



Protocol Translation Ruleset

The Protocol Translation Ruleset feature provides an effective method for creating Cisco IOS protocol translation configurations by defining a set of statements called a *ruleset*. The ruleset applies pattern matching and substitution technology to use incoming protocol elements, such as a destination address and port, to determine the outgoing protocol elements and translation options specified for originated connections. The ruleset also contains options to control the protocol translation sessions. The Protocol Translation Ruleset feature is especially useful for users that need to configure a large number of translate commands, because it makes it easy to create many individual translate configuration commands using a single ruleset-based command.

Feature History for the Protocol Translation Ruleset Feature

Release	Modification
Cisco IOS XE Gibraltar 16.10.1	This feature was introduced.

Finding Support Information for Platforms and Cisco IOS Software Images

Use Cisco Feature Navigator to find information about platform support and Cisco IOS software image support. Access Cisco Feature Navigator at <http://www.cisco.com/go/fn>. You must have an account on Cisco.com. If you do not have an account or have forgotten your username or password, click **Cancel** at the login dialog box and follow the instructions that appear.

Contents

- [Prerequisites for Using the Protocol Translation Ruleset, page 2](#)
- [Restrictions for a Protocol Translation Ruleset, page 2](#)
- [Information About the Protocol Translation Ruleset, page 2](#)
- [How to Configure a Protocol Translation Ruleset, page 6](#)
- [Configuration Examples for the Protocol Translation Ruleset Feature, page 11](#)
- [Additional References, page 14](#)
- [Command Reference, page 16](#)



Prerequisites for Using the Protocol Translation Ruleset

Understanding how to compose regular expressions for matching patterns in Cisco IOS software configurations and scripts is key to understanding the Protocol Translation Ruleset feature. Composing regular expressions is described in the [Regular Expressions](#) document.

A protocol translation ruleset does not look up the X.25 route table for a matching destination entry. An interface on which to set up the permanent virtual circuit (PVC) must be specified. Protocol translation requires a client to register for PVCs that are available for protocol translation use, whether or not a session is active for the channel. Protocol translation ruleset processing introduced by the Protocol Translation Ruleset feature allows connections only to a PVC that has been reserved for ruleset handling. You must use the **x25 pvc translate ruleset** command to reserve the PVCs.

In a Telnet-to-PAD protocol translation ruleset, an IP address must be specified with the **translate use telnet** command for the protocol translator to respond to Address Resolution Protocol (ARP) attempts for that address. The IP address that the protocol translation software listens for must be on a connected subnet; it cannot be used by another interface unless you also specify a TCP port number, and there cannot be another host that responds to ARPs for that address.

Restrictions for a Protocol Translation Ruleset

The ruleset introduced in the Protocol Translation Ruleset feature allows dynamic construction of the information needed to configure a protocol translation session. It was designed specifically to increase the flexibility of these sessions, especially in large networks with an address plan that can make effective use of pattern matching capability. However, this increased functionality may overload router memory and processing capabilities if it generates large numbers of concurrent sessions or a high aggregate volume of traffic. Memory and performance impact will vary depending upon the particulars of network design and traffic load.

Information About the Protocol Translation Ruleset

Before starting the tasks described in this document, you need to understand the following concepts:

- [Cisco IOS Protocol Translation and Translation by Ruleset, page 3](#)
- [Cisco Regular Expression Pattern Matching, page 3](#)
- [Regular Expression Pattern Matching in a Protocol Translation Ruleset, page 4](#)
- [Error Handling in the Protocol Translation Ruleset, page 6](#)

Cisco IOS Protocol Translation and Translation by Ruleset

The Cisco IOS software provides protocol translation capability that can be used in many types of networks and translate between incoming connection protocols such as TCP/IP, X.25 packet assembler/disassembler (PAD), and local-area transport (LAT), and a set of outgoing protocols that includes TCP/IP, X.25 PAD, LAT, PPP, and Serial Line Internet Protocol (SLIP). Each translation configuration is entered as a single command line, and users can choose from a lengthy list of options to define configurations for specific environments. For some users, however, it is more important to be able to quickly and efficiently define translation connections for a large number of addresses. The Protocol Translation Ruleset feature provides this capability by defining Cisco IOS protocol translation configurations in a ruleset. The ruleset is defined by using regular expression pattern matching and operations that match or ignore incoming connection requests. Substitute, set, and test string writing operations create the connection configurations based on an incoming address. This combination of pattern matching and string writing operations makes it possible to convert, for example, an IP port number to an X.121 address using just a few statements, rather than enter each configuration statement on a separate line.

The protocol translation capability introduced in the Protocol Translation Ruleset feature for Cisco IOS Release 12.3(8)T supports protocol translation from PAD to TCP and from TCP to PAD. Options are available for translations created in the ruleset to define a maximum number of sessions, require login, match an access list, and that suppress translation information messages on the session.

The Cisco IOS Release 12.3(8)T software will accept both the single-line translate commands (such as **translate pad** and **translate tcp**) and their option settings, and protocol translation statements defined in a ruleset, in the same configuration file. The ruleset configuration is applied after the incoming protocol translation connections are tested against the single line translate command configuration, so that you can make use of both the robust protocol translation capability currently available in the Cisco IOS software, and of a protocol translation ruleset that allows quick configuration of a large number of addresses.

The new ruleset environment will seem familiar to users that already know Cisco's single-line translate commands, in that many of the same keywords that are available for these commands are also used in the protocol translation ruleset. A new global configuration command, **translate ruleset**, specifies a name for the ruleset, defines the direction of translation, either from PAD to TCP or from TCP to PAD, and starts translate ruleset configuration mode. The translate ruleset configuration mode allows much flexibility in the number of statements accepted on each line. The mode also accepts multiple statements of the same type. The translate ruleset configuration mode provides **match** and **skip** commands to create statements that look at incoming connection requests to determine if they are valid, and **substitute**, **set**, and **test** commands for string writing operations that will help configure the translation session.

To assist you with writing statements that configure the connections and options needed for your network, the Protocol Translation Ruleset feature provides the **test translate** and **show translate ruleset** privileged EXEC commands. The **test translate** command is interactive and will step through the command statements to test their validity. The **show translate ruleset** command displays information about the connection rulesets to help you modify and maintain them.

Cisco Regular Expression Pattern Matching

Table 1 summarizes the basic Cisco IOS regular expression characters and their functions.

Table 1 Cisco Regular Expression Characters

Regular Expression Character	Function	Examples
.	Matches any single character.	0.0 matches 0x0 and 020 t..t matches strings such as test, text, and tart
\	Matches the character following the backslash. Also matches (escapes) special characters.	172\.1\.. matches 172.1.10.10 but not 172.12.0.0 \. allows a period to be matched as a period
[]	Matches the characters or a range of characters separated by a hyphen, within left and right square brackets.	[02468a-z] matches 0, 4, and w, but not 1, 9, or K
^	Matches the character or null string at the beginning of an input string.	^123 matches 1234, but not 01234
\$	Matches the character or null string at the end of an input string.	123\$ matches 0123, but not 1234
*	Matches zero or more sequences of the character preceding the asterisk. Also acts as a wildcard for matching any number of characters.	5* matches any occurrence of the number 5 including none 18\.* matches the characters 18. and any characters that follow 18.
+	Matches one or more sequences of the character preceding the plus sign.	8+ requires there to be at least one number 8 in the string to be matched
() []	Nest characters for matching.	(17)* matches any number of the two-character string 17 ([A-Za-z][0-9])+ matches one or more instances of letter-digit pairs: b8 and W4, as examples
	Concatenates constructs. Matches one of the characters or character patterns on either side of the vertical bar.	A(B C)D matches ABD and ACD, but not AD, ABCD, ABBD, or ACCD

The order for matching using the * or + character is longest construct first. Nested constructs are matched from the outside in. Concatenated constructs are matched beginning at the left side. If a regular expression can match two different parts of an input string, it will match the earliest part first.

Regular Expression Pattern Matching in a Protocol Translation Ruleset

Regular expressions for the Protocol Translation Ruleset feature have two uses: They match a text string against a defined pattern, and they can use information from a defined regular expression match operation to create a different string using substitution. These operations are performed by combining the characters described in [Table 1](#) with commands from the translate ruleset configuration mode.

To understand regular expression pattern matching, begin by using [Table 1](#) to interpret the following regular expression statement to match a string starting with the characters 172.18.:

```
^172\18\.*
```

The following regular expression statement matches a five-digit number starting with 10 or 11:

```
^1[0-1]...$
```

Consider the following set of actions in a ruleset named B. This ruleset listens for incoming Telnet connections from a particular IP address and port number but ignores (skips) others, decides which PAD destination address the matched incoming connections should be connected to, then finally sets the PAD connection's X.25 VC idle timer from the first digit of the port number.

```
translate ruleset B from telnet to pad
match dest-addr ^10.2.2(..)$ dest-port ^20..$
skip dest-addr ^10.2.2.11$
set pad dest-addr 4444
substitute telnet dest-port ^200(.)$ into pad idle \1
```

The caret sign anchors a match to the beginning of a string, in this example, 10.2.2 for the destination address and 20 for the destination port.

The parentheses are a powerful tool for the regular expression match operation because they identify groups of characters needed for a substitution. Combined with the substitute...into statement, the parentheses can dynamically create a broad range of string patterns and connection configurations.

In the example, the periods in the parentheses pair can be thought of as placeholders for the characters to be substituted. The dollar sign anchors the substitution match to the end of a string. The backslash preceding the number makes it a literal setting, so no substitution will be done to the idle timer setting.

The **test translate ruleset** command tests the script, and for the previous example would provide a report like the following:

```
Translate From: Telnet 10.2.2.10 Port 2000
To: PAD 4444
Ruleset B
0/1 users active
```

Consider the following, more complex expression:

```
^172\18\.(10)\.(.*)$.
```

This expression matches any string beginning with 172.18. and identifies two groups, one that matches 10 and the other that matches a wildcard character.

Let us say that the regular expression `^172\18\.(10)\.(.*)$` matched the characters 172.18.10.255 from an incoming connection. Once the match is made, the software places the character groups 10 and 255 into buffers and writes the matched groups using a substitution expression.

Regular expression substitution into the expression `0001172018\1\2` would generate the string 000117201810255.

The regular expression `\0` would write the entire matched string, and substitution into the expression `0001\0` would generate the string 0001172.18.10.255.

Error Handling in the Protocol Translation Ruleset

Configuration errors are not detected when translation ruleset commands are entered. They are tested when the connection is attempted or with the **test (ruleset)** EXEC command. In the following example, the set statement unconditionally sets the PAD's profile name to a profile that does not exist in the configuration:

```
set pad profile Bldg-1-5ess
```

This command would be accepted at the command-line interpreter, and validated only upon a connection attempt or with the **test (ruleset)** command. When the error is detected, the following messages display:

```
*%PT-3-PARAMRESULTERR: PT ruleset test protocol pad parameter profile parse error:
Bldg-1-5ess.
-Process= "PAD InCall", ipl= 3, pid= 94
*PAD: ruleset translation not generated Cause: 9 Diag: 0
```

How to Configure a Protocol Translation Ruleset

This section contains the following tasks:

- [Configuring a PVC for Protocol Translation Rulesets, page 6](#) (required)
- [Creating Protocol Translation Rulesets, page 7](#) (required)
- [Testing and Maintaining Protocol Translation Rulesets, page 10](#) (optional)

Configuring a PVC for Protocol Translation Rulesets

The protocol translation rulesets make connections only to a PVC that has been reserved for ruleset handling. Perform the following task to reserve the PVCs.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface serial** *slot/port*
4. **x25 pvc** *number* **translate ruleset** *name*
5. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Router> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> Enter your password if prompted.
Step 2	configure terminal Example: Router# configure terminal	Enters global configuration mode.
Step 3	interface serial slot/port Example: Router(config)# interface serial 2/0	Configures an interface type and enters interface configuration mode.
Step 4	x25 pvc number translate ruleset name Example: Router(config-if)# x25 pvc 4 translate ruleset A	Configures a PVC that is valid for protocol translation ruleset handling.
Step 5	exit Example: Router(config-if)# exit	Exits the current configuration mode.

What to Do Next

Perform the tasks in the section [“Creating Protocol Translation Rulesets”](#) to create the protocol translation rulesets that configure protocol translation connections. Your rulesets may be simpler or more complex than those shown in the [“Configuration Examples for the Protocol Translation Ruleset Feature” section on page 11](#), depending upon the requirements of your network.

Creating Protocol Translation Rulesets

This section describes how to create the protocol translation rulesets.

Components of a Ruleset

A protocol translation ruleset is defined by using a combination of pattern matching and commands that match or skip incoming connection requests, and then write connection configuration statements using substitute, test, and set operations. For example, telco customers that need many unique connections based on the telephone numbers in an exchange can use rulesets to generate the hundreds of specific commands as connections are established. Each generated command guides the interface and switched virtual circuit (SVC) or PVC assignment based on the incoming IP address and port selection elements.

You create the protocol translation rulesets in translate ruleset configuration mode, which is accessed when you issue the **translate ruleset** global configuration command. You define the ruleset name and the incoming and outgoing protocols to be translated using commands available in the translate ruleset configuration mode.

Numerous configuration options can be entered as part of the translation ruleset, and these options are described in the command pages for the **translate ruleset** global configuration command, and the **description**, **match**, **options**, **set**, **skip**, **substitute**, and **test** translate ruleset configuration commands.

Prerequisites

You must understand how to compose statements using Cisco regular expressions for matching patterns in a translation ruleset. See the instructions for composing regular expressions in the [“Regular Expression Pattern Matching in a Protocol Translation Ruleset”](#) section on page 4 in this document, or the [“Related Documents”](#) section on page 15.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **translate use telnet** *ip-address* (used only for Telnet-to-PAD translations statements)
4. **translate ruleset** *name* **from** *incoming-protocol* **to** *outgoing-protocol*
5. **description** *text*
6. **{match | skip}** [*line-number*] *incoming-connection-parameter* *regular-expression* [*line-number* *incoming-connection-parameter* *regular-expression* [...]]
7. **substitute** [*line-number*] **{pad | telnet}** *variable-parameter* *reg-exp-match* **into** **{pad | telnet}** *variable-parameter* [*reg-exp-write*]
8. **test** [*line-number*] **{pad | telnet}** *variable-parameter* *reg-exp-match* [**{pad | telnet}** *variable-parameter* *reg-exp-match* [...]] **set** **{pad | telnet}** *variable-parameter*
9. **set** [*line-number*] **{pad | telnet}** *variable-parameter* [**{pad | telnet}** *variable-parameter* [...]]
10. **options** *rule-option* *value* [*rule-option* *value* [...]]
11. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Router> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Router# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 3	<p>translate use telnet <i>ip-address</i></p> <p>Example: Router(config)# translate use telnet 172.30.20.15</p>	<p>Specifies a required IP address in a Telnet-to-PAD protocol translation ruleset.</p> <p>Note Required only for Telnet-to-PAD translation statements.</p>
Step 4	<p>translate ruleset <i>name</i> from <i>incoming-protocol</i> to <i>outgoing-protocol</i></p> <p>Example: Router(config)# translate ruleset Telnet-PAD from telnet to PAD</p>	<p>Defines a unique name for a translation ruleset, specifies the translated protocols, and enters translate ruleset configuration mode.</p> <ul style="list-style-type: none"> • from <i>incoming-protocol</i>—Choose telnet or pad • to <i>outgoing-protocol</i>—Choose telnet or pad
Step 5	<p>description <i>text</i></p> <p>Example: Router(cfg-pt-ruleset)# description Template Telnet-PAD for site 101</p>	<p>Adds a description about a translation ruleset.</p>
Step 6	<p>{match skip} [<i>line-number</i>] <i>incoming-connection-parameter</i> <i>regular-expression</i> [<i>line-number</i>] <i>incoming-connection-parameter</i> <i>regular-expression</i> [...]]</p> <p>Example: Router(cfg-pt-ruleset)# skip source-addr ^10\..* and Router(cfg-pt-ruleset)# match dest-addr ^172\.30\..* dest-port ^12[0-7]..\$</p>	<p>Identifies a connection for processing by a protocol translation ruleset.</p> <ul style="list-style-type: none"> • Use regular expressions to write a match or skip statement that will look at incoming connection addresses. • Up to six match or skip statements can be entered on the command line, and multiple match statements can be entered in the ruleset. <p>Note Each protocol translation ruleset must have at least one match statement.</p>
Step 7	<p>substitute [<i>line-number</i>] {pad telnet} <i>variable-parameter</i> <i>reg-exp-match</i> into {pad telnet} <i>variable-parameter</i> [<i>reg-exp-write</i>]</p> <p>Example: Router(cfg-pt-ruleset)# substitute telnet dest-port ^.(...). into pad source-addr</p>	<p>Matches an available protocol and substitutes another into the translation ruleset.</p> <ul style="list-style-type: none"> • Use this command to substitute between protocol parameters using regular expressions to match elements with a test string, and to substitute parameters into another string that can take elements from the matched string. • A substitute ... into statement will perform a regular expression match on any available protocol parameter and, if matched, substitute into any available protocol parameter. • Up to six substitute statements can be entered on the command line, and multiple substitute statements can be entered in the ruleset.

	Command or Action	Purpose
Step 8	<pre>test [line-number] {pad telnet} variable-parameter reg-exp-match [{pad telnet} variable-parameter reg-exp-match [...]] set {pad telnet} variable-parameter</pre> <p>Example: Router(cfg-pt-ruleset)# test telnet dest-addr ^172\.30\.0\.0.* telnet dest-port ^10.00 \ set pad pvc 1 telnet binary T</p>	<p>Tests parameter values in a translation ruleset using regular expressions.</p> <ul style="list-style-type: none"> • A test ... set statement conditionally sets one or more connection parameters to a given value after a successful comparison of one or more connection parameters against the regular expression. • Up to six test statements can be entered on the command line, and multiple test statements can be entered in the ruleset.
Step 9	<pre>set [line-number] {pad telnet} variable-parameter [{pad telnet} variable-parameter [...]]</pre> <p>Example: Router(cfg-pt-ruleset)# set telnet printer Y telnet binary Y</p>	<p>Sets one or more connection parameters to a fixed value for a translation ruleset.</p> <ul style="list-style-type: none"> • Once an incoming connection has been matched for processing, the ruleset generates the protocol translation parameters using a template that unconditionally sets a value defined by a set statement. • Up to six set statements can be entered on the command line, and multiple set statements can be entered in the ruleset.
Step 10	<pre>options rule-option value [rule-option value ...]</pre> <p>Example: Router(cfg-pt-ruleset)# options max-users 10 login</p>	<p>Specifies protocol translation options in the translation ruleset. Choose from the following options for the <i>rule-option value</i> arguments:</p> <ul style="list-style-type: none"> • access-class number—Defined access class number that the incoming connection must match. • login—Require login on the incoming connection (no value required). • max-users number—Maximum number of concurrent users allowed per ruleset. • quiet—Suppress translation information messages on the session (no value required).
Step 11	<pre>exit</pre> <p>Example: Router(cfg-pt-ruleset)# exit</p>	<p>Exits the current configuration mode.</p>

Testing and Maintaining Protocol Translation Rulesets

Perform this task to test and review your protocol translation rulesets.

SUMMARY STEPS

1. **enable**
2. **test translate {pad | telnet | parameter parameter} [detail]**
3. **show translate ruleset [name]**

DETAILED STEPS

	Command or Action	Purpose
Step 1	<pre>enable</pre> <p>Example: Router> enable</p>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> Enter your password if prompted.
Step 2	<pre>test translate {pad telnet parameter parameter} [detail]</pre> <p>Example: Router# test translate pad detail</p>	<p>Displays a trace of protocol translation behavior for a connection attempt.</p> <ul style="list-style-type: none"> parameter—Tests one of the translation ruleset parameters in interactive mode detail—Displays an extended trace report about the configuration and connections.
Step 3	<pre>show translate ruleset [name]</pre> <p>Example: Router# show translate ruleset</p>	<p>Displays a summary of a specific or of all configured translate rulesets and translate commands, behavioral parameters, and usage statistics.</p> <ul style="list-style-type: none"> The output of this command identifies match, skip, set, test, and substitute statement lines and numbers them; the line number can be used to reconfigure or remove any of these statements. When the optional <i>name</i> argument is used, the display includes only details about the configured ruleset and does not include information about the one-line translate commands.

Configuration Examples for the Protocol Translation Ruleset Feature

This section provides the following examples:

- [PAD-to-Telnet Translation Ruleset: Example, page 12](#)
- [SVC Conversion with Translation Ruleset Service Selection: Example, page 12](#)
- [Address Conversion in a Translation Ruleset: Example, page 12](#)
- [Reserve PVC for Protocol Translation Ruleset: Example, page 13](#)
- [SVC Conversion with Translation Ruleset Service Selection: Example, page 12](#)
- [Displaying Ruleset Configuration Parameters: Example, page 13](#)
- [Testing the Ruleset Configuration Parameters: Example, page 13](#)

PAD-to-Telnet Translation Ruleset: Example

In the following example, the incoming PAD address 55555 yields Telnet address 10.2.2.1, port 23 (default Telnet port). The local Boolean flag in the substitute statement specifies that Telnet protocol negotiations for PAD connections with destination addresses 55550 through 55555 should be forwarded, not processed.

```
translate ruleset P_to_T from pad to telnet
description forwards control sequences
match dest-addr ^5555.$
set telnet dest-addr 10.2.2.1
set telnet local n
substitute pad dest-addr ^5555([0-5])$ into telnet local Y
test telnet local n set telnet dest-port 2200
```

SVC Conversion with Translation Ruleset Service Selection: Example

The following example shows the selection of the outbound X.25 serial interface and the PAD profile for the Calling application based on the IP port number:

```
! define the profiles to be used by ruleset svc_service
x.29 profile ENG 2:0 3:128 4:0
x.29 profile DOC 2:0 3:128 4:0
x.29 profile MRKT 2:0 3:128 4:0
!
translate ruleset svc_service from telnet to pad
match dest-addr ^10.10.1.6$ dest-port ^[1]00[0-1][0-8][1-3]$
test telnet dest-port ^.0... set pad profile ENG
test telnet dest-port ^.1... set pad profile DOC
test telnet dest-port ^.2... set pad profile MRKT
substitute telnet dest-port (.)$ into pad dest-addr 765432\1
substitute telnet dest-port 0$ into pad dest-addr 76543210
```

Address Conversion in a Translation Ruleset: Example

The following translation ruleset example reduces the number of statements for converting the IP port number to an X.121 address for the following range of port numbers:

```
IP Address: 10.10.1.5 10000-19999 to X.121 Address 5559000000-9999
IP Address: 10.10.1.5 20000-29999 to X.121 Address 5559010000-9999
IP Address: 10.10.1.5 30000-39999 to X.121 Address 5559020000-9999
IP Address: 10.10.1.5 40000-49999 to X.121 Address 5559110000-9999
IP Address: 10.10.1.5 50000-59999 to X.121 Address 5559200000-9999
```

```
translate use telnet 10.10.1.5
!
translate ruleset T_to_P from telnet to pad
description Site1 10.10.1.5 Area Code 555 exchgs 900, 901, 902, 911, 920
match dest-addr ^10.10.1.5$ dest-port ^[1-5]....$
substitute telnet dest-port ^1(...) into pad dest-addr 555900\1
substitute telnet dest-port ^2(...) into pad dest-addr 555901\1
substitute telnet dest-port ^3(...) into pad dest-addr 555902\1
substitute telnet dest-port ^4(...) into pad dest-addr 555911\1
substitute telnet dest-port ^5(...) into pad dest-addr 555920\1
```

Reserve PVC for Protocol Translation Ruleset: Example

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

```
interface serial 2/0
x25 pvc 4 translate ruleset port_to_pvc
!
translate use telnet 10.10.1.6
!
translate ruleset port_to_pvc from telnet to pad
match dest-addr ^10.10.1.6$ dest-port ^([12]00[0-7])[1-3]$
substitute telnet dest-port ^..0([0-7]) into pad interface serial 0/\1
substitute telnet dest-port ^...(.) into pad pvc \1
test telnet dest-port ^.0... set pad profile TEMS
test telnet dest-port ^.1... set pad profile SQAS
test telnet dest-port ^.2... set pad profile NMA
substitute telnet dest-port (.)$ into pad dest-addr 876543\1

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

Displaying Ruleset Configuration Parameters: Example

The following example displays a summary of a configured translate ruleset named Template_1 that includes behavioral parameters, usage statistics, and line numbers for maintaining the configuration:

```
Router# show translate ruleset Template_1

PT ruleset Template_1, from telnet to pad
administrative locks: 2 (2 readers, 0 writers)
translations: 0 created, 0 active, 0 failed (0 max-user), 0 created for test
match/skip lines: 1
  #1 match on 2 telnet tests: dest-addr ^172\.18\..*, dest-port ^12(0-7)..$
options: login user, limited to 10 active sessions
set/test/substitute lines: 3
  #1 set 2 parameters: telnet/printer Y, telnet/binary Y
  #2 set 1 parameter: pad/profile cust-profile-one
  #3 test 2 parameters: telnet/dest-addr ^172\.18\.0\..*, telnet/dest-port ^10.00; to set
2: pad/pvc 1, telnet/binary T
```

Testing the Ruleset Configuration Parameters: Example

The following example shows a detailed trace of PAD ruleset configurations:

```
Router# test translate pad detail

No PAD translate command matched
(Testing translate command A ...)
Ruleset A match/skip line 1 compared: match
  (processing set/test/substitute line 1)
  (set/test/subst line 1, item 1, parameter dest-addr set to 10.2.2.1)
  (processing set/test/substitute line 2)
  (set/test/subst line 2, item 1, parameter idle set to 10)
  (parsed pad parameter idle: 10)
  (parsed telnet parameter dest-addr: 10.2.2.1)
Ruleset A; pad parameter read:
  pad/dest-addr: 55555
```

```
Parameters set:
  pad/idle: 10
  telnet/dest-addr: 10.2.2.1
  (translation requires 0 bytes variable-sized memory)

Translate From: PAD 55555
                Idle 0:10
                To:  Telnet 10.2.2.1 Port 23
                Ruleset A
                0/1 users active
```

Additional References

The following sections provide references related to the Protocol Translation Ruleset feature.

Related Documents

Related Topic	Document Title
Regular expressions	Regular Expressions
Protocol translation	Configuring Protocol Translation and Virtual Asynchronous Devices

Standards

Standards	Title
None	—

MIBs

MIBs	MIBs Link
None	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFCs	Title
None	—

Technical Assistance

Description	Link
Technical Assistance Center (TAC) home page, containing 30,000 pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/public/support/tac/home.shtml

Command Reference

The following commands are introduced or modified in the feature or features documented in this module. For information about these commands, see the *Cisco IOS Terminal Services Command Reference* at http://www.cisco.com/en/US/docs/ios/termserv/command/reference/tsv_book.html. For information about all Cisco IOS commands, go to the Command Lookup Tool at <http://tools.cisco.com/Support/CLILookup> or to the *Cisco IOS Master Commands List*.

- **description (ruleset)**
- **match (ruleset)**
- **options (ruleset)**
- **set (ruleset)**
- **show translate ruleset**
- **skip (ruleset)**
- **substitute (ruleset)**
- **test (ruleset)**
- **test translate**
- **translate ruleset**
- **translate use telnet**
- **x25 pvc translate ruleset**

Any Internet Protocol (IP) addresses used in this document are not intended to be actual addresses. Any examples, command display output, and figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses in illustrative content is unintentional and coincidental.

© 2008 Cisco Systems, Inc. All rights reserved.