

NETOPIA™ ROUTERS AND IADS

Command Line Interface Commands Reference

Firmware Versions 4.10 and 5.2



netopia®

Copyright

©2001, Netopia, Inc., v.113001

All rights reserved. Printed in the U.S.A.

This manual and any associated artwork, software, and product designs are copyrighted with all rights reserved. Under the copyright laws such materials may not be copied, in whole or part, without the prior written consent of Netopia, Inc. Under the law, copying includes translation to another language or format.

Netopia, Inc.

2470 Mariner Square Loop

Alameda, CA 94501-1010

U.S.A.

Part Number

For additional copies of this manual, order Netopia part number 6160028-00-09

Chapter 1 — Introduction.....	1-1
CLI Changes in Firmware Version 4.10	1-1
Syntax Notation	1-3
Interface Naming Conventions	1-3
Escaped Characters	1-4
Security (Configuration Access)	1-4
Entering and Editing Commands.....	1-5
 Chapter 2 — Netopia Router CLI Commands.....	 2-1
Configuration Access Commands	2-2
Interface Configuration Commands	2-8
Ethernet Interface configuration commands	2-8
NetBIOS configuration commands	2-17
Generic WAN Interface configuration commands...	2-19
Restricted WAN Interface configuration commands	2-20
ISDN WAN Interface configuration commands	2-21
ADSL WAN Interface configuration commands	2-25
SDSL WAN Interface configuration commands	2-26
PVCs.....	2-30
DSL Line Type Interface Configuration Commands	2-33
T1 WAN Interface configuration commands	2-34
T1 Statistic and Diagnostic commands	2-37
Global IP Configuration Commands.....	2-40
ARP Configuration Commands.....	2-42
Connection Profile Commands	2-43
Note on Connection Profile numbering sequence ..	2-48
PPTP commands	2-56
Manual connect/disconnect commands	2-57
Backup configuration commands.....	2-57
CompuServe Login.....	2-58
IPSec/IKE	2-59
Frame Relay Configuration Commands.....	2-65

Miscellaneous Commands.....	2-67
IP Network Address Translation (NAT) Commands.....	2-73
AppleTalk Configuration Commands.....	2-77
Backup Configuration Commands.....	2-78
RADIUS Authentication Configuration Commands	2-80
IP Filterset Configuration Commands	2-82
Hardware Acceleration Configuration Commands.....	2-84
Global IPSec/IKE Configuration Commands.....	2-85
Current Restrictions	2-90
Chapter 3 — Netopia IAD CLI Commands	3-1
Voice Commands	3-2
Directory Numbers	3-3
Tone Table.....	3-4
Logs	3-4
Outside and Operator Digits.....	3-5
Encoding Commands.....	3-5
Ring Cadence	3-5
Voice Statistics.....	3-6
Chapter 4 — Netopia Router Text Configuration Upload.....	4-1
TFTP Text Configuration Upload Overview	4-1
SNMP	4-1
VT100 Menu Console.....	4-1
VT100 Command Line Console	4-2
Example Text Configuration File	4-3
Chapter 5 — CLI Error Messages	5-1
Negative errors.....	5-1
Fatal system errors	5-1
Fatal syntax errors	5-1
Voice command errors	5-9
Fatal access control errors	5-11
Positive errors	5-12

Index of Commands

Chapter 1

Introduction

This *Command Line Interface Commands Reference* contains information on the syntax and use of the Command Line Interface for the Netopia router and IAD family. It provides information required to configure the router firmware and troubleshoot problems using the Command Line Interface.

This document is intended for small office, home office, and remote office users, and other networking professionals who administer networks using Netopia routers or IADs.

Note: The R-series data router family supports firmware versions in the 4.X series. The 4000-series data router and IAD family supports firmware versions in the 5.X series. Restrictions among firmware versions are noted in the body of this document. Where no firmware version is noted, the commands given are supported on all platforms.

- The chapter “[Netopia Router CLI Commands](#),” beginning on page 2-1, gives the command set used for configuring the Netopia data router engine, found in both the data router products and the integrated access devices (IADs).
- The chapter “[Netopia IAD CLI Commands](#),” beginning on page 3-1, gives the command set used for configuring the Netopia integrated access devices.

CLI Changes in Firmware Version 4.10

The following is a summary of some of the most important changes in the CLI in firmware version 4.10. Refer to the Table of Contents or the Index to find detailed information.

- The information shown on the System Information screen is now available via the following CLI command:

```
show system information
```

- The daughtercard firmware version is now available via the following CLI command:

```
show version [ cli ] [ firmware ] [ hardware ] [ mib ] [ wan 1 ] [ wan 2 ]
```

- You can now upgrade the daughtercard firmware via the following CLI command:

```
receive tftp [ wan { 1 | 2 } ] firmware [ servername filename ]
```

```
receive xmodem [ wan { 1 | 2 } ] firmware
```

```
show tftp status
```

```
show xmodem status
```

- It is now possible to set the SNMP system objects via the CLI. The supported system group objects are sysName, sysLocation, and sysContact.

1-2 Command Line Interface Commands Reference

```
snmp system { contact | location | name } string
```

```
show snmp system { contact | location | name }
```

```
no snmp system { contact | location | name }
```

- You can now set the WAN Ethernet MAC address on a R9100 or R910 via the following CLI command:

```
interface ethernet wan-id mac address { MAC-address | default }
```

```
show interface ethernet id mac address
```

- The Security Screen password can now be set via the following CLI command:

```
security password
```

```
no security password
```

You will be prompted to enter the old password when necessary.

- NAT MAP CLI

```
show ip nat translations
```

- The telnet port the router 'listens' to is now user configurable via the CLI.

```
telnet server port [ portnumber ]
```

```
show telnet server port
```

- You can now specify a source address when running a PING via the CLI.

```
ping [ ip ] { ip-addr | hostname } [count count] [timeout milliseconds]  
[delay milliseconds] [size bytes] [ source ip-addr ]
```

Syntax Notation

The command descriptions use formatted text to indicate various attributes of each command. The syntax is as follows:

- Required keywords and commands that must be typed literally are in **boldface**.
- Optional elements are enclosed in square brackets “[]”.
- Mutually exclusive elements are contained in braces “{ }” and separated by vertical bars “|”.
- Arguments for which you supply values are in *italics*.
- Examples of commands you type and the results of those commands are in the `courier` typeface.
- An element that may be repeated one or more times is followed by a superscripted plus sign “⁺”.
- An element that may be repeated zero or more times is followed by a superscripted asterisk “^{*}”.

Interface Naming Conventions

A number of commands described in this document require you to identify the router interface to be affected by the command. This requires specifying both an interface type (denoted *intf-type*) and an interface index (denoted *id*).

The *intf-type* argument may be replaced with one of the following keywords:

adsl | aux | dds | ethernet | isdn | modem | sds1 | t1 | wan

If a command is not specific to a particular WAN interface type, the *intf-type* **wan** may be specified; otherwise, the more specific *intf-type* must be specified.

Note: For IDSL interfaces, use the keyword **isdn**.

The *id* argument can be replaced with 0, 1, or 2, as follows:

- **0** means the motherboard
- **1** means the WAN 1 slot
- **2** means the WAN 2 slot

So, for example, the ethernet hublet is identified as “interface ethernet 0”. In some contexts, only a WAN interface may be specified, in which case the command syntax will specify *wan-id* instead of the more general *id*. The *wan-id* argument can be replaced by either **1** (the WAN 1 slot) or **2** (the WAN 2 slot).

Escaped Characters

If you wish to set a text property, such as a profile tag or a directory number, that contains space characters, backslashes (\), or double quote signs ("), you must enclose the text with double quotes. Additionally, inside the quotes, any double quote or backslash characters must be preceded by a backslash. This means that to add a backslash character to a string you must type two backslashes (\\) and to add a double quote you must type a backslash and a quote (\").

Example:

To set Connection Profile 1's "tag" property (i.e., its name) to the string

```
Profile "1" \with weird chars
```

you need to type

```
cp 1 tag "Profile \"1\" \\with weird chars"
```

Note: The underscore character may become a special character in the near future. To be safe, if you want to set a property to a string that contains underscores put double quotes around the string and precede any occurrences of an underscore with a backslash.

Security (Configuration Access)

If the device is password-protected, the device requires you to enter a name and password before you can access the menu-based or command line console interface. See the section ["Configuration Access Commands" on page 2-2](#).

Entering and Editing Commands

The device’s console user interface comes up in Menu mode by default. In this mode you use the arrow, Escape, and Return/Enter keys to navigate through a series of screens. To invoke the command line at any time, hit Control-N. The console will erase the window, and you will be presented with a # prompt. To return to Menu mode hit Control-N again.

The following table provides a description of keys that can be used when entering and editing commands. Control indicates the Control key, which must be pressed simultaneously with the associated letter key. Escape indicates the Escape key, which must be pressed and released first, followed by its associated letter key. Keys are not case-sensitive.

Command Editing Keys and Functions	
Key	Function
Control-A	Moves the cursor to the beginning of the command line.
Control-E	Moves the cursor to the end of the command line.
Control-K	Deletes all characters from the cursor to the end of the command line.
Control-N	Invokes the command line interface from the menu console. Invokes the menu console from the command line interface.
Control-U	Deletes all characters from the cursor back to the beginning of the command line.
Control-W	Deletes the word to the left of the cursor.
Escape B	Moves the cursor back one word.
Escape D	Deletes from the cursor to the end of the word.
Escape F	Moves the cursor forward one word.

Chapter 2

Netopia Router CLI Commands

This chapter describes the syntax of the supported command set of the Netopia R-series Router.

- “Configuration Access Commands” on page 2-2
- “Interface Configuration Commands” on page 2-8
- “Global IP Configuration Commands” on page 2-40
- “ARP Configuration Commands” on page 2-42
- “Connection Profile Commands” on page 2-43
 - “PPTP commands” on page 2-56
 - “Manual connect/disconnect commands” on page 2-57
 - “Backup configuration commands” on page 2-57
 - “CompuServe Login” on page 2-58
 - “IPSec/IKE” on page 2-59
- “Frame Relay Configuration Commands” on page 2-65
- “Miscellaneous Commands” on page 2-67
- “IP Network Address Translation (NAT) Commands” on page 2-73
- “AppleTalk Configuration Commands” on page 2-77
- “Backup Configuration Commands” on page 2-78
- “RADIUS Authentication Configuration Commands” on page 2-80
- “IP Filterset Configuration Commands” on page 2-82
- “Hardware Acceleration Configuration Commands” on page 2-84
- “Global IPSec/IKE Configuration Commands” on page 2-85
- “Current Restrictions” on page 2-90

Configuration Access Commands

Configuration Access Commands
<p>date <i>xx/yy/zz</i> show date</p> <p>exit</p> <p>preferences changes immediate { yes no } show preferences changes immediate no preferences changes immediate</p> <p>preferences console default { menu cli } show preferences console default</p> <p>preferences console timeout <i>seconds</i> no preferences console timeout show preferences console timeout</p> <p>preferences date format { mm/dd/yy dd/mm/yy yy/mm/dd } show preferences date format</p> <p>preferences output format { terse verbose } show preferences output format</p> <p>preferences output mask { bits dotted-quad } show preferences output mask</p> <p>preferences time format { am-pm 24-hour } show preferences time format</p> <p>security password no security password</p> <p>snmp community { ro read-only rw read-write } <i>string</i> no snmp community [ro read-only rw read-write] [<i>string</i>]</p> <p>snmp system contact <i>string</i> show snmp system contact no snmp system contact</p> <p>snmp system location <i>string</i> show snmp system location no snmp system location</p>

Configuration Access Commands (cont. 1)

<pre>snmp system name <i>string</i> show snmp system name no snmp system name</pre>

<pre>system web-server enable { yes no } no system web-server enable show system web-server enable</pre>
--

<pre>system web-server lan-only { yes no } no system web-server lan-only show system web-server lan-only</pre>
--

<pre>telnet server port [<i>port number</i>] show telnet server port</pre>
--

<pre>time <i>hh:mm</i> [am pm] show time</pre>
--

<pre>user <i>name password</i> no user <i>name</i> [<i>password</i>]</pre>
--

The **preferences** command allows you to customize certain aspects of the command line interface. Preference settings persist across restarts, and are specific to the user name, if any, you used to authenticate yourself before issuing the **preferences** command. If no users are defined, no authentication is required, and preference settings are global.

```
date xx/yy/zz
show date
```

These commands allow you to set or display the current date for the router's system clock.

```
exit
```

The **exit** command terminates your current console session. If you are connected via telnet or a modem, the connection will be closed. If you are logged in via the serial console, you will return to the command line or menu-based console based on your default console setting. (See the **preferences console default** command on [page 2-4](#).) In either case, you will be prompted either with a login prompt (if one or more users are defined), or the initial prompt for the selected console interface (if no users are defined).

```
preferences changes immediate { yes | no }  
show preferences changes immediate  
no preferences changes immediate
```

These commands allow you to specify whether or not WAN configuration changes will take effect immediately. When you specify **no**, any changes you make to the WAN configuration (except NAT) will not take effect until the router is reset.

Note: The router will reboot immediately when the value of the **changes immediate** preference item changes. No warning is given.

```
preferences console default { menu | cli }  
show preferences console default
```

The **preferences console default** command specifies the console interface that will be presented to the user on subsequent logins. When set to **menu** (the default), the user will be presented with the menu-based console interface on subsequent logins. When set to **cli**, the user will be presented with the command line console interface on subsequent logins. If the **preferences console default** command is issued and there are no users defined, the setting will determine the console interface that will be presented to all newly established console sessions (via either the serial console port or via telnet).

```
preferences console timeout seconds  
no preferences console timeout  
show preferences console timeout
```

These commands control the command-line and menu-based console auto logout. Note that the **no preferences console timeout** command sets the timeout to zero, which disables the timeout.

The command:

```
no preferences console timeout
```

is equivalent to:

```
preferences console timeout 0
```

Example:

```
preferences console timeout 300
```

```
preferences date format { mm/dd/yy | dd/mm/yy | yy/mm/dd }  
show preferences date format
```

These commands allow you to set or display your date formatting preferences for the router's system clock.

```
preferences output format { terse | verbose }  
show preferences output format
```

The **preferences output format** command affects the format of the output from show commands. When set to **verbose** (the default), the output from **show** commands is formatted as a valid command line interface command that could be entered at a command prompt. When set to **terse**, the output from show commands is *not* formatted as a valid command line interface command that could be entered at a command prompt, but rather includes only the value of the requested attribute. The **terse** mode may be more useful if the output will be processed by a computer rather than a human being.

Example:

```
#preferences output format verbose  
#show interface ethernet 0 ip address  
interface ethernet 0 ip address 192.168.1.1/24  
#preferences output format terse  
#show interface ethernet 0 ip address  
192.168.1.1/24
```

```
preferences output mask { bits | dotted-quad }  
show preferences output mask
```

The **preferences output mask** command affects the format of the output from those **show** commands that display an IP address together with a subnet mask. When set to **bits** (the default), the IP address and subnet mask are output in prefix notation – i.e., an IP address in dotted-quad notation followed by a slash followed by the number of consecutive ones-bits in the subnet mask – whereas when set to **dotted-quad**, the IP address and subnet mask are output as two consecutive dotted-quads.

Example:

```
#preferences output mask bits  
#show interface ethernet 0 ip address  
interface ethernet 0 ip address 192.168.1.1/24  
#preferences output mask dotted-quad  
#show interface ethernet 0 ip address  
interface ethernet 0 ip address 192.168.1.1 255.255.255.0
```

```
preferences time format { am-pm | 24-hour }  
show preferences time format
```

These commands allow you to set or display your time formatting preferences for the router's system clock.

security password

Enter old password: *old password*
Enter new password: *new password*
Re-enter password: *new password*

no security password

Enter old password: *old password*

Note: These commands are supported beginning with firmware versions 4.10 and 5.2.

These commands let you set and delete the **Security Options** screen password. After you enter the command the console prompts you for the existing password if you have one, then it prompts you to enter and re-enter a new password (eleven characters maximum). The **no** command will prompt you for a password if there was one, and will then delete that password.

snmp community { **ro** | **read-only** | **rw** | **read-write** } *string*
no snmp community [**ro** | **read-only** | **rw** | **read-write**] [*string*]

These commands allow you to add or delete the SNMP community Read-Only and Read-Write strings.

snmp system contact *string*
show snmp system contact
no snmp system contact

Note: These commands are supported beginning with firmware versions 4.10 and 5.2.

These commands set, display, or clear the router's SNMP system contact (sysContact) string.

snmp system location *string*
show snmp system location
no snmp system location

Note: These commands are supported beginning with firmware versions 4.10 and 5.2.

These commands set, display, or clear the router's SNMP system location (sysLocation) string.

snmp system name *string*
show snmp system name
no snmp system name

Note: These commands are supported beginning with firmware versions 4.10 and 5.2.

These commands set, display, or clear the router's SNMP system name (sysName) string.

system web-server enable { **yes** | **no** }
no system web-server enable
show system web-server enable

Note: These commands are supported beginning with firmware version 4.10 on R-Series routers only.

These commands enable or disable the router's internal web server. On R-Series routers this will take effect immediately.

```
system web-server lan-only { yes | no }
no system web-server lan-only
show system web-server lan-only
```

Note: These commands are supported beginning with firmware versions 4.10 and 5.2.

These commands enable, disable, or show the status of web server access restriction to LAN-side access only. This assumes that web server access is enabled.

```
telnet server port [ port number ]
show telnet server port
```

Note: These commands are supported beginning with firmware version 4.10 and 5.2.

These commands allow you to set or display the TCP port on which the router is currently listening for incoming telnet management sessions. If you change the port number, the router will immediately stop accepting new sessions at the old port number, and only accept incoming sessions on the new port number. All sessions currently connected to the old port number will remain connected. Allowed values for port number are 1 - 65535, except for 80 and 1723.

```
time hh:mm [ am | pm ]
show time
```

These commands allow you to set or display the current time for the router's system clock.

```
user name password
no user name [password]
```

These commands allow you to add or delete a configuration name and password pair. The firmware supports up to 4 name and password pairs.

Note: If the configuration name or password contains spaces, remember to enclose it in double quote signs ("). See ["Escaped Characters" on page 1-4](#) for more information.

Interface Configuration Commands

Ethernet Interface configuration commands

Ethernet Interface Configuration Commands
<p>interface ethernet <i>id</i> ip address [{ <i>ip-addr/ mask-bits</i> <i>ip-addr mask</i> secondary }] no interface ethernet <i>id</i> ip address [{ <i>ip-addr/ mask-bits</i> <i>ip-addr mask</i> secondary }] show interface ethernet <i>id</i> ip address</p> <p>interface ethernet <i>id</i> ip netbios proxy enable { yes no } no interface ethernet <i>id</i> ip netbios proxy enable show interface ethernet <i>id</i> ip netbios proxy enable</p> <p>interface ethernet <i>id</i> ip rip receive { no v1 v2 both } no interface ethernet <i>id</i> ip rip receive show interface ethernet <i>id</i> ip rip receive</p> <p>interface ethernet <i>id</i> ip rip transmit { no v1 v2broadcast v2multicast } no interface ethernet <i>id</i> ip rip transmit show interface ethernet <i>id</i> ip rip transmit</p> <p>interface ethernet <i>id</i> pppoe enable { yes no } no interface ethernet <i>id</i> pppoe enable show interface ethernet <i>id</i> pppoe enable</p> <p>show interface ethernet <i>id</i> statistics show interface ethernet <i>id</i> stats</p> <p>interface ethernet <i>id</i> ip filterset <i>fs-id</i> no interface ethernet <i>id</i> ip filterset show interface ethernet <i>id</i> ip filterset</p>

Ethernet Interface Configuration Commands (cont. 1)

interface ethernet *wan-id* **ip nat enable** { **yes** | **no** }
no interface ethernet *wan-id* **ip nat enable**
show interface ethernet *wan-id* **ip nat enable**

interface ethernet *wan-id* **ip nat map-list** *list-tag*
no interface ethernet *wan-id* **ip nat map-list**
show interface ethernet *wan-id* **ip nat map-list**

interface ethernet *wan-id* **ip nat server-list** *list-tag*
no interface ethernet *wan-id* **ip nat server-list**
show interface ethernet *wan-id* **ip nat server-list**

interface ethernet *wan-id* **mac address** { *MAC-address* | **default** }
show interface ethernet *wan-id* **mac address**

Ethernet Interface IP Address Serving Commands

```
interface ethernet 0 address-serve clients { any | none | { bootp | dhcp | macip | wan }+ }
no interface ethernet 0 address-serve clients { any | { bootp | dhcp | macip | wan }+ }
show interface ethernet 0 address-serve clients
```

```
interface ethernet 0 address-serve dhcp enable { yes | no }
no interface ethernet 0 address-serve dhcp enable
show interface ethernet 0 address-serve dhcp enable
```

```
interface ethernet 0 address-serve dhcp lease-time hours
show interface ethernet 0 address-serve dhcp lease-time
```

```
interface ethernet 0 address-serve gateway { gw-ip-addr | default { ip-addr/mask-bits |
    ip-addr mask } }
show interface ethernet 0 address-serve gateway
```

```
interface ethernet 0 address-serve helper ip-addr
no interface ethernet 0 address-serve helper [ip-addr]
show interface ethernet 0 address-serve helper
```

```
interface ethernet 0 address-serve mode { relay | server }
show interface ethernet 0 address-serve mode
```

```
interface ethernet 0 address-serve range { auto | from-addr to-addr }
no interface ethernet 0 address-serve range from-addr to-addr
show interface ethernet 0 address-serve range
```

```
interface ethernet 0 address-serve { no | off | on | yes }
no interface ethernet 0 address-serve
show interface ethernet 0 address-serve
```

```

interface ethernet id ip address [{ ip-addr/ mask-bits | ip-addr mask | secondary }]
no interface ethernet id ip address [{ ip-addr/ mask-bits | ip-addr mask | secondary }]
show interface ethernet id ip address

```

These commands allow you to set, delete, or show the IP subnet(s) of an Ethernet interface. If the keyword **secondary** is specified in the first command, the subnet is appended to the list of subnets (assuming that all of the allowed subnets have not yet been configured– the router supports up to eight). If the keyword **secondary** is not specified, the primary subnet configuration is replaced with the specified values. The mask may be specified either as a slash followed by the number of one-bits in the mask, or as a dotted quad.

The **no interface ethernet** *id* **ip address** command allows you to delete a particular subnet, all secondary subnets, or all subnets associated with the specified Ethernet interface.

Examples:

The following are equivalent ways to set the primary subnet of the Ethernet interface to 192.168.1.1 with a Class C subnet mask:

```

interface ethernet 0 ip address 192.168.1.1/24

interface ethernet 0 ip address 192.168.1.1 255.255.255.0

```

To set a secondary subnet of the Ethernet interface to 207.1.1.16/28 (with four host bits):

```

interface ethernet 0 ip address 207.1.1.16/28 secondary

```

To delete a particular subnet from the list of subnets, specify the particular subnet:

```

no interface ethernet 0 ip address 207.1.1.16/28

```

To delete all secondary subnets:

```

no interface ethernet 0 ip address secondary

```

To delete all subnets:

```

no interface ethernet 0 ip address

```

To show the IP subnets of the Ethernet interface:

```

show interface ethernet 0 ip address

```

```
interface ethernet id ip netbios proxy enable { yes | no }  
no interface ethernet id ip netbios proxy enable  
show interface ethernet id ip netbios proxy enable
```

Note: These commands are supported beginning with firmware version 4.9.4.

These commands allow you to enable, disable, or show the NetBIOS proxy status for the specified Ethernet interface. The NetBIOS proxy enables the ability to forward Windows Networking NetBIOS broadcasts. This is useful for, for example, a Virtual Private Network, in which you want to be able to browse the remote network to which you are tunnelling, as part of your Windows Network Neighborhood.

Routed connections, such as VPNs, can not use NetBEUI to carry the Network Neighborhood information. They need to use NetBIOS, because NetBEUI cannot be routed. This feature will allow browsing the Network Neighborhood without any additional workstation configuration.

Note: Microsoft Network browsing is available with or without a Windows Internet Name Service (WINS) server. Shared volumes on the remote network are accessible with or without a WINS server. Local LAN shared volumes that have Port Address Translation (PAT) applied to them are *not* available to hosts on the remote LAN. For tunnelled traffic, NAT on the WAN has no effect on the Microsoft Networking traffic.

```
interface ethernet 0 address-serve dhcp enable { yes | no }  
no interface ethernet 0 address-serve dhcp enable  
show interface ethernet 0 address-serve dhcp enable
```

These commands allow you to enable, disable, or show the DHCP IP address serving behavior of the specified Ethernet interface. These commands do not affect the DHCP server mode. Consequently, if the router is set to DHCP relay these commands have no effect.

The **show interface ethernet 0 address-serve dhcp** command may also include the following keywords: **available**, **leased**, **offered**, and **reserved**. These return the count of client IP addresses in their respective states.

Examples:

```
show interface ethernet 0 address-serve dhcp report available  
  
show interface ethernet 0 address-serve dhcp report leased  
  
show interface ethernet 0 address-serve dhcp report offered  
  
show interface ethernet 0 address-serve dhcp report reserved
```

```

interface ethernet id ip rip receive { no | v1 | v2 | both }
no interface ethernet id ip rip receive
show interface ethernet id ip rip receive

```

These commands allow you to set, delete, or show the RIP receive behavior of the specified Ethernet interface.

Example:

```
show interface ethernet 0 ip rip receive
```

```

interface ethernet id ip rip transmit { no | v1 | v2broadcast | v2multicast }
no interface ethernet id ip rip transmit
show interface ethernet id ip rip transmit

```

These commands allow you set, delete, or show the RIP transmit behavior of the specified Ethernet interface.

Examples:

```
show interface ethernet 0 ip rip transmit
```

```

interface ethernet id pppoe enable { yes | no }
no interface ethernet id pppoe enable
show interface ethernet id pppoe enable

```

These commands allow you enable, disable, or show the PPP over Ethernet behavior of the specified interface.

```

show interface ethernet id statistics
show interface ethernet id stats

```

These commands allow you to display statistics for the specified Ethernet interface, including receive frames, octets, and errors, and transmit frames, octets, and errors.

```

interface ethernet id ip filterset fs-id
no interface ethernet id ip filterset
show interface ethernet id ip filterset

```

Note: These commands are supported beginning with firmware versions 4.10 and 5.2.

These commands allow you to enable, disable, or show an IP filterset identified by *fs-id* on the specified Ethernet interface. *fs-id* is specified as an ASCII string corresponding to the name of a filterset. See [“IP Filterset Configuration Commands” on page 2-82](#) for more information.

```

interface ethernet wan-id ip nat enable { yes | no }
no interface ethernet wan-id ip nat enable
show interface ethernet wan-id ip nat enable

```

These commands allow you to enable, disable, or show the Network Address Translation behavior for the specified WAN interface.

```
interface ethernet wan-id ip nat map-list list-tag  
no interface ethernet wan-id ip nat map-list  
show interface ethernet wan-id ip nat map-list
```

These commands allow you to set, delete, or show a NAT map list for the specified WAN interface.

```
interface ethernet wan-id ip nat server-list list-tag  
no interface ethernet wan-id ip nat server-list  
show interface ethernet wan-id ip nat server-list
```

These commands allow you to set, delete, or show a NAT server list for the specified WAN interface.

```
interface ethernet wan-id mac address { MAC-address | default }  
show interface ethernet wan-id mac address
```

Note: These commands are supported beginning with firmware version 4.10.

The first command allows you to set the MAC Address for the WAN on an R9100 or an R910 Ethernet Router. You can return it to the default by typing in a MAC Address consisting of all zeros or by typing **default**. The **show** command applies to the LAN of *all* models, as well as the WAN on the R9100 and R910.

```
interface ethernet 0 address-serve clients { any | none | { bootp | dhcp | macip | wan } }  
no interface ethernet 0 address-serve clients { any | { bootp | dhcp | macip | wan }+ }  
show interface ethernet 0 address-serve clients
```

The **interface ethernet 0 address-serve clients** command allows you to configure the types of clients that may request IP addresses from the address server. If you specify the keyword **any**, the address server will accept requests from clients of any type supported by the router. Otherwise, you may specify one or more of the keywords **bootp**, **dhcp**, **macip**, or **wan**, in which case the address server will accept requests from only the specified types of clients. If you specify the keyword **none**, the address server will not accept requests from clients of any type.

The **no interface ethernet 0 address-serve clients** command removes the specified client types from those from which the address server will accept requests.

```
interface ethernet 0 address-serve dhcp lease-time hours  
show interface ethernet 0 address-serve dhcp lease-time
```

These commands allow you to set or show the address serving DHCP lease time to any number of hours, up to and including 168 (one week). The default DHCP lease time is one hour.

```

interface ethernet 0 address-serve gateway { gw-ip-addr | default { ip-addr/mask-bits |
    ip-addr mask } }
show interface ethernet 0 address-serve gateway

```

This command allows you to specify the gateway IP address that will be served to clients requesting an address via an address serving protocol that can serve a gateway address. You may specify a gateway IP address for each Ethernet subnet for which you have configured an address-serving pool. (See the description of the

interface ethernet 0 address-serve range command on [page 2-16](#).)

If you specify the keyword **default**, you must also specify an Ethernet subnet; the gateway IP address for the specified subnet will be reset to its default value. The default gateway IP address for a particular subnet is either the router's default gateway (if that gateway is on the specified subnet) or the router's address on the subnet.

```

interface ethernet 0 address-serve helper ip-addr
no interface ethernet 0 address-serve helper [ip-addr]
show interface ethernet 0 address-serve helper

```

These commands allow you to configure or display the addresses of up to four remote DHCP servers to which the router will forward DHCP requests when it is acting as a DHCP relay agent. The **interface ethernet 0 address-serve helper** command adds the specified IP address to the server list. The **no interface ethernet 0 address-serve helper** command removes the specified IP address from the server list; if you omit the IP address, all configured DHCP server IP addresses are removed.

Examples:

```

#show interface ethernet 0 address-serve helper
#interface ethernet 0 address-serve helper 10.0.0.1
#interface ethernet 0 address-serve helper 20.0.0.1
#interface ethernet 0 address-serve helper 30.0.0.1
#no interface ethernet 0 address-serve helper 20.0.0.1
#show interface ethernet 0 address-serve helper
interface ethernet 0 address-serve helper 10.0.0.1
interface ethernet 0 address-serve helper 30.0.0.1
#

```

```

interface ethernet 0 address-serve mode { relay | server }
show interface ethernet 0 address-serve mode

```

These commands allow you to specify or display the address serving mode for the specified Ethernet interface. The keyword **relay** causes the router to act as a DHCP relay agent. The keyword **server** enables address serving from one or more locally configured address pools.

Examples:

```

#interface ethernet 0 address-serve mode server
#show interface ethernet 0 address-serve mode
interface ethernet 0 address-serve mode server
#

```

```
interface ethernet 0 address-serve range { auto | from-addr to-addr }  
no interface ethernet 0 address-serve range from-addr to-addr  
show interface ethernet 0 address-serve range
```

This command configures a pool of IP addresses for use by the address server. You may specify one address pool for each configured Ethernet subnet (primary and secondary). The total number of addresses in all configured pools may not exceed 512 addresses.

If you specify the keyword **auto** instead of an IP address range, the router will automatically configure IP address pools for each configured Ethernet subnet. An automatically configured pool will include one-half of the number of addresses available in the corresponding subnet, and will be located in the opposite half of the subnet from the router's IP address on the subnet. If the total number of addresses required would exceed the maximum of 512 total addresses, the 512 available addresses will be allocated on a pro-rata basis across all pools.

```
interface ethernet 0 address-serve { no | off | on | yes }  
no interface ethernet 0 address-serve  
show interface ethernet 0 address-serve
```

These commands enable, disable, or display the status of address-serving for the specified Ethernet interface.

NetBIOS configuration commands

Note: The commands in this section are supported beginning with firmware version 4.8.

NetBIOS Configuration Commands

```
interface ethernet 0 address-serve netbios mode type { b-node | p-node | m-node | h-node }
show interface ethernet 0 address-serve netbios mode type
```

```
interface ethernet 0 address-serve netbios mode enable { yes | no }
no interface ethernet 0 address-serve netbios mode enable
show interface ethernet 0 address-serve netbios mode enable
```

```
interface ethernet 0 address-serve netbios scope enable { yes | no }
no interface ethernet 0 address-serve netbios scope enable
show interface ethernet 0 address-serve netbios scope enable
```

```
interface ethernet 0 address-serve netbios scope name domain-name
show interface ethernet 0 address-serve netbios scope name
```

```
interface ethernet 0 address-serve netbios name-server enable { yes | no }
no interface ethernet 0 address-serve netbios name-server enable
show interface ethernet 0 address-serve netbios name-server enable
```

```
interface ethernet 0 address-serve netbios name-server address xxx.xxx.xxx.xxx [secondary]
show interface ethernet 0 address-serve netbios name-server address
```

```
interface ethernet 0 address-serve netbios mode enable { yes | no }
no interface ethernet 0 address-serve netbios mode enable
show interface ethernet 0 address-serve netbios mode enable
```

These commands allow you to enable, delete, or show the router's IP address serving capability on the Ethernet interface in NetBIOS mode.

```
interface ethernet 0 address-serve netbios mode type { b-node | p-node | m-node | h-node }
show interface ethernet 0 address-serve netbios mode type
```

These commands allow you to set or show the router's NetBIOS mode type of IP address serving on the Ethernet interface.

```
interface ethernet 0 address-serve netbios scope enable { yes | no }
no interface ethernet 0 address-serve netbios scope enable
show interface ethernet 0 address-serve netbios scope enable
```

These commands allow you to set, delete, or show whether NetBIOS scope is enabled.

```
interface ethernet 0 address-serve netbios scope name domain-name  
show interface ethernet 0 address-serve netbios scope name
```

These commands allow you to set or show the domain name under which the NetBIOS scope is enabled.

```
interface ethernet 0 address-serve netbios name-server enable { yes | no }  
no interface ethernet 0 address-serve netbios name-server enable  
show interface ethernet 0 address-serve netbios name-server enable
```

These commands allow you to set, delete, or show whether a NetBIOS name server address is served to NetBIOS clients.

```
interface ethernet 0 address-serve netbios name-server address xxx.xxx.xxx.xxx [secondary]  
show interface ethernet 0 address-serve netbios name-server address
```

These commands allow you to set or show the IP address of the NetBIOS name server.

If the keyword **secondary** is specified and there is no primary WINS server the command will be rejected as **CLI_NO_CFG_SUPPORT_ERR**, with the error message "; error 2: not supported with current configuration".

Generic WAN Interface configuration commands

Note: For possible values of *intf-type*, refer to “Interface Naming Conventions” on page 1-3. Generic WAN Interface Commands may be applied to any router WAN interface by specifying the *intf-type* **wan** together with the appropriate interface *id*. Alternatively, you can specify the more specific *intf-type* if you choose.

Generic WAN Interface Configuration Commands
<pre>interface <i>intf-type</i> <i>id</i> dle { hdlc ppp [{vcmux vcmultiplexed llcsnap}] rfc1483 [{ bridged routed }] rfc1490 } show interface <i>intf-type</i> <i>id</i> dle show interface <i>intf-type</i> <i>id</i> statistics show interface <i>intf-type</i> <i>id</i> stats interface wan 0 tracking { yes no }</pre>
Restricted WAN Interface Configuration Commands
<pre>interface { adsl ethernet isdn sdsl } <i>id</i> pppoe enable { yes no } no interface { adsl ethernet isdn sdsl } <i>id</i> pppoe enable show interface { adsl ethernet isdn sdsl } <i>id</i> pppoe enable</pre>

```
interface intf-type id dle { hdlc | ppp [{vcmux | vcmultiplexed | llcsnap}] |  
      rfc1483 [{ bridged | routed }] | rfc1490 }  
show interface intf-type id dle
```

These commands allow you to set or show the global data link encapsulation type of the interface specified by *intf-type id*. At this time you can generally think of the data link encapsulation of interface 1 as the global data link encapsulation of the router itself.

Note: **atmfuni** is accepted as a synonym for **rfc1483** and **frame-relay** is accepted as a synonym for **rfc1490**. For **ppp**, the default mode is **vcmux**. For **rfc1483**, the default mode for frame-based SDSL (R7100) interfaces is **bridged**, while the default mode for cell-based SDSL (R7200) interfaces is **routed**.

Example:

```
interface wan 1 dle frame-relay
```

```
show interface intf-type id statistics  
show interface intf-type id stats
```

These commands allow you to display statistics for the specified interface, including receive frames, octets, and errors, and transmit frames, octets, and errors. For switched ISDN interfaces, the statistics are broken down by channel.

interface wan 0 tracking { yes | no }

For D-Series CSU/DSU equipment, this command allows you to track or not track the primary interface speed. Specifying **yes** means the primary interface (AUX) speed will be tracked, which is the default. Changing this to **no** currently means that we will be running at 1.5 MHz.

Restricted WAN Interface configuration commands

interface { adsl | ethernet | isdn | sdsl } id pppoe enable { yes | no }
no interface { adsl | ethernet | isdn | sdsl } id pppoe enable
show interface { adsl | ethernet | isdn | sdsl } id pppoe enable

These commands allow you enable, disable, or show the PPP over Ethernet behavior of the specified interface.

ISDN WAN Interface configuration commands

ISDN WAN Interface Configuration Commands
Generic ISDN
interface isdn <i>id</i> mode { switched leased idsl-ascend idsl-cmn } show interface isdn <i>id</i> mode show interface isdn <i>id</i> status [b1 b2]
Permanent ISDN (IDSL) only
interface isdn <i>id</i> imux mode { mlppp dml } show interface isdn <i>id</i> imux mode interface isdn <i>id</i> speed { b1 b2 2b 2b+d } show interface isdn <i>id</i> speed
Switched ISDN only
interface isdn <i>id</i> switch { auto ni1 5esspttopt 5essmultipt dms100 ts013 euroisdn japanntt uk-euro } interface isdn <i>id</i> dn { 1 2 } <i>string</i> no interface isdn <i>id</i> dn { 1 2 } show interface isdn <i>id</i> dn { 1 2 } interface isdn <i>id</i> spid { 1 2 } <i>string</i> no interface isdn <i>id</i> spid { 1 2 } show interface isdn <i>id</i> spid { 1 2 }

```
interface isdn id imux mode { mlppp | dml }
show interface isdn id imux mode
```

These commands allow you set or show the ISDN interface IMUX bonding mode: Multilink PPP or DML (for Copper Mountain Networks central office equipment).

Example:

```
interface isdn 1 imux mode dml
```

```
interface isdn id mode { switched | leased | idsl-ascend | idsl-cmn }
show interface isdn id mode
```

These commands allow you set or show the ISDN interface mode: switched, leased, idsl-ascend (IDSL for Lucent/Ascend Communications central office equipment), or idsl-cmn (IDSL for Copper Mountain Networks central office equipment).

Example:

```
interface isdn 1 mode leased
```

```
show interface isdn id status [ b1 | b2 ]
```

This command allows you display the status of the specified ISDN/IDSL interface. For a switched ISDN interface, you may specify the optional keyword **b1** or **b2**, in which case the status of the specified B-channel is displayed rather than the status of the interface itself.

For a leased ISDN/IDSL interface, the possible status strings and their meanings are:

Status String	Meaning
Inactive	The interface is not yet active.
Waiting for rate negotiation	The interface is in the process of sensing the data rate configured for the IDSL line at the central office. This status applies only to an interface set to idsl-cmn mode, in which the router can sense the data rate automatically.
Backup recovery in progress	The interface is in the process of recovering back to the primary interface from a backup interface after a failure.
Connected at xxx Kbps	The interface is connected to the DSLAM or other end device at the specified data rate. (xxx will be one of 64, 128, or 144.)

For a switched ISDN interface, the possible status strings and their meanings are:

Status String	Meaning
Inactive	The interface is not yet active.
Active	The interface is active, and this is an interface that does not require SPIDs.
Active, <i>n</i> of <i>m</i> SPIDs registered	The interface is active, and this is an interface that requires SPIDs. <i>n</i> indicates the number of SPIDs that have been successfully registered so far, and <i>m</i> indicates the total number of SPIDs to be registered. If <i>n</i> is less than <i>m</i> , the device is still in the process of registering some of the SPIDs.

Status String	Meaning
Active, <i>n</i> of <i>m</i> SPIDs registered (<i>p</i> failed)	The interface is active, and this is an interface that requires SPIDs. <i>n</i> indicates the number of SPIDs that have been successfully registered so far, <i>m</i> indicates the total number of SPIDs to be registered, and <i>p</i> indicates the number of SPIDs that failed registration. If the sum of <i>n</i> and <i>p</i> is less than <i>m</i> , the device is still in the process of registering some of the SPIDs.

For one of the B-channels of a switched ISDN interface, the possible status strings and their meanings are:

Status String	Meaning
Inactive	The associated interface is not yet active.
Idle	The channel is not currently in use.
Speech Call	The channel is in use by a speech call.
64 Kbps Data Call	The channel is in use for a 64 Kbps data call.
56 Kbps Data Call	The channel is in use for a 56 Kbps data call.
3.1 Khz Call	The channel is in use for a 3.1 Khz call.

Example:

```
#show interface isdn 1 status
Connected at 144 Kbps
```

```
interface isdn id speed { b1 | b2 | 2b | 2b+d }
show interface isdn id speed
```

These commands, which apply only to permanent ISDN (i.e., IDSL), allow you to set or show the data rate (and B Channel usage) of the ISDN line. **b1** means use Channel B1 at 64 Kbps, **b2** means use Channel B2 at 64 Kbps, **2b** means use both Channels B1 and B2 at 128 Kbps, and **2b+d** means use all three channels at 144 Kbps.

```
interface isdn id switch { auto | ni1 | 5esspttopt | 5essmultipt | dms100 | ts013 | euroisdn |
japanntt | uk-euro }
```

This command allows you to change the ISDN switch type. This command applies only to switched ISDN. The only currently supported *id* is 1, which identifies the ISDN interface in the WAN 1 slot of the Netopia router. The WAN 2 slot (id 2) cannot be populated with an ISDN wanlet at this time, and the Motherboard (id 0) is incapable of supporting ISDN internally.

Under many circumstances it is unnecessary to explicitly set the switch type, particularly in Europe. This is because for “S/T” ISDN routers the default switch type is **euroisdn**, and for “U” ISDN routers the default switch type is **ni1**.

auto is appropriate only in the United States and allows the router to auto-determine the switch type, SPIDs, and directory numbers (DNs).

uk-euro sets the switch type to Euro-ISDN, and as a side effect sets the console's clock time display type to 24 hour (i.e., "17:45" instead of "5:45 PM").

Example:

The command to set the switch type of the wanlet for the correct value in Japan is:

```
interface isdn 1 switch japanntt
```

```
interface isdn id dn { 1 | 2 } string  
no interface isdn id dn { 1 | 2 }  
show interface isdn id dn { 1 | 2 }
```

These commands allow you to set, change, delete, or show the directory numbers associated with the specified ISDN interface. These commands apply only to switched ISDN. The only currently supported *id* is 1. The string parameter can contain up to 32 characters. Non-dialable characters are allowed (and are ignored).

Example:

```
interface isdn 1 dn 1234567
```

```
interface isdn id spid { 1 | 2 } string  
no interface isdn id spid { 1 | 2 }  
show interface isdn id spid { 1 | 2 }
```

These commands allow you to set, change, delete, or show the SPIDs associated with the specified ISDN interface. These commands apply only to switched ISDN. The only currently supported *id* is 1. The string parameter can contain up to 23 characters. Illegal characters are allowed (for instance, for formatting) and are ignored by the interface.

ADSL WAN Interface configuration commands

Note: The commands in this section are supported beginning with firmware release 4.8.

ADSL WAN Interface Configuration Commands
interface adsl <i>id</i> pvc <i>vpi-value vci-value</i> show interface adsl <i>id</i> pvc show interface adsl <i>id</i> status

interface adsl *id* **pvc** *vpi-value vci-value*
show interface adsl *id* **pvc**

These commands allow you to set, change, or show the PVC VPI and VCI values associated with the ADSL WAN interface.

show interface adsl *id* **status**

This command allows you to display the status of the specified ADSL interface. For an ADSL (R6100) interface, the possible status strings and their meanings are:

Status String	Meaning
Connected at <i>xxx rx / yyy tx</i> Kbps	The interface is connected to the DSLAM at the specified speeds, where <i>xxx</i> is the downstream (receive) speed and <i>yyy</i> is the upstream (transmit) speed, each in Kbps.
Activation Backoff	The ADSL interface is between connection attempts.
Down	The ADSL interface is not yet initialized.
No signal from DSLAM	The ADSL interface is not detecting a signal from a DSLAM.

SDSL WAN Interface configuration commands

SDSL WAN Interface Configuration Commands
<pre>interface sdsl <i>id</i> clock source { internal network } show interface sdsl <i>id</i> clock source interface sdsl <i>id</i> clock rate <i>rate-specification</i> show interface sdsl <i>id</i> clock rate interface sdsl <i>id</i> operation mode { generic lucent nokia-eoc-fast nokia-fixed paradyne promatory } [default] show interface sdsl <i>id</i> operation mode interface sdsl <i>id</i> pvc <i>vpi-value</i> <i>vci-value</i> show interface sdsl <i>id</i> pvc interface sdsl <i>id</i> priority-queuing enable { yes no } no interface sdsl <i>id</i> priority-queuing enable show interface sdsl <i>id</i> priority-queuing enable interface { sdsl isdn } <i>id</i> rfc1973 dlci { 16 .. 991 } show interface { sdsl isdn } <i>id</i> rfc1973 dlci interface { sdsl isdn } <i>id</i> rfc1973 enable { yes no } no interface ethernet { sdsl isdn } <i>id</i> rfc1973 enable show interface { sdsl isdn } <i>id</i> rfc1973 enable interface { sdsl isdn } <i>id</i> rfc1973 lmi { none lmi ccitt ansi annexa annexd } no interface { sdsl isdn } <i>id</i> rfc1973 lmi show interface { sdsl isdn } <i>id</i> rfc1973 lmi show interface sdsl <i>id</i> status</pre>

```
interface sdsl id clock source { internal | network }
show interface sdsl id clock source
```

These commands allow you to set, change, or show the clock source associated with the SDSL WAN interface.

Note: These commands apply only to frame-based SDSL (R7100) interfaces.

```
interface sdsl id clock rate rate-specification  
show interface sdsl id clock rate
```

These commands allow you to set, change, or show the data rate associated with the SDSL WAN interface.

Note: The permissible values for *rate-specification* depend on the type of SDSL WAN interface. For frame-based SDSL (R7100) interfaces, *rate-specification* may be replaced with:

{ 160 | 208 | 320 | 416 | 784 | 1040 | 1568 }

For cell-based SDSL (R7200) interfaces, *rate-specification* may be replaced with:

{ 144...2320 } [{ hunt | locked }]

See the table on the next page for possible rate specifications.

Also, **data rate** is accepted as a synonym for **clock rate**.

```
interface sdsl id operation mode { generic | lucent | nokia-eoc-fast | nokia-fixed | paradyne | nortel }  
[ default ]  
show interface sdsl id operation mode
```

Note: These commands apply only to ATM-based SDSL interfaces.

If the optional default token is included in the command, various WAN interface parameters will be set to appropriate default values, given the particular mode setting. The parameters and their values are enumerated in the table below. In addition, the data rates accepted by the **interface sdsl id data rate** command depend on what the operation mode is, and correspond to the values available from the Data Rate pop-up menu on the SDSL Line Configuration screen in the menu console. These acceptable data rates are enumerated below as well.

	Nokia	Lucent	Paradyne	Nortel
VPI	0	0	0	0
VCI	38	35	35	38
RFC 1483 Mode	Routed	Routed	Routed	Routed
Data Rate	384k	784k	784k	1536k
Data Rate Mode	HUNT	LOCKED	LOCKED	LOCKED
Clock Source	Network	Network	Network	Network
DLE	rfc1483	rfc1483	rfc1483	rfc1483
Data Rates	192k 384k 768k 1152k 1536k	144k 160k 192k 208k 272k 384k 400k 416k 528k 768k 784k 1040k 1152k 1168k 1536k 1552k 1568k 2320k	144k 272k 400k 528k 784k 1168k 1552k 2320k	144k 160k 192k 208k 272k 384k 400k 416k 528k 768k 784k 1040k 1152k 1168k 1536k 1552k 1568k 2320k

Note that setting the mode value to generic will not change any other WAN interface module parameter; thus, the following command:

```
interface sdsl 1 operation mode generic default
```

will be rejected as a syntax error.

```
interface sdsl id pvc vpi-value vci-value  
show interface sdsl id pvc
```

These commands allow you to set, change, or show the PVC VPI and VCI values associated with the SDSL WAN interface.

Note: These commands apply only to cell-based SDSL (R7200) interfaces.

```
interface sdsl id priority-queuing enable { yes | no }  
no interface sdsl id priority-queuing enable  
show interface sdsl id priority-queuing enable
```

These commands allow you to enable, disable, or show the priority queuing (QOS) behavior on the SDSL WAN interface.

```
interface { sdsl | isdn } id rfc1973 dlci { 16 .. 991 }  
show interface { sdsl | isdn } id rfc1973 dlci
```

These commands allow you to set or show an RFC 1973 DLCI for the SDSL or ISDN WAN interface.

Note that the only WAN interface modules that currently support RFC 1973 are the U/ISDN (31xx) and Copper Mountain SDSL (71xx). Attempts to set or show RFC 1973 parameters on any other WAN interface module will return an error.

```
interface { sdsl | isdn } id rfc1973 enable { yes | no }  
no interface { sdsl | isdn } id rfc1973 enable  
show interface { sdsl | isdn } id rfc1973 enable
```

These commands allow you to enable, disable, or show RFC 1973 (PPP) behavior on the SDSL or ISDN WAN interface.

```
interface { sdsl | isdn } id rfc1973 lmi { none | lmi | ccitt | ansi | annexa | annexd }  
no interface { sdsl | isdn } id rfc1973 lmi  
show interface { sdsl | isdn } id rfc1973 lmi
```

These commands allow you to specify, disable, or show the RFC 1973 (PPP) Local Management Interface (LMI) type on the SDSL or ISDN WAN interface.

The keywords **ccitt** and **annexa** are synonyms, as are the keywords **ansi** and **annexd**.

show interface sdsl id status

This command allows you to display the status of the specified SDSL interface. For a cell-based SDSL (R7200) interface, the possible status strings and their meanings are:

Status String	Meaning
Connected at xxx Kbps	The interface is connected to the DSLAM at the specified speed.
Trying xxx Kbps	The SDSL interface is attempting to connect to the DSLAM at the specified speed.
Activation Backoff	The SDSL interface is between connection attempts.
Down	The SDSL interface is not yet initialized.
No signal from DSLAM	The SDSL interface is not detecting a signal from a DSLAM.

PVCs

Note: The commands in this section are supported beginning with firmware release 4.8.

PVC Configuration Commands
<pre>interface sdsl id pvc { id tag } no interface sdsl id pvc { id tag } show interface sdsl id pvc { id tag } interface sdsl id pvc { id tag } tag tag show interface sdsl id pvc { id tag } tag interface sdsl id pvc { id tag } enable { yes no } no interface sdsl id pvc { id tag } enable show interface sdsl id pvc { id tag } enable interface sdsl id pvc { id tag } vpi vpi-val show interface sdsl id pvc { id tag } vpi interface sdsl id pvc { id tag } vci vci-val show interface sdsl id pvc { id tag } vci interface sdsl id pvc { id tag } cp { profile-id profile-tag default } show interface sdsl id pvc { id tag } cp</pre>

```

interface sdsl id pvc { id | tag }
no interface sdsl id pvc { id | tag }
show interface sdsl id pvc { id | tag }

```

These commands allow you to set, disable, or show a permanent virtual circuit. You can specify an optional circuit **tag** of up to 14 ASCII characters. The **tag** is used only to identify the circuit for management purposes, and has no significance on the wire; it is merely a convenience to aid in selecting circuits from lists. The default circuit name is "Circuit <n>", where <n> is replaced with a single decimal ASCII digit (between one and eight) corresponding to the circuit's position in the list of up to eight circuits.

tag

```

interface sdsl id pvc { id | tag } tag tag
show interface sdsl id pvc { id | tag } tag

```

These commands allow you to set or show a permanent virtual circuit identified by **tag**.

enable

```

interface sdsl id pvc { id | tag } enable { yes | no }
no interface sdsl id pvc { id | tag } enable
show interface sdsl id pvc { id | tag } enable

```

These commands allow you to enable, disable, or show a permanent virtual circuit.

vpi and vci

```

interface sdsl id pvc { id | tag } vpi vpi-val
show interface sdsl id pvc { id | tag } vpi

```

These commands allow you to set or show the Virtual Path Identifier value **vpi** for a permanent virtual circuit.

```

interface sdsl id pvc { id | tag } vci vci-val
show interface sdsl id pvc { id | tag } vci

```

These commands allow you to set or show the Virtual Channel Identifier value **vci** for a permanent virtual circuit.

The **vpi** and **vci** allow you to configure the Virtual Path Identifier and Virtual Channel Identifier which together identify the ATM permanent virtual circuit used between the router and the remote device. The values configured for these items must match those configured in the remote device for data to flow between the devices. The **vpi** may be set to any value between zero (0) and 255. (Earlier firmware versions allowed only a VPI value of zero (0). This restriction has been removed beginning with router firmware version 4.7.1. However, the ability to set a non-zero VPI value depends on both the router firmware revision and the WAN interface module firmware revision. In order to use a non-zero **vpi** value, the WAN interface module firmware version must be version 1.0.20 or later. If an earlier version of WAN interface module firmware is present on the SDSL WAN interface module, the **vpi** value will be restricted to zero (0).) The **vci** may be set to any value between 0 and 65535.

profile

```
interface sdsi id pvc { id | tag } cp { profile-id | profile-tag | default }  
show interface sdsi id pvc { id | tag } cp
```

These commands allow you to set or show the connection profile assigned to the specified PVC.

Note: **default** means that the router will use the first appropriate connection profile or the Default Profile if an appropriate connection profile is not found.

DSL Line Type Interface Configuration Commands

Note: The commands in this section are supported beginning with firmware version 5.2 for router and IAD platforms that support multiple line types.

DSL Line Type Interface Configuration Commands
<pre>interface dsl <i>id</i> line type { g.shdsl sdsl-atm sdsl-hdlc idsl-cmn idsl-leased } show interface dsl <i>id</i> line type</pre>

```
interface dsl id line type { g.shdsl | sdsl-atm | sdsl-hdlc | idsl-cmn | idsl-leased }
show interface dsl id line type
```

These commands allow you to set or show the line type for the specified DSL interface.

The **line type** command is supported for platforms that can be configured for different types of DSL connections. These include certain 4000-Series data routers and 4000-Series IADs:

- For SDSL IADs, **sdsl-atm** and **sdsl-hdlc** line types are supported.
- For G.SHDSL data routers and IADs, all of the available line types are supported.

T1 WAN Interface configuration commands

Beginning with the version 4.9 firmware release, the Command Line Interface supports the following new interface configuration commands for configuring T1 WAN interfaces:

T1 WAN Interface Configuration Commands
<pre>interface t1 id buildout { auto 0-0.6 7.5 15.0 22.5 } show interface t1 id buildout interface t1 id channels count integer [start integer] [{ alternating contiguous }] [rate { 56 64 56k 64k Nx56k Nx64k }] show interface t1 id channels interface t1 id clock source { internal network } show interface t1 id clock source interface t1 id dle { ppp hdlc rfc1490 } show interface t1 id dle interface t1 id framing { d4 esf } show interface t1 id framing interface t1 id encoding { ami b8zs } show interface t1 id encoding interface t1 id operation mode { normal copper-mountain } show interface t1 id operation mode interface t1 id prm-enable { yes no } show interface t1 id prm-enable no interface t1 id prm-enable interface t1 id rfc1973 enable { yes no } show interface t1 id rfc1973 enable no interface t1 id rfc1973 enable interface t1 id rfc1973 dlci { 16..991 } show interface t1 id rfc1973 dlci interface t1 id rfc1973 lmi { annexa annexd ansi ccitt lmi none } show interface t1 id rfc1973 lmi no interface t1 id rfc1973 lmi</pre>

```
interface t1 id buildout { auto | 0-0.6 | 7.5 | 15.0 | 22.5 }
show interface t1 id buildout
```

These commands set or display the line buildout for the specified T1 WAN interface.

```
interface t1 id channels
    count integer
    [ start integer ]
    [ { alternating | contiguous } ]
    [ rate { 56 | 64 | 56k | 64k | Nx56k | Nx64k } ]
show interface t1 id channels
```

These commands set or display which DS0 channels are utilized on the specified T1 WAN interface, and the rate of those DS0 channels. The **count** clause is always required. The **start** clause is required unless the **count** clause specifies 24 channels, in which case if the **start** clause is not present, the starting channel number is assumed to be channel 1. If neither the **alternating** nor the **contiguous** keyword is specified, the **contiguous** keyword is assumed unless the line encoding is AMI and the **count** clause specifies two or more channels, in which case the **alternating** keyword is assumed. The **rate** clause is always optional. If the **rate** clause is not present, the value **Nx64k** is assumed, unless the line encoding is AMI, the **count** clause specifies two or more channels, and the **contiguous** keyword is specified, in which case the value **Nx56k** is assumed.

```
interface t1 id clock source { internal | network }
show interface t1 id clock source
```

These commands set or display the clock source for the specified T1 WAN interface.

```
interface t1 id dle { ppp | hdlc | rfc1490 }
show interface t1 id dle
```

These commands set or display the data link encapsulation (DLE) for the specified T1 WAN interface.

Note: **frame-relay** is accepted as a synonym for **rfc1490**.

```
interface t1 id framing { d4 | esf }
show interface t1 id framing
```

These commands set or display the framing mode for the specified T1 WAN interface.

```
interface t1 id encoding { ami | b8zs }
show interface t1 id encoding
```

These commands set or display the line encoding for the specified T1 WAN interface.

Note: If this command changes the line encoding from **b8zs** to **ami** and there are two or more contiguous Nx64k channels in use, the channel data rate will be changed to Nx56k.

```
interface t1 id operation mode { normal | copper-mountain }  
show interface t1 id operation mode
```

These commands set or display the operation mode for the specified T1 WAN interface. The keyword **copper-mountain** should be specified when connected to a Copper Mountain DSLAM T1 line card; the keyword **normal** should be specified in all other situations.

```
interface t1 id prm-enable { yes | no }  
show interface t1 id prm-enable  
no interface t1 id prm-enable
```

These commands set or display whether or not ANSI PRMs are sent on the specified T1 WAN interface.

```
interface t1 id rfc1973 enable { yes | no }  
show interface t1 id rfc1973 enable  
no interface t1 id rfc1973 enable
```

These commands set or display whether or not PPP in Frame Relay (RFC1973) is enabled on the specified T1 WAN interface.

```
interface t1 id rfc1973 dlci { 16..991 }  
show interface t1 id rfc1973 dlci
```

These commands set or display the DLCI used for PPP in Frame Relay (RFC1973) on the specified T1 WAN interface.

```
interface t1 id rfc1973 lmi { annexa | annexd | ansi | ccitt | lmi | none }  
show interface t1 id rfc1973 lmi  
no interface t1 id rfc1973 lmi
```

These commands set or display the Local Management Interface (LMI) type for PPP in Frame Relay (RFC1973) on the specified T1 WAN interface.

T1 Statistic and Diagnostic commands

Beginning with the version 4.9 firmware release, the Command Line Interface supports the following new statistic and diagnostic commands for T1 WAN interfaces:

T1 Statistic and Diagnostic Commands
<pre>show interface t1 id errors { current interval 1..96 total } interface t1 id diagnostic mode { local loopback normal remote loopback send { all ones blue alarm loopback } } show interface t1 id diagnostic mode show interface t1 id line status show interface t1 id loopback mode show interface t1 id loopback status</pre>

show interface t1 id errors { current | interval 1..96 | total }

This command displays the error statistics for the specified T1 WAN interface for a particular 15-minute interval during the previous 24-hour period, or the total for the past 24 hours. Specifying the keyword **current** displays the error statistics for the current 15-minute interval. Specifying the keyword **interval** followed by an integer between 1 and 96 displays the error statistics for a prior 15-minute interval. Interval 1 is the most recently completed 15-minute interval, while interval 96 is the interval completed 23 hours and 45 minutes prior to interval 1. Specifying the keyword **total** displays the total error statistics for the last 24 hours.

Example:

```
#show interface t1 1 errors interval 1
15 minutes ending 16:32:44
Errored Seconds          001
Unavailable Seconds      000
Severely Errored Seconds 001
Bursty Errored Seconds   001
Loss of Frame Count      000
Bipolar Violation Count  001

#show interface t1 1 errors total
24 hours ending 16:32:44
Errored Seconds          001
Unavailable Seconds      000
Severely Errored Seconds 001
Bursty Errored Seconds   001
Loss of Frame Count      000
Bipolar Violation Count  001

#show interface t1 1 errors current
Current Interval elapsed time 02:45
Errored Seconds          002
Unavailable Seconds      000
Severely Errored Seconds 001
Bursty Errored Seconds   001
Loss of Frame Count      000
Bipolar Violation Count  000
```

```
interface t1 id diagnostic mode { local loopback | normal | remote loopback |
    send { all ones | blue alarm | loopback } }
show interface t1 id diagnostic mode
```

This command sets or displays the diagnostic mode for the specified T1 interface. Specifying **local loopback** puts the near end in local payload loopback mode. Specifying **remote loopback** instructs the far end to put itself in payload loopback mode. Specifying **send all ones** or **send blue alarm** (which are synonyms) causes the near end to start sending an all-ones pattern, which puts the far end in the red alarm state, causing it to send back a yellow alarm. Specifying **send loopback** causes the near end to begin sending loopback packets. Specifying **normal** cancels the effect of any previous diagnostic mode command.

After issuing the **diagnostic mode send loopback command**, the loopback progress can be monitored by issuing the **loopback status** command (see below).

show interface t1 id line status

This command displays the line status on the specified T1 interface. This will display one of the following strings:

```
Red Alarm
Yellow Alarm
Blue Alarm
Normal Operation
```

show interface t1 id loopback mode

This command displays the loopback mode of the specified T1 interface. This will display one of the following strings:

```
Layer 1 Activation Not Present
Local Payload Loopback Enabled
Remote Line Loopback Enabled
Remote Payload Loopback Enabled
Clear - No Loopback Enabled
```

show interface t1 id loopback status

This command displays the progress of the loopback test on the specified T1 interface. This will display one of the following strings:

```
Loopback Not Active
PASS (xxxxxx good, yyyyyy bad packets)
FAIL (xxxxxx good, yyyyyy bad packets)
```

Examples:

```
#show interface t1 1 loopback status
Loopback Not Active
#interface t1 1 diagnostic mode send loopback
#show interface t1 1 loopback status
PASS (00255 good, 00000 bad packets)
#show interface t1 1 loopback status
FAIL (00000 good, 00256 bad packets)
```

Global IP Configuration Commands

Global IP Configuration Commands
<pre>ip dns { 1 2 } ip-addr no ip dns [{ 1 2 } [ip-addr]] show ip dns [{ 1 2 }] ip domain-name string no ip domain-name [string] show ip domain-name ip gateway ip-addr [backup] no ip gateway [ip-addr backup { ip-addr backup }] show ip gateway ip route { ip-addr/mask-bits ip-addr mask } gw-ip-addr [{ high low }] [advertise [{no distance}] [{enable disable}] no ip route { ip-addr/mask-bits ip-addr mask } gw-ip-addr show ip route [{ static ip-addr ip-addr/mask-bits ip-addr mask }]</pre>

```
ip dns { 1 | 2 } ip-addr
no ip dns [ { 1 | 2 } [ip-addr] ]
show ip dns [ { 1 | 2 } ]
```

These commands allow you to set, change, delete, or show the router’s primary and secondary domain name server addresses.

```
ip domain-name string
no ip domain-name [string]
show ip domain-name
```

These commands allow you to set, change, delete, or show the domain name of the router. *string* can be up to 64 characters in length and may contain only valid domain name characters (alpha-numeric characters, dot (“.”), and dash (“-”). Note that email addresses contain the at symbol ‘@’ and are not valid domain names.

```
ip gateway ip-addr [ backup ]
no ip gateway [ ip-addr ] [ ip-addr | backup | { ip-addr backup } ]
show ip gateway
```

These commands allow you to set, change, delete, or show the router’s default gateway. Specifying **backup** applies changes to the gateway used for backup in routers so equipped.

```

ip route { ip-addr/mask-bits | ip-addr mask } gw-ip-addr [{ high | low }]
               [advertise [{no | distance}] [{enable | disable}]
no ip route { ip-addr/mask-bits | ip-addr mask } gw-ip-addr
show ip route [{ static | ip-addr | ip-addr/mask-bits | ip-addr mask }]

```

The **ip route** and **no ip route** commands allow you to add, change, or delete static routes. The **show ip route static** form of the **show ip route** command displays the configured static routes (including invalid or disabled ones), while the other forms of the **show ip route** command display the router's IP routing table (including any installed (i.e., valid and enabled) static routes).

The destination network may be specified as an IP address and mask in either prefix or dotted-quad notation. *gw-ip-addr* is the IP address of the next-hop router, and should be on one of the router's directly connected IP subnets.

The keywords **high** and **low** control the priority of the static route relative to an identical route learned via RIP. A static route with **high** priority (the default) takes precedence over an identical route learned via RIP, while an identical route learned via RIP takes precedence over a static route with **low** priority.

The keyword **advertise** controls whether or not the router will advertise (redistribute) the static route via RIP. The keyword **advertise** may be followed by a RIP metric (*distance*) between 1 (the default) and 15 inclusive.

If the **show ip route** command includes an optional IP address or IP address and mask, the route, if any, in the IP routing table that pertains to the specified destination network, subnetwork, or host address is displayed.

Examples:

```

ip route 192.168.2.0/24 192.168.1.123 low advertise

no ip route 192.168.2.0 255.255.255.0 192.168.1.123

show ip route

show ip route static

```

ARP Configuration Commands

ARP Configuration Commands	
arp <i>ip-addr hw-address</i>	
arp <i>ip-addr hw-address interface-id</i>	(D7100 CSU only)
no arp <i>ip-addr hw-address</i>	
show arp static	
clear arp-cache	
show arp-cache	

Note: The *hw-address* format for an Ethernet MAC address is six hexadecimal values between 00 and FF inclusive separated by colons. Thus, the following is an example of a valid Ethernet MAC address *hw-address*:

00:00:C5:70:00:04

arp *ip-addr hw-address*
arp *ip-addr hw-address interface-id* (D7100 CSU only)

This command allows you to create or modify a global ARP cache entry. If the model number of the Netopia router is D7100 (CSU), then the *interface-id* is required so that the device knows the interface with which to associate the ARP entry.

no arp *ip-addr hw-address*

This command allows you to remove a global ARP cache entry. Note that this does not affect entries in the interface-specific caches acquired via ARP requests and responses. To flush the interface-specific ARP caches, use the **clear arp-cache** command, described below.

show arp static

This command displays global ARP cache entries configured using the **arp** command described above.

clear arp-cache

This command allows you to flush all of the interface-specific ARP caches. It does not affect any entries in the global ARP cache, described above.

show arp-cache

This command allows you to display the global ARP cache as well as the ARP cache for each active interface that supports ARP.

Connection Profile Commands

Connection Profile Commands

```

cp { name | index }
no cp { name | index }
show cp { name | index }

cp { name | index } enable { yes | no }
no cp { name | index } enable
show cp { name | index } enable

cp { name | index } tag string
show cp { name | index } tag

cp { name | index } dle { hdlc | ppp | frame-relay | rfc1483 | atmp | pptp | ipsec }
show cp { name | index } dle

cp { name | index } filterset string
no cp { name | index } filterset [string]
show cp { name | index } filterset

cp { name | index } ip addressing { numbered | unnumbered }
show cp { name | index } ip addressing

cp { name | index } ip address local { ip-addr | ip-addr/mask-bits | ip-addr mask }
no cp { name | index } ip address local
show cp { name | index } ip address local

cp { name | index } ip address remote { ip-addr | ip-addr/mask-bits | ip-addr mask }
no cp { name | index } ip address remote
show cp { name | index } ip address remote

cp { name | index } ip dhcp client mode { standard | copper-mountain | cmn }
show cp { name | index } ip dhcp client mode

cp { name | index } ip mask local ip-mask
no cp { name | index } ip mask local
show cp { name | index } ip mask local

cp { name | index } ip mask remote ip-mask
no cp { name | index } ip mask remote
show cp { name | index } ip mask remote

```

Connection Profile Commands (continued, 1)

cp { *name* | *index* } **ip negotiate-lan** { yes | no }

no cp { *name* | *index* } **ip negotiate-lan**

show cp { *name* | *index* } **ip negotiate-lan**

cp { *name* | *index* } **ip netbios proxy enable** { yes | no }

no cp { *name* | *index* } **ip netbios proxy enable**

show cp { *name* | *index* } **ip netbios proxy enable**

cp { *name* | *index* } **ip rip receive** { no | v1 | v2 | both }

no cp { *name* | *index* } **ip rip receive**

show cp { *name* | *index* } **ip rip receive**

cp { *name* | *index* } **ip rip transmit** { no | v1 | v2broadcast | v2multicast }

no cp { *name* | *index* } **ip rip transmit**

show cp { *name* | *index* } **ip rip transmit**

cp { *name* | *index* } **ppp authentication type** { none | pap | chap }

no cp { *name* | *index* } **ppp authentication type**

show cp { *name* | *index* } **ppp authentication type**

cp { *name* | *index* } **ppp authentication** { send | receive } *name string*

no cp { *name* | *index* } **ppp authentication** { send | receive } *name*

show cp { *name* | *index* } **ppp authentication** { send | receive } *name*

cp { *name* | *index* } **ppp authentication** { send | receive } **password** *string*

no cp { *name* | *index* } **ppp authentication** { send | receive } **password**

cp { *name* | *index* } **ppp usage** { 1 | 2 [preemptible] [dynamic] }

show cp { *name* | *index* } **ppp usage**

cp { *name* | *index* } **frame-relay dlci auto-detect** { yes | no }

no cp { *name* | *index* } **frame-relay dlci auto-detect**

show cp { *name* | *index* } **frame-relay dlci auto-detect**

cp { *name* | *index* } **frame-relay dlci multicast-number** { 0 | 16 ... 991 }

no cp { *name* | *index* } **frame-relay multicast-number**

show cp { *name* | *index* } **frame-relay dlci multicast-number**

cp { *name* | *index* } **telco direction** { in | out | both }

show cp { *name* | *index* } **telco direction**

cp { *name* | *index* } **telco dn** [1 | 2] *string*

no cp { *name* | *index* } **telco dn** [1 | 2]

show cp { *name* | *index* } **telco dn** [1 | 2]

Connection Profile Commands (continued, 2)

cp { *name* | *index* } **telco prefix** *string*
no cp { *name* | *index* } **telco prefix**
show cp { *name* | *index* } **telco prefix**

cp { *name* | *index* } **telco callback** { **yes** | **no** }
no cp { *name* | *index* } **telco callback**
show cp { *name* | *index* } **telco callback**

cp { *name* | *index* } **ip nat enable** { **yes** | **no** }
no cp { *name* | *index* } **ip nat enable**
show cp { *name* | *index* } **ip nat enable**

cp { *name* | *index* } **ip nat map-list** *list-tag*
no cp { *name* | *index* } **ip nat map-list**
show cp { *name* | *index* } **ip nat map-list**

cp { *name* | *index* } **ip nat server-list** *list-tag*
no cp { *name* | *index* } **ip nat server-list**
show cp { *name* | *index* } **ip nat server-list**

Note: The last six commands above are supported beginning with firmware release 4.4.

show cp { *name* | *index* } **id**

cp { *name* | *index* } **connection demand** { **yes** | **no** }
cp { *name* | *index* } **connection timeout** *seconds*

Note: The last three commands above are supported beginning with firmware release 4.6.

Connection Profile Commands (continued, 3)
Connection Profile PPTP Commands
<p>cp { <i>name</i> <i>index</i> } pptp ip partner <i>ip-addr</i></p> <p>cp { <i>name</i> <i>index</i> } pptp ip via <i>ip-addr</i></p> <p>cp { <i>name</i> <i>index</i> } pptp authentication type { pap chap mschap }</p> <p>cp { <i>name</i> <i>index</i> } pptp compression { none standardlzs }</p> <p>no cp { <i>name</i> <i>index</i> } pptp compression</p> <p>cp { <i>name</i> <i>index</i> } pptp encryption { none mppe }</p> <p>no cp { <i>name</i> <i>index</i> } pptp encryption</p> <p>cp { <i>name</i> <i>index</i> } pptp authentication { send receive } name <i>string</i></p> <p>no cp { <i>name</i> <i>index</i> } pptp authentication { send receive } name</p> <p>show cp { <i>name</i> <i>index</i> } pptp authentication { send receive } name</p> <p>cp { <i>name</i> <i>index</i> } pptp authentication { send receive } password <i>string</i></p> <p>no cp { <i>name</i> <i>index</i> } pptp authentication { send receive } password</p>

Connection Profile Commands (continued, 4)
--

Connection Profile Manual Connect/Disconnect Commands

connect cp { <i>name</i> <i>index</i> } disconnect cp { <i>name</i> <i>index</i> }

Connection Profile Backup Configuration Commands
--

cp { <i>name</i> <i>index</i> } interface-group { primary backup auxiliary } show cp { <i>name</i> <i>index</i> } interface-group
--

Note on Connection Profile numbering sequence

The Easy Setup Profile is always assumed to be the Primary Connection Profile, whether or not it exists. The menu console reserves the index number 1 (one) for the Easy Setup Profile, even if you do not create an Easy Setup Profile.

If you do not create an Easy Setup Profile using the Easy Setup screens, but instead use the WAN Configuration/Add Connection Profile screen to create a Connection Profile, the menu console will name it *Profile 1* by default (you can rename it anything you want). Nevertheless, the router will always assign such a profile the index number 2 (two). Profiles added subsequently are internally indexed incrementally.

This can be confusing, when issuing CLI commands because it is possible for *Profile 1* to be indexed by the router as the second profile. *Profile 2* is indexed as the third, and so on.

This is illustrated in the following menu console screen:

```

                                WAN Configuration
+--Profile Name-----IP Address----IPX Network--+
+
Easy Setup Profile          0.0.0.0
Profile 1                   0.0.0.0
Profile 2                   0.0.0.0
Profile 3                   0.0.0.0
+-----+

Up/Down Arrow Keys to select, ESC to dismiss, Return/Enter to Edit.
```

```
cp { name | index }
```

This command allows you to create a connection profile, which is a structure used to define WAN connections. *index* can be any value from 1 to 16. The command has no effect if the profile already exists. The name of the profile defaults to "profile *index*" (e.g., "profile 1").

```
no cp { name | index }
```

This command allows you to delete a connection profile.

```
cp { name | index } enable { yes | no }
```

This command allows you to enable or disable connection profile index. Use this command if you want to temporarily disable a profile but do not want to lose its configuration.

```
cp { name | index } tag string
```

This command allows you to name connection profile string, which can be up to 32 characters long.

```
cp { name | index } dle { hdlc | ppp | frame-relay | rfc1483 | atmp | pptp | ipsec }
```

This command allows you to set the encapsulation type that the connection profile will use when establishing a WAN connection. Note that a profile whose encapsulation type is incompatible with the global encapsulation type is essentially disabled and cannot be used. Also, when a profile is created, it inherits the global encapsulation type by default and thus it is not usually necessary to explicitly set this parameter.

Note: **atmfuni** is accepted as a synonym for **rfc1483**, and **frame-relay** is accepted as a synonym for **rfc1490**. **ipsec** is supported beginning with firmware version 4.8.

```
cp { name | index } filterset string  
no cp { name | index } filterset [string]  
show cp { name | index } filterset
```

These commands allow you to associate a filter set with the a connection profile. The filter set is identified by name.

```
cp { name | index } ip addressing { numbered | unnumbered }
```

This command allows you to specify whether or not the WAN interface using this profile has an IP address. With point-to-point connections, a WAN IP address is not necessary for the router to function properly, but may be required, depending on how the equipment at the other end is configured.

In firmware versions prior to firmware release 4.4, the local WAN IP address was closely associated with Network Address Translation, in that it was the exterior address used by NAT to do Port Address Translation. Beginning with firmware release 4.4, this constraint has been removed, although NAT can still make use of an address assigned via PPP or DHCP. See the description of the **ip nat public** command on [page 2-74](#) for more information.

```
cp { name | index } ip address local { ip-addr | ip-addr/mask-bits | ip-addr mask }
```

This command allows you to set the profile's local WAN IP address.

```
cp { name | index } ip address remote { ip-addr | ip-addr/mask-bits | ip-addr mask }
```

This command allows you to set the profile's remote WAN IP address.

```
cp { name | index } ip dhcp client mode { standard | copper-mountain | cmn }  
show cp { name | index } ip dhcp client mode
```

Note: These commands are supported beginning with firmware version 4.8.

These commands allow you to set or show the router's DHCP mode, whether **standard**, **copper-mountain**, or **cmn**.

The connection profile, default profile, and IP configuration structures now include a **dhcp client mode** setting that selects between the **standard** RFC 2131 standards-based mode of operation (the default), and the **copper-mountain** or **cmn** proprietary mode of operation.

When the DHCP client is activated on a RFC1483 MER interface, it examines the **dhcp client mode** in the associated connection profile (or the default profile there was no explicitly configured connection profile). If the **dhcp client mode** specifies **standard**, the DHCP client initializes the htype and hlen fields in the header of its DHCP requests to the appropriate values for an RFC1483 MER interface (htype = 1 and hlen = 6). If the **dhcp client mode** specifies **copper-mountain** or **cmn**, the DHCP client initializes the htype and hlen fields in the header of its DHCP requests to zero.

When the DHCP client is activated on an Ethernet WAN interface, it examines the **dhcp client mode** in the associated IP configuration structure, and behaves as described above for the RFC1483 MER DHCP client.

Note: **cmn** is accepted as a synonym for **copper-mountain**.

```
cp { name | index } ip mask local ip-mask
```

This command allows you to set the profile's local WAN IP mask.

```
cp { name | index } ip mask remote ip-mask
```

This command allows you to set the profile's remote WAN IP mask.

```

cp { name | index } ip negotiate-lan { yes | no }
no cp { name | index } ip negotiate-lan
show cp { name | index } ip negotiate-lan

```

Note: These commands are supported beginning with firmware version 4.8.

These commands allow you to set, delete, or show whether the specified connection profile will attempt to negotiate the router hub's IP address and subnet mask from the central site router.

Beginning with version 4.8, the firmware adds PPP support for the IPCP Subnet Mask option documented in *PPP Internet Protocol Control Protocol Extensions for IP Subnet*, draft-helenius-ppp-subnet-00.txt. This option, together with the IPCP IP Address option, allows a central site router to supply an entire IP subnet, rather than a single IP address, for use by a CPE router.

PPP Ethernet LAN reconfiguration is controlled by an **ip negotiate-lan** connection profile flag. If the applicable connection profile specifies an unnumbered, non-NAT connection and the **ip negotiate-lan** flag is **yes**, PPP will attempt to negotiate both an IP Address and subnet mask.

```

cp { name | index } ip netbios proxy enable { yes | no }
no cp { name | index } ip netbios proxy enable
show cp { name | index } ip netbios proxy enable

```

Note: These commands are supported beginning with firmware version 4.9.4.

These commands allow you to enable, disable, or show the NetBIOS proxy status for the specified Connection Profile. The NetBIOS proxy enables the ability to forward Windows Networking NetBIOS broadcasts. This is useful for, for example, a Virtual Private Network, in which you want to be able to browse the remote network to which you are tunnelling, as part of your Windows Network Neighborhood.

Routed connections, such as VPNs, can not use NetBEUI to carry the Network Neighborhood information. They need to use NetBIOS, because NetBEUI cannot be routed. This feature will allow browsing the Network Neighborhood without any additional workstation configuration.

Note: Microsoft Network browsing is available with or without a Windows Internet Name Service (WINS) server. Shared volumes on the remote network are accessible with or without a WINS server. Local LAN shared volumes that have Port Address Translation (PAT) applied to them are *not* available to hosts on the remote LAN. For tunnelled traffic, NAT on the WAN has no effect on the Microsoft Networking traffic.

```

cp { name | index } ip rip receive { no | v1 | v2 | both }

```

This command allows you to set the RIP receive behavior when the profile is used for a WAN connection.

```

cp { name | index } ip rip transmit { no | v1 | v2broadcast | v2multicast }

```

This command allows you to set the RIP transmit behavior when the profile is used for a WAN connection.

Note: If network address translation is enabled, RIP transmit is disabled regardless of the current setting of this parameter.

cp { *name* | *index* } **ppp authentication type** { **none** | **pap** | **chap** }

This command allows you to configure the type of authentication used by the profile.

cp { *name* | *index* } **ppp authentication** { **send** | **receive** } **name** *string*

This command allows you to configure the send or receive PPP authentication name. The send name is used when the remote side attempts to authenticate the Netopia router, and the receive name is used when the Netopia router is attempting the authentication (for instance, if a WAN connection is being established to the router).

cp { *name* | *index* } **ppp authentication** { **send** | **receive** } **password** *string*

This command allows you to configure the send or receive PPP authentication password (or secret) associated with the send or receive names.

cp { *name* | *index* } **ppp usage** { **1** | **2** [**preemptible**] [**dynamic**] }

This command allows you to configure the characteristics of how the channels of the interface are used. The number indicates the maximum number of channels to use.

If you specify the keyword **preemptible** and more than one channel is being used for the connection, additional calls (both data and voice, when applicable) may borrow a channel for their own use.

If you specify the keyword **dynamic** channels are added and removed from the connection based on bandwidth usage. If traffic exceeds a certain threshold for a certain amount of time, and if there is a free channel available, it will be used for the connection. Conversely, if more than one channel is being used by the connection and traffic drops below a certain level for a certain amount of time, a channel will be dropped.

The keywords **dynamic** and **preemptible** may be specified only if the number of channels is 2.

Note: With the current firmware, a dynamic 2B Channel profile will also be preemptible, regardless of whether or not the **preemptible** keyword is specified.

Examples:

These examples illustrate all forms of the command that you are likely to use:

```
cp 1 ppp usage 1
```

```
cp 1 ppp usage 2
```

```
cp 1 ppp usage 2 preemptible
```

```
cp 1 ppp usage 2 dynamic
```

cp { *name* | *index* } **frame-relay dlci auto-detect** { **yes** | **no** }

This command allows you to enable or disable the automatic detection of Frame Relay DLCIs when the profile is used to establish a WAN connection.

cp { *name* | *index* } **frame relay dlci multicast-number** { **0** | **16 ... 991** }

This command allows you to specify the DLCI multicast number for the profile.

cp { *name* | *index* } **telco direction** { **in** | **out** | **both** }

This command allows you to set whether this profile will be used to establish WAN connections (keyword **out**), to establish inbound connections (keyword **in**), or to establish both (keyword **both**).

cp { *name* | *index* } **telco dn** [**1** | **2**] *string*

This command allows you to set the profile's directory number, or number-to-dial (DN). The number can be up to 32 characters in length and may contain non-dialable characters, which are ignored when placing a call.

Beginning with firmware version 4.9.1, the profile directory number command can handle both the primary [1] and alternate [2] DN.

Note: For the **cp** and **no cp** versions of this command, if no DN index is specified, **1** is assumed. For the **show cp** version of this command, if no DN index is specified, both DNs will be displayed, each on its own line.

cp { *name* | *index* } **telco prefix** *string*

This command allows you to set the profile's dial prefix. The prefix can be up to three characters long and may contain non-dialable characters, which are ignored when placing a call. The prefix field is prepended to the directory number when placing a call.

Note: This parameter is used ONLY by routers with analog modem interfaces installed.

cp { *name* | *index* } **telco callback** { **yes** | **no** }

This command allows you to configure a profile so that when it is used to accept an incoming call, the router will hang up that call and use its (prefix and) directory number to call back the device that originated the initial call. This is useful when you want a particular party to be billed for WAN connections.

cp { *name* | *index* } **ip nat enable** { **yes** | **no** }

This command allows you to enable or disable Network Address Translation for the profile. Note that for firmware versions 4.4 and newer, enabling NAT is not sufficient – you must also attach a rule list and optionally a server list using the commands below.

Note: The remaining connection profile NAT commands are supported beginning with firmware release 4.4.

cp { *name* | *index* } **ip nat rule-list** *list-tag*

This command allows you to attach a previously configured IP NAT rule list to a particular profile. *list-tag* should be the name of the desired rule list to use for this profile.

no cp { *name* | *index* } **ip nat rule-list**

This command allows you to detach an IP NAT rule list from a particular profile.

cp { *name* | *index* } **ip nat server-list** *list-tag*

This command allows you to attach a previously configured IP NAT server list to a particular profile. *list-tag* should be the name of the desired server list to use for this profile.

no cp { *name* | *index* } **ip nat server-list**

This command allows you to detach an IP NAT Server List from a particular profile.

show cp { *name* | *index* } id

This command displays a connection profile's name and index number. *name* can be any unique descriptive alphanumeric string. *index* can be any value from 1 to 16.

cp { *name* | *index* } connection demand { **yes | **no** }**

This command allows you to specify whether or not a connection profile will connect "on demand".

cp { *name* | *index* } connection timeout *seconds*

This command allows you to specify the idle timeout value in seconds for a connection profile.

PPTP commands

```
cp { name | index } pptp ip partner ip-addr
```

This command allows you to specify a PPTP partner IP address for a particular connection profile specified by *name* or *index*.

```
cp { name | index } pptp ip via ip-addr
```

This command allows you to specify a gateway by which the PPTP partner IP address can be reached when the partner address is in the same subnet as the remote IP address.

If you do not specify the PPTP partner IP address, the router will use the default gateway to reach the partner. If the partner should be reached via an alternate port (i.e. the LAN instead of the WAN), the Tunnel Via Gateway field allows this path to be resolved.

```
cp { name | index } pptp authentication type { pap | chap | mschap }
```

This command allows you to specify a PPTP authentication type, PAP, CHAP, or MS-CHAP, for a particular connection profile specified by *name* or *index*.

```
cp { name | index } pptp compression { none | standardlzs }  
no cp { name | index } pptp compression
```

These commands allow you to specify or delete a PPTP compression algorithm, either none or Standard LZS, for a particular connection profile specified by *name* or *index*.

```
cp { name | index } pptp encryption { none | mppe }  
no cp { name | index } pptp encryption
```

These commands allow you to specify or delete a PPTP encryption algorithm, either none or MPPE, for the specified connection profile.

```
cp { name | index } pptp authentication { send | receive } name string  
no cp { name | index } pptp authentication { send | receive } name
```

These commands allow you to set or delete the user name as an alphanumeric string that the specified connection profile will use for PPTP authentication.

```
show cp { name | index } pptp authentication { send | receive } name
```

This command allows you to show the user name as an alphanumeric string that the specified connection profile uses for PPTP authentication.

```
cp { name | index } pptp authentication { send | receive } password string  
no cp { name | index } pptp authentication { send | receive } password
```

These commands allow you to set or delete the password as an alphanumeric string that the specified connection profile will use for PPTP authentication.

Manual connect/disconnect commands

connect cp { *name* | *index* }

Invoking this command with a valid, applicable connection profile will cause the router to attempt to make the appropriate connection, using the profile's settings. A valid, applicable connection profile must be either a profile that matches the primary WAN interface's data link encapsulation, or a tunnel profile.

If the specified profile is valid in this context, the console remains in a modal state until one of the following occurs:

- you type Control-C
- the connection is established, in which case the word "connect" is displayed
- the connection fails, in which case the word "down" is displayed, followed by an appropriate error message

disconnect cp { *name* | *index* }

This command allows you to disconnect the connection, if any, associated with the specified profile. If no connection is in place an error message is displayed. This command returns immediately; the connection disconnect process may still be in progress since it is asynchronous.

Backup configuration commands

Note: The commands in this section are supported beginning with firmware release 4.8.

cp { *name* | *index* } **interface-group** { **primary** | **backup** | **auxiliary** }
show cp { *name* | *index* } **interface-group**

These commands allow you to set or show the interface group to which the dial backup feature is applied.

Note: **auxiliary** is only allowed if the router is an Ethernet-to-Ethernet router that has the dial-in kit installed.

CompuServe Login

Beginning with the version 4.9 firmware release, the Command Line Interface supports the following new and modified Connection Profile configuration commands:

CompuServe Login Connection Profile Commands
cp { name index } telco compuserve login { yes no } show cp { name index } telco compuserve login no cp { name index } telco compuserve login
cp { name index } telco compuserve hostname string show cp { name index } telco compuserve hostname no cp { name index } telco compuserve hostname
cp { name index } telco compuserve username string show cp { name index } telco compuserve username no cp { name index } telco compuserve username
cp { name index } telco compuserve password string no cp { name index } telco compuserve password

cp { name | index } telco compuserve login { yes | no }
show cp { name | index } telco compuserve login
no cp { name | index } telco compuserve login

These commands set, display, or disable the specified connection profile’s CompuServe login enable setting.

cp { name | index } telco compuserve hostname string
show cp { name | index } telco compuserve hostname
no cp { name | index } telco compuserve hostname

cp { name | index } telco compuserve username string
show cp { name | index } telco compuserve username
no cp { name | index } telco compuserve username

cp { name | index } telco compuserve password string
no cp { name | index } telco compuserve password

These commands set, display, or disable the specified connection profile’s CompuServe login host-name, user-name, or password string. For security reasons, there is no show variant of the **cp { name | index } telco compuserve password** command.

IPSec/IKE

Note: The commands in this section are supported beginning with firmware release 4.8.

Connection Profile IPSec Configuration Commands

```
cp { name / index } ipsec suite encryption { des | 3des | null }
    authentication { esp | ah } { md5 | sha1 } [compression { none | lz } ]

cp { name / index } ipsec ip
    [remote {[members {xxx.xxx.xxx.xxx/nn | xxx.xxx.xxx.xxx xxx.xxx.xxx.xxx}] [tep x.x.x.x]}]
    [local tep x.x.x.x] [via x.x.x.x]

cp { name / index } ipsec spi rx-esp-spi [ tx-esp-spi [ rx-ah-spi [ tx-ah-spi ]]]

cp { name / index } ipsec authentication key string

cp { name / index } ipsec encryption key 1234567890123456 [1234567890123456
1234567890123456 ]
```

```
cp { name / index } ipsec suite encryption { des | 3des | null }
    authentication { esp | ah } { md5 | sha1 } [compression { none | lz } ]
```

This command allows you to specify the IPSec suite encryption type and authentication method for an IPSec tunnel.

Note: Beginning with firmware version 4.10, the **3des** option is available for all R-Series routers. For firmware versions earlier than 4.10, the **3des** option is only available if the VPN accelerator module is installed in the router.

```
cp { name / index } ipsec ip
    [remote {[members {xxx.xxx.xxx.xxx/nn | xxx.xxx.xxx.xxx xxx.xxx.xxx.xxx}] [tep x.x.x.x]}]
    [local tep x.x.x.x] [via x.x.x.x]
```

This command sets all the pertinent IP values for the IPSec tunnel. There are three sub-sections of this command, the **remote**, **local**, and **via**. The **remote** section, if it exists, may contain a **members** or a **tep** (“tunnel endpoint”) parameter, or both. The **local** section, if it exists, may contain only a **tep** parameter. The optional **via** section sets the next hop gateway.

```
cp { name / index } ipsec spi rx-esp-spi [ tx-esp-spi [ rx-ah-spi [ tx-ah-spi ]]]
```

This command allows you to specify the security parameters indexes for an IPSec tunnel.

cp { *name* / *index* } **ipsec authentication key** *string*

This command allows you to specify the authentication secret for an IPsec tunnel. You must specify an authentication secret if the authentication type is anything other than None.

Note: The key is a hexadecimal entry of 16 bytes (32 characters of input) for **md5** and 20 bytes (40 characters of input) for **sha1**. It is not possible to retrieve the encryption keys or authentication key once they have been set.

cp { *name* / *index* } **ipsec encryption key** *1234567890123456* [*1234567890123456 1234567890123456*]

This command allows you to specify the authentication key for an IPsec tunnel. You must specify an authentication key if the authentication type is anything other than None. The key must be an ASCII string of up to 48 characters for both **md5** and **sha1**.

Note: For DES the key is one group of 16 hexadecimal characters; for 3DES the key is three groups of 16 hexadecimal characters each.

Beginning with the version 4.9 firmware release, the Command Line Interface supports the following new and modified Connection Profile configuration commands **on the R-Series Router platform only**:

IKE/IPSec Connection Profile Commands

```

cp { name | index } ipsec dead-peer-detection { yes | no }
show cp { name | index } ipsec dead-peer-detection
no cp { name | index } ipsec dead-peer-detection

cp { name | index } ipsec idle-timeout { non-negative-integer | none }
show cp { name | index } ipsec idle-timeout
no cp { name | index } ipsec idle-timeout

cp { name | index } ipsec key-manager { manual | ike }
show cp { name | index } ipsec key-manager

cp { name | index } ipsec ike phase1 { name | index | none }
show cp { name | index } ipsec ike phase1
no cp { name | index } ipsec ike phase1

cp { name | index } ipsec pfs { yes | no }
show cp { name | index } ipsec pfs
no cp { name | index } ipsec pfs

cp { name | index } ipsec suite encapsulation { esp | ah | esp+ah }
    [ encryption { des | 3des | null } ]
    [ authentication esp { md5 | hmac-md5-96 | sha1 | hmac-sha1-96 } ]
    [ authentication ah { md5 | hmac-md5-96 | sha1 | hmac-sha1-96 } ]
    [ compression lzs ]
show cp { name | index } ipsec suite

cp { name | index } ipsec ip
    [remote
        [members {a.b.c.d | a.b.c.d/n | a.b.c.d e.f.g.h | a.b.c.d-e.f.g.h}]
        [tep a.b.c.d] ]
    [local
        [members {a.b.c.d | a.b.c.d/n | a.b.c.d e.f.g.h | a.b.c.d-e.f.g.h}]
        [tep a.b.c.d] ]
    [via a.b.c.d]
show cp { name | index } ipsec ip

cp { name | index } ipsec sa lifetime { seconds | kbytes } { non-negative-integer | none }
show cp { name | index } ipsec sa lifetime [ { seconds | kbytes } ]
no cp { name | index } ipsec sa lifetime [ { seconds | kbytes } ]

```

```

cp { name | index } ipsec dead-peer-detection { yes | no }
show cp { name | index } ipsec dead-peer-detection
no cp { name | index } ipsec dead-peer-detection

```

Note: These commands are supported beginning with firmware versions 4.10 and 5.2.

These commands set, display, or disable the status of dead peer detection for the specified IPsec Phase 2 profile. Dead peer detection counts the outbound packets on a tunnel. If 256 packets go out without a single packet coming in, the tunnel SAs are expired and a rekey is started. Rekeying is first attempted on the previous Phase 1 SA. If the Phase 1 request times out, then the Phase 1 SA is expired and Phase 1 rekeying is begun over again.

```

cp { name | index } ipsec idle-timeout { non-negative-integer | none }
show cp { name | index } ipsec idle-timeout
no cp { name | index } ipsec idle-timeout

```

These commands set or display the idle timeout associated with the specified IPSec connection profile. If the IPSec **key-manager** associated with the connection profile is **manual**, then the idle-timeout value is meaningful only if the **remote sg** is 0.0.0.0 or the empty string. In that case, the idle-timeout value specifies the period in seconds during which the SPI (or SPIs) are bound to a particular remote peer in the absence of outbound traffic through the IPSec tunnel. The value zero (or the keyword **none**) causes the SPI (or SPIs) to be permanently bound to the first remote peer that sends traffic through the tunnel using the SPI (or SPIs). If the IPSec **key-manager** associated with the connection profile is **ike**, then the idle-timeout value specifies the period prior to SA expiration during which there must be at least one outbound packet through the IPSec tunnel for a re-key to be performed one second prior to SA expiration. The value zero (or the keyword **none**) indicates that a re-key should always be performed one second prior to SA expiration even if there has been no outbound traffic through the tunnel.

```

cp { name | index } ipsec key-manager { manual | ike }
show cp { name | index } ipsec key-manager

```

These commands set or display the IPSec key manager associated with the specified connection profile.

```

cp { name | index } ipsec ike phase1 { name | index | none }
show cp { name | index } ipsec ike phase1
no cp { name | index } ipsec ike phase1

```

These commands set, display, or disable the IKE Phase1 profile associated with the specified connection profile. The IKE Phase1 profile may be specified either by index or by name.

```

cp { name | index } ipsec pfs { yes | no }
show cp { name | index } ipsec pfs
no cp { name | index } ipsec pfs

```

Note: These commands are supported beginning with firmware versions 4.10 and 5.2.

These commands set, display, or change the Phase 2 perfect forward secrecy setting for the specified IPsec Phase 2 profile.

```

cp { name | index } ipsec suite encapsulation { esp | ah | esp+ah }
  [ encryption { des | 3des | null } ]
  [ authentication esp { md5 | hmac-md5-96 | sha1 | hmac-sha1-96 } ]
  [ authentication ah { md5 | hmac-md5-96 | sha1 | hmac-sha1-96 } ]
  [ compression lzs ]
show cp { name | index } ipsec suite

```

Note: This is an extended version of an existing CLI command. The existing command is modified to add an encapsulation clause and to allow for one or two authentication clauses. See [“IPSec/IKE” on page 2-59](#) for more information.

These commands set or display the IPSec encapsulation, encryption, authentication, and compression parameters for the specified connection profile.

Note: The authentication clause may appear either one or two times; if it appears twice, one occurrence must specify ah and the other must specify esp.

The keywords **md5** and **hmac-md5-96** are synonyms, although the latter keyword is preferred, the former being retained only for backwards compatibility. The keywords **sha1** and **hmac-sha1-96** are synonyms, although the latter keyword is preferred, the former being retained only for backwards compatibility.

```

cp { name | index } ipsec ip
  [remote
    [members {a.b.c.d | a.b.c.d/n | a.b.c.d e.f.g.h | a.b.c.d-e.f.g.h}]
    [tep a.b.c.d] ]
  [local
    [members {a.b.c.d | a.b.c.d/n | a.b.c.d e.f.g.h | a.b.c.d-e.f.g.h}]
    [tep a.b.c.d] ]
  [via a.b.c.d]
show cp { name | index } ipsec ip

```

Note: This is an extended version of an existing CLI command. The existing command is modified to allow a members specification to appear in the local clause and to allow for a host address or an IP address range (rather than a network address and subnet mask) in the remote and local members clauses. See [“IPSec/IKE” on page 2-59](#) for more information.

This command sets the pertinent IP values for the IPSec tunnel, and may contain zero or one instances of each of three possible clauses: **remote**, **local**, and **via**. The **remote** clause, if specified, may include a members specification or a tunnel endpoint (“tep”) specification, or both. The **local** clause, if specified, may contain a members specification or a tunnel endpoint specification, or both. The optional **via** clause sets the next hop gateway. The keyword **sg** (short for “security-gateway”) is an acceptable synonym for the keyword **tep**.

```
cp { name | index } ipsec sa lifetime { seconds | kbytes } { non-negative-integer | none }  
show cp { name | index } ipsec sa lifetime [ { seconds | kbytes } ]  
no cp { name | index } ipsec sa lifetime [ { seconds | kbytes } ]
```

These commands set, display, or disable one or both of the two IKE Phase 2 SA lifetimes (in seconds and/or kbytes protected) for the specified IPSec protocol for the specified connection profile. Specifying neither the keyword **seconds** nor the keyword **kbytes** with the show variant of this command displays both lifetime values for the specified protocol. Specifying neither the protocol nor the keyword **seconds** or **kbytes** displays all four possible lifetime values. The keyword **none** is equivalent to the value zero, and indicates that there is no lifetime of the specified type.

Note: It is a run-time checked error if both of the IKE Phase 2 SA lifetime values for a particular protocol are set to zero or **none**.

Frame Relay Configuration Commands

Frame Relay Configuration Commands

```

frame-relay dlci number [ tag tag ] [ ip-addr { 0.0.0.0 | remote-ip-addr } ]
    [ cir { default | 1-accessrate } ] [ bc { default | 1-accessrate } ] [ be { default | 0-accessrate } ]
    [ {disable | enable} ]
no frame-relay dlci number

frame-relay lmi type { none | annexa | annexd | ansi | ccitt | lmi }
no frame-relay lmi type
show frame-relay lmi type

frame-relay tim { none | standard | buffered }

show frame-relay lmi statistics

show frame-relay pvc

```

```

frame-relay dlci number [ tag tag ] [ ip-addr { 0.0.0.0 | remote-ip-addr } ]
    [ cir { default | 1-accessrate } ] [ bc { default | 1-accessrate } ] [ be { default | 0-accessrate } ]
    [ {disable | enable} ]

```

This command allows you to manually configure a DLCI. If the IP address is 0.0.0.0 the router will attempt to auto-discover the remote IP address. The router supplies a default DLCI 16 with a 0.0.0.0 address, so in many cases manually configuring a DLCI is unnecessary.

Examples:

To add a DLCI using all of the default values, type:

```
frame-relay dlci 17
```

The tag (i.e., name) of the DLCI defaults, in this case, to "DLCI 17". To specify a different name, type:

```
frame-relay dlci 17 tag "My DLCI"
```

For descriptions of the other parameters available for configuration (and their default values) see the *Netopia Router User's Reference Guide*.

no frame-relay dlci *number*

This command deletes the DLCI identified by *number*.

```
frame-relay lmi type { none | annexa | annexd | ansi | ccitt | lmi }  
no frame-relay lmi type  
show frame-relay lmi type
```

These commands allow you to change or display the Frame Relay Local Management Interface (LMI) type. The keywords **ccitt** and **annexa** are synonyms, as are the keywords **ansi** and **annexd**.

```
frame-relay tim { none | standard | buffered }
```

This command allows you to set the routers's Frame Relay transmit injection management **tim**.

```
show frame-relay lmi statistics
```

This command displays Frame Relay Local Management Interface (LMI) statistics.

Example:

```
#show frame-relay lmi statistics  
LMI Tx Status Enquiries:      10  
LMI Rx Status Responses:     10  
LMI Timeouts:                 0  
LMI Errors:                   0  
LMI Failures:                 0
```

```
show frame-relay pvc
```

This command displays the status of each Frame Relay permanent virtual circuit (pvc).

Example:

```
#show frame-relay pvc  
DLCI:    16, status: active  
DLCI:    17, status: active
```

Miscellaneous Commands

Miscellaneous Commands

```
clear
ping [ ip ] { ip-addr | hostname } [count count] [timeout milliseconds] [delay milliseconds]
[size bytes] [source ip-addr]
ping oam interface sds1 id pvc {id | tag | vpi/vci } [count count] [timeout seconds]
receive tftp config [server-name file-name]
receive tftp [ wan { 1 | 2 } ] firmware [server-name [file-name]]
send tftp config [server-name file-name]
show tftp last error
show tftp status

receive xmodem [ wan { 1 | 2 } ] firmware
show xmodem status

reset [factory]
show history
show model
show system information
[show] version [ cli ] [ firmware ] [ hardware ] [ mib ] [ wan 1 ] [ wan 2 ]

upgrade key-value
```

clear

This command erases the screen.

```
ping [ ip ] { ip-addr | hostname } [count count] [timeout milliseconds] [delay milliseconds] [size bytes]
[source ip-addr]
```

The **ping** command allows you to send ICMP echo requests to another network device. You can specify the destination using either an IP address in dotted-quad notation or a hostname. The default **count** is 5, the default **timeout** and **delay** are 1000 milliseconds (1 second) each, and the default **size** is 100 bytes. The **size** value you specify includes the size of the IP and ICMP packet headers. The minimum **size** value is 28 and the maximum **size** value is 1664. **source** specifies which router interface IP address *ip-addr* to use as the source IP address. If the ping goes out through an interface that has NAT (Network Address Translation) enabled, then the source address will be translated.

For each timely ICMP echo response received, an exclamation point ("!") is displayed; for each timeout, a period (".") is displayed; and for each ICMP destination unreachable received, an uppercase letter U ("U") is displayed. When the ping operation completes, statistics are displayed including the total number of ICMP echo requests sent and ICMP echo responses received, the success rate as a percentage, and the minimum, average, and maximum round trip times.

To abort a ping operation while it is in progress, type Control-C.

Example:

```
#ping www.netopia.com
Translating "www.netopia.com"... (163.176.4.31) [OK]
Type Control-C to abort.
Sending 5, 100-byte ICMP Echos to 163.176.4.32, timeout is 1 second:
!!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 0/4/10 ms
#
```

ping oam interface sdsl *id pvc {id | tag | vpi/vci }* [**count** *count*] [**timeout** *seconds*]

Note: This command is supported beginning with firmware version 5.2. This is an extension of the existing **ping** command. If neither **ip** nor **oam** follows the **ping** token, then an ICMP ping is assumed (See previous command.)

This command does not support the **delay** parameter. The lesser of the response time or the timeout value will be the delay.

This command initiates a process by which up to *n* (*count*) OAM loopback tests are performed at *n* (*seconds*) delay between each.

For each individual OAM loopback test failure, a period '.' will be output. For each individual OAM loopback test success, an exclamation mark '!' will be output. A running total of successes and failures will be maintained for the duration of the test. The success-to-failure rate is output at the conclusion. If any OAM loopback test fails for a reason other than a timeout, the command will be aborted immediately.

Example:

```
ping oam interface sdsl 1 pvc count 17

Type Control-C to abort.

Performing 17 OAM loopback tests; timeout is 5 seconds.

.....

Success rate is 0 percent (0/17)
```

The default count is 5, and the default timeout is 0.

receive tftp config [*server-name file-name*]

This command allows you to initiate a configuration file upload from the command line interface. If the TFTP server name and config file name were set previously, either by a previous invocation of this command or via the menu console, then you do not need to supply the server and file name. If the upload is successful, the router will automatically reboot. If the upload is unsuccessful, in menu mode you can go to:



for information about what went wrong.

receive tftp [wan { 1 | 2 }] **firmware** [server-name [file-name]]

This command allows you to receive a firmware file that is located remotely on a TFTP server into the router by means of a TFTP file transfer.

Beginning with firmware version 4.10 for R-Series equipment, the **wan 1** and **wan 2** options allow you to specify that the firmware upgrade is to be sent one of the WAN interface modules. As with the regular **receive tftp firmware** comand, *server-name* and *file-name* are optional if they are already specified. If *server-name* or *file-name* are not already specified, it is necessary to specify them. (Specifying them with this command will set them.)

send tftp config [server-name file-name]

The configuration file will be stored on the TFTP server as a binary image. Note that the file must already exist on the server and be writable.

show tftp last error

An empty string indicates no error. Other possible return values are: File not found, Unauthorized, Disk Full, Illegal Operation, Bad TID, File Already Exists, No Such User, and Unknown Error.

show tftp status

The possible return values are: Idle, Reading Firmware, Reading Wanlet Firmware, Reading Config, Writing Config, Idle, ****TRANSFER FAILED****.

receive xmodem [wan { 1 | 2 }] **firmware**

This command allows you to receive a firmware file that is located locally into the router by means of an XMODEM file transfer.

Beginning with firmware version 4.10 for R-Series equipment, the **wan 1** and **wan 2** options allow you to specify that the firmware upgrade is to be sent one of the WAN interface modules.

The underlying routine that is invoked blocks until either the file has been successfully received or some condition caused the transfer to abort. Therefore when this command is invoked the following will appear:

```
#receive xmodem firmware
```

```
This will take a long time, and the console will be inactive during this time.
If the transfer is successful, the router will reboot automatically.
```

```
Press Ctl-C to abort, or...
```

```
Press Return/Enter to continue; you will have 10 seconds to start the transfer.
```

Once you press Return the console is inactive until the XMODEM routine returns. If the transfer is unsuccessful, the following message displays:

```
XModem transfer Failed!  
#
```

show xmodem status

Note: These commands are supported beginning with firmware versions 4.10 and 5.2.

This command displays the status of an xmodem firmware file transfer. Possible values are:

- Reading Wanlet Firmware
- Sending firmware to wanlet
- Sending block x of y to wanlet
- Validating Firmware
- Decompressing Firmware
- Checking CRC
- Updating Firmware: x% complete
- ****Upgrade SUCCESSFUL****
- Insufficient Memory
- ****Upgrade FAILED****

reset [factory]

This command allows you to reset the router from the command line interface. If the optional keyword **factory** is specified, the parameters of the router are reset to their initial, factory default values. This command must be completely typed out; it may not be abbreviated.

WARNING: In many cases a factory reset will cause you to be unable to communicate with the router over the network, since the default IP address of the router will revert to 192.168.1.1.

show history

This command displays the command history buffer, which contains a record of the commands that have been entered on the console. The list is limited to the most recent ten commands entered. The oldest command appears first, and the most recent command appears last.

show model

This command displays the device's model number.

Example:

```
#show model  
Netopia R7200 Router
```

show system information

Note: These commands are supported beginning with firmware versions 4.10 and 5.2.

This command shows the same information as is shown in the System Information screen in the console menu user interface:

- Serial Number
- Firmware Version
- Processor Speed (MHz)
- Flash ROM Capacity (MBytes)
- DRAM Capacity (MBytes)
- Ethernet
- WAN Interface

show version [cli] [firmware] [hardware] [mib] [wan 1] [wan 2]

Note: These commands are supported beginning with firmware versions 4.10 and 5.2.

This command allows you to display some or all of the router's version strings. If you don't specify anything after the keyword **version**, the router displays a list of all of the version strings; otherwise it displays the specified version string.

Beginning with firmware version 4.10, the R-Series WAN interface module firmware version is displayed for the specified WAN module, **wan 1** or **wan 2**. **wan1** is an acceptable substitute for **wan 1**; **wan2** is an acceptable substitute for **wan 2**.

The CLI, hardware, and product-specific SNMP MIB version strings consist of a major version, minor version, release stage, and revision, displayed in the (decimal) form "MM.mmsrr", where:

'MM' is the major version,

'mm' is the minor version,

's' is the release stage -- e (experimental), d (development), a (alpha), b (beta), or f (final),

'rr' is the revision.

The firmware version string consists of a major version, minor version, point release version, release stage, and revision, displayed in the (decimal) form "MM.mm.ppsrr", where:

'MM' is the major version,

'mm' is the minor version,

'pp' is the point release version,

's' is the release stage -- d (development), a (alpha), b (beta), fc (final candidate), or f (final),

'rr' is the revision.

Examples:

```
#show version
cli      version:  01.00d00
firmware version:  04.10.00
hardware version:  01.00f00
mib      version:  01.00f00
wan1     version:  fw v1.0.6
wan1     version:  V2.210-N-V90_2M_DLS
#
```

upgrade *key-value*

If the upgrade operation is successful the router will reboot. Otherwise, an error will be returned. The *key-value* argument is the upgrade key that was received when the upgrade was purchased.

IP Network Address Translation (NAT) Commands

Note: The commands in this section are supported beginning with firmware release 4.4, with the exception of the **ip nat public tag dynamic** command which is supported beginning with firmware release 4.5.

IP Network Address Translation (NAT) Commands
<pre>ip nat public tag dynamic from-address to-address ip nat public tag static pub-from-address pub-to-address ip nat public tag pat { pub-address 0.0.0.0 } [from-port to-port] no ip nat public [tag] ip nat map list-tag priv-from-address priv-to-address pool-tag no ip nat map [list-tag] show ip nat map list-tag ip nat server list-tag priv-ip-addr pub-ip-addr { port-name port [end-port] } no ip nat server [list-tag] show ip nat server list-tag show ip nat translations</pre>

Overview

Network Address Translation (NAT) makes use of five basic structures:

- “public” or externally visible address ranges
- “map” rules that bind an interior, private address range with a public address range
- “map-lists”, which are ordered lists of maps
- “servers”, which define a mapping between a private IP address and port and a public IP address (and the same port)
- “server-lists” which are lists of servers.

ip nat public tag dynamic from-address to-address

This command allows you to allocate a dynamic range of exterior, public addresses for use by Network Address Translation. This range of addresses will be associated dynamically with private addresses you will define when you create a map. *tag* is the name you assign to the range, and it can contain up to 16 characters.

Example:

The following creates a dynamic public range of 8 addresses starting at 163.176.12.1:

```
ip nat public "my second range" dynamic 163.176.12.1 163.176.12.8
```

ip nat public *tag* static *pub-from-address* *pub-to-address*

This command allows you to allocate a range of exterior, public addresses for use by Network Address Translation. This range of addresses will be associated one-to-one with private addresses you will define when you create a map, described later. *tag* is the name you assign to the range, and it can contain up to 16 characters. The address range is defined to be from the first address to the last address, inclusive.

Example:

The following creates a static public range of 8 addresses starting at 163.176.12.1:

```
ip nat public "my first range" static 163.176.12.1 163.176.12.8
```

ip nat public *tag* pat { *pub-address* | 0.0.0.0 } [*from-port* *to-port*]

This command allows you to configure a PAT public range. Since PAT allows you to map multiple private addresses to a single public address, you specify only the public address. Optionally you can specify a range of ports to be used by PAT. The lowest allowed port is 1025, and the highest allowed port is 65535. If you do not specify the port range explicitly the default range is 49152 to 65535 inclusive.

If you specify 0.0.0.0 as the public address, whatever address is negotiated by PPP or DHCP when the WAN connection is established, this pool will adopt. Since neither PPP nor DHCP are capable of assigning more than one address to a single client you should have at most one (active) PAT public range whose public address is 0.0.0.0.

Example:

The following command creates a PAT public range at 163.176.12.1, with a port range from 10000 to 65535:

```
ip nat public "my pat range" pat 163.176.12.1 10000 65535
```

no ip nat public [*tag*]

This command allows you to delete a public range whose name is *tag*. Since tags must be unique this command works for both static and PAT public ranges. If the *tag* is omitted then all public ranges will be deleted.

Note: Use of this command can have significant side effects, because any map in any map list that refers to a public range that is deleted will also be deleted.

ip nat map *list-tag* *priv-from-address* *priv-to-address* *pub-range-tag*

This command allows you to map a range of private addresses (from *priv-from-address* to *priv-to-address* inclusive) to the public address range named *pub-range-tag*. This map is appended to the list of maps named *list-tag*. If a map list of that name doesn't exist it is created.

no ip nat map [*list-tag*]

This command allows you to delete the map list named *list-tag*. This also deletes all of the maps contained in the list. No public range referred to by any of the contained maps is deleted. If a connection profile has been bound to this map list it will be updated to reflect the fact that the list no longer exists, and will thus be bound to no map list.

If the *list-tag* is omitted then all map lists will be deleted.

show ip nat map *list-tag*

This command allows you to see the rules contained in the nat map list named *list-tag*.

Example:

```
show ip nat map Easy-PAT
```

ip nat server *list-tag* *priv-ip-addr* *pub-ip-addr* { *port-name* | *port* [*end-port*] }

This command allows you to map a private server address, *priv-ip-addr*, and port or ports to a public address *pub-ip-addr* and the same port or ports. In the Netopia router's earlier firmware releases this feature was called "Exported Services." Its primary use is to allow servers to be accessed through a WAN interface to which PAT has been applied.

portname currently may be one of **ftp**, **telnet**, **smtp**, **tftp**, **gopher**, **finger**, **www-http**, **pop2**, **pop3**, **snmp**, **timbuktu**, **pptp**, or **irc**.

Examples:

The two commands below would append mappings from 207.3.3.3 to 192.168.1.2 for the ftp ports (ports 20 and 21) and ports 7249 through 7253 to the server list named "my servers".

```
ip nat server "my servers" 192.168.1.2 207.3.3.3 ftp
```

```
ip nat server "my servers" 192.168.1.2 207.3.3.3 7249 7253
```

no ip nat server [*list-tag*]

This command allows you delete a list of NAT server maps. If the *list-tag* is omitted then all server lists will be deleted.

show ip nat server *list-tag*

This command allows you to see the rules contained in the nat server list named *list-tag*.

Example:

```
show ip nat server Easy-Servers
```

show ip nat translations

Note: This command is supported beginning with firmware versions 4.10 and 5.2.

This command displays the current sessions passing through network address translation.

Example:

```
#show ip nat translations
LAN IP address--Port--WAN IP address--Port--Rem IP Address--Port--Dir-Prot----
192.168.1.100:1550      163.176.56.86:49163    216.115.102.80:80      out TCP
163.176.56.255:137     163.176.56.255:137    163.176.56.41:137     in  UDP
163.176.56.255:138     163.176.56.255:138    163.176.56.41:138     in  UDP
192.168.1.100:1556     163.176.56.86:49165    216.115.107.144:80     out TCP
192.168.1.100:1552     163.176.56.86:49164    216.115.102.46:80      out TCP
192.168.1.100:1564     163.176.56.86:49166    216.115.110.45:80      out TCP
192.168.1.100:1566     163.176.56.86:49167    216.115.102.46:80      out TCP
163.176.56.255:138     163.176.56.255:138    163.176.56.231:138     in  UDP
Total entries in NAT cache: 8
```

AppleTalk Configuration Commands

AppleTalk Configuration Commands
macip static-range { none <i>from-addr to-addr</i> }
no macip static-range
show macip static-range

macip static-range { none | *from-addr to-addr* }**no macip static-range****show macip static-range**

These commands allow you to set, clear, or display the IP address pool reserved for MacIP static clients. The specified address pool must be a subset of one of the IP address pools configured for Ethernet address-serving.

Backup Configuration Commands

Note: The commands in this section are supported beginning with firmware release 4.8.

Backup Configuration Commands
<div>backup enable { no manual automatic yes } no backup show backup status</div> <div>backup [no]</div> <div>backup delay 1..65535</div> <div>backup ping host { ip-address host-name }</div> <div>backup recovery delay 1..65535</div> <div>backup recovery idle delay 1..65535</div> <div>backup recovery idle only { yes no }</div> <div>backup recovery layer-2-loss { yes no }</div> <div>backup recovery mode { manual automatic }</div>

backup enable { no | manual | automatic | yes }
no backup
show backup status

These commands allow you to set, disable, or display the status of the dial backup feature.

Note: yes = automatic

backup delay 1..65535

This command allows you to set the number of minutes before the router invokes the dial backup feature in the event of loss of connectivity. This allows you to determine how long you want the system to wait before the backup port becomes enabled in the event of primary line failure, ensuring that the primary WAN connection is not merely briefly interrupted before the router switches to backup mode.

backup [no]

This command allows you directly to invoke or cancel the backup mode.

backup ping host { *ip-address* | *host-name* }

This command allows you to enter an IP address or resolvable DNS name that the router will ping. This is an optional item that is particularly useful for testing if the remote end of a VPN connection has gone down. Should this address become unreachable the router will treat this as a loss of connectivity and begin the backup timer. This loss is a Layer 2 loss.

backup recovery delay 1..65535

This command allows you to set the number of minutes before the router attempts to recover back to the primary WAN connection. This allows you to determine how long you want the system to wait before the primary WAN port becomes enabled after connectivity is restored, ensuring that the backup connection is not merely briefly interrupted before the router switches to primary mode.

backup recovery idle delay 1..65535

This command allows you to set the number of seconds for the backup link to be idle, i.e. passing no traffic, before the router attempts to recover back to the primary WAN connection.

backup recovery idle only { **yes** | **no** }

This command allows you to toggle the idle delay on or off.

backup recovery layer-2-loss { **yes** | **no** }

This command allows you to specify whether the router should try to auto-recover when the backup is invoked because of a layer 2 loss, for example, no valid Connection Profile. (Layer 1 is still available, and this is what recovery checks.) Use this setting with caution. Setting it to **yes** may induce alternating switching between backup and recovery mode. This setting will determine the recovery behavior of a manual backup and ping failure backup. These two failures are treated as layer 2 failures.

backup recovery mode { **manual** | **automatic** }

This command allows you to specify the backup recovery mode to be either manually or automatically invoked.

RADIUS Authentication Configuration Commands

Note: The commands in this section are supported beginning with firmware version 4.8.

RADIUS Authentication Configuration Commands
<pre>console authentication { local radius radius-local [serial-only] local-radius } show console authentication radius-server { 1 2 } { ip-address hostname } [secret secret] no radius-server { 1 2 } show radius-server [1 2] radius identifier identifier</pre>

```
console authentication { local | radius | radius-local [ serial-only ] | local-radius }
show console authentication
```

These commands allow you to set or show how the router will authenticate users seeking console configuration access using a remote authentication database maintained by a RADIUS server. It supports four security database modes: **local**, **radius**, **radius-local**, and **local-radius**.

Specifying **local** selects the pre-4.8 authentication mechanism.

Specifying **radius** causes the router to ignore the local database and to authenticate users using the configured RADIUS server.

Specifying **radius-local** causes the router to attempt to authenticate a user first using a RADIUS server and then, if that fails, using the local authentication database.

Beginning with firmware version 4.9.1, specifying **radius-local serial-only** causes the router to attempt to authenticate a user first using the configured RADIUS server(s) and then, if that fails and the user is accessing the router via the built-in serial console port, using the local authentication database. If the user is accessing the router via telnet or asynchronous dial-in (via a modem on the AUX port), and RADIUS authentication fails, the local authentication database is not consulted and the user is refused access to the router.

Note: **radius-local serial-only** represents a modification of the existing command. Firmware versions earlier than 4.9.1 accepted only the **local**, **radius**, **radius-local**, and **local-radius** modes.

Specifying **local-radius** causes the router to attempt to authenticate a user first using the local authentication database, and then, if that fails using the configured RADIUS server.

In those modes that involve both RADIUS and the local database, if the local database includes no user-name/password pairs, authentication will succeed only if the RADIUS server authenticates the user. This differs from the **local** mode where no authentication is performed when the local database is empty.

```
radius-server { 1 | 2 } { ip-address | hostname } [ secret secret ]  
no radius-server { 1 | 2 }  
show radius-server [ 1 | 2 ]
```

These commands allow you to specify, delete, or show a RADIUS **server** either by using an IP address in dotted-quad notation or by using a hostname to be resolved using the Domain Name System (DNS) information configured in the router. In addition to specifying the server's IP address or hostname, you must also specify a shared-secret known to both the router and the RADIUS server. The **secret** is used to encrypt RADIUS transactions in transit.

radius identifier *identifier*

This command allows you to specify the RADIUS **identifier** as either an IP address in dotted-quad notation (to be used as the value of the NAS-IP-Address (4) attribute), or an arbitrary string (to be used as the value of the NAS-Identifier (32) attribute), in the router's outgoing Access-Request packets. The RADIUS **identifier** is limited to 63 characters.

IP Filterset Configuration Commands

Note: The commands in this section are supported beginning with firmware version 4.8.

IP Filterset Configuration Commands
<pre>ip filterset <i>fs-tag</i> {in out} [filter] <i>filter-id</i> [enable {yes no}] [forward {yes no}] [source { <i>ip-addr/mask-bits</i> <i>ip-addr mask</i> }] [destination { <i>ip-addr/mask-bits</i> <i>ip-addr mask</i> }] [protocol { 1..65535 any gre { {tcp 6 } [{source port-compare}] [{destination port-compare}] [established all] } { {udp 17 } [{source port-compare}] [{destination port-compare}]} { {icmp 1 } [{type port-compare}] [{code port-compare}]} }] no ip filterset <i>fs-id</i> [{in out} [<i>filter-id</i>]] show ip filterset <i>fs-id</i> [{in out} [<i>filter-id</i>]]</pre>

The CLI for filters is fairly complex. More explanation follows the command itself.

compare-op = { **nc** | **ne** | <> | **lt** | < | **le** | <= | **eq** | = | **ge** | >= | **gt** | > }

port-compare = { **nc** | { *compare-op digits* } }

filter-id = { 1..255 | **new** | **last** }

```
ip filterset fs-tag {in | out} [filter] filter-id [enable {yes | no}] [forward {yes | no}]
[source { ip-addr/mask-bits | ip-addr mask }]
[destination { ip-addr/mask-bits | ip-addr mask }]
[ protocol { 1..65535 |
    any |
    gre |
    { {tcp | 6 } [{source port-compare}] [{destination port-compare}]
      [established | all] } |
    { {udp | 17 } [{source port-compare}] [{destination port-compare}]} |
    { {icmp | 1 } [{type port-compare}] [{code port-compare}]} } ]
no ip filterset fs-id [{in | out} [filter-id] ]
show ip filterset fs-id [{in | out} [filter-id] ]
```

set

A Filter set, as with NAT Server and Rule Lists, is instantiated by creating its first contained object. This first filter can be identified by its ones-based index, 1, or with the special **new** keyword. Subsequent modifications to this filter, assuming no more filters have been added to the filter set yet, must be done by referring to the filter either by id (1), or by the other special keyword **last**. Subsequent filters can be added using either **new** or by the next integer filter id. You can always specify the last filter in the set by using **last**. It is an error to attempt to create a new filter whose id is not 1 greater than the id of the last filter.

Using **new** and **last** allow you to create filter sets without using filter indices.

show

You can show the contents of all filter sets by typing:

```
show ip filterset
```

Or you can show the contents of a filter set by typing (for example):

```
show ip filterset "My Filters"
```

Or all of the input or output filters of a filter set by adding the {in | out} keyword:

```
show ip filterset "My Filters" in
```

Or a particular filter by specifying {in | out} and the tag:

```
show ip filterset "My Filters" in 3
```

Since the command line console is currently constrained to 78 characters per line, the show command breaks each filter up into four separate lines, for example:

```
show ip filterset "Basic Firewall" in 2
ip filterset "Basic Firewall" in 2 enable yes forward no
ip filterset "Basic Firewall" in 2 source 0.0.0.0/0
ip filterset "Basic Firewall" in 2 destination 0.0.0.0/0
ip filterset "Basic Firewall" in 2 protocol tcp source nc destination
eq 6000 any
```

Note: Some commands, when dumping existing canned filters, exceed 78 characters and will wrap. To work around this limitation use truncated keywords.

no

Syntax corresponds to the syntax for **show**.

Hardware Acceleration Configuration Commands

Hardware Acceleration Configuration Commands
hardware acceleration enable { yes no } no hardware acceleration enable show hardware acceleration enable

hardware acceleration enable { yes | no }
no hardware acceleration enable
show hardware acceleration enable

These commands allow you to enable, disable, or show the status of hardware acceleration if the XL acceleration/encryption daughtercard is installed in the router.

In the unlikely event of a hardware acceleration card failure, the **no hardware acceleration enable** command allows you to turn off hardware acceleration. This will disable IPcomp compression.

Global IPSec/IKE Configuration Commands

Beginning with version 4.9 firmware, the Command Line Interface supports the following global IPSec configuration commands for creating, displaying, modifying, and deleting Internet Key Exchange (IKE) Phase1 Profiles **on the R-Series Router platform only**:

IKE Configuration Commands

```

ike phase1 { name | index } [ { yes | no } ]
no ike phase1 { name | index }
show ike phase1 { name | index }

show ike phase1 { name | index } id

ike phase1 { name | index } tag string
show ike phase1 { name | index } tag

ike phase1 { name | index } mode { main | aggressive }
show ike phase1 { name | index } mode

ike phase1 { name | index } identity { remote | local } { ipv4-address | ipv4-subnet | ipv4-range |
    hostname | e-mail-address | ascii-key-id | hex-key-id } string
show ike phase1 { name | index } identity [ { remote | local } ]

ike phase1 { name | index } authentication method { shared-secret }
show ike phase1 { name | index } authentication method

ike phase1 { name | index } authentication shared-secret { ascii | hexadecimal } string

ike phase1 { name | index } dangling-sas { yes | no }
show ike phase1 { name | index } dangling-sas
no ike phase1 { name | index } dangling-sas

ike phase1 { name | index } encryption { des | 3des }
show ike phase1 { name | index } encryption

ike phase1 { name | index } group { 1 | 2 | 5 | dh-768-bits | dh-1024-bits | dh-1536-bits }
show ike phase1 { name | index } group

ike phase1 { name | index } hash { sha1 | md5 }
show ike phase1 { name | index } hash

```

Global IPSec Configuration Commands
<pre>ike phase1 { name index } independent rekeys { yes no } show ike phase1 { name index } independent rekeys no ike phase1 { name index } independent rekeys ike phase1 { name index } initial-contact { yes no } show ike phase1 { name index } initial-contact no ike phase1 { name index } initial-contact ike phase1 { name index } negotiation { normal initiate-only respond-only } show ike phase1 { name index } negotiation ike phase1 { name index } pfs { yes no } show ike phase1 { name index } pfs no ike phase1 { name index } pfs ike phase1 { name index } port policy { "strict" "permissive" } show ike phase1 { name index } port policy ike phase1 { name index } sa lifetime { seconds kbytes } { non-negative-integer none } show ike phase1 { name index } sa lifetime [{ seconds kbytes }] no ike phase1 { name index } sa lifetime [{ seconds kbytes }] ike phase1 { name index } sa use-policy { new-sas-immediately old-sas-until-expired } show ike phase1 { name index } sa use-policy ike phase1 { name index } vendor-id { yes no } show ike phase1 { name index } vendor-id no ike phase1 { name index } vendor-id</pre>

```
ike phase1 { name | index } [ { yes | no } ]
no ike phase1 { name | index }
show ike phase1 { name | index }
```

These commands create, delete, or show the specified IKE Phase1 profile, which may be identified by index or by name. The show version of this command displays the value **yes** if the specified IKE Phase 1 Profile exists, and **no** otherwise.

```
show ike phase1 { name | index } id
```

This command displays the index of the specified IKE Phase1 profile. This command is useful only when referring to a profile by name.

```
ike phase1 { name | index } tag string
show ike phase1 { name | index } tag
```

These commands name or display the specified IKE Phase1 profile.

```
ike phase1 { name | index } mode { main | aggressive }
show ike phase1 { name | index } mode
```

These commands set or display whether the specified IKE Phase1 profile uses main mode or aggressive mode.

```
ike phase1 { name | index } identity { remote | local } { ipv4-address | ipv4-subnet | ipv4-range | hostname |
e-mail-address | ascii-key-id | hex-key-id } string
show ike phase1 { name | index } identity [ { remote | local } ]
```

These commands set or display the specified IKE Phase1 profile's local or remote identity type and value.

The **identity** type specifies the type of Identity value to be used. Possible types are: **ipv4-address**, **ipv4-subnet**, **ipv4-range**, **hostname**, **e-mail-address**, **ascii-key-id**, and **hex-key-id**. The identity value specifies a value of the specified type as follows:

Identity Type	Format of Identity Value
IPv4 Address	A single IPv4 address in the familiar dotted-quad notation (a.b.c.d)
IPv4 Subnet	A single IPv4 network address in the familiar dotted-quad notation (a.b.c.d) followed by a mask specified EITHER by a slash and a bit-count between 0 and 32 OR by a second dotted-quad.
IPv4 Range	Two IPv4 addresses in the familiar dotted quad notation (a.b.c.d) separated by a space.
Host Name	A fully-qualified domain name (FQDN)
E-Mail Address	An RFC 822 e-mail address in the form user@hostname
Key ID (ASCII)	An opaque string consisting of printable ASCII characters represented as a sequence of printable ASCII characters
Key ID (HEX)	An opaque string consisting of arbitrary 8-bit ASCII values represented as a sequence of HEXADECIMAL digits, each of which corresponds to one nibble of the string value

```
ike phase1 { name | index } authentication method { shared-secret }
show ike phase1 { name | index } authentication method
```

These commands set or display the specified IKE Phase1 profile's authentication method. Currently, the only supported method is shared-secret; others may be added in the future.

```
ike phase1 { name | index } authentication shared-secret { ascii | hexadecimal } string
```

This command sets the specified IKE Phase1 profile's shared secret. For security reasons no **show** variant of this command exists.

```
ike phase1 { name | index } dangling-sas { yes | no }
show ike phase1 { name | index } dangling-sas
no ike phase1 { name | index } dangling-sas
```

These commands set, display, or disable whether or not Phase 2 SAs may persist after the underlying Phase 1 SAs have expired.

```
ike phase1 { name | index } encryption { des | 3des }
show ike phase1 { name | index } encryption
```

These commands set or display the specified IKE Phase1 profile's encryption algorithm.

```
ike phase1 { name | index } group { 1 | 2 | 5 | dh-768-bits | dh-1024-bits | dh-1536-bits }
show ike phase1 { name | index } group
```

These commands set or display the specified IKE Phase1 profile's Diffie-Hellman group.

Note: **1** and **dh-768-bits**, **2** and **dh-1024-bits**, and **5** and **dh-1536-bits**, respectively, are synonyms.

```
ike phase1 { name | index } hash { sha1 | md5 }
show ike phase1 { name | index } hash
```

These commands set or display the specified IKE Phase1 profile's hash algorithm.

```
ike phase1 { name | index } independent rekeys { yes | no }
show ike phase1 { name | index } independent rekeys
no ike phase1 { name | index } independent rekeys
```

These commands set or display the specified IKE Phase1 profile's independent phase 2 re-keys setting.

```
ike phase1 { name | index } initial-contact { yes | no }
show ike phase1 { name | index } initial-contact
no ike phase1 { name | index } initial-contact
```

These commands set or display the specified IKE Phase1 profile's send initial-contact message setting.

```
ike phase1 { name | index } negotiation { normal | initiate-only | respond-only }
show ike phase1 { name | index } negotiation
```

These commands set or display the specified IKE Phase1 profile's negotiation setting.

```
ike phase1 { name | index } pfs { yes | no }
show ike phase1 { name | index } pfs
no ike phase1 { name | index } pfs
```

These commands set, display, or disable the specified IKE Phase1 profile's perfect forward secrecy setting.

```
ike phase1 { name | index } port policy { "strict" | "permissive" }
show ike phase1 { name | index } port policy
```

These commands set or display whether or not IKE requires packets to originate from the IANA port (500).

```
ike phase1 { name | index } sa lifetime { seconds | kbytes } { non-negative-integer | none }
show ike phase1 { name | index } sa lifetime [ { seconds | kbytes } ]
no ike phase1 { name | index } sa lifetime [ { seconds | kbytes } ]
```

These commands set, display, or disable one or both of the specified IKE Phase1 profile's two SA lifetimes (in seconds and/or kilobytes protected). Specifying neither the keyword **seconds** nor the keyword **kbytes** with the show variant of this command displays both lifetime values. The keyword **none** is equivalent to the value zero, and indicates that there is no lifetime of the specified type. The Phase1 SA lifetime minimum is 300 (seconds) and the maximum is 1 (leap) year (31622400 seconds).

Note: It is a run-time checked error if both of the IKE Phase 1 profile's SA lifetime values are set to zero or **none**.

```
ike phase1 { name | index } sa use-policy { new-sas-immediately | old-sas-until-expired }
show ike phase1 { name | index } sa use-policy
```

These commands set or display the specified IKE Phase1 profile's SA use policy.

```
ike phase1 { name | index } vendor-id { yes | no }
show ike phase1 { name | index } vendor-id
no ike phase1 { name | index } vendor-id
```

These commands set, display, or disable the specified IKE Phase1 profile's send vendor-id payload setting.

Current Restrictions

None.

Chapter 3

Netopia IAD CLI Commands

This chapter describes the syntax of the supported voice command set of the Netopia Internet Access Devices. Commands supported by a given device depend on whether or not it supports PBX features. All IADs support the basic voice commands.

- [“Voice Commands” on page 3-2](#)

Voice Commands

Voice Commands
Directory Numbers
<div>show voice dn <i>digit-string</i></div> <div>voice dn <i>digits-string</i> <i>port-number</i> [ringtype { dial busy congestion stutter call-waiting acknowledge normal urgent }} [forward { no <i>digit-string</i> {after4 now no }}] [huntgroup {yes no}]</div> <div>no voice dn <i>digit-string</i></div>
Tone Table
<div>show voice tonetable country</div> <div>voice tonetable country { Belgium Denmark France Germany Italy Netherlands Norway Sweden UK US }</div> <div>show voice tonetable current { dial busy congestion stutter call-waiting acknowledge normal urgent }</div> <div>voice tone {dial busy congestion stutter call-waiting acknowledge normal urgent } {{frequency <i>f1 f2 f3 f4</i> }}{{loopcount <i>number</i>}}{{delay <i>d1 d2 d3 d4</i>}}</div> <div>show voice tone { dial busy congestion stutter call-waiting acknowledge normal urgent }{{frequency loopcount delay}}</div>
Logs
<div>show voice log voice</div> <div>show voice log voice enable</div> <div>voice log voice enable { yes no }</div> <div>no voice log voice enable</div> <div>clear voice log voice</div> <div>show voice log accounting</div> <div>show voice log accounting enable</div> <div>voice log enable accounting { yes no }</div> <div>no voice log accounting enable</div> <div>clear voice log accounting</div> <div>show voice log error</div> <div>show voice log error enable</div> <div>voice log error enable { yes no }</div> <div>no voice log error enable</div> <div>clear voice log error</div>

Voice Commands
Outside and Operator Digits
show voice digit outside voice digit outside <i>0-9</i>
show voice digit operator voice digit operator <i>0-9</i>
Encoding Commands
show voice coding voice coding { <i>alaw</i> <i>mulaw</i> }
show voice overlap enable voice overlap enable { <i>yes</i> <i>no</i> } no voice overlap enable
Ring Cadence Commands
show voice ringcadence voice ringcadence { <i>20</i> <i>25</i> }
Voice Statistics Commands
show voice payload statistics show voice port [<i>1..numVoxPorts</i>] hang-ups show voice port [<i>1..muxVoxPorts</i>] debug statistics
Voice Port On/Off Hook Test Commands
do voice port <i>n</i> { <i>on-hook</i> <i>off-hook</i> }

Directory Numbers

show voice dn *digit-string*
voice dn *digits-string port-number* [**ringtype** { *dial* | *busy* | *congestion* | *stutter* |
 call-waiting | *acknowledge* | *normal* | *urgent* }]
 [**forward** { *no* | *digit-string* {*after4* | *now* | *no* } }] [**huntgroup** {*yes* | *no*}]
no voice dn *digit-string*

These commands allow you to show, set, or disable the directory numbers that correspond to each available voice phone port. External Directory numbers are mapped into particular internet extensions/ports, and optionally into the hunt group.

Note: The number of Pots Ports = 8 for SDSL, 4 for ADSL

Tone Table

```
show voice tonetable country
voice tonetable country { Belgium | Denmark | France | Germany | Italy | Netherlands |
    Norway | Sweden | UK | US }
show voice tonetable current { dial | busy | congestion | stutter | call-waiting | acknowledge |
    normal | urgent }
```

These commands allow you to set or show the tone table for the specified country. The **show voice tonetable country** command returns the name of the country whose tone table is currently active.

```
voice tone {dial | busy | congestion | stutter | call-waiting | acknowledge | normal | urgent }
    {{frequency f1 f2 f3 f4 }}{{loopcount number}}{{delay d1 d2 d3 d4}}
show voice tone { dial | busy | congestion | stutter | call-waiting | acknowledge | normal |
    urgent }{{frequency | loopcount | delay}}
```

These commands allow you to set or show the tone to be generated for the specified function.

Logs

```
show voice log voice
show voice log voice enable
voice log voice enable { yes | no }
no voice log voice enable
clear voice log voice
```

These commands allow you to enable, disable, clear or show the logging of voice-related events for the IAD. The **show voice log voice** command returns a dump of the log of voice-related events.

```
show voice log accounting
show voice log accounting enable
voice log enable accounting { yes | no }
no voice log accounting enable
clear voice log accounting
```

These commands allow you to enable, disable, clear or show the logging of voice-related accounting events for the IAD. The **show voice log accounting** command returns a dump of the log of voice-related accounting events.

```
show voice log error
show voice log error enable
voice log error enable { yes | no }
no voice log error enable
clear voice log error
```

These commands allow you to enable, disable, clear or show the logging of voice-related errors for the IAD. The **show voice log error** command returns a dump of the log of voice-related errors.

Outside and Operator Digits

show voice digit outside
voice digit outside 0-9

These commands allow you to set or show the digit used to obtain an outside line. Legal digits can range from 0 to 9.

show voice digit operator
voice digit operator 0-9

These commands allow you to set or show the digit used to dial the operator. Legal digits can range from 0 to 9.

Encoding Commands

show voice coding
voice coding { alaw | mulaw }

These commands allow you to set or show the voice coding method you will be using. The default is mu-law, which is the standard 8-bit, 8 kHz, mono format intended primarily for the requirements of voice in North America. You can also choose a-law, a more common audio format outside North America.

show voice overlap enable
voice overlap enable { yes | no }
no voice overlap enable

These commands allow you to set, show, or disable the overlap behavior of the tones generated by the device. If overlap is enabled, and the line is unavailable, you hear a busy signal when you pick up the handset.

Ring Cadence

show voice ringcadence
voice ringcadence {20 | 25}

These commands allow you to set or show the ring cadence for the attached phones. Legal values are 20Hz (the default) or 25Hz for compliance with several non-North American telephone systems.

Voice Statistics

Note: These commands are supported beginning with firmware version 5.2.

show voice payload statistics

This command allows you to display a dump of a variety of voice transmit and receive statistics for diagnostic purposes. This command produces output similar to the following:

Chan & cmprss	Receive			Transmit		Tx Drop
	-----DSP-----	WAN---	Last Sec.	-----DSP-----	WAN	--Bytes--
1 G711	123456789	123456789	123456789	123456789	123456789	123456789
2 G711	123456789	123456789	123456789	123456789	123456789	123456789
3 G711	123456789	123456789	123456789	123456789	123456789	123456789
4 G711	123456789	123456789	123456789	123456789	123456789	123456789
5 G711	123456789	123456789	123456789	123456789	123456789	123456789
6 G711	123456789	123456789	123456789	123456789	123456789	123456789
7 G711	123456789	123456789	123456789	123456789	123456789	123456789
8 G711	123456789	123456789	123456789	123456789	123456789	123456789

On the Receive side a payload is delivered first to the WAN, and then to the DSP. On the Transmit side this is reversed.

show voice port [1..numVoxPorts] hang-ups

This command allows you to display voice transmit and receive statistics for the specified voice port(s). This command produces output similar to the following:

Port	-----Near-----	Far---	Payload
1	123456789	123456789	123456789

If no voice port index is specified, statistics for all ports are displayed:

Port	-----Near-----	Far---	Payload
1	123456789	123456789	123456789
2	123456789	123456789	123456789
3	123456789	123456789	123456789
4	123456789	123456789	123456789
5	123456789	123456789	123456789
6	123456789	123456789	123456789
7	123456789	123456789	123456789
8	123456789	123456789	123456789

show voice port [1..muxVoxPorts] debug statistics

If the token after **port** is a valid port index this command produces output similar to the following:

```
Port N Local Call State: Local Call State description
Port N WAN Call State: WAN Call State description
Port N VoDSL Call State: VoDSL Call State description
```

If the token after **port** is **debug**, that is, no index is provided, the output is a dump of all of the ports. The following table lists the internal call state enumerations and their corresponding text descriptions:

Local Call State	
PHONE_IDLE	idle
INITIATING	initiating
INITIATING_OUTSIDE	initiating, inbound call
DIALING_LOCAL	initiating, extension to extension call
DIALING_OUTSIDE	initiating, placing outbound call
DIALED	dialed
FAR_END_ALERTED	far end contacted
BUSY	busy
INVALID_NUMBER	invalid number
ACTIVE	active
ACT_WITH_2	Active with 2 calls (other call is 'on hold')
ACT_CALL_XFER_COLLECTING_DIGITS	call transfer, collecting digits
ACT_CALL_XFER_INITO	call transfer, initiating outbound WAN call
ACT_CALL_XFER_DIALO	call transfer, dialing outbound WAN call
ACT_CALL_XFER_DIALED	call transfer, dialed
ACT_CALL_XFER_FAR_ALERT	call transfer, far end contacted
ACT_INTERNAL_XFER	call transfer, extension to extension
ACT_WAIT_FOR_FLASH_PRESS	waiting for hook flash
ALERTING	inbound call, ringing
OFF_HOOK	off-hook
WAIT_FOR_ON_HOOK	waiting for on-hook
WAN Call State	
ISDN_B_IDLE	idle
INITIATING	initiating
ISDN_B_DIALING	dialing
DIALED	dialed
FAR_END_ALERTED	far end contacted
FAR_END_PICKED_UP	far end picked up

ACTIVE	active
CALL_PROCEEDING	call proceeding
ISDN_B_ALERTING	inbound call, ringing
OFF_HOOK	off-hook
RELEASING_NEAR1	releasing call, near-end disconnect
RELEASING_FAR1	releasing call, far-end disconnect
VoDSL Call State	
IDLE	idle
RX_RING_RCVD	inbound call, ring received
RX_RING_WAIT	inbound call, ring waiting
RX_RING_CONFIRM	inbound call, ring confimed
RX_RINGING	inbound call, ringing
RX_RING_ABORT	inbound call, ring aborted
RX_CONNECT	inbound call, connect
TX_OFFHOOK	transmitting off-hook
TX_CONNECT	transmitting connect
HANGUP	hanging-up
HANGUP_FINAL	hang-up completed

do voice port [1..numVoxPorts] { on-hook | off-hook }

This command allows you to take a specified voice port virtually on and off hook, for testing purposes.

Chapter 4

Netopia Router Text Configuration Upload

This chapter describes the supported TFTP text configuration upload process.

TFTP Text Configuration Upload Overview

With the release of Netopia router firmware version 4.3.5 you can configure many of the basic features of the router by uploading a text-based configuration file to the router. This file can be either a Macintosh- or PC-formatted text file. There must be no formatting information in the file – it must contain only raw text. Generally this means that you must save the file in Text Only (.txt) format when using word processing applications that support text formatting.

The file must be located on a TFTP Server. The Netopia router needs the IP Address or DNS Name of the TFTP Server in order to start the file upload. There are at least three ways to accomplish this:

- SNMP
- VT100 Menu Console (Serial or Telnet)
- VT100 Command Line Console (Serial or Telnet)

The supported character set for TFTP text configuration files is the set of US-ASCII printable characters (ASCII values from 32 to 126 inclusive), including the space character. This means that characters containing diacritical marks, such as 'À', are not supported. Such characters will be translated to the character '%' when processing a text configuration file. The reason for this is that the Netopia router web server supports only the US-ASCII character set.

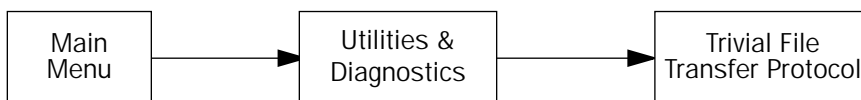
Note: All commands, including the last, must be followed by an appropriate end-of-line sequence (Carriage Return, Line Feed, or Carriage Return/Line Feed pair).

SNMP

Three SNMP objects are associated with TFTP text configuration upload. They are tftpServerName, tftpConfigFileName, and tftpReadConfig. All three objects are defined in the Netopia MIB. Refer to this document found on the Netopia CD for more information.

VT100 Menu Console

The path to the TFTP menu is:



4-2 Command Line Interface Commands Reference

You will need to set the TFTP Server Name (IP address or domain name) and the Config File Name, and then invoke the transfer using GET CONFIG FROM SERVER.

VT100 Command Line Console

The router's console user interface comes up in Menu mode by default. In this mode you use the arrow, Escape, and Return/Enter keys to navigate through a series of screens. To invoke the command line at any time, hit Control-N. The console will erase the window, and you will be presented with a # prompt. The procedure for invoking the file transfer via the command line console is described in the section "[Miscellaneous Commands](#)" on page 2-67. To return to Menu mode hit Control-N again.

A command that contains a syntax error will cause the configuration file processing to terminate. Any valid commands that were processed before the error was detected will modify the configuration of the router. The error will be reported in the Device Event History, which will display as much of the text of the offending command as possible.

Example Text Configuration File

The following text file is provided for your use as an example. Make your own appropriate substitutions.

Note: All commands, including the last, must be followed by an appropriate end-of-line sequence (Carriage Return, Line Feed, or Carriage Return/Line Feed pair).

```
;Example config file

;LAN configuration
;set ethernet address
interface ethernet 0 ip address 163.176.227.1/24

;set a secondary ethernet address
interface ethernet 0 ip address 163.176.254.1/24 secondary

;set default gateway to 163.176.224.1
ip gateway 163.176.224.1

;set Rx and Tx RIP
interface ethernet 0 ip rip receive both
interface ethernet 0 ip rip transmit v1

;set dns 1 and dns 2
ip dns 1 163.176.4.10
ip dns 2 163.176.4.31

;set domain name
ip domain-name netopia.com

;configure IP address serving to serve 100 addresses off the 163.176.227.0/24 subnet
ip address-serve 163.176.227.101 163.176.227.201 dhcp
ip address-serve mode server

;other possible commands include:
;interface wan 1 dle ppp
;interface wan 1 dle rfc1483
;user mysecurname mysecurpass

;WAN configuration
;instantiate profile 1 with name "My ISP" and DLE type RFC1483
cp 1
cp 1 tag "My ISP"

;set up profile addressing
```

4-4 Command Line Interface Commands Reference

```
cp 1 ip address local 163.176.224.2
```

```
;other possible commands include:
```

```
;cp 2
```

```
;cp 16
```

```
;cp 1 ip nat enabled no
```

```
;cp 1 ip nat enabled yes
```

```
;cp 1 ip addressing unnumbered
```

```
;cp 1 ip addressing numbered
```

```
;cp 1 ip address remote 163.176.224.1
```

```
;cp 1 ip mask local 255.255.255.0
```

```
;cp 1 ip mask remote 255.255.255.0
```

```
;cp 1 dle rfc1483
```

```
;cp 1 dle ppp
```

```
;cp 1 ppp authentication pap
```

```
;cp 1 ppp authentication chap
```

```
;cp 1 ppp authentication send name "My Name"
```

```
;cp 1 ppp authentication send password "My Password"
```

Chapter 5

CLI Error Messages

This chapter describes the CLI error messages and their meaning.

Negative errors

Negative errors are fatal. They will terminate processing of a TFTP configuration file upload if the command that caused the error was executed as part of a TFTP configuration file upload.

Fatal system errors

; error -1: unknown error

This error indicates that an internal error occurred within the command line processor. This error should not occur under normal circumstances.

; error -2: memory allocation failed

This error indicates that the command line processor ran out of memory attempting to complete the requested operation. This error should not occur under normal circumstances.

Fatal syntax errors

; error -101: no match

This error indicates that you entered an unrecognized command.

; error -120: syntax error

This error indicates that you entered a command with a syntactic error for which the command line processor was unable to provide a more specific error message. For example, you may have misspelled or omitted a keyword or interchanged two keywords.

Example:

```
#frame-relay dlci 22 default
; error -120: syntax error
```

; error -121: illegal operation

This error indicates that you attempted to perform an unsupported operation, or one that does not make sense.

Example:

```
#no version
; error -121: illegal operation
#clear version
; error -121: illegal operation
```

; error -122: illegal value

This error indicates that you entered a properly formatted command, but that one of the values specified in the command is not a valid value for the attribute you are trying to set. For example, this error would be generated if you entered the interface intf-type id dle command and requested a datalink encapsulation that isn't supported by the specified wan interface.

Example:

```
#interface sds1 1 pvc 0 65536
; error -122: illegal value
```

; error -123: illegal ip address

This error indicates that you supplied an invalid value where an IP address is required. IP addresses should be specified in “dotted-quad” notation: four decimal values, each between 0 and 255 inclusive, separated by dots (e.g., 192.168.1.1).

Example:

```
#interface ethernet 0 ip address xyz
; error -123: illegal ip address
#interface ethernet 0 192.168.256.1/24
; error -123: illegal ip address
```

; error -124: illegal ip mask

This error indicates that you supplied an invalid value where an IP mask is required. An IP mask always may be entered in "dotted-quad" notation -- four decimal values, each between 0 and 255 inclusive, separated by dots (e.g., 255.255.255.0). Whenever the IP mask is being entered in conjunction with an IP address, it may also be entered in "prefix" notation -- a slash ("/") immediately following the IP address, followed by a value between 0 and 32 inclusive indicating the number of contiguous ones-bits in the mask (e.g., /24). Note that IP mask values entered in dotted-quad notation must consist of a contiguous number of ones-bits beginning with the most significant bit. IP masks with discontinuous ones-bits (such as 255.0.255.0) are invalid.

Example:

```
#int e0 ip address 163.176.1.1/40
; error -124: illegal ip mask
```

; error -125: invalid index

This error indicates that the command referenced a currently non-existent instance of an indexed object. The specified index value might be valid at some point if the appropriate instance of the object were created.

Example:

```
#show frame-relay dlci 22
; error -125: invalid index
#frame-relay dlci 22 tag "My DLCI"
#show frame-relay dlci 22
frame-relay dlci 22 tag "My DLCI" ip-address 0.0.0.0 cir
default bc default be default enable
```

; error -126: number required

This error indicates that a non-numeric value was entered where a numeric one was required.

Example:

```
#ping 192.168.1.1 count abc
; error -126: number required
```

; error -127: index out of bounds

This error indicates that an out of range value was specified in a command that requires an index, such as attempting to access connection profile 17 on a router that supports only sixteen connection profiles. The value supplied is never a valid index value in the context in which it was used.

Example:

```
#sh cp 17
; error -127: index out of bounds
```

; error -128: 'yes' or 'no' required

This error indicates that a value other than 'yes' or 'no' was entered where only 'yes' or 'no' are acceptable.

Example:

```
#cp 1 enable foo
; error -128: 'yes' or 'no' required
```

; error -129: text too long

This error indicates that a string value was entered that was longer than the permissible length for the particular string.

Example:

```
#cp 1 tag "A overly long connection profile name"
; error -129: text too long
```

; error -130: text can't be empty

This error indicates that an empty string was supplied for a string item that must contain at least one character, such as a connection profile name.

Example:

```
#cp 1 tag ""
; error -130: text can't be empty
```

; error -133: invalid address/mask

This error indicates either than you omitted a required IP address and/or mask.

Example:

```
#interface ethernet 0 ip address 192.168.1.2
; error -133: invalid address/mask
```

; error -135: missing required text

This error indicates that you omitted a portion of the command.

Example:

```
#ip route
; error -135: missing required text
```

; error -136: illegal or conflicting range

This error indicates that you entered an invalid or an inconsistent range of values. For example, you would receive this error if you entered a range of IP addresses with a starting address that is greater than the ending address.

Example:

```
#ip addr 163.176.12.100 163.176.12.50
; error -136: illegal or conflicting range
```

; error -137: no arp cache entry

This error indicates that you attempted to delete a non-existent arp cache entry.

Example:

```
#arp 192.168.1.1 00:00:C5:70:00:04
#no arp 192.168.1.2 00:00:C5:70:00:04
; error -137: no arp cache entry
```

; error -138: invalid wan port

This error indicates that you entered an invalid interface index. See [“Interface Naming Conventions” on page 1-3](#).

Example:

```
#arp 192.168.1.1 00:00:C5:70:00:04 3
; error -138: invalid wan port
```

; error -140: illegal hardware address

This error indicates that you entered an improperly formatted hardware address. The format for an Ethernet MAC address is six hexadecimal values between 00 and FF inclusive separated by colons (e.g., 00:00:C5:70:00:04).

Example:

```
#arp 192.168.1.1 00:00:C5:70:00:
; error -140: illegal hardware address
```

; error -142: address and mask required

This error indicates that you omitted an IP address and/or mask where both are required.

Example:

```
#no interface ethernet 0 ip address 192.168.1.1
; error -142: address and mask required
```

; error -143: can't add

This error indicates that you attempted to add more than the allowed number of some object.

Example:

```
#interface ethernet 0 address-serve helper 10.0.0.1
#interface ethernet 0 address-serve helper 20.0.0.1
#interface ethernet 0 address-serve helper 30.0.0.1
#interface ethernet 0 address-serve helper 40.0.0.1
#interface ethernet 0 address-serve helper 50.0.0.1
; error -143: can't add
```

; error -144: incomplete command

This error indicates that you omitted a portion of the command.

Example:

```
#interface ethernet 0 ip address
; error -144: incomplete command
```

; error -145: ip address required

This error indicates that you failed to supply an IP address where one is required.

Example:

```
#cp 1 ip address remote
; error -145: ip address required
```

; error -146: ip mask required

This error indicates that you failed to supply an IP mask where one is required.

Example:

```
#cp 1 ip mask remote
; error -146: ip mask required
```

; error -147: too many tokens

This error indicates that you entered more items than are allowed as part of the command. This could result from failing to quote a string value that contains one or more spaces.

Example:

```
#cp 1 telco dn 555 1212
; error -147: unexpected text following command
#cp 1 telco dn "555 1212"
#sh cp 1 telco dn
cp 1 telco dn "555 1212"
#
```

; error -148: text too short

This error indicates that you supplied a string value that is shorter than the minimum permissible length for a string item.

; error -149: no such subnet

This error indicates that you specified (or implicitly referenced) an unknown subnet.

Example:

```
#show interface ethernet 0 ip address
interface ethernet 0 ip address 192.168.1.1/24
#interface ethernet 0 address-serve range 192.168.1.128 192.168.1.159
#interface ethernet 0 address-serve gateway 192.168.2.1
; error -149: no such subnet
```

; error -200: execution failed

This error indicates that a requested operation, such as a TFTP configuration file upload, failed.

Example:

```
#receive tftp config 192.168.1.1 myconfig.txt
; error -200: execution failed
```

Voice command errors

; error -250: bad extension number

; error -251: extension number does not exist

; error -252: extension number already exists

; error -253: wrong Auto-Attendant time

; error -254: directory number doesn't exist

; error -255: port has not extension number,set phonemap first

; error -256: Directory is full,can't add new one

; error -257: Caller ID list is full

; error -258: Bad Caller ID

; error -259: No such carrier name

; error -260: Carrier Name too long

; error -261: prefix Name too long

; error -262 Pin Name too long

; error -263: Carrier table is full,can't add new one

; error -264: No such dialed digits

; error -265: Route table is full,can't add new one

; error -266: Duplicated dialed string

; error -267: Dialed digits too long

; error -268: invalid digit

; error -269: only single digit allowed

Fatal access control errors

; error -400: access denied

This error indicates that you attempted to display an attribute that may not be displayed, or to change an attribute to which you do not have access.

Example:

```
#show cp 1 ppp authentication send password  
; error -400: access denied
```

Positive errors

Positive (non-fatal) errors do not terminate TFTP configuration file upload processing.

; error 1: not supported with current hardware

This error indicates that you entered a command that is not supported by the particular the router model you have, or the specified WAN interface module(s). For example, you may have issued the interface sdsl id pvc command, but the SDSL wan interface module in the specified slot is a frame-based SDSL (R7100) interface rather than a cell-based SDSL (R7200) interface.

; error 2: not supported with current configuration

This error indicates that you entered a command that is not compatible with the current configuration of the router. For example, you may have issued a command specific to an ISDN interface in switched mode, but the specified ISDN interface is currently configured for leased mode.

Example:

```
#sh int isdn 1 mode
interface isdn 1 mode idsl-cmn
#int isdn 1 spid 555-1212
; error 2: not supported with current configuration
```

; error 3: not supported

This error indicates that you entered a command in a context in which it was not supported. For example, you may have included the clear command in a text configuration file uploaded via TFTP.

; error 102: can't delete

This error indicates that you attempted to delete an item that doesn't exist.

Example:

```
#show interface ethernet 0 address-serve helper
interface ethernet 0 address-serve helper 10.0.0.1
interface ethernet 0 address-serve helper 20.0.0.1
interface ethernet 0 address-serve helper 30.0.0.1
interface ethernet 0 address-serve helper 40.0.0.1
#no interface ethernet 0 address-serve helper 50.0.0.1
; error 102: can't delete
```

; error 103: incomplete command

This error indicates that you omitted a portion of the command.

Example:

```
#interface ethernet 0 ip
; error 103: incomplete command
```

; error 104: ambiguous

This error indicates that you didn't enter enough of the text of a keyword such that the keyword as entered was ambiguous.

Example:

```
#sh cp 1 t
; error 104: ambiguous
#sh cp 1 tag
cp 1 tag "Profile 01"
```

; error 106: arp cache is full delete an entry to make room

This error indicates that you attempted to add an entry to the global arp cache when it already contained the maximum number of entries (16).

Index of Commands

A

arp 2-42

B

backup delay 2-78
 backup enable 2-78
 backup ping host 2-79
 backup recovery delay 2-79
 backup recovery idle delay 2-79
 backup recovery idle only 2-79
 backup recovery layer-2-loss 2-79
 backup recovery mode 2-79

C

clear 2-67
 clear arp-cache 2-42
 clear voice log accounting 3-4
 clear voice log error 3-4
 clear voice log voice 3-4
 console authentication 2-80
 cp { *name* | *index* } 2-49
 cp { *name* | *index* } connection demand 2-55
 cp { *name* | *index* } connection timeout seconds 2-55
 cp { *name* | *index* } dle 2-49
 cp { *name* | *index* } enable 2-49
 cp { *name* | *index* } filterset 2-49
 cp { *name* | *index* } frame relay dlci multicast-number 2-53
 cp { *name* | *index* } frame-relay dlci auto-detect 2-53
 cp { *name* | *index* } interface-group 2-57
 cp { *name* | *index* } ip address local 2-49
 cp { *name* | *index* } ip address remote 2-50

cp { *name* | *index* } ip addressing 2-49
 cp { *name* | *index* } ip dhcp client mode 2-50
 cp { *name* | *index* } ip mask local 2-50
 cp { *name* | *index* } ip mask remote 2-50
 cp { *name* | *index* } ip nat enable 2-54
 cp { *name* | *index* } ip nat rule-list 2-54
 cp { *name* | *index* } ip nat server-list 2-54
 cp { *name* | *index* } ip negotiate-lan 2-51
 cp { *name* | *index* } ip netbios proxy enable 2-51
 cp { *name* | *index* } ip rip receive 2-51
 cp { *name* | *index* } ip rip transmit 2-51
 cp { *name* | *index* } ipsec authentication key 2-60
 cp { *name* | *index* } ipsec
 dead-peer-detection 2-62
 cp { *name* | *index* } ipsec encryption key 2-60
 cp { *name* | *index* } ipsec idle-timeout 2-62
 cp { *name* | *index* } ipsec ike phase1 2-62
 cp { *name* | *index* } ipsec ip 2-59, 2-63
 cp { *name* | *index* } ipsec key-manager 2-62
 cp { *name* | *index* } ipsec pfs 2-62
 cp { *name* | *index* } ipsec sa lifetime 2-64
 cp { *name* | *index* } ipsec spi 2-59
 cp { *name* | *index* } ipsec suite 2-63
 cp { *name* | *index* } ipsec suite encryption 2-59
 cp { *name* | *index* } ppp authentication 2-52
 cp { *name* | *index* } ppp authentication type 2-52
 cp { *name* | *index* } ppp usage 2-53
 cp { *name* | *index* } pptp authentication 2-56
 cp { *name* | *index* } pptp authentication { send | receive } password 2-56
 cp { *name* | *index* } pptp authentication type 2-56
 cp { *name* | *index* } pptp compression 2-56
 cp { *name* | *index* } pptp encryption 2-56
 cp { *name* | *index* } pptp ip partner 2-56

cp { *name* | *index* } pptp ip via 2-56
 cp { *name* | *index* } tag 2-49
 cp { *name* | *index* } telco callback 2-54
 cp { *name* | *index* } telco compuserve hostname
 2-58
 cp { *name* | *index* } telco compuserve login 2-58
 cp { *name* | *index* } telco compuserve
 password 2-58
 cp { *name* | *index* } telco compuserve username
 2-58
 cp { *name* | *index* } telco direction 2-53
 cp { *name* | *index* } telco dn 2-54
 cp { *name* | *index* } telco prefix 2-54

D

date 2-3
 do voice port { on-hook | off-hook } 3-8

E

exit 2-3

F

frame-relay dlci default 2-65
 frame-relay lmi type 2-66
 frame-relay tim 2-66

H

hardware acceleration enable 2-84

I

ike phase1 2-86
 ike phase1 { *name* | *index* } authentication
 method 2-87
 ike phase1 { *name* | *index* } authentication
 shared-secret 2-88
 ike phase1 { *name* | *index* } dangling-sas 2-88
 ike phase1 { *name* | *index* } encryption 2-88
 ike phase1 { *name* | *index* } group 2-88
 ike phase1 { *name* | *index* } hash 2-88
 ike phase1 { *name* | *index* } identity 2-87
 ike phase1 { *name* | *index* } independent rekeys
 2-88
 ike phase1 { *name* | *index* } initial-contact 2-88

ike phase1 { *name* | *index* } mode 2-87
 ike phase1 { *name* | *index* } negotiation 2-88
 ike phase1 { *name* | *index* } pfs 2-88
 ike phase1 { *name* | *index* } port policy 2-89
 ike phase1 { *name* | *index* } sa lifetime 2-89
 ike phase1 { *name* | *index* } sa use-policy 2-89
 ike phase1 { *name* | *index* } tag 2-87
 ike phase1 { *name* | *index* } vendor-id 2-89
 interface { adsl | ethernet | isdn | sdsl } *id* pppoe
 enable 2-20
 interface { sdsl | isdn } *id* rfc1973 dlci 2-29
 interface { sdsl | isdn } *id* rfc1973 enable 2-29
 interface { sdsl | isdn } *id* rfc1973 lmi 2-29
 interface adsl *id* pvc 2-25
 interface dsl *id* line type 2-33
 interface ethernet 0 address-serve 2-16
 interface ethernet 0 address-serve clients 2-14
 interface ethernet 0 address-serve dhcp enable
 2-12
 interface ethernet 0 address-serve dhcp
 lease-time 2-14
 interface ethernet 0 address-serve gateway 2-15
 interface ethernet 0 address-serve helper 2-15
 interface ethernet 0 address-serve mode 2-15
 interface ethernet 0 address-serve netbios mode
 enable 2-17
 interface ethernet 0 address-serve netbios mode
 type 2-17
 interface ethernet 0 address-serve netbios
 name-server address 2-18
 interface ethernet 0 address-serve netbios
 name-server enable 2-18
 interface ethernet 0 address-serve netbios
 scope enable 2-17
 interface ethernet 0 address-serve netbios
 scope name 2-18
 interface ethernet 0 address-serve range 2-16
 interface ethernet *id* ip address 2-11
 interface ethernet *id* ip filterset 2-13
 interface ethernet *id* ip netbios proxy enable 2-12
 interface ethernet *id* ip rip receive 2-13
 interface ethernet *id* ip rip transmit 2-13
 interface ethernet *id* pppoe enable 2-13

interface ethernet *wan-id* ip nat enable 2-13
 interface ethernet *wan-id* ip nat map-list 2-14
 interface ethernet *wan-id* ip nat server-list 2-14
 interface ethernet *wan-id* mac address 2-14
 interface *intf-type id* dle 2-19
 interface isdn *id* dn 2-24
 interface isdn *id* imux mode 2-21
 interface isdn *id* mode 2-22
 interface isdn *id* speed 2-23
 interface isdn *id* spid 2-24
 interface isdn *id* switch 2-23
 interface sdsl *id* clock rate 2-27
 interface sdsl *id* clock source 2-26
 interface sdsl *id* operation mode 2-28
 interface sdsl *id* priority-queuing enable 2-29
 interface sdsl *id* pvc 2-29, 2-31
 interface sdsl *id* pvc { *id* | *tag* } cp { profile-id |
 profile-tag | default } 2-32
 interface sdsl *id* pvc { *id* | *tag* } enable 2-31
 interface sdsl *id* pvc { *id* | *tag* } tag 2-31
 interface sdsl *id* pvc { *id* | *tag* } vci 2-31
 interface sdsl *id* pvc { *id* | *tag* } vpi 2-31
 interface t1 *id* buildout 2-35
 interface t1 *id* channels 2-35
 interface t1 *id* clock source 2-35
 interface t1 *id* diagnostic mode 2-38
 interface t1 *id* dle 2-35
 interface t1 *id* encoding 2-35
 interface t1 *id* framing 2-35
 interface t1 *id* operation mode 2-36
 interface t1 *id* prm-enable 2-36
 interface t1 *id* rfc1973 dlci 2-36
 interface t1 *id* rfc1973 enable 2-36
 interface t1 *id* rfc1973 lmi 2-36
 interface wan 0 tracking 2-20
 ip dns 2-40
 ip domain-name 2-40
 ip filterset 2-82
 ip gateway 2-40
 ip nat map 2-74
 ip nat public *tag* dynamic 2-73
 ip nat public *tag* pat 2-74
 ip nat public *tag* static 2-74

ip nat server 2-75

ip route 2-41

M

macip static-range 2-77

N

backup 2-78

no arp 2-42

no backup 2-78

no cp { *name* | *index* } 2-49

no cp { *name* | *index* } filterset 2-49

no cp { *name* | *index* } ip nat rule-list 2-54

no cp { *name* | *index* } ip nat server-list 2-54

no cp { *name* | *index* } ip negotiate-lan 2-51

no cp { *name* | *index* } ip netbios proxy
 enable 2-51

no cp { *name* | *index* } ipsec dead-peer-detection
 2-62

no cp { *name* | *index* } ipsec idle-timeout 2-62

no cp { *name* | *index* } ipsec ike phase1 2-62

no cp { *name* | *index* } ipsec pfs 2-62

no cp { *name* | *index* } ipsec sa lifetime 2-64

no cp { *name* | *index* } pptp authentication 2-56

no cp { *name* | *index* } pptp authentication { send |
 receive } password 2-56

no cp { *name* | *index* } pptp compression 2-56

no cp { *name* | *index* } pptp encryption 2-56

no cp { *name* | *index* } telco compuserve
 hostname 2-58

no cp { *name* | *index* } telco compuserve
 login 2-58

no cp { *name* | *index* } telco compuserve
 password 2-58

no cp { *name* | *index* } telco compuserve
 username 2-58

no frame-relay dlci 2-65

no frame-relay lmi type 2-66

no hardware acceleration enable 2-84

no ike phase1 2-86

no ike phase1 { *name* | *index* } dangling-sas 2-88

no ike phase1 { *name* | *index* } independent
 rekeys 2-88

- no ike phase1 { *name* | *index* } initial-contact 2-88
- no ike phase1 { *name* | *index* } pfs 2-88
- no ike phase1 { *name* | *index* } sa lifetime 2-89
- no ike phase1 { *name* | *index* } vendor-id 2-89
- no interface { adsl | ethernet | isdn | sdsl } *id*
pppoe enable 2-20
- no interface { sdsl | isdn } *id* rfc1973 enable 2-29
- no interface { sdsl | isdn } *id* rfc1973 lmi 2-29
- no interface ethernet 0 address-serve 2-16
- no interface ethernet 0 address-serve
clients 2-14
- no interface ethernet 0 address-serve dhcp
enable 2-12
- no interface ethernet 0 address-serve
helper 2-15
- no interface ethernet 0 address-serve netbios
mode enable 2-17
- no interface ethernet 0 address-serve netbios
name-server enable 2-18
- no interface ethernet 0 address-serve netbios
scope enable 2-17
- no interface ethernet 0 address-serve range 2-16
- no interface ethernet *id* ip address 2-11
- no interface ethernet *id* ip filterset 2-13
- no interface ethernet *id* ip netbios proxy enable
2-12
- no interface ethernet *id* ip rip receive 2-13
- no interface ethernet *id* ip rip transmit 2-13
- no interface ethernet *id* pppoe enable 2-13
- no interface ethernet *wan-id* ip nat enable 2-13
- no interface ethernet *wan-id* ip nat map-list 2-14
- no interface ethernet *wan-id* ip nat
server-list 2-14
- no interface isdn *id* dn 2-24
- no interface isdn *id* spid 2-24
- no interface sdsl *id* priority-queuing enable 2-29
- no interface sdsl *id* pvc 2-31
- no interface sdsl *id* pvc { *id* | *tag* } enable 2-31
- no interface t1 *id* prm-enable 2-36
- no interface t1 *id* rfc1973 enable 2-36
- no interface t1 *id* rfc1973 lmi 2-36
- no ip dns 2-40
- no ip domain-name 2-40

- no ip filterset 2-82
- no ip gateway 2-40
- no ip nat map 2-75
- no ip nat public 2-74
- no ip nat server 2-75
- no ip route 2-41
- no macip static-range 2-77
- no preferences changes immediate 2-4
- no preferences console timeout 2-4
- no radius-server 2-81
- no security password 2-6
- no snmp community 2-6
- no snmp system contact 2-6
- no snmp system location 2-6
- no snmp system name 2-6
- no system web-server enable 2-6
- no system web-server lan-only 2-7
- no user 2-7
- no voice dn 3-3
- no voice log accounting enable 3-4
- no voice log error enable 3-4
- no voice log voice enable 3-4
- no voice overlap enable 3-5

P

- ping 2-67
- ping oam interface sdsl 2-68
- preferences changes immediate 2-4
- preferences console default 2-4
- preferences console timeout 2-4
- preferences date format 2-4
- preferences output format 2-5
- preferences output mask 2-5
- preferences time format 2-5

R

- radius identifier 2-81
- radius-server 2-81
- receive tftp config 2-68
- receive tftp firmware 2-69
- receive xmodem firmware 2-69
- reset 2-70
- reset factory 2-70

S

security password 2-6
 send tftp config 2-69
 show arp static 2-42
 show arp-cache 2-42
 show backup status 2-78
 show console authentication 2-80
 show cp { *name* | *index* } filterset 2-49
 show cp { *name* | *index* } id 2-55
 show cp { *name* | *index* } interface-group 2-57
 show cp { *name* | *index* } ip dhcp client mode 2-50
 show cp { *name* | *index* } ip negotiate-lan 2-51
 show cp { *name* | *index* } ip netbios proxy enable 2-51
 show cp { *name* | *index* } ipsec
 dead-peer-detection 2-62
 show cp { *name* | *index* } ipsec idle-timeout 2-62
 show cp { *name* | *index* } ipsec ike phase1 2-62
 show cp { *name* | *index* } ipsec ip 2-63
 show cp { *name* | *index* } ipsec key-manager 2-62
 show cp { *name* | *index* } ipsec pfs 2-62
 show cp { *name* | *index* } ipsec sa lifetime 2-64
 show cp { *name* | *index* } ipsec suite 2-63
 show cp { *name* | *index* } pptp authentication 2-56
 show cp { *name* | *index* } telco compuserve
 hostname 2-58
 show cp { *name* | *index* } telco compuserve login 2-58
 show cp { *name* | *index* } telco compuserve
 username 2-58
 show date 2-3
 show frame-relay lmi statistics 2-66
 show frame-relay lmi type 2-66
 show frame-relay pvc 2-66
 show hardware acceleration enable 2-84
 show history 2-70
 show ike phase1 2-86
 show ike phase1 { *name* | *index* } authentication
 method 2-87
 show ike phase1 { *name* | *index* }
 dangling-sas 2-88
 show ike phase1 { *name* | *index* } encryption 2-88
 show ike phase1 { *name* | *index* } group 2-88
 show ike phase1 { *name* | *index* } hash 2-88
 show ike phase1 { *name* | *index* } id 2-86
 show ike phase1 { *name* | *index* } identity 2-87
 show ike phase1 { *name* | *index* } independent
 rekeys 2-88
 show ike phase1 { *name* | *index* }
 initial-contact 2-88
 show ike phase1 { *name* | *index* } mode 2-87
 show ike phase1 { *name* | *index* }
 negotiation 2-88
 show ike phase1 { *name* | *index* } pfs 2-88
 show ike phase1 { *name* | *index* } port policy 2-89
 show ike phase1 { *name* | *index* } sa lifetime 2-89
 show ike phase1 { *name* | *index* } sa
 use-policy 2-89
 show ike phase1 { *name* | *index* } tag 2-87
 show ike phase1 { *name* | *index* } vendor-id 2-89
 show interface { adsl | ethernet | isdn | sdsl } *id*
 pppoe enable 2-20
 show interface { sdsl | isdn } *id* rfc1973 dlci 2-29
 show interface { sdsl | isdn } *id* rfc1973
 enable 2-29
 show interface { sdsl | isdn } *id* rfc1973 lmi 2-29
 show interface adsl *id* pvc 2-25
 show interface adsl *id* status 2-25
 show interface dsl *id* line type 2-33
 show interface ethernet 0 address-serve 2-16
 show interface ethernet 0 address-serve clients 2-14
 show interface ethernet 0 address-serve dhcp
 enable 2-12
 show interface ethernet 0 address-serve dhcp
 lease-time 2-14
 show interface ethernet 0 address-serve gateway 2-15
 show interface ethernet 0 address-serve helper 2-15
 show interface ethernet 0 address-serve
 mode 2-15
 show interface ethernet 0 address-serve netbios
 mode enable 2-17
 show interface ethernet 0 address-serve netbios

- mode type 2-17
- show interface ethernet 0 address-serve netbios name-server address 2-18
- show interface ethernet 0 address-serve netbios name-server enable 2-18
- show interface ethernet 0 address-serve netbios scope enable 2-17
- show interface ethernet 0 address-serve netbios scope name 2-18
- show interface ethernet 0 address-serve range 2-16
- show interface ethernet *id* ip address 2-11
- show interface ethernet *id* ip filterset 2-13
- show interface ethernet *id* ip netbios proxy enable 2-12
- show interface ethernet *id* ip rip receive 2-13
- show interface ethernet *id* ip rip transmit 2-13
- show interface ethernet *id* pppoe enable 2-13
- show interface ethernet *id* statistics 2-13
- show interface ethernet *id* stats 2-13
- show interface ethernet *wan-id* ip nat enable 2-13
- show interface ethernet *wan-id* ip nat map-list 2-14
- show interface ethernet *wan-id* ip nat server-list 2-14
- show interface ethernet *wan-id* mac address 2-14
- show interface *intf-type id* dle 2-19
- show interface *intf-type id* statistics 2-19
- show interface *intf-type id* stats 2-19
- show interface isdn *id* dn 2-24
- show interface isdn *id* imux mode 2-21
- show interface isdn *id* mode 2-22
- show interface isdn *id* speed 2-23
- show interface isdn *id* spid 2-24
- show interface isdn *id* status 2-22
- show interface sdsl *id* clock rate 2-27
- show interface sdsl *id* clock source 2-26
- show interface sdsl *id* operation mode 2-28
- show interface sdsl *id* priority-queuing enable 2-29
- show interface sdsl *id* pvc 2-29, 2-31
- show interface sdsl *id* pvc { *id* | *tag* } cp 2-32
- show interface sdsl *id* pvc { *id* | *tag* } enable 2-31
- show interface sdsl *id* pvc { *id* | *tag* } tag 2-31
- show interface sdsl *id* pvc { *id* | *tag* } vci 2-31
- show interface sdsl *id* pvc { *id* | *tag* } vpi 2-31
- show interface sdsl *id* status 2-30
- show interface t1 *id* buildout 2-35
- show interface t1 *id* channels 2-35
- show interface t1 *id* clock source 2-35
- show interface t1 *id* diagnostic mode 2-38
- show interface t1 *id* dle 2-35
- show interface t1 *id* encoding 2-35
- show interface t1 *id* errors 2-37
- show interface t1 *id* framing 2-35
- show interface t1 *id* line status 2-39
- show interface t1 *id* loopback mode 2-39
- show interface t1 *id* loopback status 2-39
- show interface t1 *id* operation mode 2-36
- show interface t1 *id* prm-enable 2-36
- show interface t1 *id* rfc1973 dlci 2-36
- show interface t1 *id* rfc1973 enable 2-36
- show interface t1 *id* rfc1973 lmi 2-36
- show ip dns 2-40
- show ip domain-name 2-40
- show ip filterset 2-82
- show ip gateway 2-40
- show ip nat map 2-75
- show ip nat server 2-75
- show ip nat translations 2-76
- show ip route 2-41
- show macip static-range 2-77
- show model 2-70
- show preferences changes immediate 2-4
- show preferences console default 2-4
- show preferences console timeout 2-4
- show preferences date format 2-4
- show preferences output format 2-5
- show preferences output mask 2-5
- show preferences time format 2-5
- show radius-server 2-81
- show snmp system contact 2-6
- show snmp system location 2-6
- show snmp system name 2-6

- show system information 2-71
- show system web-server enable 2-6
- show system web-server lan-only 2-7
- show telnet server port 2-7
- show tftp last error 2-69
- show tftp status 2-69
- show time 2-7
- show version 2-71
- show voice coding 3-5
- show voice digit operator 3-5
- show voice digit outside 3-5
- show voice dn 3-3
- show voice log accounting 3-4
- show voice log accounting enable 3-4
- show voice log error 3-4
- show voice log error enable 3-4
- show voice log voice 3-4
- show voice log voice enable 3-4
- show voice overlap enable 3-5
- show voice payload statistics 3-6
- show voice port debug statistics 3-7
- show voice port hang-ups 3-6
- show voice ringcadence 3-5
- show voice tone 3-4
- show voice tonetable country 3-4
- show voice tonetable current 3-4
- show xmodem status 2-70
- snmp community 2-6
- snmp system contact 2-6
- snmp system location 2-6
- snmp system name 2-6
- system web-server enable 2-6
- system web-server lan-only 2-7

T

- telnet server port 2-7
- time 2-7

U

- user 2-7

V

- version 2-71

- voice coding 3-5
- voice digit operator 3-5
- voice digit outside 3-5
- voice dn 3-3
- voice log enable accounting 3-4
- voice log error enable 3-4
- voice log voice enable 3-4
- voice overlap enable 3-5
- voice ringcadence 3-5
- voice tone 3-4
- voice tonetable country 3-4