is treated like a PAD when it accepts a call. By default, the terminal server behaves like a PAD for calls that it initiates, and behaves 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 terminal server, and then treats it like a PAD.

Protocol Translation Configuration Examples

This section provides examples of how to enter the **translate** command, and then how to create a protocol translation configuration file.

Translate Command Examples

The following examples illustrate various ways in which the **translate** command can be entered. (The lines denote parts of the command, and are for illustration purposes only.)

Example 1:

This example illustrates a simple X.25 to TCP translation command. Packets coming in X.25 address *652365123* arrive via PVC 1 and are translated to TCP packets and transmitted out IP address *131.108.1.1*.

```
translate x25 652365123 pvc 1 tcp 131.108.1.1
          incoming      option  outgoing
```

Example 2:

This example illustrates incoming LAT to outgoing TCP translations. The **unadvertised** keyword prevents broadcast of service advertisements to other servers. Outgoing translated packets are transmitted out IP address *rubble* via TCP port *4005*.

```
translate lat pt-printer1 unadvertised tcp rubble port 4005
          incoming      option      outgoing  option
```

Example 3:

This example illustrates a more complex **translate** command that calls an X.29 profile and swaps the default PAD operation of the terminal server to that of an X.25 host.

```
x29 profile fullpackets 2:0 3:0 4:100 7:21
translate x25 217536124 profile fullpackets tcp rubble port 4006 swap
          incoming option outgoing option global
```

Protocol Translation Application Examples

This section provides protocol translation examples for these applications:

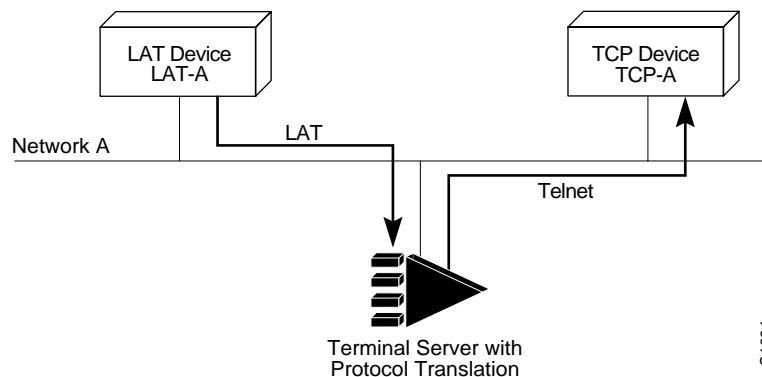
- Local LAT to TCP translations
- Standalone LAT to TCP translations
- LAT to TCP over a wide area network (WAN)
- LAT to LAT over a WAN

Note: In the application illustrations that follow, source and destination device icons used to illustrate the flow of translated information are shown with black type in outlined shapes. Other elements in the environment are shown with reverse type on solid black shapes.

Local LAT to TCP Translation

This example illustrates a simple LAT to TCP translation across an Ethernet network. The name *TCPA* is the logical name given to device TCP-A.

Figure 1-1 Translating Local LAT to TCP



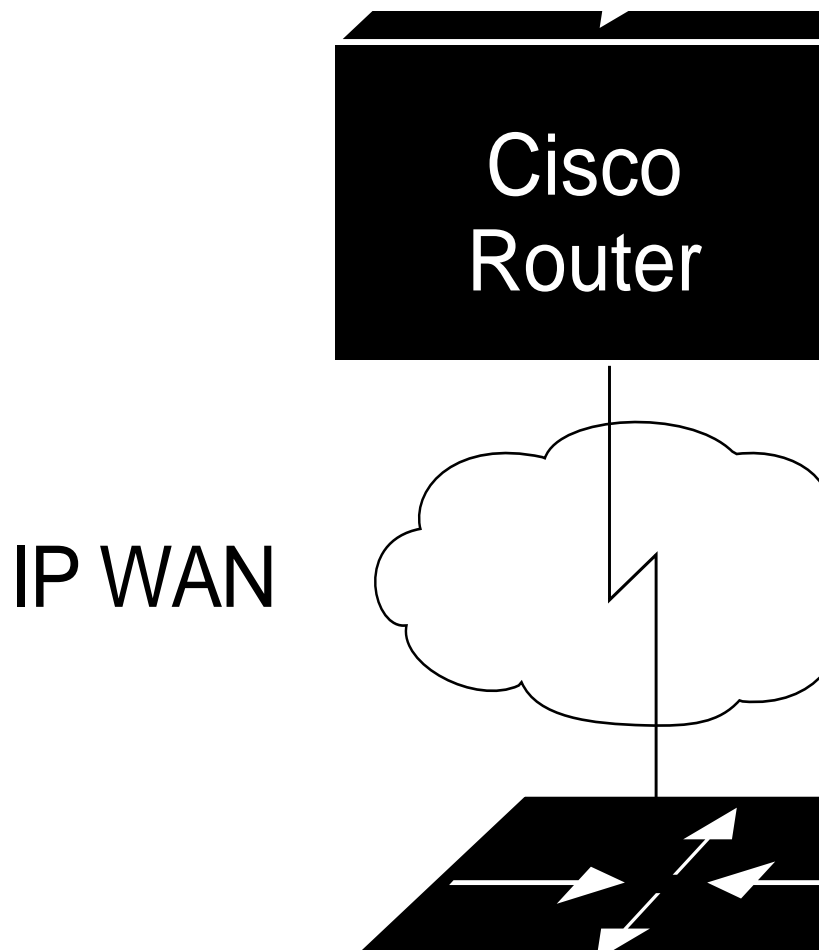
Configuration:

```
interface ethernet 0
ip address 1.0.0.2 255.255.0.0
!
! enable LAT on this interface
lat enabled
!
translate lat TCPA tcp TCP-A
```

LAT to TCP over a WAN

This example illustrates a configuration that allows translation of LAT to TCP and transmission across an IP-based WAN. The logical name *distant-TCP* is the name given to device TCP-B.

Figure 1-2 Translating LAT to TCP over a WAN



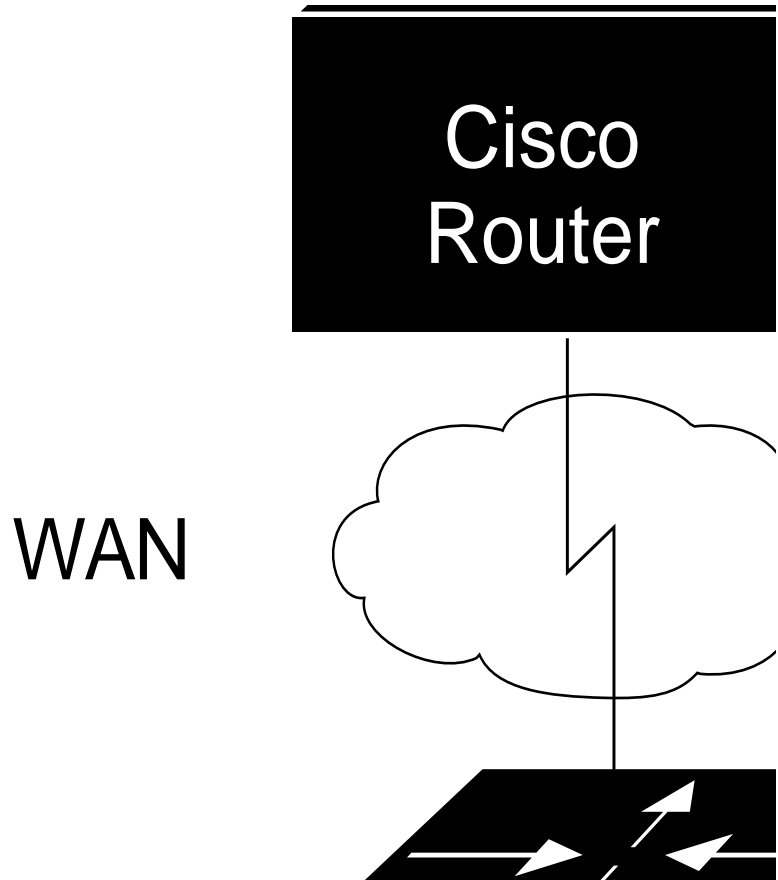
Configuration for TS-A:

```
interface ethernet 0
ip address 1.0.0.2 255.255.0.0
!
! enable LAT on this interface
lat enabled
!
translate lat distant-TCP tcp TCP-B
```

LAT to LAT over a WAN

This configuration allows LAT to be transported to a remote LAT device by translating the packets to TCP format and using Telnet to send them across the WAN. The logical name *TS-B1* is the name given to device TS-B.

Figure 1-3 Translating LAT to LAT over a WAN



Configuration for TS-A:

```
interface ethernet 0
ip address 1.0.0.2 255.255.0.0
!
! enable LAT on this interface
lat enabled
!
translate lat distant-LAT tcp TS-B1
```

Configuration for TS-B:

```
interface ethernet 0
ip address 2.0.0.2 255.255.0.0
!
! enable LAT on this interface
lat enabled
!
translate lat TS-B1 lat LAT-B
```

Monitoring Translation

Use the EXEC **show translate** command described in this section to obtain displays of translation activity.

Examining Translation Configuration

The command **show translate** is used to show information about any protocol translation information configured using the **translate** configuration command. The command syntax is:

show translate

Sample output of the command follows:

```
Translate From: X25 3137005406515 access-class 5
To:    TCP 199.31.7.17 Port 23
0/10 users active, 5 peak, 20 total, 1 failures
```

The first line contains the source protocol for the translation, and the second line contains the destination protocol and global options, all as specified in the original **translate** configuration command.

The last line in the display describes usage statistics for that particular translation. The first number 0 is the number of currently active users acting through the translate command. Following the slash, 10 is the max-users global parameter from the configuration (zero if no maximum was specified). The peak number 5 is the largest number of users concurrently using this translation. The total 20 is the total number of connections through this translation, and failures 1 is the number of attempted translations that failed for any reason (destination host not responding, access failure, login failure, resource exhaustion, and so on).

Additional displays of output follow:

```
Translate From: LAT LUCY
To:    TCP 132.100.2.113 Port 23
1/0 users active, 1 peak, 1 total, 0 failures
Translate From: X25 3137005406515 access-class 5
To:    TCP 199.31.7.17 Port 23
0/0 users active, 0 peak, 0 total, 0 failures
Translate From: LAT MATHOML
To:    TCP 199.31.7.17 Port 23
2/0 users active, 2 peak, 3 total, 0 failures
```

Debugging Translation

There is one privileged debugging command for translation: **debug translate**. It is described in this section and is used to troubleshoot translation activity. Generally, you enter this command during troubleshooting sessions with Cisco Customer Engineers.

Use the **undebug translate** command to turn the display off.

debug translate

The **debug translate** command shows the startups of each type of protocol translation, and indicates failures of the **translate** command.

Translation Global Configuration Command Summary

This section summarizes the global system configuration command **translate**:

translate *inaddress* [*inoptions*]*outaddress* [*outoptions*][*globaloptions*]

Causes incoming connections to the specified input protocol and address to result in the automatic formation of an outgoing connection to the specified protocol and address.

Protocol Options (*inaddress* | *outaddress*)

- **lat** *servicename*—Specifies LAT. The argument *service-name* identifies a LAT service.
- **x25** *X.121address*—Specifies X.25. The argument *X.121address* is an X.121 address, up to 14 decimal numbers in length.
- **tcp** *IP-address*—Specifies TCP/IP Telnet. The argument *IP-address* is a standard dotted-decimal IP address or the name of an IP host that can be resolved via Domain Name Service, IEN116 name service, or by explicit specification in an **ip host** command.

Telnet/TCP Translation Options (*inoptions* | *outoptions*)

- **port** *number*—Specifies outgoing TCP connections, or for matching only this port *number* on incoming connections. Default is port 23 (Telnet). For incoming connections, port zero will match *any* port. The argument *number* specifies the desired port number.
- **binary**—Negotiates Telnet binary mode on the Telnet connection.

LAT Translation Options (*inoptions* | *outoptions*)

- **node** *nodename*—Causes a connection to be made to the specified node (*nodename*) which offers a service. By default, the connection is made to the highest-rated node which offer the service.
- **port** *portname*—Specifies a destination LAT port name (argument *portname*) in the format of the remote system.
- **unadvertised**—Prevents service advertisements from being broadcast to the network.

X.25 Translation Options (*inoptions* | *outoptions*)

- **cud** *c-u-data*—Sends the specified Call User Data (CUD) text (*c-u-data*) as part of an outgoing call request after the protocol identification bytes.
- **profile** *profile*—Causes the X.3 PAD parameters to be set as defined in the profile created by the **x29 profile** subcommand.
- **reverse**—Provides reverse charging for X.25 on a per-call rather than a per-interface basis.
- **pvc** *number*—Specifies that the incoming connection (identified by the argument *number*) is actually a permanent virtual circuit (PVC).

Global Translation Options (*globaloptions*)

- **access-class** *number*—Incoming-only option that allows the **translate** subcommand to be used by source hosts that match the access list parameters. The argument *number* is the number (integer) previously assigned to an access list.
- **local**—Allows Telnet protocol negotiations to *not* be translated.
- **login**—Requires that the user log in before the outgoing connection is made. This type of login is specified on the VTY's using the **login** command.
- **quiet**—Suppresses printing of user-information messages.
- **max-users** *number*—Limits the number of simultaneous users of the translation to *number* (an integer you specify).
- **swap**—For X.25 to TCP translations only. Allows X.3 parameters to be set on the terminal server by the host originating the X.25 call, or by an X.29 profile. Allows connections from an X.25 host that wants to connect to the terminal server, and then treat it like a PAD.