

SpeedTouch™

500Series

Multi-User ADSL Gateways

CLI Reference Guide

Release R4.2



THOMSON

speedtouch™

SpeedTouch™

500Series

CLI Reference Guide

Release R4.2

Status Released

Change Note PeckelbeenS

Short Title CLI ST500Series R4.2

Copyright

© 2003 THOMSON. All rights reserved. Passing on, and copying of this document, use and communication of its contents is not permitted without written authorization from THOMSON. The content of this document is furnished for informational use only, may be subject to change without notice, and should not be construed as a commitment by THOMSON. THOMSON assumes no responsibility or liability for any errors or inaccuracies that may appear in this document.

Content

About this Document	17
CLI Navigation	19
Accessing the Command Line Interface	20
Basic Navigation and Manipulation.....	21
Command Line Interface Top Level Structure	25
Command Line Interface Commands	26
Menu-driven CLI Navigation	27
Configuration Profile Files and the CLI.....	28
ADSL Commands	31
adsl config	32
adsl info	33
ATM Commands.....	35
atm aal5stats.....	36
atm gstats.....	37
atm portstats.....	38
atm vcstats.....	39
atm oam ccconfig.....	40
atm oam cclist.....	42
atm oam ccsend.....	43
atm oam config.....	45
atm oam mode.....	46
atm oam ping.....	47
atm oam status	48
AutoPVC Commands	49
autopvc config.....	50
autopvc info.....	52

Bridge Commands	53
bridge config.....	54
bridge flush	55
bridge ifadd	56
bridge ifattach.....	59
bridge ifconfig.....	62
bridge ifdelete	64
bridge ifdetach	65
bridge iflist	66
bridge macadd.....	68
bridge macdelete.....	69
bridge maclist.....	70
CIP Commands	71
cip flush.....	72
cip ifadd	73
cip ifdelete.....	74
cip iflist.....	75
cip pvcadd	76
cip pvcdelete	77
cip pvclist.....	78
Config Commands	79
config backup.....	80
config dump	81
config erase.....	82
config flush	83
config list	84
config load.....	85
config save.....	87

DHCP Commands	89
dhcp client clear	91
dhcp client config	92
dhcp client flush.....	93
dhcp client ifadd.....	94
dhcp client ifattach.....	95
dhcp client ifconfig.....	96
dhcp client ifdelete.....	98
dhcp client iflist.....	99
dhcp client ifrelease.....	100
dhcp client ifrenew	102
dhcp client stats.....	104
dhcp relay add.....	105
dhcp relay config	106
dhcp relay delete.....	107
dhcp relay flush.....	108
dhcp relay ifconfig.....	109
dhcp relay iflist.....	110
dhcp relay list.....	111
dhcp relay stats.....	112
dhcp server clear.....	113
dhcp server config.....	114
dhcp server flush.....	115
dhcp server policy.....	116
dhcp server stats	117
dhcp server lease add.....	119
dhcp server lease delete.....	121
dhcp server lease flush.....	122
dhcp server lease list.....	123
dhcp server pool add	124
dhcp server pool config.....	125
dhcp server pool delete.....	127
dhcp server pool flush	128
dhcp server pool list.....	129

DNS Commands 131

dns add 132

dns clear 133

dns clrstats 134

dns delete 135

dns domain 136

dns flush 137

dns fwdadd 138

dns fwddelete 139

dns fwdlist 140

dns fwdtable 141

dns list 142

dns nslookup 143

dns start 144

dns stats 145

dns status 146

dns stop 147

dns toutfwd 148

dns troff 149

dns tron 150

Env Commands 151

env flush 152

env get 154

env list 156

env set 157

env unset 158

Eth Commands 159

eth config 160

eth ifconfig 161

eth iflist 162

ETHoA Commands 163

ethoa flush..... 164
ethoa ifadd 165
ethoa ifattach..... 166
ethoa ifconfig..... 167
ethoa ifdelete 169
ethoa ifdetach..... 170
ethoa iflist..... 171

Firewall Commands..... 173

firewall assign 174
firewall flush..... 176
firewall list..... 177
firewall troff..... 178
firewall tron..... 179
firewall unassign..... 180
firewall chain create..... 181
firewall chain delete..... 182
firewall chain flush..... 183
firewall chain list..... 184
firewall rule clear..... 185
firewall rule create..... 186
firewall rule delete 190
firewall rule flush..... 191
firewall rule list..... 192
firewall rule stats..... 193

IP Commands 195

ip apadd	196
ip apdelete.....	198
ip apelist.....	199
ip arpadd	200
ip arpdelete.....	201
ip arplist.....	202
ip config	203
ip flush.....	206
ip ifconfig.....	207
ip iflist.....	208
ip ifwait	209
ip ping.....	210
ip rtadd	211
ip rtdelete	212
ip rtlist.....	213
ip sendto.....	214
ip traceroute	215
ip auto flush	217
ip auto ifadd.....	218
ip auto ifattach	219
ip auto ifconfig.....	220
ip auto ifdelete	221
ip auto ifdetach	222
ip auto iflist	223

IPoA Commands 225

ipoa flush	226
ipoa ifadd.....	227
ipoa ifattach	228
ipoa ifconfig.....	229
ipoa ifdelete	231
ipoa ifdetach	232
ipoa iflist	233

Label Commands 235

label add236

label config237

label delete.....238

label flush.....239

label list.....240

label troff.....241

label tron.....242

label chain create.....243

label chain delete.....244

label chain flush.....245

label chain list.....246

label rule clear.....247

label rule create.....248

label rule delete.....250

label rule flush.....251

label rule list.....252

label rule stats.....253

Language Commands 255

language config.....256

language list.....257

language remove.....258

NAT Commands	259
nat applist.....	260
nat bind.....	261
nat bindlist.....	264
nat clear.....	265
nat config.....	266
nat create.....	267
nat defserver	269
nat delete	270
nat disable	272
nat enable.....	273
nat flush	274
nat list	275
nat multinatadd.....	276
nat multinatdelete	277
nat multinatlist	278
nat unbind	279
Phonebook Commands	281
phonebook add.....	282
phonebook autolist.....	284
phonebook delete	285
phonebook flush.....	286
phonebook list.....	287
PPPoA Commands	289
pppoa flush.....	290
pppoa ifadd	291
pppoa ifattach.....	293
pppoa ifconfig.....	294
pppoa ifdelete.....	298
pppoa ifdetach.....	299
pppoa iflist.....	300
pppoa rtadd	301
pppoa rtdelete	303

PPPoE Commands.....	305
pppoe flush	306
pppoe ifadd	307
pppoe ifattach.....	309
pppoe ifconfig.....	311
pppoe ifdelete	316
pppoe ifdetach	317
pppoe iflist	318
pppoe ifscan.....	319
pppoe relay add.....	320
pppoe relay delete	321
pppoe relay flush	322
pppoe relay portlist.....	323
pppoe relay sesslist.....	324
pppoe rtadd.....	325
pppoe rtdelete	327
PPTP Commands.....	329
pptp ifadd	330
pptp flush.....	331
pptp list.....	332
pptp profadd.....	333
pptp profdelete.....	334
pptp proflist.....	335
QoSBook Commands	337
qosbook add.....	338
qosbook config	340
qosbook delete.....	341
qosbook flush.....	342
qosbook list	343
Script Commands	345

SNMP Commands.....	347
snmp config.....	348
snmp get.....	349
snmp getNext.....	350
snmp list.....	351
snmp walk.....	352
Software Commands.....	353
software version.....	354
software upgrade.....	355
Switch Commands.....	357
switch group flush.....	358
switch group list.....	359
switch group move.....	360
switch mirror capture.....	361
switch mirror egress.....	362
switch mirror ingress.....	363
System Commands.....	365
system clearpassword.....	366
system config.....	367
system flush.....	368
system reboot.....	369
system reset.....	370
system setpassword.....	371
system stats.....	372
Systemlog Commands.....	373
systemlog show.....	374
systemlog send.....	375
TD Commands.....	377
td call.....	378

UPnP Commands.....	379
upnp config	380
upnp flush.....	381
upnp list.....	382
 USB Commands.....	 383
usb add	384
usb config.....	385
usb delete.....	386
usb info	387
usb list.....	388
 Wireless Commands	 389
wireless aclconfig.....	390
wireless aclentry.....	391
wireless aclflush	392
wireless acllist	393
wireless counters	394
wireless flush	395
wireless ifconfig	396
wireless scan	397
wireless status.....	398
wireless wepconfig.....	399
 Abbreviations	 401

Syslog Messages.....	405
Auto-PVC Module	406
Configuration Module	406
DHCP Client Module	407
DHCP Relay Module	407
DHCP Server Module.....	408
Firewall Module.....	408
HTTP Module	408
Kernel Module	409
Linestate Module.....	409
Login Module.....	409
NAPT Module.....	409
PPP Module.....	410
PPTP Module.....	410
Routing Module	410
Software Module.....	411
UPnP Module	411
Supported Key Names	413
Supported Internet Protocol (IP) Protocol Names	413
Supported TCP/UDP Port Names.....	414
Supported ICMP Type Names	417
Supported Syslog Facilities.....	418
Supported Syslog Severities.....	419

About this Document

Introduction

Welcome to the SpeedTouch™500Series Command Line Interface (CLI) Reference Guide!

For the Service Provider, this Reference Guide aims to give a concise and practical document for creating a customized configuration profile file, based on the SpeedTouch™500Series CLI command set, to be used by the end-user to configure the SpeedTouch™500Series and PC(s).

For the fastidious user this Reference Guide aims to give a handbook for advanced and detailed configuration and troubleshooting of the SpeedTouch™500Series via its character based Command Line Interface.

This CLI Reference Guide covers the CLI commands of the following Digital Subscriber Line (DSL) SpeedTouch™ products:

- SpeedTouch™ 510 ADSL/POTS
- SpeedTouch™ 510i ADSL/ISDN
- SpeedTouch™ 530
- SpeedTouch™ 570 Wireless
- SpeedTouch™ 570i Wireless

Contents

The Reference Guide consists of two main parts:

- **CLI Navigation**
This chapter familiarises the user with the SpeedTouch™500Series CLI. It describes general manipulations to navigate through and to perform some operations on the CLI.
- **CLI Command Description**
The other chapters describe all the available CLI commands of the SpeedTouch™500Series per command group and in alphabetical order.
Each command is described in a systematic manner:
 - The full name of the CLI command (including the group selection)
 - A short description of the CLI command, if needed completed by a description of the possible impact on the user and/or the SpeedTouch™500Series
 - The syntax of the command with a description of each parameter
 - An example to demonstrate the use of the CLI command
 - A list of related CLI commands.

Trademarks

The following trademarks are used in this document:

- SpeedTouch™ is a trademark of THOMSON
- Netscape and Netscape Navigator are registered trademarks of Netscape Communications Corporation
- Windows and Internet Explorer are trademarks of Microsoft Corporation
- Apple and MacOS are registered trademarks of Apple Computer Inc.
- UNIX is a registered trademark of UNIX System Laboratories, Inc.
- Ethernet is a trademark of Xerox Corporation.

Other products may be trademarks or registered trademarks of their respective manufacturers.

Updates

Due to the continuous evolution of SpeedTouch™ technology, existing products are regularly upgraded.

For more information on the latest technological innovations, software upgrades, and documents, please visit the SpeedTouch™ web site at:

www.speedtouch.com

CLI Navigation

Contents

This chapter covers the following topics:

Topic	Page
Accessing the Command Line Interface	20
Basic Navigation and Manipulation	21
Command Line Interface Top Level Structure	25
Command Line Interface Commands	26
Menu-driven CLI Navigation	27
Configuration Profile Files and the CLI	28

Basic Navigation and Manipulation

Manipulation commands are commands that manipulate operations on the command line, for example changing the command group, go to the beginning of the command line, go to the end of the command line, etc.

Command group Navigation

From top level, you can change to a command group by executing the name of the desired command group.

To obtain a list of all available command groups, execute help from the top level.

The example below shows every possible CLI command group for the SpeedTouch™510.

```
=>help
Following commands are available :
help          : Displays this help information
menu          : Displays menu
?             : Displays this help information
exit          : Exits this shell.
..            : Exits group selection.
saveall       : Saves current configuration.

Following command groups are available :

ads1          atm          autopvc       bridge        cip
config        dhcp          dns           env           eth
ethoa         firewall     ip            ipoa          label
language      nat          phonebook    pppoa         pppoe
pptp          qosbook     script        snmp          software
switch        system      systemlog    td            upnp
=>
```

The following command groups are specific for certain variants:

- usb is only available for the SpeedTouch™530
- wireless is only available for the SpeedTouch™570

To return to top level, or to descend one level (in case of nested command groups) execute ..

EXAMPLE:

```
=>phonebook
[phonebook]=>
[phonebook]=>..
=>
```

The Help Command

Execute help or ? from top level to list all available command groups for the SpeedTouch™ 500Series.

EXAMPLE for the SpeedTouch™ 510:

```
=>help
Following commands are available :

help          : Displays this help information
menu          : Displays menu
?             : Displays this help information
exit          : Exits this shell.
..           : Exits group selection.
saveall       : Saves current configuration.

Following command groups are available :

ads1          atm          autopvc        bridge         cip
config        dhcp          dns            env            eth
ethoa         firewall      ip            ipoa           label
language      nat           password      phonebook     pppoa
pppoe         pptp          qosbook       script         snmp
software      system       systemlog     td            upnp

=>
```

You can execute the help or ? command from each command group selection. This results in a list of the available commands (and nested command groups, if available) in this particular command group.

EXAMPLE:

```
=>firewall
[firewall]=>
[firewall]=>?
Following commands are available :

tron          : Enables verbose console messaging.
troff         : Disables verbose console messaging.
match         : Defines an ip packet match.
assign        : Assign a chain to an entry point.
list          : Shows a list of all the hooks with the chain attached.
flush         : Clears all hooks. If a hook is provided, that hook is cleared.

Following command groups are available :

chain         rule

[firewall]=>
```

As both help and ? have the exact same functionality in the SpeedTouch™ 500Series CLI, the help command may always be equally replaced by the ? command.

Executing e.g. help firewall from top level gives the same result as executing help from the firewall command group selection.

EXAMPLE:

```
=>firewall help
Following commands are available :

assign          : Assign a chain to an entry point.
unassign        : Clear a specific hook
list            : Shows a list of all the hooks with the chains attached.
flush           : Clears all hooks, chains and rules
tron           : Enables verbose console messaging.
troff           : Disables verbose console messaging.

Following command groups are available :

chain          rule

=>
```

Entering help followed by a specific command, e.g. help firewall assign (starting from top level) or help assign (executed from the firewall command group selection) results in a description of the syntax for the command.

EXAMPLE:

```
=>help firewall assign
Assign a chain to an entry point.
Syntax : assign hook = <{input|sink|forward|source|output}> chain = <string>

Parameters :
  hook = <{input|sink|forward|source|output}>
        Name of hook to assign chain to.
  chain = <string>
        Name of chain to use.

=>
```

Executing help all will generate the complete listing of all available CLI commands with syntax description.

Command Completion

The CLI features command completion, which means that when starting to enter a command it can be completed by pressing the TAB key.

For the completion to be successful, the part to be added must be unique. Completion works for the command groups, for the commands, for the options, but not for values.

For example, pressing a and TAB at the firewall command group selection results in the full assign command being completed. Entering firewall a and pressing the TAB key from top level gives the same result.

EXAMPLE:

```
=>firewall
[firewall]=>"a+TAB"
[firewall]=>assign
```

Going to the beginning or end of the Command Line

To go to ...	Press ...
the beginning of the Command Line	CTRL+A
the end of the Command Line	CTRL+E

Breaking off Commands

Press CTRL+G to break off a command. This can be useful in a situation where a user is prompted to enter a value which he does not know and wants to quit the command. Instead of being prompted over and over again for the same value, this allows to quit the command and return to the command line prompt.

In the example below CTRL+G is pressed after the third prompt chain =

```
[firewall]=>match
chain =
chain =
chain = "CTRL+G"
[firewall]=>
```

History of Commands

Use the UP and DOWN ARROW keys to select a previously executed command. Press ENTER to execute the selected command.

EXAMPLE:

```
=>firewall
[firewall]=>list
assign    hook=input chain=input
assign    hook=sink chain=sink
assign    hook=forward chain=forward
assign    hook=source chain=source
[firewall]=> "UP ARROW"
[firewall]=>:firewall list
```


Command Line Interface Top Level Structure

The table below shows the available command groups for the different SpeedTouch™500Series variants:

Command group	Availability
adsl	Only applicable for the SpeedTouch™510 and SpeedTouch™570 ADSL/POTS, and the Speed-Touch™510i and SpeedTouch™570i ADSL/ISDN variants.
atm	Always
autopvc	Always
bridge	Always
cip	Always
config	Always
dhcp	Always
dns	Always
env	Always
eth	Always
ethoa	Always
firewall	Always
ip	Always
ipoa	Always
label	Always
language	Always
nat	Always
phonebook	Always
pppoa	Always
pppoe	Always
pptp	Always
qosbook	Always
script	Always
snmp	Always
software	Always
switch	Only applicable for SpeedTouch™ devices with a four port switch.
system	Always
systemlog	Always
td	Always
upnp	Always
usb	Only applicable for the SpeedTouch™530 variant.
wireless	Only applicable for the SpeedTouch™570 Wireless variant.

Command Line Interface Commands

All CLI commands are commands that operate on, or configure, the SpeedTouch™ 500Series settings.

You can execute these commands from top level, preceded by the name of the command group from which the command should be executed (e. g. firewall list).

You can also execute the commands from the command group itself, using the reduced form of the command (e.g. list at the firewall command group selection).

! in a command means 'NOT', e.g. the [!]syn parameter in the firewall rule create command.

EXAMPLE:

```
=>firewall list
assign hook=input chain=input
assign hook=sink chain=sink
assign hook=forward chain=forward
assign hook=source chain=source
=>firewall
[firewall]=>list
assign hook=input chain=input
assign hook=sink chain=sink
assign hook=forward chain=forward
assign hook=source chain=source
[firewall]=>
```

Instead of entering a completely built-up command with all its parameters, you can also enter just the command itself, without its parameters. After this you are prompted to complete the command with the required and the optional parameters. For the optional parameters you can simply press enter without giving a value.

The example below is the equivalent of firewall assign hook=input chain=input. To break of such incomplete command press CTRL+G.

EXAMPLE:

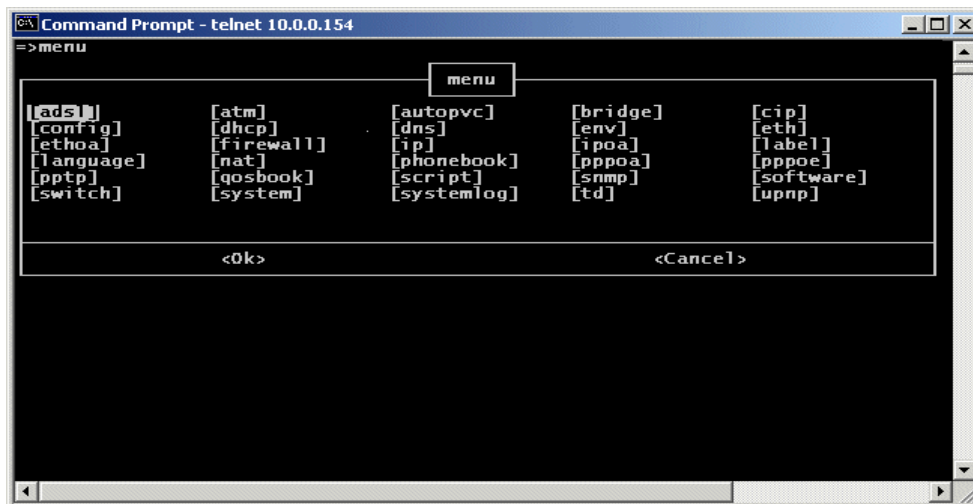
```
=>firewall assign
hook = input
chain = input
=>
```

Menu-driven CLI Navigation

To improve the user-friendliness of the SpeedTouch™ 500Series CLI, the CLI features a menu-driven interface.

To enter the menu-driven interface, simply execute menu from the CLI prompt:

EXAMPLE (for the SpeedTouch™ 510):



The semi-graphical CLI offers you an attractive and easy-to-use configuration environment for the CLI.

You can browse through the CLI command groups via the ARROW keys. Press ENTER to execute your selection. From each level you can select '..' to go back one level up.

Use the TAB key to change from the CLI command menu to the control menu and vice versa.

To setup a CLI command, simply press ENTER on its name. You can configure and overview its various parameters at one time. In case the parameter provides preset values, you can go through these via the ARROW keys.

If you are satisfied, use the TAB key to go to the OK field and press ENTER.

Note Do not forget to save your changes by executing saveall (from any CLI prompt).

Configuration Profile Files and the CLI

Configuration Profile Files

The configuration profile files, used by the Setup wizard, or directly uploaded via the SpeedTouch™500Series web pages, consist of a set of CLI commands. Only CLI commands, which are part of the SpeedTouch™500Series CLI command set may be used in the .ini and .def files. For readability, configuration profile files are partitioned in paragraphs. These paragraphs reflect a CLI command group selection in the SpeedTouch™500Series CLI. However, due to internal processing, not all paragraph names are the same as their corresponding CLI command group selection.

In the following overview the relationship between the available SpeedTouch™500Series CLI commands and the [<configuration>.ini] paragraphs is given:

[<configuration>.ini] paragraphs	Available SpeedTouch™500Series CLI commands
[adslisdn.ini]	The adsl CLI command group in case of a ADSL/ISDN variant
[adslpots.ini]	The adsl CLI command group in case of a ADSL/POTS variant
[autoip.ini]	The ip auto CLI command subgroup
[autopvc.ini]	The autopvc CLI command subgroup
[bridge.ini]	The bridge CLI command group
[cip.ini]	The cip CLI command group
[dhcc.ini]	The dhcp client CLI command subgroup
[dhcp.ini]	The dhcp server CLI command subgroup
[dhcr.ini]	The dhcp relay CLI command subgroup
[dnsd.ini]	The dns CLI command group
[env.ini]	The env CLI command group
[eth.ini]	The eth CLI command group
[ethoa.ini]	The ethoa CLI command group
[ip.ini]	The ip CLI command group
[ipoa.ini]	The cip CLI command group
[label.ini]	The label CLI command group
[language.ini]	The language CLI command group
[nat.ini]	The nat CLI command group
[oam.ini]	The atm oam CLI command subgroup
[pfilter.ini]	<ul style="list-style-type: none"> • The firewall chain CLI command subgroup • The firewall rule CLI command subgroup
[pfirewall.ini]	The firewall CLI command group (except those part of the firewall chain and firewall rule CLI command subgroups).

[<configuration>.ini] paragraphs	Available SpeedTouch™500Series CLI commands
[phone.ini]	The phonebook CLI command group
[ppp.ini]	The ppp CLI command group
[pppoa.ini]	The pppoa CLI command group
[pppoe.ini]	The pppoe CLI command group
[pppoerelay.ini]	The pppoe relay CLI command subgroup
[pptp.ini]	The pptp CLI command group
[qos.ini]	The qosbook CLI command group
[script.ini]	The script CLI command group
[snmp.ini]	The snmp CLI command group
[system.ini]	The system CLI command group
[upnp.ini]	The upnp CLI command group
[usb.ini]	The usb CLI command group
[wireless.ini]	The wireless CLI command group

CLI Commands in Configuration Profile Files

CLI commands in a paragraph of a configuration profile file should always be constructed in their complete form. Uncompleted CLI commands, i.e. commands in which required parameters are not specified, will be discarded by the CLI command interpreter. This may result in a wrongly configured SpeedTouch™500Series.

In configuration profile files, the use of customization variables, allow the Setup wizard to invite the end-user to provide some input regarding the settings of the SpeedTouch™500Series. The declaration of such variables must be done in the [env.ini] paragraph. Further use of the resulting variable TAG is allowed through all other paragraphs, even multiple times.

To make sure that a variable always will result in a valid variable TAG, the [env.ini] paragraph also allows to declare a preset value for the variable.

In case the variable TAG is used in a CLI command, the value of the variable should always be in conformity with the syntax of the CLI command.

Customizing Configuration Profile Files

For more information on the customization possibilities of the SpeedTouch™500Series, the Setup wizard and the configuration profile files, please check the SpeedTouch™ support pages at:

www.speedtouch.com

ADSL Commands

Contents

This chapter covers the following commands:

Topic	Page
adsl config	32
adsl info	33

adsl config

Show/set the Asymmetric Digital Subscriber Line (ADSL) configuration.

Although the same command for both SpeedTouch™ ADSL/POTS and SpeedTouch™ ADSL/ISDN variants, the command features specific parameter values per variant:

SYNTAX for ADSL/Plain Old Telephone Service (POTS) variants:

```
adsl config [opermode = <{ansi|g.dmt_annex_a|g.lite|multimode}>]
            [maxbitspertoneUS = <number{10-14}>]
            [trace = <{off|on}>]
```

opermode	The operational mode of the SpeedTouch™ modem. Choose between: <ul style="list-style-type: none"> • ansi • g.dmt_annex_a • g.lite • multimode By default the SpeedTouch™ will start in multimode.	OPTIONAL
maxbitspertoneUS	A number between 10 and 14 (bits per tone). Represents the maximum number of bits which can be allocated to each ADSL DMT tone in the upstream direction. By default the modem will use up to 13 bits per tone.	OPTIONAL
trace	Enable/Disable adsl tracing	OPTIONAL

SYNTAX for ADSL/Integrated Services Digital Network (ISDN) variants:

```
adsl config [opermode = <{etsi|g.dmt_annex_b|multimode}>]
            [maxbitspertoneUS = <number{10-14}>] [trace = <{off|on}>]
```

opermode	The operational mode of the SpeedTouch™ modem. Choose between: <ul style="list-style-type: none"> • etsi • g.dmt_annex_b • multimode By default the SpeedTouch™ will start in multimode.	OPTIONAL
maxbitspertoneUS	A number between 10 and 14 (bits per tone). Represents the maximum number of bits which can be allocated to each ADSL DMT tone in the upstream direction. By default the modem will use up to 13 bits per tone.	OPTIONAL
trace	Enable/Disable adsl tracing	OPTIONAL

Note Executing the :adsl config without specifying parameters, shows the current ADSL configuration.

adsl info

Show ADSL statistics and information about the SpeedTouch™ DSL line status.

Although the same command for both SpeedTouch™ ADSL/POTS and SpeedTouch™ ADSL/ISDN variants, the command features specific output parameters and counters per variant:

SYNTAX:

```
adsl info
```

EXAMPLE (for a SpeedTouch™500Series ADSL/POTS variant):

```
=>adsl info
Modemstate           : up
Operation Mode       : G.DMT Annex A [POTS Overlay Mode]
Channel Mode         : fast
Number of resets     : 1

Vendor (ITU)         : Local      Remote
Country              : 0f         0f
Vendor               : ALCB       ALCB
VendorSpecific       : 0000      0000
StandardRevisionNr  : 01         01

Margin [dB]          : Downstream  Upstream
Attenuation [dB]     : 25         31
                    : 26         15

Available Bandwidth  : Cells/s    Kbit/s
Downstream           : 7924      3360
Upstream             : 452       192

Transfer statistics
Total since power on : Cells      Kbit
Downstream           : 10153    4304
Upstream             : 3399     1441

Current Connection
Downstream           : 10153    4304
Upstream             : 3399     1441

Errors
Received FEC        : 0
Received CRC        : 0
Received HEC        : 0
```

.. Continued output on following page ..

.. Continued output ..

```
Far End Failure
No Failure
Near end failure
No failure
Far end failures since reset
Loss of frame:      0 failures
Loss of signal:    0 failures
Loss of power:     0 failures
Loss of link:      0 failures
Errored seconds:   0 seconds
Far end failures last 15 minutes
Loss of frame:     0 seconds
Loss of signal:    0 seconds
Loss of power:     0 seconds
Loss of link:      0 seconds
Errored seconds:   0 seconds
Far end failures current day
Errored seconds:   0 seconds
Far end failures previous day
Errored seconds:   0 seconds
Near end failures since reset
Loss of frame:     0 failures
Loss of signal:    0 failures
Loss of power:     0 failures
Errored seconds:   0 seconds
Near end failures last 15 minutes
Loss of frame:     0 seconds
Loss of signal:    0 seconds
Loss of power:     0 seconds
Errored seconds:   0 seconds
Near end failures current day
Errored seconds:   0 seconds
Near end failures previous day
Errored seconds:   0 seconds
```

=>

ATM Commands

Contents

This chapter covers the following commands:

Topic	Page
atm aal5stats	36
atm gstats	37
atm portstats	38
atm vcstats	39
atm oam ccconfig	40
atm oam cclist	42
atm oam ccsend	43
atm oam config	45
atm oam mode	46
atm oam ping	47
atm oam status	48

atm aal5stats

Display AAL5 port specific Asynchronous Transfer Mode (ATM) statistics

SYNTAX:

```
atm aal5stats      port = <{dsl0|dsl1|atm2|atm3|aal5|atm5} or number>
                   vpi = <number{0-15}>
                   [vci = <number{0-511}>]
                   [reset = <{no|yes}>]
```

port	<p>Port for which statistics will be retrieved. Choose between:</p> <ul style="list-style-type: none"> • DSL0 • DSL1 • ATM2 • ATM3 • AAL5 • ATM5 <p>Or specify a port number (dsl0 has port number 0, ..., atm5 has port number 5).</p> <p>The parameter DSL0 is replaced by the parameter ADSL0 for the SpeedTouch™570.</p>	REQUIRED
vpi	<p>A number between 0 and 15. Represents the VPI number for which statistics will be retrieved.</p>	REQUIRED
vci	<p>A number between 0 and 511. Represents the VCI number for which statistics will be retrieved.</p>	OPTIONAL
reset	<p>Reset the statistics after request</p>	OPTIONAL

atm gstats

Display ATM global statistics.

SYNTAX:

```
atm gstats [reset = <{no|yes}>]
```

reset	Reset the statistics after request	OPTIONAL
-------	------------------------------------	----------

Note The command :atm portstats is replaced by the command :atmstats portstats for the SpeedTouch™570. The command :atmstats portstats is the only ATM command for the SpeedTouch™570.

EXAMPLE:

```
=>atm gstats
# of received octets = 170554.
# of transmitted octets = 519665.
# of received cells = 3218.
# of transmitted cells = 9805.
# of unknown cells = 0.
# of errors on the input = 0.
# of errors on output = 0.
=>
```

atm portstats

Show port specific ATM statistics.

SYNTAX:

```
atm portstats      port = <{dsl0|dsl1|atm2|atm3|aal5|atm5} or number>
                   [reset = <{no|yes}>]
```

port	Port for which statistics will be retrieved. Choose between: <ul style="list-style-type: none"> • DSL0 • DSL1 • ATM2 • ATM3 • AAL5 • ATM5 Or specify a port number (dsl0 has port number 0, ..., atm5 has port number 5). The parameter DSL0 is replaced by the parameter ADSL0 for the SpeedTouch™570.	REQUIRED
reset	Reset the statistics after request.	OPTIONAL

EXAMPLE:

```
=>atm portstats port=DSL0
      # of received octets = 547649.
      # of transmitted octets = 191648.
      # of received cells = 10333.
      # of transmitted cells = 3616.
      # of unknown cells = 0.
      # of errors on the input = 0.
      # of errors on output = 0.
=>
```

atm vcstats

Display VC specific atm statistics

SYNTAX:

```
atm vcstats      port = <{dsl0|dsl1|atm2|atm3|aal5|atm5} or number>
                  vpi = <number{0-15}>
                  [vci = <number{0-511}>]
                  [reset = <{no|yes}>]
```

port	Port for which statistics will be retrieved. Choose between: <ul style="list-style-type: none"> • DSL0 • DSL1 • ATM2 • ATM3 • AAL5 • ATM5 Or specify a port number (dsl0 has port number 0, ..., atm5 has port number 5). The parameter DSL0 is replaced by the parameter ADSL0 for the SpeedTouch™570.	REQUIRED
vpi	A number between 0 and 15. Represents the VPI number for which statistics will be retrieved.	REQUIRED
vci	A number between 0 and 511. Represents the VCI number for which statistics will be retrieved.	OPTIONAL
reset	Reset the statistics after request.	OPTIONAL

atm oam ccconfig

Configure Operation and Maintenance (OAM) Continuity Check (CC) on the connection.

SYNTAX:

```
atm oam ccconfig    port = <{dsl0|dsl1|atm2|atm3|aal5|atm5} or number>
                   vpi = <number{0-15}>
                   [vci = <number{0-511}>]
                   [transmit = <{disabled|enabled}>]
                   [receive = <{disabled|enabled}>]
                   [auto = <{disabled|enabled}>]
                   [span = <{segment|end2end}>]
```

port	<p>ATM port number. Choose between:</p> <ul style="list-style-type: none"> • DSL0 • DSL1 • ATM2 • ATM3 • AAL5 • ATM5 <p>Or specify a port number (dsl0 has port number 0, ..., atm5 has port number 5).</p> <p>The parameter DSL0 is replaced by the parameter ADSL0 for the SpeedTouch™570.</p>	REQUIRED
vpi	<p>A number between 0 and 15. Represents the Virtual Path Identifier (VPI)</p>	REQUIRED
vci	<p>A number between 0 and 511. Represents the Virtual Channel identifier (VCI). Use VCI=0, or do not specify for a VP cross-connection.</p>	OPTIONAL
transmit	<p>Enable (enabled) or disable (disabled) transmission of CC cells. By default transmission of CC cells is disabled.</p>	OPTIONAL
receive	<p>Enable (enabled) or disable (disabled) loss of continuity. By default loss of continuity is disabled.</p>	OPTIONAL
auto	<p>Enable (enabled) or disable (disabled) remote CC activation and deactivation. By default remote CC activation/deactivation is disabled.</p>	OPTIONAL
span	<p>End2end or segment continuity check</p>	OPTIONAL

RELATED COMMANDS:

`atm oam cclist` Show current CC configuration.

atm oam ccsend

Send CC activate/deactivate to connection.

atm oam cclist

Show current OAM CC configuration

SYNTAX:

```
atm oam cclist
```

EXAMPLE (default configuration):

```
=>atm oam cclist
PORT = 0 VPI = 15 VCI = 64 End2End Mode = Auto Segment Mode = Auto
PORT = 0 VPI = 15 VCI = 16 End2End Mode = Auto Segment Mode = Auto
PORT = 0 VPI = 0 VCI = 16 End2End Mode = Auto Segment Mode = Auto
=>
```

RELATED COMMANDS:

`atm oam ccconfig`

Configure CC on the connection.

`atm oam ccsend`

Send CC activate/deactivate to connection.

atm oam ccsend

Send CC activate/deactivate to connection.

SYNTAX:

```
atm oam ccsend      port = <{dsl0|dsl1|atm2|atm3|aal5|atm5} or number>
                    vpi = <number{0-15}>
                    [vci = <number{0-511}>]
                    [span = <{segment|end2end}>]
                    [action = <{activate|deactivate}>]
                    [direction = <{source|sink|both}>]
```

port	<p>ATM port number. Choose between:</p> <ul style="list-style-type: none"> • DSL0 • DSL1 • ATM2 • ATM3 • AAL5 • ATM5 <p>Or specify a port number (dsl0 has port number 0, ..., atm5 has port number 5).</p> <p>The parameter DSL0 is replaced by the parameter ADSL0 for the SpeedTouch™570.</p>	REQUIRED
vpi	<p>A number between 0 and 15. Represents the Virtual Path Identifier (VPI)</p>	REQUIRED
vci	<p>A number between 0 and 511. Represents the Virtual Channel identifier (VCI). Use VCI=0, or do not specify for a VP cross-connection.</p>	OPTIONAL
span	<p>End2end or segment continuity check</p>	OPTIONAL
action	<p>Enable or disable (default) CC.</p>	OPTIONAL
direction	<p>Indicates the direction of CC activity Select either:</p> <ul style="list-style-type: none"> • source • sink • both. <p>If not specified, CC is activated/deactivated for both directions.</p>	OPTIONAL

RELATED COMMANDS:

`atm oam ccconfig` Configure CC on the connection.

`atm oam cclist`

Show current CC configuration.

atm oam config

Configure OAM cell settings.

SYNTAX:

```
atm oam config      [clp = <number{0-1}>]
                   [loopbackid = <string>]
```

clp	The CLP bit value of the OAM cells. Select either: <ul style="list-style-type: none">• 0• 1	OPTIONAL
loopbackid	The loopback id (hexadecimal string) for processing of segment loopback cells. Default is 6a 6a 6a 6a 6a 6a 6a 6a 6a 6a 6a 6a 6a 6a 6a.	OPTIONAL

atm oam mode

Configure the OAM data blocking mode.

SYNTAX:

atm oam mode	port = <{ds10 ds11 atm2 atm3 aa15 atm5} or number> blocking = <{disabled enabled}>
--------------	---

port	Port for which OAM blocking is configured. Choose between: <ul style="list-style-type: none">• DSL0• DSL1• ATM2• ATM3• AAL5• ATM5 Or specify a port number (dsl0 has port number 0, ..., atm5 has port number 5).	REQUIRED
blocking	Enable (enabled) or disable (disabled) the OAM data blocking mode on this port. By default OAM data blocking is enabled on all ports.	REQUIRED

atm oam ping

Sends ATM loopback cells.

SYNTAX:

```
atm oam ping      dest = <string>
                  [count = <number{1-1000000}>]
                  [interval = <number{100-1000000}>]
```

dest	The destination address for the request. Can be any phonebook entry.	REQUIRED
count	A number between 1 and 1000000. Represents the number of pings to send.	OPTIONAL
interval	A number between 1 and 1000000. Represents the interval in milliseconds between packets.	OPTIONAL

EXAMPLE:

```
=>atm oam ping dest=Sascha count=10 interval=200
loopback: successful, sequence: 1 time: 7762 usec
loopback: successful, sequence: 2 time: 8239 usec
loopback: successful, sequence: 3 time: 11100 usec
loopback: successful, sequence: 4 time: 9384 usec
loopback: successful, sequence: 5 time: 7209 usec
loopback: successful, sequence: 6 time: 20008 usec
loopback: successful, sequence: 7 time: 9651 usec
loopback: successful, sequence: 8 time: 9593 usec
loopback: successful, sequence: 9 time: 8411 usec
loopback: successful, sequence: 10 time: 41656 usec

--- loopback statistics ---
10 loopbacks transmitted, 10 successful, 0% loss, time 600 ms
rtt min/avg/max = 7209/13301/41656
=>
```

atm oam status

Show OAM data blocking mode of all ports.

SYNTAX:

```
atm oam status
```

EXAMPLE:

```
=>atm oam status
  OAM config dump
-----
      CLP bit value : 1
      Loopback id   : 6a 6a 6a 6a 6a 6a 6a 6a 6a 6a 6a 6a 6a 6a 6a
OAM data blocking mode
-----
      Port ds10: blocking
      Port ds11: blocking
      Port atm2: blocking
      Port atm3: blocking
      Port aa15: blocking
      Port atm5: blocking
=>
```

AutoPVC Commands

Contents

This chapter covers the following commands:

Topic	Page
autopvc config	50
autopvc info	52

autopvc config

Configure autopvc

SYNTAX:

```
autopvc config [mode = <{pseudo|passive|active}>]
               [type = <{bridge|pppoerelay}>]
               [opmode = <{partial|full}>]
               [overwrite = <{disabled|enabled}>]
               [peakrate = <number{0-27786}>]
```

mode	<p>Select the autopvc mode:</p> <ul style="list-style-type: none"> • pseudo: enable only pseudo-ILMI (VP/VC 15/16). When connection parameters are written to the MIB, display this information on CLI or web-interface but do not use these parameters for configuration. • passive: enable both ILMI (VP/VC 0/16) and pseudo-ILMI (VP/VC 15/16). When connection parameters are written to the MIB, display this information on CLI or web-interface but do not use these parameters for configuration. (default) • active: enable both ILMI (VP/VC 0/16) and pseudo-ILMI (VP/VC 15/16). When connection parameters are written to the MIB, use these parameters to configure phonebook entries, qosbook profiles and bind bridge or PPPoE interfaces on top. 	OPTIONAL
type	<p>Type of autopvc:</p> <ul style="list-style-type: none"> • bridge (default) • pppoerelay: an ETHoA interface will be created, will be bound to the ILMI PVC and will be added to the PPPoE relay as relay port. 	OPTIONAL
opmode	<p>Operational autopvc mode:</p> <ul style="list-style-type: none"> • partial: only read the ILMI VPC and VCC MIB tables (default) • full: read all supported MIB tables 	OPTIONAL
overwrite	<p>Enable/disable UBR peak rate overwrite.</p>	OPTIONAL
peakrate	<p>A number between 0 and 27786. Represents the UBR peak rate (in kilobits per second).</p> <p>Note 0 indicates the linerate</p>	OPTIONAL

EXAMPLE (default configuration):

```
=>autopvc config
Autopvc mode   : pseudo
Autopvc opmode : partial
Autopvc type   : bridge
Autopvc standard: down
Autopvc pseudo : up
UBR overwrite  : disabled
UBR peak rate  : linerate
=>
```

RELATED COMMANDS:

`autopvc info` Show retrieved information.

autopvc info

Show retrieved information.

SYNTAX:

```
autopvc info [table = <{Port|AtmLayer|Vpc|Vcc|Address|
AddressRegistrationAdmin|AtmServiceType|
AtmServiceConnectionInfo|AAL1Profile|
AAL34Profile|AAL5Profile|AAL2CommonProfile|
AAL2TrunkingProfile|AAL2LESPProfile}>]
```

table Choose the autopvc table for which the information must be shown. **OPTIONAL**
 Choose between:

- Port
- AtmLayer
- Vpc
- Vcc
- Address
- AddressRegistrationAdmin
- AtmServiceType
- AtmServiceConnectionInfo
- AAL1Profile
- AAL34Profile
- AAL5Profile
- AAL2CommonProfile
- AAL2TrunkingProfile
- AAL2LESPProfile

EXAMPLE (default configuration):

```
=>autopvc info
Address Type BestEff Par1 Par2 Par3 Par4 Par5
8.35 ubr Enabled Tx: 451 0 0 0 0
Rx: 7923 0 0 0 0
=>
```

RELATED COMMANDS:

autopvc config Configure autopvc

Bridge Commands

Contents

This chapter covers the following commands:

Topic	Page
bridge config	54
bridge flush	55
bridge ifadd	56
bridge ifattach	59
bridge ifconfig	62
bridge ifdelete	64
bridge ifdetach	65
bridge iflist	66
bridge macadd	68
bridge macdelete	69
bridge maclist	70

bridge config

Show/set bridge ageing policy for dynamically learned Medium Access Control (MAC) addresses.

SYNTAX:

```
bridge config [age = <number {10 - 100000}>]
              [filter = <{no_WAN_broadcast|PPPoE_only|none}>]
```

age	A number between 10 and 100000 (seconds). Represents the lifetime of a dynamically learned MAC address. By default the ageing timer is 300 seconds.	OPTIONAL
filter	The bridge filter to be applied for all WAN bridge ports. Choose between: <ul style="list-style-type: none"> • no_WAN_broadcast (default): broadcasts from the SpeedTouch™ itself to the WAN are filtered out, broadcasts from the LAN to the WAN are still passed through. • PPPoE_only: only PPPoE broadcasts are passed through, all others are filtered out • none: no broadcasts are filtered out. 	OPTIONAL

EXAMPLE:

```
=>bridge config
Ageing : 300
Filter : no_WAN_broadcast
=>bridge config age=600
=>bridge config
Ageing : 600
Filter : no_WAN_broadcast
=>
```

RELATED COMMANDS:

bridge ifadd	Create a bridged Ethernet interface.
bridge ifattach	Attach a bridge interface.
bridge ifdelete	Delete a bridge interface.
bridge ifdetach	Detach a bridge interface.
bridge iflist	Show current bridge configuration.

bridge flush

Flush bridge interfaces and parameters.

Note The flush command does not impact previously saved configurations.

SYNTAX:

```
bridge flush
```

bridge ifadd

Create a bridged Ethernet interface.

SYNTAX:

```
bridge ifadd      [intf = <intfname>]
                  [dest = <string>]
```

intf	The bridged Ethernet interface name. If not specified, the destination name will double as interface name.	OPTIONAL
dest	The destination address for the new interface. Typically a phonebook entry. Browse through the available entries via the ARROW UP and ARROW DOWN keys.	OPTIONAL

EXAMPLE:

```
=>bridge iflist
OBC      : Internal
          Connection State: connected Port: OBC PortState: forwarding
          RX bytes: 75783 frames: 572
          TX bytes: 82768372 frames: 341221 dropframes: 0
eth0     : Internal
          Connection State: connected Port: eth0 PortState: forwarding
          RX bytes: 156344216 frames: 5899238
          TX bytes: 75689 frames: 425 dropframes: 5558017
usb_bridge : dest : usb_port
          Retry : 10 QoS : default Encaps : llc/snap Fcs : off
          Connection State: connected
          Port: wan2 PortNr: 4 PortState: forwarding
          RX bytes: 0 frames: 0
          TX bytes: 0 frames: 0 dropframes: 0
=>bridge ifadd intf=TestBridge dest=Br1
=>bridge iflist
OBC      : Internal
          Connection State: connected Port: OBC PortState: forwarding
          RX bytes: 75783 frames: 572
          TX bytes: 82843610 frames: 341554 dropframes: 0
eth0     : Internal
          Connection State: connected Port: eth0 PortState: forwarding
          RX bytes: 156472129 frames: 5903256
          TX bytes: 75689 frames: 425 dropframes: 5561702
usb_bridge : dest : usb_port
          Retry : 10 QoS : default Encaps : llc/snap Fcs : off
          Connection State: connected
          Port: wan2 PortNr: 4 PortState: forwarding
          RX bytes: 0 frames: 0
          TX bytes: 0 frames: 0 dropframes: 0
TestBridge : dest : Br1
          Retry: 10 QoS: default Encaps: llc/snap Fcs: off
          Connection State: not-connected Port: (Unassigned) PortState: forwarding
=>
```

RELATED COMMANDS:

bridge ifattach	Attach a bridge interface.
bridge ifconfig	Configure a bridge interface.
bridge ifdelete	Delete a bridge interface.

`bridge ifdetach`

Detach a bridge interface.

`bridge iflist`

Show current bridge configuration.

bridge ifattach

Attach (i.e. connect) a bridged Ethernet interface.

SYNTAX:

```
bridge ifattach      intf = <intfname>
```

intf	The name of the interface to attach. Browse through the available entries via the ARROW UP and ARROW DOWN keys.	REQUIRED
------	--	----------

EXAMPLE:

```
=>bridge iflist
OBC      : Internal
          Connection State: connected Port: OBC PortState: forwarding
          RX bytes: 75783 frames: 572
          TX bytes: 82843610 frames: 341554 dropframes: 0
eth0     : Internal
          Connection State: connected Port: eth0 PortState: forwarding
          RX bytes: 156472129 frames: 5903256
          TX bytes: 75689 frames: 425 dropframes: 5561702
usb_bridge : dest : usb_port
          Retry : 10 QoS : default Encaps : llc/snap Fcs : off
          Connection State: connected
          Port: wan2 PortNr: 4 PortState: forwarding
          RX bytes: 0 frames: 0
          TX bytes: 0 frames: 0 dropframes: 0
TestBridge : dest : Br1
          Retry: 10 QoS: default Encaps: llc/snap Fcs: off
          Connection State: not-connected Port: (Unassigned) PortState: forwarding
=>bridge ifattach intf=TestBridge
=>bridge iflist
OBC      : Internal
          Connection State: connected Port: OBC PortState: forwarding
          RX bytes: 75783 frames: 572
          TX bytes: 82843610 frames: 341554 dropframes: 0
eth0     : Internal
          Connection State: connected Port: eth0 PortState: forwarding
          RX bytes: 156472129 frames: 5903256
          TX bytes: 75689 frames: 425 dropframes: 5561702
usb_bridge : dest : usb_port
          Retry : 10 QoS : default Encaps : llc/snap Fcs : off
          Connection State: connected
          Port: wan2 PortNr: 4 PortState: forwarding
          RX bytes: 0 frames: 0
          TX bytes: 0 frames: 0 dropframes: 0
TestBridge : dest : Br1
          Retry: 10 QoS: default Encaps: llc/snap Fcs: off
          Connection State: connected Port: wan0 PortState: forwarding
          RX bytes: 75 frames: 12
          TX bytes: 30246 frames: 91 dropframes: 0
=>
```

RELATED COMMANDS:

- bridge ifadd Create a bridged Ethernet interface.
- bridge ifconfig Configure a bridge interface.
- bridge ifdelete Delete a bridge interface.

<code>bridge ifdetach</code>	Detach a bridge interface.
<code>bridge iflist</code>	Show current bridge configuration.

bridge ifconfig

Configure a bridge interface.

SYNTAX:

```
bridge ifconfig      intf = <intfname>
                    [dest = <string>]
                    [qos = <string>]
                    [encaps = <{llc/snap|vcmux}>]
                    [fcs = <{off|on}>]
                    [portstate = <{disabled|learning|forwarding}>]
                    [retry = <number {0-65535}>]
```

intf	The name of the bridge interface to configure.	REQUIRED
dest	The destination for this interface. Typically a phonebook entry. This parameter only needs to be specified when an interface has been created without specified destination.	OPTIONAL
qos	The name of the Quality Of Service (QoS) book entry to apply on this bridge interface.	OPTIONAL
encaps	The type of encapsulation to be used for this bridge interface. Choose between: <ul style="list-style-type: none"> • llc/snap • vc mux 	OPTIONAL
fcs	Whether or not to include the Ethernet FCS in the packet header on the WAN side. Choose between: <ul style="list-style-type: none"> • off (default) • on <p>Note FCS is normally left off.</p>	OPTIONAL
portstate	The bridge portstate for this interface. Choose between: <ul style="list-style-type: none"> • disabled • learning • forwarding 	OPTIONAL
retry	A number between 0 and 65535. Represents the number of times the SpeedTouch™ retries to set up a WAN connection before giving up. By default the retry value is 10.	OPTIONAL

Note In case of a SpeedTouch™530 device, never change, or delete the usb_bridge interface!

EXAMPLE:

```
=>bridge iflist intf=TestBridge
TestBridge : dest : Br1
             Retry: 10   QoS: default   Encaps: llc/snap   Fcs: off
             Connection State: connected Port: wan0 PortState: forwarding
             RX bytes: 75      frames: 12
             TX bytes: 30246   frames: 91   dropframes: 0
=>bridge ifconfig intf=TestBridge encaps=vcmux retry=15
=>bridge iflist intf=TestBridge
TestBridge : dest : Br1
             Retry: 15   QoS: default   Encaps: vcmux      Fcs: off
             Connection State: connected Port: wan0 PortState: forwarding
             RX bytes: 83      frames: 13
             TX bytes: 30740   frames: 102  dropframes: 0
=>
```

RELATED COMMANDS:

<code>bridge ifadd</code>	Create a bridged Ethernet interface.
<code>bridge ifattach</code>	Attach a bridge interface.
<code>bridge ifdelete</code>	Delete a bridge interface.
<code>bridge ifdetach</code>	Detach a bridge interface.
<code>bridge iflist</code>	Show current bridge configuration.

bridge ifdelete

Delete a bridge interface.

SYNTAX:

```
bridge ifdelete    intf = <intfname>
```

intf	The name of the interface name to delete. Browse through the available entries via the ARROW UP and ARROW DOWN keys.	REQUIRED
------	---	-----------------

EXAMPLE:

```
=>bridge ifdelete intf=TestBridge
=>bridge iflist
OBC      : Internal
          Connection State: connected Port: OBC PortState: forwarding
          RX bytes: 75783 frames: 572
          TX bytes: 82843610 frames: 341554 dropframes: 0
eth0     : Internal
          Connection State: connected Port: eth0 PortState: forwarding
          RX bytes: 156472129 frames: 5903256
          TX bytes: 75689 frames: 425 dropframes: 5561702
usb_bridge : dest : usb_port
          Retry : 10 QoS : default Encaps : llc/snap Fcs : off
          Connection State: connected
          Port: wan2 PortNr: 4 PortState: forwarding
          RX bytes: 0 frames: 0
          TX bytes: 0 frames: 0 dropframes: 0
=>
```

Note In case of a SpeedTouch™530 device, never change, or delete the usb_bridge interface!

RELATED COMMANDS:

- bridge ifadd Create a bridged Ethernet interface.
- bridge ifattach Attach a bridge interface.
- bridge ifconfig Configure a bridge interface.
- bridge ifdetach Detach a bridge interface.
- bridge iflist Show current bridge configuration.

bridge ifdetach

Detach (i.e. disconnect) a bridge interface.

SYNTAX:

```
bridge ifdetach    intf = <intfname>
```

intf	The name of the bridge interface to detach. Browse through the available entries via the ARROW UP and ARROW DOWN keys.	REQUIRED
------	---	----------

EXAMPLE:

```
=>bridge iflist intf=TestBridge
TestBridge :  dest : Br1
              Retry: 10  QoS: default  Encaps: llc/snap  Fcs: off
              Connection State: connected  Port: wan0      PortState: forwarding
              RX bytes: 75      frames: 12
              TX bytes: 30246   frames: 91      dropframes: 0
=>bridge ifdetach intf=TestBridge
=>bridge iflist intf=TestBridge
TestBridge :  dest : Br1
              Retry: 10  QoS: default  Encaps: llc/snap  Fcs: off
              Connection State: not-connected  Port: (unassigned)  PortState: forwarding
=>
```

Note In case of a SpeedTouch™530 device, never change, or delete the usb_bridge interface!

RELATED COMMANDS:

bridge ifadd	Create a bridged Ethernet interface.
bridge ifattach	Attach a bridge interface.
bridge ifconfig	Configure a bridge interface.
bridge ifdelete	Delete a bridge interface.
bridge iflist	Show current bridge configuration.

bridge iflist

Show the current state of all or the selected bridge interfaces.

SYNTAX:

```
bridge iflist [intf = <intfname>]
```

intf	The name of the bridge interface for which the configuration must be shown. Browse through the available entries via the ARROW UP and ARROW DOWN keys. If no interface is specified, all bridge interfaces are shown.	OPTIONAL
------	---	----------

EXAMPLE:

```
=>bridge iflist
OBC      : Internal
          Connection State: connected Port: OBC PortState: forwarding
          RX bytes: 75783 frames: 572
          TX bytes: 82768372 frames: 341221 dropframes: 0
eth0     : Internal
          Connection State: connected Port: eth0 PortState: forwarding
          RX bytes: 156344216 frames: 5899238
          TX bytes: 75689 frames: 425 dropframes: 5558017
usb_bridge : dest : usb_port
          Retry : 10 QoS : default Encaps : llc/snap Fcs : off
          Connection State: connected
          Port: wan2 PortNr: 4 PortState: forwarding
          RX bytes: 0 frames: 0
          TX bytes: 0 frames: 0 dropframes: 0
TestBridge : dest : Br1
          Retry: 15 QoS: default Encaps: vcmux Fcs: off
          Connection State: connected Port: wan0 PortState: forwarding
          RX bytes: 83 frames: 13
          TX bytes: 30740 frames: 102 dropframes: 0
=>
```

Note In case of a SpeedTouch™530 device, never change, or delete the usb_bridge interface!

DESCRIPTION:

RX bytes	The number of Received bytes
TX bytes	The number of Transmitted bytes
OBC	On Board Controller: indicates the physical bridge port.

RELATED COMMANDS:

bridge ifadd	Create a bridged Ethernet interface.
--------------	--------------------------------------

<code>bridge ifattach</code>	Attach a bridge interface.
<code>bridge ifconfig</code>	Configure a bridge interface.
<code>bridge ifdelete</code>	Delete a bridge interface.
<code>bridge ifdetach</code>	Detach a bridge interface.

bridge macadd

Add a static MAC address to the filtering database. Allows to manually add static addresses, which should normally be dynamically discovered by the bridge itself.

SYNTAX:

```
bridge macadd      intf = <intfname>
                   hwaddr = <hardware-address>
```

intf	The name of the bridge interface to add the MAC address for.	REQUIRED
hwaddr	The Ethernet MAC address of the new entry.	REQUIRED

EXAMPLE:

```
=>bridge maclist
00:90:d0:01:02:03 -- static, OBC
ff:ff:ff:ff:ff:ff -- static, OBC
01:80:c2:00:00:00 -- static, OBC
01:80:c2:00:00:01 -- static, OBC
...
01:80:c2:00:00:10 -- static, OBC
00:01:42:5f:7d:81 -- dynamic, eth0, 597 seconds
00:50:8b:31:cc:aa -- dynamic, eth0, 513 seconds
08:00:20:c1:9a:12 -- dynamic, eth0, 600 seconds
...
=>bridge macadd intf=eth0 hwaddr=00:80:9f:01:23:45
=>bridge maclist
00:90:d0:01:02:03 -- static, OBC
ff:ff:ff:ff:ff:ff -- static, OBC
01:80:c2:00:00:00 -- static, OBC
01:80:c2:00:00:01 -- static, OBC
...
01:80:c2:00:00:10 -- static, OBC
00:80:9f:01:23:45 -- permanent, eth0
00:01:42:5f:7d:81 -- dynamic, eth0, 598 seconds
00:50:8b:31:cc:aa -- dynamic, eth0, 379 seconds
08:00:20:c1:9a:12 -- dynamic, eth0, 600 seconds
00:08:c7:c3:5f:fc -- dynamic, eth0, 215 seconds
...
=>
```

RELATED COMMANDS:

- bridge macdelete Delete a MAC address entry.
- bridge maclist Show current filtering database.

bridge macdelete

Remove a MAC address from the filtering database.

SYNTAX:

```
bridge macdelete hwaddr = <hardware-address>
```

hwaddr	The Ethernet MAC address of the entry to delete.	REQUIRED
--------	--	----------

EXAMPLE:

```
=>bridge maclist
00:90:d0:01:02:03 -- static, OBC
ff:ff:ff:ff:ff:ff -- static, OBC
01:80:c2:00:00:00 -- static, OBC
01:80:c2:00:00:01 -- static, OBC
...
01:80:c2:00:00:10 -- static, OBC
00:80:9f:01:23:45 -- permanent, OBC
00:01:42:5f:7d:81 -- dynamic, eth0, 597 seconds
00:50:8b:31:cc:aa -- dynamic, eth0, 513 seconds
08:00:20:c1:9a:12 -- dynamic, eth0, 600 seconds
...
=>bridge macdelete hwaddr=00:80:9f:01:23:45
=>bridge maclist
00:90:d0:01:02:03 -- static, OBC
ff:ff:ff:ff:ff:ff -- static, OBC
01:80:c2:00:00:00 -- static, OBC
01:80:c2:00:00:01 -- static, OBC
...
01:80:c2:00:00:10 -- static, OBC
00:01:42:5f:7d:81 -- dynamic, eth0, 598 seconds
00:50:8b:31:cc:aa -- dynamic, eth0, 379 seconds
08:00:20:c1:9a:12 -- dynamic, eth0, 600 seconds
00:08:c7:c3:5f:fc -- dynamic, eth0, 215 seconds
...
=>
```

RELATED COMMANDS:

bridge macadd	Add a static MAC address to the filtering database.
bridge maclist	Show current filtering database.

bridge maclist

Shows the current MAC address filtering database.

SYNTAX:

```
bridge maclist
```

EXAMPLE:

```
=>bridge maclist
=>bridge maclist
00:90:d0:01:02:03 -- static, OBC
ff:ff:ff:ff:ff:ff -- static, OBC
01:80:c2:00:00:00 -- static, OBC
01:80:c2:00:00:01 -- static, OBC
...
01:80:c2:00:00:10 -- static, OBC
00:80:9f:24:ab:cf -- static, OBC
00:01:42:5f:7d:81 -- dynamic, eth0, 598 seconds
00:50:8b:31:cc:aa -- dynamic, eth0, 379 seconds
08:00:20:c1:9a:12 -- dynamic, eth0, 600 seconds
00:08:c7:c3:5f:fc -- dynamic, eth0, 215 seconds
08:00:20:a8:f4:34 -- dynamic, eth0, 600 seconds
08:00:20:83:b7:26 -- dynamic, eth0, 600 seconds
00:10:83:1b:13:18 -- dynamic, eth0, 599 seconds
...
=>
```

RELATED COMMANDS:

- `bridge macadd` Add a static MAC address to the filtering database.
- `bridge macdelete` Delete a MAC address entry.

CIP Commands

Contents

This chapter covers the following commands:

Topic	Page
cip flush	72
cip ifadd	73
cip ifdelete	74
cip iflist	75
cip pvcadd	76
cip pvdelete	77
cip pvclist	78

cip flush

Flush complete Classical IP over ATM (IP oA) configuration.

Note The flush command does not impact previously saved configurations.

SYNTAX:

```
cip flush
```


cip ifadd

Create a Classical IP oA interface at the local side of the Logical IP Subnet (LIS).

SYNTAX:

```
cip ifadd          addr = <ip-address>
                  [netmask = <ip-mask (dotted or cidr)>]
                  [uniaddr = <portspec:address[.selector]>]
```

addr	The Classical IP oA interface's local IP address in the LIS.	REQUIRED
netmask	The LIS's subnetmask.	OPTIONAL
uniaddr	The UNI-address/port specification for incoming connections, e.g. 'A0:*.03': ADSL port, any address, selector 3. Only applicable in an Switched Virtual Channel (SVC) environment. In most cases the Classical IP oA LIS is built in a Permanent Virtual Channel (PVC) environment.	OPTIONAL

EXAMPLE:

```
=>cip iflist
cip1      addr = 172.16.0.5  mask = 255.255.255.0
          UNI address = A0:*.04
          inarp_reqs_in  = 0  inarp_repl_in  = 0  inarp_inv_in  = 0
          inarp_reqs_out = 0  inarp_repl_out = 0  inarp_inv_out = 0
=>cip ifadd addr=172.16.1.1 netmask=255.255.255.0
=>cip iflist
cip1      addr = 172.16.0.5  mask = 255.255.255.0
          UNI address = A0:*.04
          inarp_reqs_in  = 0  inarp_repl_in  = 0  inarp_inv_in  = 0
          inarp_reqs_out = 0  inarp_repl_out = 0  inarp_inv_out = 0
cip0      addr = 172.16.1.1  mask = 255.255.255.0
          UNI address = A0:*.03
          inarp_reqs_in  = 0  inarp_repl_in  = 0  inarp_inv_in  = 0
          inarp_reqs_out = 0  inarp_repl_out = 0  inarp_inv_out = 0
=>
```

RELATED COMMANDS:

cip ifdelete	Delete a Classical IP oA interface.
cip iflist	Show current Classical IP oA configuration.

cip ifdelete

Delete a Classical IP oA interface at the local side of the LIS.

SYNTAX:

```
cip ifdelete      addr = <ip-address>
```

addr	The Classical IP oA interface's local IP address in the LIS.	REQUIRED
------	--	----------

EXAMPLE:

```
=>cip iflist
cip0      addr = 172.16.1.1  mask = 255.255.255.0
          UNI address = A0:*.03
          inarp_reqs_in  = 0  inarp_repl_in  = 0  inarp_inv_in  = 0
          inarp_reqs_out = 0  inarp_repl_out = 0  inarp_inv_out = 0
cip1      addr = 172.16.0.5  mask = 255.255.255.0
          UNI address = A0:*.04
          inarp_reqs_in  = 0  inarp_repl_in  = 0  inarp_inv_in  = 0
          inarp_reqs_out = 0  inarp_repl_out = 0  inarp_inv_out = 0
=>cip ifdelete addr=172.16.1.1
=>cip iflist
cip1      addr = 172.16.0.5  mask = 255.255.255.0
          UNI address = A0:*.04
          inarp_reqs_in  = 0  inarp_repl_in  = 0  inarp_inv_in  = 0
          inarp_reqs_out = 0  inarp_repl_out = 0  inarp_inv_out = 0
=>
```

RELATED COMMANDS:

cip ifadd	Create a Classical IP oA interface at the local side of the Logical IP Subnet.
cip iflist	Show current Classical IP oA configuration.

cip iflist

Show current Classical IP oA configuration.

SYNTAX:

```
cip iflist
```

EXAMPLE:

```
=>cip iflist
cip0      addr = 172.16.1.1  mask = 255.255.255.0
          UNI address = A0:*.03
          inarp_reqs_in = 0  inarp_repl_in = 0  inarp_inv_in = 0
          inarp_reqs_out = 0  inarp_repl_out = 0  inarp_inv_out = 0
cip1      addr = 172.16.0.5  mask = 255.255.255.0
          UNI address = A0:*.04
          inarp_reqs_in = 0  inarp_repl_in = 0  inarp_inv_in = 0
          inarp_reqs_out = 0  inarp_repl_out = 0  inarp_inv_out = 0
=>
```

DESCRIPTION:

inarp_reqs_in/inarp_reqs_out Incoming/outgoing inverse ARP requests
inarp_repl_in/inarp_repl_out Incoming/outgoing inverse ARP replies
inarp_inv_in/inarp_inv_out Incoming/outgoing invalid inverse ARP messages

EXAMPLE INPUT/OUTPUT: EVOLUTION OF ARP REQUESTS IN A NETWORKED ENVIRONMENT:

```
=>cip iflist
cip0      addr = 200.200.200.138  mask = 255.255.255.0
          UNI address = A0:*.03
          inarp_reqs_in = 18  inarp_repl_in = 75  inarp_inv_in = 0
          inarp_reqs_out = 18  inarp_repl_out = 75  inarp_inv_out = 0
=>cip iflist
cip0      addr = 200.200.200.138  mask = 255.255.255.0
          UNI address = A0:*.03
          inarp_reqs_in = 22  inarp_repl_in = 75  inarp_inv_in = 0
          inarp_reqs_out = 22  inarp_repl_out = 75  inarp_inv_out = 0
=>cip iflist
cip0      addr = 200.200.200.138  mask = 255.255.255.0
          UNI address = A0:*.03
          inarp_reqs_in = 22  inarp_repl_in = 76  inarp_inv_in = 0
          inarp_reqs_out = 22  inarp_repl_out = 76  inarp_inv_out = 0
=>
```

RELATED COMMANDS:

cip ifadd Create a Classical IP oA interface at the local side of the Logical IP Subnet.
cip ifdelete Delete a Classical IP oA interface.

cip pvcadd

Create a PVC Address Resolution Protocol (ARP) entry for destinations which are not RFC1577/RFC2225 compliant.

SYNTAX:

```
cip pvcadd          dest = <phonebookname>
                   [destaddr = <ip-address>]
                   [mtu = <number {273-20000}>]
```

dest	The ATM address (hardware address) of the destination host. Typically a phonebook name.	REQUIRED
destaddr	The IP address of the destination host.	OPTIONAL
mtu	A number between 273 and 20000 (bytes). Represents the maximum ATM Adaption Layer 5 (AAL5) packet size for this connection. By default the mtu is 9180 bytes.	OPTIONAL

EXAMPLE:

```
=>phonebook list
Name      Type   Use  Address
Br1       bridge 1    8.35
Br2       bridge 1    8.36
Br3       bridge 1    8.37
Br4       bridge 0    8.38
RELAY_PPP1 ppp    0    8.48
RELAY_PPP2 ppp    0    8.49
RELAY_PPP3 ppp    0    8.50
RELAY_PPP4 ppp    0    8.51
PPP1      ppp    1    8.64
PPP2      ppp    1    8.65
PPP3      ppp    1    8.66
DHCP_SPOOF ppp    1    8.67
CIPPVC1   cip    0    8.80
CIPPVC2   cip    0    8.81
CIPPVC3   cip    0    8.82
CIPPVC4   cip    0    8.83
=>cip pvclist
=>cip pvcadd dest CIPPVC1 destaddr 172.16.1.2 mtu 546
=>cip pvclist
CIPPVC1      atmport = 0      vpi = 8      vci = 80      dest_ip = 172.16.1.2
              encaps = 11c      mtu = 546
=>
```

RELATED COMMANDS:

<code>cip pvcdelete</code>	Delete a PVC ARP entry.
<code>cip pvclist</code>	Show current PVC ARP entries.

cip pvdelete

Delete a PVC ARP entry.

SYNTAX:

```
cip pvdelete      dest = <phonebookname>
```

dest	Typically a phonebook entry name. Represents the ATM address (hardware address) or name of the entry to delete.	REQUIRED
------	--	----------

EXAMPLE:

```
=>cip pvclist
CIPPVC1      atmport = 0      vpi = 8      vci = 80      dest_ip = 172.16.1.2
              encaps  = 11c   mtu  = 546
=>cip pvdelete dest=CIPPVC1
=>cip pvclist
=>
```

RELATED COMMANDS:

cip pvadd	Creates a PVC ARP entry.
cip pvclist	Show current PVC ARP entries.

cip pvclist

Show current PVC ARP entries.

SYNTAX:

```
cip pvclist
```

EXAMPLE:

```
=>cip pvclist
CIPPVC1      atmport = 0      vpi = 8      vci = 80      dest_ip = 172.16.1.2
              encaps  = 11c      mtu = 546
=>
```

EXAMPLE INPUT/OUTPUT IN A NETWORKED ENVIRONMENT:

```
=>cip iflist
cip0          addr = 200.200.200.138  mask = 255.255.255.0
              UNI address = A0:*.03
              inarp_reqs_in = 0   inarp_repl_in = 75   inarp_inv_in = 0
              inarp_reqs_out = 0  inarp_repl_out = 75  inarp_inv_out = 0
=>cip pvclist
699          atmport = 0      vpi = 6      vci = 99      dest_ip = 172.16.1.3
              encaps  = 11c      mtu = 9180
8.50         atmport = 0      vpi = 8      vci = 50      dest_ip = 200.200.200.14
              encaps  = 11c      mtu = 9180
=>
```

RELATED COMMANDS:

- `cip pvccadd` Creates a PVC ARP entry.
- `cip pvccdelete` Delete a PVC ARP entry.

Config Commands

Contents

This chapter covers the following commands:

Topic	Page
config backup	80
config dump	81
config erase	82
config flush	83
config list	84
config load	85
config save	87

config backup

Store current configuration to backup file.

SYNTAX:

```
config backup filename = <user configuration filename>
```

filename	Filename for backup file of current configuration	REQUIRED
----------	---	----------

RELATED COMMANDS:

- `config dump` Shows the saved configuration file
- `config list` Shows the current configuration set.

config dump

Shows the saved configuration file.

SYNTAX:

```
config dump
```

RELATED COMMANDS:

[config backup](#)

Store current configuration to backup file.

[config list](#)

Shows the current configuration set.

config erase

Physically remove all saved configurations, i.e. the user configuration file, saved in the SpeedTouch™ permanent storage, is deleted.

SYNTAX:

```
config erase [filename = <user configuration filename>]
```

filename	Name of the configuration file to erase.	OPTIONAL
----------	--	----------

RELATED COMMANDS:

- config flush Flush complete runtime configuration.
- config load Load complete saved or default configuration.
- config save Save complete runtime configuration.

config flush

Flush complete current configuration without affecting saved configurations.

This combines all flush commands: bridge flush, cip flush, dhcp client flush, dhcp relay flush, dhcp server flush, dhcp server lease flush, dhcp server pool flush, dns flush, env flush, ethoa flush, firewall flush, firewall chain flush, firewall rule flush, ip flush, ip auto flush, ipoa flush, label flush, label chain flush, label rule flush, nat flush, phonebook flush, pppoa flush, pppoe flush, pptp flush, qosbook flush, script flush, snmp flush, system flush, upnp flush and wireless flush.

SYNTAX:

```
config flush [flush_ip = <{no|yes}>]
```

flush_ip	Flush current IP configuration (yes) or keep current IP configuration (no).	OPTIONAL
	Not keeping the IP settings could cause lost IP connectivity in the LAN. By default IP settings are preserved.	

EXAMPLE:

```
=>ip rtlist
  Destination      Source      Gateway      Intf      Mtrc
  10.0.0.0/24      10.0.0.0/24 10.0.0.140   eth0      0
  172.16.0.5/32    0.0.0.0/0   172.16.0.5   cip1      0
  10.0.0.140/32    0.0.0.0/0   10.0.0.140   eth0      0
  127.0.0.1/32     0.0.0.0/0   127.0.0.1    loop      0
  10.0.0.0/24      0.0.0.0/0   10.0.0.140   eth0      0
  172.16.0.0/24    0.0.0.0/0   172.16.0.5   cip1      1
=>config flush flush_ip=no
=>ip rtlist
  Destination      Source      Gateway      Intf      Mtrc
  10.0.0.0/24      10.0.0.0/24 10.0.0.140   eth0      0
  10.0.0.140/32    0.0.0.0/0   10.0.0.140   eth0      0
  127.0.0.1/32     0.0.0.0/0   127.0.0.1    loop      0
  10.0.0.0/24      0.0.0.0/0   10.0.0.140   eth0      0
=>config flush flush_ip=yes

##### ALL TCP/IP CONNECTIVITY IS LOST #####
```

RELATED COMMANDS:

config erase	Physically remove all saved configurations.
config load	Load complete saved or default configuration.
config save	Save current runtime configuration.

config list

Shows the current configuration set.

SYNTAX:

```
config list [templates = <{no|yes}>]
```

templates Select yes to include the template configuration files in the list. OPTIONAL

EXAMPLE:

```
=>config list templates=yes
Configurations file(s):
  user.ini
Template file(s)
  No template files present
Factory template file(s) :
  pppoe.tpl
  pppoa.tpl
  br.tpl
=>
```

RELATED COMMANDS:

- config backup Store current configuration to backup file.
- config dump Shows the saved configuration file

config load

Load complete saved (backup) or default configuration file.

Note Execute config flush prior to config load.

SYNTAX:

```
config load      [load_ip = <{no|yes}>]
                 [defaults = <{yes|no}>]
                 [flush = <{yes|no}>]
                 [echo = <{no|yes}>]
                 [filename = <string>]
```

load_ip	Load IP settings (yes) or not (no). Not keeping the IP settings could cause lost IP connectivity in the LAN.	OPTIONAL
defaults	Load default configuration (yes) or saved configuration (no). Not specifying this parameter loads the saved configuration	OPTIONAL
flush	Flush current configuration before loading new one.	OPTIONAL
echo	Echo each command string when loaded.	OPTIONAL
filename	Configuration filename.	OPTIONAL

EXAMPLE:

```
=>ip rtlist
  Destination      Source      Gateway      Intf      Mtrc
  10.0.0.0/24      10.0.0.0/24 10.0.0.140   eth0      0
  172.16.0.5/32   0.0.0.0/0   172.16.0.5   cip1      0
  10.0.0.140/32   0.0.0.0/0   10.0.0.140   eth0      0
  127.0.0.1/32    0.0.0.0/0   127.0.0.1    loop      0
  10.0.0.0/24     0.0.0.0/0   10.0.0.140   eth0      0
  172.16.0.0/24   0.0.0.0/0   172.16.0.5   cip1      1
=>config flush flush_ip=no
=>ip rtlist
  Destination      Source      Gateway      Intf      Mtrc
  10.0.0.0/24      10.0.0.0/24 10.0.0.140   eth0      0
  10.0.0.140/32   0.0.0.0/0   10.0.0.140   eth0      0
  127.0.0.1/32    0.0.0.0/0   127.0.0.1    loop      0
  10.0.0.0/24     0.0.0.0/0   10.0.0.140   eth0      0
=>config load load_ip=yes
=>ip rtlist
  Destination      Source      Gateway      Intf      Mtrc
  10.0.0.0/24      10.0.0.0/24 10.0.0.140   eth0      0
  10.0.0.140/32   0.0.0.0/0   10.0.0.140   eth0      0
  172.16.0.5/32   0.0.0.0/0   172.16.0.5   cip1      0
  127.0.0.1/32    0.0.0.0/0   127.0.0.1    loop      0
  10.0.0.0/24     0.0.0.0/0   10.0.0.140   eth0      0
  172.16.0.0/24   0.0.0.0/0   172.16.0.5   cip1      1
=>
```

RELATED COMMANDS:

<code>config erase</code>	Physically remove all saved configurations.
<code>config flush</code>	Flush complete runtime configuration.
<code>config save</code>	Save current runtime configuration.

config save

Save all existing configurations and modifications entered by the user.

The result of executing this command is a user.ini file saved in the SpeedTouch™ permanent storage. This file can be downloaded via the SpeedTouch™ web pages or via an FTP session.

SYNTAX:

```
config save
```

EXAMPLE:

```
=>config save  
=>
```

RELATED COMMANDS:

config erase	Physically remove all saved configurations.
config flush	Flush complete runtime configuration.
config load	Load complete saved or default configuration.

DHCP Commands

Contents

This chapter covers the following commands

Topic	Page
dhcp client clear	91
dhcp client config	92
dhcp client flush	93
dhcp client ifadd	94
dhcp client ifattach	95
dhcp client ifconfig	96
dhcp client ifdelete	98
dhcp client iflist	99
dhcp client ifrelease	100
dhcp client ifrenew	102
dhcp client stats	104
dhcp relay add	105
dhcp relay config	106
dhcp relay delete	107
dhcp relay flush	108
dhcp relay ifconfig	109
dhcp relay iflist	110
dhcp relay list	111
dhcp relay stats	112
dhcp server clear	113
dhcp server config	114
dhcp server flush	115
dhcp server policy	116
dhcp server stats	117
dhcp server lease add	119
dhcp server lease delete	121

Topic	Page
dhcp server lease flush	122
dhcp server lease list	123
dhcp server pool add	124
dhcp server pool config	125
dhcp server pool delete	127
dhcp server pool flush	128
dhcp server pool list	129

dhcp client clear

Clear Dynamic Host Configuration Protocol (DHCP) client statistics.

SYNTAX:

```
dhcp client clear
```

EXAMPLE:

```
=>dhcp client stats
DHCP client statistics:
Corrupted packet recv : 0
OFFERS recv           : 0
ACKs recv             : 0
NAKS recv             : 0
Pure BOOTP REPLIES   : 0
Other message types   : 0
DISCOVERs sent       : 253
REQUESTs sent        : 9
DECLINES sent        : 0
RELEASES sent        : 0
INFORMs sent         : 0
Number of dynamic interfaces: 1
Memory usage:
Table size of dyn leases: 19, in use: 1, free: 94 %
=>dhcp client clear
=>dhcp client stats
DHCP client statistics:
Corrupted packet recv : 0
OFFERS recv           : 0
ACKs recv             : 0
NAKS recv             : 0
Pure BOOTP REPLIES   : 0
Other message types   : 0
DISCOVERs sent       : 0
REQUESTs sent        : 0
DECLINES sent        : 0
RELEASES sent        : 0
INFORMs sent         : 0
Number of dynamic interfaces: 1
Memory usage:
Table size of dyn leases: 19, in use: 1, free: 94 %
=>
```

RELATED COMMANDS:

[dhcp client stats](#)

Show DHCP client statistics.

dhcp client config

Show/set DHCP client configuration.

SYNTAX:

```
dhcp client config [trace = <{off|on}>]
```

trace

Enable tracing (on) or not (off).

OPTIONAL

EXAMPLE:

```
=>dhcp client config
tracing: off
=>dhcp client config trace=on
=>dhcp client config
tracing: on
=>
```

RELATED COMMANDS:

[dhcp client ifconfig](#)

Configure a DHCP lease created for a specific interface.

dhcp client flush

Flush complete DHCP client configuration and dynamic interfaces.

Note The flush command does not impact previously saved configurations.

SYNTAX:

```
dhcp client flush
```

EXAMPLE:

```
=>dhcp client iflist
NewETHoA      : [SELECTING]
               flags= uc
               IP address  : 10.0.0.10
               HW address  : 0:90:d0:01:47:de
               DHCP server : 255.255.255.255
               hostname    : NewLease
               req.leasetime= 10800 s
               trying to get a lease for 8 min, 32 sec
               transmission of DISCOVER in 57 sec
               retransmission timeout: 64
               nbr of retransmissions: 14
Number of leases: 1
Total size of table: 19, in use: 1, free: 94 %
=>dhcp client flush
=>dhcp client iflist
No dynamic interfaces defined.
=>
```

dhcp client ifadd

Create a DHCP lease for a specific interface.

SYNTAX:

```
dhcp client ifadd    intf = <interface name>
```

intf	The name of an existing interface, e.g. created via :mer ifadd.	REQUIRED
------	---	----------

EXAMPLE:

```
=>dhcp client iflist
No dynamic interfaces defined.
=>dhcp client ifadd intf=NewEthoa
=>dhcp client iflist
NewETHoA      : [INIT]
                flags= uc
                IP address   : 0.0.0.0
                HW address   : 00:90:d0:01:47:de
                DHCP server  : 255.255.255.255
Number of leases: 1
Total size of table: 19, in use: 1, free: 94 %
=>
```

RELATED COMMANDS:

dhcp client ifattach	Attach a DHCP lease to an interface.
dhcp client ifconfig	Configure a DHCP lease created for a specific interface.
dhcp client ifdelete	Delete a dynamic interface.
dhcp client iflist	Show all dynamic interfaces.

dhcp client ifattach

Attach a DHCP lease to a dynamic interface.

Note Create the interface first with :dhcp client ifadd.

SYNTAX:

```
dhcp client ifattach   intf = <interface name>
```

intf	The name of the dynamic interface.	REQUIRED
------	------------------------------------	----------

EXAMPLE:

```
=>dhcp client iflist
NewETHoA   : [INIT]
            flags= uc
            IP address   : 0.0.0.0
            HW address   : 00:90:d0:01:47:de
            DHCP server  : 255.255.255.255
Number of leases: 1
Total size of table: 19,   in use: 1,   free: 94 %
=>dhcp client ifattach intf=NewETHoA
=>dhcp client iflist
NewETHoA   : [SELECTING]
            flags= uc
            IP address   : 10.0.0.10
            HW address   : 0:90:d0:01:47:de
            DHCP server  : 255.255.255.255
            hostname    : NewLease
            req.lease time = 10800 s
            trying to get a lease for 8 min, 32 sec
            transmission of DISCOVER in 57 sec
            retransmission timeout: 64
            nbr of retransmissions: 14
Number of leases: 1
Total size of table: 19,   in use: 1,   free: 94 %
=>
```

RELATED COMMANDS:

dhcp client ifadd	Create a DHCP lease for a specific interface.
dhcp client ifconfig	Configure a DHCP lease created for a specific interface.
dhcp client ifrelease	Release a lease attached to a dynamic interface.
dhcp client iflist	Show all dynamic interfaces.

dhcp client ifconfig

Show/set the configuration of DHCP lease created for a specific interface.

Note Execute the dhcp client ifrelease command before configuring the dhcp client.

SYNTAX:

```
dhcp client ifconfig    intf = <interface name>
                        [clientid = <{client-id|none}>]
                        [hostname = <hostname|"">]
                        [addr = <ip-address>]
                        [leasetime = <number>]
                        [addrtrans = <{none|pat}>]
                        [dns = <{off|on}>]
                        [gateway = <{off|on}>]
                        [metric = <number{0-100}>]
                        [dnsmetric = <number{0-100}>]
```

intf	The name of the dynamic interface to be configured.	REQUIRED
clientid	The client identity to be associated with the lease. Use none in case no clientid should be associated with this lease.	OPTIONAL
hostname	The host name of the client to be associated with the lease. Use "" in case no hostname should not be associated with this lease.	OPTIONAL
addr	The preferred dynamic IP address.	OPTIONAL
leasetime	A number between 0 and 1814400 (seconds). Represents the preferred time the client wants to use an address. By default the leasetime is 7200 seconds (2 hours). Specifying -l makes the lease permanent.	OPTIONAL
addrtrans	Automatically enable address translation for this dynamic interface (pat) or not (none).	OPTIONAL
dns	Request (and accept) DNS server IP addresses (on) or not (off).	OPTIONAL
gateway	Request (and accept) gateway IP addresses (on) or not (off).	OPTIONAL
metric	A number between 0 and 100. Represents the gateway route metric By default the gateway route metric is 1.	OPTIONAL
dnsmetric	A number between 0 and 100. Represents the DNS route metric By default the DNS route metric is 1.	OPTIONAL

EXAMPLE:

```
=>dhcp client iflist
NewETHoA   : [INIT]
            flags= uc
            IP address   : 0.0.0.0
            HW address   : 00:90:d0:01:47:de
            DHCP server  : 255.255.255.255
Number of leases: 1
Total size of table: 36, in use: 1, free: 97%
=>dhcp client ifconfig intf=NewETHoA hostname=NewLease addr=10.0.0.10 leasetime=10800
=>dhcp client iflist
NewETHoA   : [INIT]
            flags= uc
            IP address   : 10.0.0.10
            HW address   : 00:90:d0:01:47:de
            DHCP server  : 255.255.255.255
            hostname     : NewLease
            req.leasetime = 10800 s
Number of leases: 1
Total size of table: 36, in use: 1, free: 97%
=>
```

RELATED COMMANDS:

- | | |
|------------------------------------|--|
| <code>dhcp client ifadd</code> | Create a DHCP lease for a specific interface. |
| <code>dhcp client ifdelete</code> | Delete a dynamic interface. |
| <code>dhcp client iflist</code> | Show all dynamic interfaces. |
| <code>dhcp client ifrelease</code> | Release a lease attached to a dynamic interface. |

dhcp client ifdelete

Delete a dynamic interface.

SYNTAX:

```
dhcp client ifdelete   intf = <interface name>
```

intf	The name of the dynamic interface.	REQUIRED
------	------------------------------------	----------

EXAMPLE:

```
=>dhcp client iflist
NewETHoA   : [SELECTING]
           flags= uc
           IP address   : 10.0.0.10
           HW address   : 00:90:d0:01:47:de
           DHCP server  : 255.255.255.255
           hostname     : NewLease
           req.leasetime = 10800 s
           trying to get a lease for 8 min, 32 sec
           transmission of DISCOVER in 57 sec
           retransmission timeout: 64
           nbr of retransmissions: 14
Number of leases: 1
Total size of table: 19, in use: 1, free: 94 %
=>dhcp client ifdelete intf NewETHoA
=>dhcp client iflist
No dynamic interfaces defined.
=>
```

RELATED COMMANDS:

- | | |
|---------------------------------------|--|
| dhcp client ifadd | Create a DHCP lease for a specific interface. |
| dhcp client ifattach | Attach a DHCP lease to an interface. |
| dhcp client ifconfig | Configure a DHCP lease created for a specific interface. |
| dhcp client iflist | Show all dynamic interfaces. |
| dhcp client ifrelease | Release a lease attached to a dynamic interface. |

dhcp client iflist

Show all dynamic interfaces.

SYNTAX:

```
dhcp client iflist
```

EXAMPLE:

```
=>dhcp client iflist
NewETHoA      : [INIT]
              flags= uc
              IP address   : 0.0.0.0
              HW address   : 00:90:d0:01:47:de
              DHCP server  : 255.255.255.255
Number of leases: 1
Total size of table: 19,  in use: 1,  free: 94 %
=>
```

EXAMPLE INPUT/OUTPUT IN A NETWORKED ENVIRONMENT:

The SpeedTouch™ is configured as DHCP client on its Ethernet interface eth0.

```
=>dhcp client iflist
eth0          : [BOUND]
              flags= uc
              IP address   : 10.0.0.3
              HW address   : 00:90:d0:01:47:f1
              DHCP server  : 10.10.1.1
              lease renewal in 5 days, 1 h, 26 min, 45 sec
              lease rebinding in 8 days, 20 h, 34 min, 15 sec
              lease expires in 10 days, 2 h, 56 min, 45 sec
Number of leases: 1
Total size of table: 18,  in use: 1,  free: 94 %
=>dhcp client iflist
eth0          : [BOUND]
              flags= uc
              IP address   : 10.0.0.3
              HW address   : 00:90:d0:01:47:f1
              DHCP server  : 10.10.1.1
              lease renewal in 5 days, 1 h, 25 min, 27 sec
              lease rebinding in 8 days, 20 h, 32 min, 57 sec
              lease expires in 10 days, 2 h, 55 min, 27 sec
Number of leases: 1
Total size of table: 18,  in use: 1,  free: 94 %
=>
```

RELATED COMMANDS:

- | | |
|-----------------------------------|---|
| <code>dhcp client ifadd</code> | Create a DHCP lease for a specific interface. |
| <code>dhcp client ifdelete</code> | Delete a dynamic interface. |

dhcp client ifrelease

Release a lease attached to a dynamic interface.

SYNTAX:

```
dhcp client ifrelease intf = <interface name>
```

intf	The name of the dynamic interface.	REQUIRED
------	------------------------------------	----------

EXAMPLE:

```
=>dhcp client iflist
NewETHoA : [SELECTING]
          flags= uc
          IP address : 10.0.0.10
          HW address : 00:90:d0:01:47:de
          DHCP server : 255.255.255.255
          hostname : NewLease
          req.leasetime = 10800 s
          trying to get a lease for 8 min, 32 sec
          transmission of DISCOVER in 57 sec
          retransmission timeout: 64
          nbr of retransmissions: 14
Number of leases: 1
Total size of table: 19, in use: 1, free: 94 %
=>dhcp client ifattach intf=NewETHoA
=>dhcp client iflist
NewETHoA : [SELECTING]
          flags= uc
          IP address : 0.0.0.0
          HW address : 00:90:d0:01:47:de
          DHCP server : 255.255.255.255
          hostname : NewLease
          req.leasetime = 10800 s
Number of leases: 1
Total size of table: 19, in use: 1, free: 94 %
=>
```

EXAMPLE INPUT/OUTPUT IN A NETWORKED ENVIRONMENT:

The SpeedTouch™ is configured as DHCP client on its Ethernet interface eth0.

```

=>dhcp client iflist
eth0      : [BOUND]
           flags= uc
           IP address   : 10.0.0.3
           HW address   : 00:90:d0:01:47:f1
           DHCP server  : 10.10.1.1
           lease renewal in    5 days, 58 min, 45 sec
           lease rebinding in  8 days, 20 h, 6 min, 18 sec
           lease expires in   10 days, 2 h, 28 min, 48 sec
Number of leases: 1
Total size of table: 18,  in use: 1,  free: 94 %
=>dhcp client stats
DHCP client statistics:
Corrupted packet recv   :           0
DECLINES sent           :           0
RELEASES sent           :           0
INFORMS sent            :           0
Number of dynamic interfaces: 1
Memory usage:
Table size of dyn leases: 19,  in use: 1,  free: 94 %
=>dhcp client ifrelease intf=eth0
=>(CTRL + Q)
=>STATE ACTIVATE !
STATE IDLE !
STATE ACTIVATE !
dhcc: intf 1 releases 10.0.0.3 to server 10.10.1.1.
dhcc: 10.0.0.3 deleted: ok.
STATE IDLE !
STATE ACTIVATE !
.....
dhcc: intf 1 in init state.
n_send() broadcast triggered; To be verified
dhcc: broadcast discover on intf 1.
=>(CTRL + S)
=>dhcp client stats
DHCP client statistics:
Corrupted packet recv   :           0
DECLINES sent           :           0
RELEASES sent           :           1
INFORMS sent            :           0
Number of dynamic interfaces: 1
Memory usage:
Table size of dyn leases: 19,  in use: 1,  free: 94 %
=>

```

RELATED COMMANDS:

- | | |
|-----------------------------------|--|
| <code>dhcp client ifattach</code> | Attach a DHCP lease to an interface. |
| <code>dhcp client ifconfig</code> | Configure a DHCP lease created for a specific interface. |
| <code>dhcp client ifdelete</code> | Delete a dynamic interface. |

dhcp client ifrenew

Renew the lease of a dynamic interface.

SYNTAX:

```
dhcp client ifrenew intf = <interface name>
```

intf	The name of the dynamic interface.	REQUIRED
------	------------------------------------	----------

EXAMPLE:

```
=>dhcp client iflist
NewETHoA : [BOUND]
          flags= uc
          IP address : 10.0.0.10
          HW address : 00:90:d0:01:47:f1
          DHCP server : 255.255.255.255
          hostname : NewLease
          req.lease time = 10800 s
          lease renewal in 5 days, 58 min, 48 sec
          lease rebinding in 8 days, 20 h, 6 min, 18 sec
          lease expires in 10 days, 2 h, 28 min, 48 sec
Number of leases: 1
Total size of table: 19, in use: 1, free: 94 %
=>dhcp client ifrenew intf=NewETHoA
=>dhcp client iflist
NewETHoA : [SELECTING]
          flags= uc
          IP address : 10.0.0.10
          HW address : 00:90:d0:01:47:de
          DHCP server : 255.255.255.255
          hostname : NewLease
          req.lease time = 10800 s
          trying to get a lease for 12 sec
          transmission of DISCOVER in 24 sec
          retransmission timeout: 64
          nbr of retransmissions: 11
Number of leases: 1
Total size of table: 19, in use: 1, free: 94 %
=>
```

EXAMPLE INPUT/OUTPUT IN A NETWORKED ENVIRONMENT:

The SpeedTouch™ is configured as DHCP client on its Ethernet interface eth0.

```

=>dhcp client stats
DHCP client statistics:
Corrupted packet recv :          0
OFFERS   recv         :          0
ACKS     recv         :          0
NAKS     recv         :          0
Pure BOOTP REPLIES   :          0
Other message types  :          0
DISCOVERs sent       :          0
REQUESTs sent        :          0
DECLINEs sent        :          0
RELEASEs sent        :          1
INFORMs sent         :          0
Number of dynamic interfaces: 1
Memory usage:
Table size of dyn leases: 18,   in use: 1,   free: 94 %
=>dhcp client ifrenew intf=eth0
=>dhcp client stats
DHCP client statistics:
Corrupted packet recv :          0
OFFERS   recv         :          1
ACKS     recv         :          1
NAKS     recv         :          0
Pure BOOTP REPLIES   :          0
Other message types  :          0
DISCOVERs sent       :          1
REQUESTs sent        :          1
DECLINEs sent        :          0
RELEASEs sent        :          1
INFORMs sent         :          0
Number of dynamic interfaces: 1
Memory usage:
Table size of dyn leases: 18,   in use: 1,   free: 94 %
=>(CTRL + Q)
.....
STATE IDLE !
STATE ACTIVATE !
dhcc: intf 1 renews lease 10.0.0.3.
dhcc: intf 1 requests 10.0.0.3 from 10.10.1.1
dhcc: 10.10.1.1 acks 10.0.0.3 to intf 1.
dhcc: lease 10.0.0.3 bound to intf 1.
STATE IDLE !
STATE ACTIVATE !
.....
=>(CTRL + S)

```

RELATED COMMANDS:

- `dhcp client ifadd` Create a DHCP lease for a specific interface.
- `dhcp client ifattach` Attach a DHCP lease to an interface.

dhcp client stats

Show DHCP client statistics.

SYNTAX:

```
dhcp client stats
```

EXAMPLE:

```
=>dhcp client stats
DHCP client statistics:
Corrupted packet rcv :          0
OFFERS   rcv         :          1
ACKs     rcv         :          1
NAKS     rcv         :          0
Pure BOOTP REPLIES   :          0
Other message types  :          0
DISCOVERs sent       :         244
REQUESTs sent        :          9
DECLINEs sent        :          0
RELEASES sent        :          1
INFORMs  sent        :          0
Number of dynamic interfaces: 1
Memory usage:
Table size of dyn leases: 19,   in use: 1,   free: 94 %
=>
```

RELATED COMMANDS:

`dhcp client clear`

Clear DHCP client statistics.

dhcp relay add

Adds a DHCP server to the DHCP server list.

SYNTAX:

```
dhcp relay add      addr = <ip-address>
                   [intf = <none|interface_name>]
                   [giaddr = <ip-address>]
```

addr	The DHCP server IP address.	REQUIRED
intf	The name of the relay interface, 'None' to indicate no interface is specified.	OPTIONAL
giaddr	The giaddr field to be used in relayed DHCP packets.	OPTIONAL

RELATED COMMANDS:

dhcp relay config	Sets the relay configuration settings.
dhcp relay delete	Deletes a DHCP server from the DHCP server list.

dhcp relay config

Sets the relay configuration settings.

SYNTAX:

```
dhcp server config [agentinfo = <{off|on}>]
                  [agentmismatch = <{off|on}>]
                  [trace = <{off|on}>]
```

agentinfo	Sets the relay agent info status (RFC3046) off or on. Default is off.	OPTIONAL
agentmismatch	Forward/Drop DHCP reply packet when a relay agent info mismatch is detected (RFC3046). Default is off	OPTIONAL
trace	Disable verbose console logging and generation of debug traces (off) or enable verbose console logging and generation of debug traces (on). By default tracing is disabled.	OPTIONAL

EXAMPLE:

```
=>dhcp relay config
Agent info status : off
Drop agent info mismatch status : off
Verbose console logging : off
=>dhcp relay config agentinfo=on
=>dhcp relay config
Agent info status : on
Drop agent info mismatch status : off
Verbose console logging : off
=>
```

RELATED COMMANDS:

dhcp relay add	Adds a DHCP server to the DHCP server list.
dhcp relay delete	Deletes a DHCP server from the DHCP server list.

dhcp relay delete

Deletes a DHCP server from the DHCP server list.

SYNTAX:

```
dhcp relay delete      addr = <ip-address>
                       [intf = <interface name>]
```

addr	The DHCP server IP address.	REQUIRED
intf	The name of the dynamic interface.	OPTIONAL

RELATED COMMANDS:

dhcp relay add	Adds a DHCP server to the DHCP server list.
dhcp relay config	Sets the relay configuration settings.

dhcp relay flush

Flushes the DHCP relay settings.

Note The flush command does not impact previously saved configurations.

SYNTAX:

```
dhcp relay flush
```

EXAMPLE:

```
=>dhcp relay list
  DHCP server      Interface      giaddr
-----
  127.0.0.1
=>dhcp relay flush
=>dhcp relay list
No dynamic interfaces defined.
=>
```

dhcp relay ifconfig

Configures a relay interface.

SYNTAX:

```
dhcp relay ifconfig    intf = <interface name>
                      [relay = <{off|on}>]
                      [maxhops = <number{0-16}>]
                      [remoteid = <password>]
                      [trusted = <{no|yes}>]
```

intf	The name of the dynamic interface to be configured.	REQUIRED
relay	Sets the relay status.	OPTIONAL
maxhops	A number between 0 and 16. Represents the maximum number of hops allowed in the DHCP packet. Default is 4.	OPTIONAL
remoteid	Sets the remote id as specified in RFC3046.	OPTIONAL
trusted	Drop/Forward DHCP request packet when the DHCP Relay Agent Option is enabled (with the command :dhcp relay config agentinfo=on) and the giaddr field is 0 (RFC3046).	OPTIONAL

EXAMPLE:

```
=>dhcp relay ifconfig
intf = pppoa_pppoa
[relay] = off
[maxhops] = 4
[remoteid] =
[trusted] = no
=>
```

RELATED COMMANDS:

dhcp relay config	Sets the relay configuration settings.
dhcp relay iflist	Show all dynamic interfaces.

dhcp relay iflist

Show all dynamic interfaces.

SYNTAX:

```
dhcp relay iflist [intf = <interface name>]
```

intf	The name of the dynamic interface to be configured.	OPTIONAL
------	---	----------

EXAMPLE:

```
=>dhcp relay iflist
pppoa_pppoa :
  admin state = down   oper state = down
  max hops = 4
  trusted = no        remote id =

eth0 :
  admin state = up     oper state = up
  max hops = 4
  trusted = no        remote id =

=>
```

RELATED COMMANDS:

dhcp relay ifconfig	Configures a relay interface.
---------------------	-------------------------------

dhcp relay list

Lists the DHCP server list.

SYNTAX:

```
dhcp relay list
```

EXAMPLE:

```
=>dhcp relay list
  DHCP server      Interface      giaddr
-----
  127.0.0.1
=>
```

RELATED COMMANDS:

dhcp relay add	Adds a DHCP server to the DHCP server list.
dhcp relay config	Sets the relay configuration settings.
dhcp relay delete	Deletes a DHCP server from the DHCP server list.

dhcp relay stats

Shows DHCP relay statistics.

SYNTAX:

```
dhcp relay stats
```

EXAMPLE:

```
=>dhcp relay stats
  DHCP relay statistics
-----
Client packet relayed   :      64
Server packet relayed  :       0
Bogus relay agent      :       0
Bogus giaddr recv     :       0
Corrupt agent option   :       0
Missing agent option   :       0
Bad circuit id        :       0
Missing circuit id     :       0
=>
```

dhcp server clear

Clear SpeedTouch™ DHCP server statistics.

SYNTAX:

```
dhcp server clear
```

EXAMPLE:

```
=>dhcp client stats
Corrupted packet recv :          0
OFFERS   recv         :        9575
ACKS     recv         :         121
NAKS     recv         :           0
Pure BOOTP REPLIES   :           0
Other message types  :           0
DISCOVERs sent      :        9552
REQUESTs sent       :         142
DECLINEs sent       :           0
RELEASEs sent       :           0
INFORMs sent        :           0
Number of dynamic interfaces: 1
Memory usage:
Table size of dyn leases: 19,   in use: 1,   free: 94 %
=>dhcp server clear
=>dhcp client stats
DHCP client statistics:
Corrupted packet recv :          0
OFFERS   recv         :           0
ACKS     recv         :           0
NAKS     recv         :           0
Pure BOOTP REPLIES   :           0
Other message types  :           0
DISCOVERs sent      :           0
REQUESTs sent       :           0
DECLINEs sent       :           0
RELEASEs sent       :           0
INFORMs sent        :           0
Number of dynamic interfaces: 1
Memory usage:
Table size of dyn leases: 19,   in use: 1,   free: 94 %
=>
```

RELATED COMMANDS:

[dhcp server stats](#)

Show DHCP server statistics.

dhcp server config

Show/set SpeedTouch™ DHCP server configuration settings.

SYNTAX:

```
dhcp server config  [autodhcp = <{off|on}>]
                   [scantime = <number>]
                   [state = <{disabled|enabled}>]
                   [trace = <{off|on}>]
```

autodhcp	Allow the SpeedTouch™ to present itself as DHCP client (AutoDHCP mode) at boot time and probe for another DHCP server on the network for some time before starting its own DHCP server (yes) or immediately start the DHCP server (no).	OPTIONAL
scantime	A number between 0 and 1814400 (seconds). Represents the time the SpeedTouch™ scans for another DHCP server to be active in the network. By default the scantime is 20 seconds.	OPTIONAL
state	State of the DHCP server (enabled/disabled).	OPTIONAL
trace	Disable verbose console logging and generation of debug traces (off) or enable verbose console logging and generation of debug traces (on). By default tracing is disabled.	OPTIONAL

EXAMPLE:

```
=>dhcp server config
autodhcp: on
scantime: 20s
state: disabled
tracing: off
=>dhcp server config scantime=30 tracing=on
=>dhcp server config
autodhcp: on
scantime:30s
state: disabled
tracing: on
=>
```

RELATED COMMANDS:

`dhcp server stats` Show current DHCP server state and statistics.

dhcp server flush

Flushes all DHCP server pool and lease entries.

Note The flush command does not impact previously saved configurations.

SYNTAX:

```
dhcp server flush
```

RELATED COMMANDS:

dhcp server stats	Show current DHCP server state and statistics.
dhcp server config	Show/set current DHCP server configuration.

dhcp server policy

Show/set SpeedTouch™ DHCP server policy.

SYNTAX:

```
dhcp server policy [verifyfirst = <off|on>]
                  [trustclient = <off|on>]
```

verifyfirst	Probe the network for conflicting IP addresses before giving a suggested IP address to the requesting DHCP client (on) or not (off).	OPTIONAL
trustclient	Take the IP address suggested by a DHCP client into account (on) or not (off).	OPTIONAL

EXAMPLE:

```
=>dhcp server policy
Verify first:  off
Trust client:  on
=>dhcp server policy verifyfirst=on trustclient=off
=>dhcp server policy
Verify first:  on
Trust client:  off
=>
```

RELATED COMMANDS:

dhcp server stats	Show current DHCP server state and statistics.
dhcp server config	Show/set current DHCP server configuration.

dhcp server stats

Show SpeedTouch™ DHCP server statistics.

SYNTAX:

```
dhcp server stats
```

EXAMPLE:

```
=>dhcp server stats
DHCP Server State:  Stopped
DHCP server statistics:
Corrupted packet recv      :      0
DISCOVER                   :    2451
REQUEST                    :      28
DECLINE                    :       0
RELEASE                    :      22
INFORM                     :       1
Pure BOOTP REQUESTS       :       2
Other message types       :       0
OFFERS sent                 :    2451
ACKs sent                  :      19
NAKs sent                  :       0
Relay agent options dropped :       0
Lease table got full      : no
Ping table got full       : no
Second dhcp server seen  : no
Total size of lease table: 32, in use: 16, free: 50 %
=>
```

DESCRIPTION:

DHCP server state	Indicates the state of the SpeedTouch™ DHCP server.
Corrupted packet recv	Indicates the number of corrupted packets (not compliant to RFC2131) received from the LAN.
DISCOVER	Indicates the number of DHCP server discovery packets received from the Local Area Network (LAN). These broadcasts are sent by potential DHCP clients to locate available DHCP servers.
REQUEST	Indicates the number of DHCP address lease requests received from the LAN.
DECLINE	Indicates the number of DHCP address lease requests declined.
RELEASE	Indicates the number of DHCP address release requests received from DHCP clients.
INFORM	Indicates the number of information requests received from DHCP clients.
Pure BOOTP requests	Indicates the number of BOOTP requests received from the LAN.
Other message types	Indicates the number of other messages received from the LAN.
OFFERs sent	Indicates the number of IP address offers sent in reply to DHCP requests.
ACKs sent	Indicates the number of ACKnowledgement replies sent to successfully configured DHCP clients.
NAKs sent	Indicates the number of Not-AcKnowledge ment replies sent to wrongly configured DHCP clients.
Relay agent options dropped	
Lease table got full	Indicates whether the maximum number of DHCP leases is reached or not.
Ping table got full	Indicates whether the history list of IP address pings got full or not. These pings are sent by the SpeedTouch™ DHCP server to verify whether the IP address is already in use on the LAN or not. (dhcp server policy verifyfirst=yes)
Second DHCP server seen	Indicates whether a concurrent DHCP server was found on the LAN or not.

RELATED COMMANDS:

dhcp server clear	Clear DHCP server statistics.
-------------------	-------------------------------

dhcp server lease add

Assign a DHCP server lease to a DHCP host in the local network.

SYNTAX:

```
dhcp server lease add  clientid = <client-id>
                        pool = <string>
                        [addr = <ip-address>]
                        [offset = <number>]
                        [leasetime = <number>]
                        [hostname = <{hostname|""}>]
```

clientid	The DHCP client identification string of the booting host.	REQUIRED
pool	The name of the DHCP server pool from which the DHCP lease should be taken from. Use :dhcp server pool list for a list of available DHCP server pools.	REQUIRED
addr	The favoured IP address for this DHCP host. This IP address, if specified, must be in the range of the DHCP pool specified.	OPTIONAL
offset	A number between 0 and the integer number defined by the number of available IP addresses in the DHCP server pool. Represents the IP address offset in the DHCP server pool preserved for this host. Not specifying this parameter does not preserve an IP address for the host.	OPTIONAL
leasetime	A number between 0 and 1814400 (seconds). Represents the time the host is allowed to use this address, before renewing. Specifying -1 makes the lease permanent.	OPTIONAL
hostname	The hostname to add to the local Domain Name System (DNS) table for this host. Use "" if no hostname is associated with this lease.	OPTIONAL

EXAMPLE:

```
=>dhcp server lease list
Lease      Pool      TTL      State    Clientid
0 0.0.0.0  dhcp_pool_1 00:26:40  FREE    00:90:D0:12:34:56
=>dhcp server lease add clientid=01:23:55:67:89:ab pool=Local_pool leasetime=3600
=>dhcp server lease list
Lease      Pool      TTL      State    Clientid
0 0.0.0.0  dhcp_pool_1 00:26:40  FREE    00:90:D0:12:34:56
1 10.0.0.1  local_pool 00:59:22  USED    01:23:45:67:89:AB
=>
```

RELATED COMMANDS:

- `dhcp server lease delete` Delete a DHCP lease.
- `dhcp server lease flush` Delete all DHCP leases.
- `dhcp server lease list` Show current DHCP leases.

dhcp server lease delete

Delete a DHCP lease.

SYNTAX:

```
dhcp server lease delete [clientid = <clientid|none>]
                        [index = <number>]
```

clientid	The DHCP client identification string of the DHCP lease. If no DHCP client is specified, all DHCP clients are deleted.	OPTIONAL
index	The index number of the entry to be deleted. Use :dhcp server lease list to see a list of the index numbers of all current DHCP leases.	OPTIONAL

EXAMPLE:

```
=>dhcp server lease list
Lease      Pool      TTL      State      Clientid
0 0.0.0.0  dhcp_pool_1  00:26:40  FREE      00:90:D0:12:34:56
1 10.0.0.1  local_pool  00:59:22  USED      01:23:45:67:89:AB
=>dhcp server lease delete index=0
=>dhcp server lease list
Lease      Pool      TTL      State      Clientid
1 10.0.0.1  local_pool  00:59:22  USED      01:23:45:67:89:AB
=>
```

RELATED COMMANDS:

dhcp server lease add	Add a DHCP lease manually.
dhcp server lease flush	Delete all DHCP leases.
dhcp server lease list	Show current DHCP leases.

dhcp server lease flush

Flush all DHCP server leases.

Note The flush command does not impact previously saved configurations.

SYNTAX:

```
dhcp server lease flush [pool = <string>]
```

pool	The name of the DHCP server pool. Only the leases belonging to this pool will be deleted.	OPTIONAL
------	---	----------

EXAMPLE:

```
=>dhcp server lease list
Lease      Pool      TTL      State      Clientid
0 0.0.0.0   dhcp_pool_1 00:26:40   FREE      00:90:D0:12:34:56
1 10.0.0.1  local_pool 00:59:22   USED      01:23:45:67:89:AB
2 10.0.0.101 local_pool 00:21:01   USED      01:23:89:AB:80:CD
3 10.0.0.132 local_pool 00:45:37   USED      09:D0:25:CE:F1:31
5 10.0.0.5  local_pool 00:21:11   USED      AB:33:A1:7C:89:DD
4 10.0.0.6  local_pool 00:59:01   USED      E3:81:9F:11:11:11
8 10.0.0.8  local_pool 00:01:00   USED      08:80:09:90:AB:DC
9 10.0.0.15 local_pool 00:00:23   USED      08:93:DA:AE:01:AF
=>dhcp server lease flush
=>dhcp server lease list
=>
```

RELATED COMMANDS:

dhcp server lease add	Add a DHCP lease manually.
dhcp server lease delete	Delete a DHCP lease.
dhcp server lease list	Show current DHCP leases.

dhcp server lease list

List current DHCP leases, indicated by their index number.

SYNTAX:

```
dhcp server lease list [clientid = <clientid|none>]
                      [index = <number>]
```

clientid	The DHCP client identification string of the DHCP lease. If no DHCP client is specified, all DHCP clients are listed.	OPTIONAL
index	The index number of the entry to be deleted.	OPTIONAL

EXAMPLE:

```
=>dhcp server lease list
Lease      Pool      TTL      State      Clientid
0 0.0.0.0  dhcp_pool_1 00:26:40  FREE      00:90:D0:12:34:56
1 10.0.0.1  local_pool 00:59:22  USED      01:23:45:67:89:AB
2 10.0.0.101 local_pool 00:21:01  USED      01:23:89:AB:80:CD
3 10.0.0.132 local_pool 00:45:37  USED      09:D0:25:CE:F1:31
5 10.0.0.5  local_pool 00:21:11  USED      AB:33:A1:7C:89:DD
4 10.0.0.6  local_pool 00:59:01  USED      E3:81:9F:11:11:11
8 10.0.0.8  local_pool 00:01:00  USED      08:80:09:90:AB:DC
9 10.0.0.15 local_pool 00:00:23  USED      08:93:DA:AE:01:AF
=>
```

RELATED COMMANDS:

dhcp server lease add	Add a DHCP lease manually.
dhcp server lease delete	Delete a DHCP lease.
dhcp server lease flush	Delete complete DHCP server configuration and dynamic leases.

dhcp server pool add

Add a DHCP server pool.

SYNTAX:

```
dhcp server pool add [name = <string>]
                    [index = <number>]
```

name	A name for the DHCP server pool. If not specified, the name is "dhcp_pool_x", where x is a subsequent number.	OPTIONAL
index	The number of the pool before which you want the new pool to be added.	OPTIONAL

EXAMPLE:

```
=>dhcp server pool list
Pool      Start      End        State      PPP
0 dhcp_pool_1  0.0.0.0    0.0.0.0    FREE
1 My_LAN_Pool 10.0.0.1    10.0.0.254 USED
2 dhcp_pool_2  0.0.0.0    0.0.0.0    FREE
=>dhcp server pool add
=>dhcp server pool list
Pool      Start      End        State      PPP
0 dhcp_pool_1  0.0.0.0    0.0.0.0    FREE
1 My_LAN_Pool 10.0.0.1    10.0.0.254 USED
2 dhcp_pool_2  0.0.0.0    0.0.0.0    FREE
3 dhcp_pool_3  0.0.0.0    0.0.0.0    FREE
=>dhcp server pool add name=POOL_EXTRA1
=>dhcp server pool list
Pool      Start      End        State      PPP
0 dhcp_pool_1  0.0.0.0    0.0.0.0    FREE
1 My_LAN_Pool 10.0.0.1    10.0.0.254 USED
2 dhcp_pool_2  0.0.0.0    0.0.0.0    FREE
3 dhcp_pool_3  0.0.0.0    0.0.0.0    FREE
4 POOL_EXTRA1 0.0.0.0    0.0.0.0    FREE
=>ppp ifconfig name=PPP_Test pool=POOL_EXTRA1
=>dhcp server pool list
Pool      Start      End        State      PPP
0 dhcp_pool_1  0.0.0.0    0.0.0.0    FREE
1 My_LAN_Pool 10.0.0.1    10.0.0.254 USED
2 dhcp_pool_2  0.0.0.0    0.0.0.0    FREE
3 dhcp_pool_3  0.0.0.0    0.0.0.0    FREE
4 POOL_EXTRA1 0.0.0.0    0.0.0.0    FREE      PPP_Test
=>
```

RELATED COMMANDS:

dhcp server pool delete	Delete a DHCP pool.
dhcp server pool flush	Delete all DHCP pools.
dhcp server pool list	Show current DHCP pools.

dhcp server pool config

Configure an existing DHCP pool. Before you are able to configure the DHCP pool, you must create it via :dhcp server pool add.

SYNTAX:

```
dhcp server pool config name = <string>
                        [index = <number>]
                        [intf = <string>]
                        [poolstart = <ip-address>]
                        [poolend = <ip-address>]
                        [netmask = <ip-mask(dotted or cidr)>]
                        [gateway = <ipaddress | 0>]
                        [server = <ipaddress | 0>]
                        [primdns = <ipaddress | 0>]
                        [secdns = <ipaddress | 0>]
                        [dnsmetric = <number{0-100}>]
                        [primwins = <ipaddress | 0>]
                        [secwins = <ipaddress | 0>]
                        [leasetime = <number>]
                        [unnumbered = <{no|yes}>]
                        [localgw = <{off|on}>]
```

name	The name of the DHCP server pool to configure.	REQUIRED
index	A number between 0 (highest priority) and the highest number (lowest priority) found in the list of existing DHCP server pools. Represents a (higher) priority for the DHCP server pool.	OPTIONAL
intf	The interface for which the pool is allowed to lease IP addresses.	OPTIONAL
poolstart	The lowest IP address in the DHCP address range to use for leasing. Default value of this parameter is 0.0.0.0 (not specified), which means that the lowest IP address of the pool will be defined by the remote server via Internet Protocol Control Protocol (IPCP) as soon as the Point-to-Point Protocol (PPP) IPCP subnetmasking connection is established.	OPTIONAL
poolend	The highest IP address in the DHCP address range to use for leasing. Default value of this parameter is 0.0.0.0 (not specified), which means that the highest IP address of the pool will be defined by the remote server via IPCP as soon as the PPP IPCP subnetmasking connection is established.	OPTIONAL
netmask	The applicable netmask for the DHCP leases.	OPTIONAL

gateway	The IP address of the default gateway for the DHCP clients. Default value of this parameter is 0 (not specified), which means that the gateway IP address will be communicated by the remote server as soon as the PPP IPCP subnetmasking connection is established or that the SpeedTouch™ acts as the LAN default gateway.	OPTIONAL
server	The IP address of the DHCP server for DHCP clients.	OPTIONAL
primdns	The IP address of the primary DNS server for the DHCP clients. Default value of this parameter is 0 (not specified), which means that the IP address of the DNS server will be communicated by the remote server as soon as the PPP IPCP subnetmasking connection is established or that the SpeedTouch™ acts as the LAN DNS server.	OPTIONAL
secdns	The IP address of the optional secondary DNS server for DHCP clients. Default value of this parameter is 0 (not specified), which means that the gateway IP address will be communicated by the remote server as soon as the PPP IPCP subnetmasking connection is established.	OPTIONAL
dnsmetric	The DHCP server pool DNS route metric.	OPTIONAL
primwins	The IP address of the primary WINS server for DHCP clients.	OPTIONAL
secwins	The IP address of the secondary WINS server for DHCP clients.	OPTIONAL
leasetime	A number between 0 and 1814400 (seconds). Represents the time for which a client can use its dynamically allocated IP address. By default the leasetime is 2 hours (7200 seconds). Specifying -l makes the lease permanent.	OPTIONAL
unnumbered	Assign an IP address from this pool to the DHCP server or not (dynamic pools only).	OPTIONAL
localgw	Proxy for a virtual default gateway residing in same subnet of DHCP client instead of the remote peer address.	OPTIONAL

EXAMPLE:

```

=>dhcp server pool list
Pool      Start      End        Intf      State
0 LAN_Private 10.0.0.1   10.0.0.254 eth0      USED
=>dhcp server pool config name=My_Pool poolstart=192.6.11.101 |
poolend=192.6.11.254 netmask=255.255.255 gateway=192.6.11.100 leasetime=21600
=>dhcp server pool list
Pool      Start      End        Intf      State
0 LAN_Private 10.0.0.1   10.0.0.254 eth0      USED
1 My_Pool   192.6.11.101 192.6.11.254 eth0      USED
=>

```

RELATED COMMANDS:

dhcp server pool list Show current DHCP pools.

dhcp server pool delete

Delete a DHCP pool.

SYNTAX:

```
dhcp server pool delete      name = <string>
```

name	The name of the DHCP server pool to delete. Use :dhcp server pool list to see a list of all current DHCP leases.	REQUIRED
------	---	----------

EXAMPLE:

```
=>dhcp server pool list
Pool      Start      End      Intf      State
0 LAN_Private  10.0.0.1  10.0.0.254  eth0      USED
1 My_Pool    192.6.11.101  192.6.11.254  eth0      USED
=>dhcp server pool delete name=My_Pool
=>dhcp server pool list
Pool      Start      End      Intf      State
0 LAN_Private  10.0.0.1  10.0.0.254  eth0      USED
=>
```

RELATED COMMANDS:

dhcp server pool add	Add a DHCP pool.
dhcp server pool flush	Delete all DHCP pools.
dhcp server pool list	Show current DHCP pools.

dhcp server pool flush

Flush all DHCP pools

Note The flush command does not impact previously saved configurations.

SYNTAX:

```
dhcp server pool flush
```

EXAMPLE:

```
=>dhcp server pool list
Pool      Start          End            Intf          State
0 LAN_Private 10.0.0.1       10.0.0.254   eth0         USED
1 My_Pool    192.6.11.101  192.6.11.254 eth0         USED
=>dhcp server pool flush
=>dhcp server pool list
=>
```

RELATED COMMANDS:

<code>dhcp server pool add</code>	Add a DHCP pool.
<code>dhcp server pool delete</code>	Delete a DHCP pool.
<code>dhcp server pool list</code>	Show current DHCP pools.

dhcp server pool list

List current DHCP pools.

SYNTAX:

```
dhcp server pool list
```

EXAMPLE:

```
=>dhcp server pool list
Pool      Start      End        Intf      State
0 LAN_Private 10.0.0.1   10.0.0.254 eth0      USED
1 My_Pool   192.6.11.101 192.6.11.254 eth0      USED
=>
```

RELATED COMMANDS:

dhcp server pool add	Add a DHCP pool.
dhcp server pool delete	Delete a DHCP pool.
dhcp server pool flush	Delete all DHCP pools.

DNS Commands

Contents

This chapter covers the following commands:

Topic	Page
dns add	132
dns clear	133
dns clrstats	134
dns delete	135
dns domain	136
dns flush	137
dns fwdadd	138
dns fwddelete	139
dns fwdlist	140
dns fwdtable	141
dns list	142
dns nslookup	143
dns start	144
dns stats	145
dns status	146
dns stop	147
dns toutfwd	148
dns troff	149
dns tron	150

dns add

Adds an entry to the local Domain Name System (DNS) table.

SYNTAX:

```
dns add          hostname = <string>
                 [addr = <ip-address>]
```

hostname	The name of the IP host to add (without the (sub)domain name).	REQUIRED
addr	The IP address of the host (without mask).	OPTIONAL
Note	If this parameter is not specified, the hostname applies to the SpeedTouch™ itself.	

EXAMPLE:

```
=>dns list
Domain: business.lan
Nr.      Hostname      IP Address
0        SpeedTouch™    *.*.*.*
1        TestHost       10.0.0.140
2        HTTP_Server  10.0.0.8
Total Table Size: 73 entries
Amount used: 3 (4%)
=>dns add hostname=FTP_Server addr=10.0.0.7
=>dns list
Domain: business.lan
Nr.      Hostname      IP Address
0        SpeedTouch™    *.*.*.*
1        TestHost       10.0.0.140
2        HTTP_Server  10.0.0.8
3        FTP_Server    10.0.0.7
Total Table Size: 73 entries
Amount used: 4 (5%)
=>
```

RELATED COMMANDS:

dns list	Lists the contents of the local DNS table.
dns delete	Deletes an entry from the local DNS table by index.

dns clear

Clears the local DNS table.

SYNTAX:

```
dns clear
```

EXAMPLE:

```
=>dns list
Domain: business.lan
Nr.      Hostname      IP Address
0        SpeedTouch™   *.*.*.*
1        TestHost      10.0.0.140
2        HTTP_Server   10.0.0.8
3        FTP_Server    10.0.0.7
Total Table Size: 73 entries
Amount used: 4 (5%)
=>dns clear
=>dns list
Domain: business.lan
Nr.      Hostname      IP Address
Total Table Size: 73 entries
Amount used: 0 (0%)
=>
```

RELATED COMMANDS:

`dns list`

Lists the contents of the local DNS table.

dns clrstats

Clears the DNS statistics.

SYNTAX:

```
dns clrstats
```

EXAMPLE:

```
=>dns stats
DNS Statistics:
Corrupted packets recv      :    0
Local questions resolved   :    0
Local neg answers sent     :    4
Total DNS packets fwd     :    0
External answers recv     :    0
Fwd table full, discard   :    0
Spurious answers          :    0
Unknown query types       :    0

Total number of packets received :    4

=>dns clrstats
DNS statistics cleared.
=>dns stats
DNS Statistics:
Corrupted packets recv      :    0
Local questions resolved   :    0
Local neg answers sent     :    0
Total DNS packets fwd     :    0
External answers recv     :    0
Fwd table full, discard   :    0
Spurious answers          :    0
Unknown query types       :    0

Total number of packets received :    0

=>
```

RELATED COMMANDS:

[dns stats](#)

Prints the DNS server/forwarder statistics.

dns delete

Deletes an entry from the local DNS table by index.

SYNTAX:

```
dns delete          index = <number>
```

index	The index of the entry to be deleted. Use dns list to see a list of the index numbers of all current DNS entries.	REQUIRED
-------	--	----------

EXAMPLE:

```
=>dns list
Domain: business.lan
Nr.      Hostname      IP Address
0        SpeedTouch™   *.*.*.*
1        TestHost      10.0.0.140
2        HTTP_Server   10.0.0.8
3        FTP_Server    10.0.0.7
Total Table Size: 73 entries
Amount used: 4 (5%)
=>dns delete index=2
=>dns list
Domain: business.lan
Nr.      Hostname      IP Address
0        SpeedTouch™   *.*.*.*
1        TestHost      10.0.0.140
3        FTP_Server    10.0.0.7
Total Table Size: 73 entries
Amount used: 3 (4%)
=>
```

RELATED COMMANDS:

`dns add` Adds an entry to the local DNS table.

`dns list` List current DNS entries.

dns domain

Set the local DNS domain name.

SYNTAX:

```
dns domain          domain = <string>
```

domain	The local DNS domain name.	REQUIRED
--------	----------------------------	----------

EXAMPLE:

```
=>dns list
Domain: business.lan
Nr.      Hostname      IP Address
0        SpeedTouch™   *.*.*.*
1        TestHost      10.0.0.140
2        HTTP_Server   10.0.0.8
3        FTP_Server    10.0.0.7
Total Table Size: 73 entries
Amount used: 4 (5%)
=>dns domain domain=office.home.lan
=>dns list
Domain: office.home.lan
Nr.      Hostname      IP Address
0        SpeedTouch™   *.*.*.*
1        TestHost      10.0.0.140
2        HTTP_Server   10.0.0.8
3        FTP_Server    10.0.0.7
Total Table Size: 73 entries
Amount used: 4 (5%)
```

RELATED COMMANDS:

dns list	Lists the contents of the local DNS table.
--------------------------	--

dns flush

Flush the complete SpeedTouch™ DNS server/forwarder configuration and static entries.

Note The flush command does not impact previously saved configurations.

SYNTAX:

```
dns flush
```

EXAMPLE:

```
=>dns list
Domain: office.home.lan
Nr.      Hostname      IP Address
4*       Z7V1D8       10.0.0.29
0        SpeedTouch™  *.*.*.*
1        TestHost     10.0.0.140
2        Default     10.0.0.8
3        ftpserver   172.16.0.1
Total Table Size: 73 entries
Amount used: 5 (6%)
=>dns flush
=>dns list
Domain: lan
Nr.      Hostname      IP Address
3*       Z7V1D8       10.0.0.29
Total Table Size: 73 entries
Amount used: 1 (1%)
=>
```

dns fwdadd

Add a DNS forwarding entry. The entries in the forwarding list determine which DNS server should be used for which PC. If an identification cannot be established within the local LAN, the request is forwarded to another DNS server, on another network (Internet/LAN to LAN connection). The connection is negotiated within a PPP link.

SYNTAX:

```
dns fwdadd      dns = <ip-address>
                src = <ip-address>
                mask = <ip-mask (dotted or cidr)>
                [metric = <number{0-100}>]
                [direct = <no/yes>]
```

dns	The IP address of the (remote) DNS server.	REQUIRED
src	The source IP address (pool) of the host(s) using this DNS server.	REQUIRED
mask	The appropriate source IP (sub)netmask.	REQUIRED
metric	A number between 0 and 100 Represents the metric (weight factor) for this DNS route.	OPTIONAL
direct	Selects whether DNS replies are sent directly back to the client versus being relayed by the DNS forwarder.	OPTIONAL

EXAMPLE:

```
=>dns fwdlist
DNS          SRC          Metric Direct Permanent
10.0.0.138   10.0.0.2          1      no    yes
=>dns fwdadd dns=10.0.0.138 src=10.0.0.3 mask=24 direct=1
Dns forwarding server added.
=>dns fwdlist
DNS          SRC          Metric Direct Permanent
10.0.0.138   10.0.0.2          1      no    yes
10.0.0.138   10.0.0.3          1      yes   yes
=>
```

RELATED COMMANDS:

<code>dns fwddelete</code>	Deletes a DNS forwarding entry.
<code>dns fwdlist</code>	Lists all forwarding entries.

dns fwdelete

Delete a DNS forwarding entry.

SYNTAX:

```
dns fwdelete    src = <ip-address>
                mask = <ip-mask (dotted or cidr)>
                [dns = <ip-address>]
```

src	The source IP address (pool) of the hosts to remove the entry for.	REQUIRED
mask	The source IP (sub)netmask.	REQUIRED
dns	The IP address of the (remote) DNS server (in case of multiple DNS server entries).	OPTIONAL

EXAMPLE:

```
=>dns fwdlist
DNS          SRC          Metric Direct Permanent
10.0.0.138   10.0.0.2          1      no     yes
10.0.0.138   10.0.0.3          1      yes    yes
=>dns fwdelete dns=10.0.0.138 src=10.0.0.3 mask=24 direct=1
Dns forwarding server deleted.
=>dns fwdlist
DNS          SRC          Metric Direct Permanent
10.0.0.138   10.0.0.2          1      no     yes
=>
```

RELATED COMMANDS:

- `dns fwdadd` Add a DNS forwarding entry.
- `dns fwdlist` Lists all forwarding entries.

dns fwdlist

Lists all forwarding entries.

SYNTAX:

```
dns fwdlist
```

EXAMPLE:

```
=>dns fwdlist
DNS forwarding servers:
DNS      SRC          MASK          Direct
10.0.0.138  10.0.0.0      255.255.255.0  yes
192.6.11.150  192.6.11.0   255.255.255.0  yes
=>
```

RELATED COMMANDS:

- | | |
|-------------------------------|--------------------------------|
| dns fwdadd | Add a DNS forwarding entry. |
| dns fwddelete | Delete a DNS forwarding entry. |
| dns fwdtable | Show DNS forwarding table. |

dns fwdtable

Show DNS forwarding table, i.e. list all currently unresolved DNS requests.

SYNTAX:

```
dns fwdtable
```

EXAMPLE:

```
=>dns fwdtable
Forwarding table:
Nr.  Ip Address      (port#):id(hex)  (expiry)      dns server      tries
0    10.10.10.12     (54751):8331     (13 sec)      10.10.10.112   1
Timeout: 15 seconds
Table size: 10
amount of table used: 1 (10%)
=>
```

RELATED COMMANDS:

`dns fwdlist` Show current DNS forwarding entries.

dns list

Lists the contents of the local DNS table.

SYNTAX:

```
dns list
```

EXAMPLE:

```
=>dns list
Domain: office.home.lan
Nr.      Hostname      IP Address
4*       Z7V1b8       10.0.0.29
0        SpeedTouch™   *.*.*.*
1        TestHost     10.0.0.140
2        Default      10.0.0.8
3        ftpserver    172.16.0.1
Total Table Size: 73 entries
Amount used: 5 (6%)
=>
```

EXAMPLE INPUT/OUTPUT IN A NETWORKED ENVIRONMENT:

The SpeedTouch™ is configured as DNS server.

```
=>dns list
Domain: SpeedLAN.local
Nr.      Hostname      IP Address
0        SpeedTouch™   *.*.*.*
1        Server        10.10.1.1
2        Client        10.0.0.3
Total Table Size: 73 entries
Amount used: 3 (4%)
=>
```

RELATED COMMANDS:

- `dns add` Adds an entry to the local DNS table.
- `dns delete` Deletes an entry from the local DNS table by index.

dns nslookup

Looks up a name or an IP address via local DNS.

SYNTAX:

```
dns nslookup      lookup = <string>
```

lookup	The DNS hostname or IP address to query.	REQUIRED
--------	--	----------

EXAMPLE:

```
=>dns list
Domain: office.home.lan
Nr.      Hostname      IP Address
4*       Z7V1D8       10.0.0.29
0        SpeedTouch™   *.*.*.*
1        TestHost     10.0.0.140
2        Default     10.0.0.8
3        ftpserver    172.16.0.1
Total Table Size: 73 entries
Amount used: 5 (6%)
=>dns nslookup lookup=TestHost
Name:    TestHost
Address: 10.0.0.140
=>dns nslookup lookup=10.0.0.29
Name:    Z7V1D8
Address: 10.0.0.29
=>
```

RELATED COMMANDS:

dns list	Lists the contents of the local DNS table.
----------	--

dns start

Starts the local DNS server and forwarder.

SYNTAX:

```
dns start
```

EXAMPLE:

```
=>dns status
DNS server status: Stopped
DNS table size           : 73, in use: 4, free: 94 %
DNS forwarding table size : 10, in use: 0, free: 100 %
DNS forwarding dns servers table size : 25, in use: 4, free: 84 %
No dns cache.
Tracing: off
=>dns start
DNS server started.
=>dns status
DNS server status: started
DNS table size           : 73, in use: 4, free: 94 %
DNS forwarding table size : 10, in use: 0, free: 100 %
DNS forwarding dns servers table size : 25, in use: 4, free: 84 %
No dns cache.
Tracing: off
=>
```

RELATED COMMANDS:

- `dns status` Shows the current status of the DNS server/forwarder.
- `dns stop` Stops the local DNS server/forwarder.

dns stats

Show SpeedTouch™ DNS server/forwarder statistics.

SYNTAX:

```
dns stats
```

EXAMPLE INPUT/OUTPUT IN A NETWORKED ENVIRONMENT:

The SpeedTouch™ is configured as DNS server.

```
=>dns list
Domain: SpeedLAN.local
Nr.      Hostname      IP Address
0        SpeedTouch™  *.*.*.*
1        Server       10.10.1.1
2        Client       10.0.0.3
Total Table Size: 73 entries
Amount used: 3 (4%)
=>dns stats
DNS Statistics:
Corrupted packets recv      :    0
Local questions resolved   :    1
Local neg answers sent     :    0
Total DNS packets fwd      :    0
External answers recv      :    0
Fwd table full, discard    :    0
Spurious answers           :    0
Unknown query types        :    0
Total number of packets received :    1
=>(Ping Client.SpeedLAN.local)
=>(CTRL + Q)
dnisd: Internet class type A request received from 10.10.1.1.
dnisd: Client.SpeedLAN.local found in local database.
dnisd: Client.SpeedLAN.local resolved into 10.0.0.3.
=>(Ping Server.SpeedLAN.local)
dnisd: Internet class type A request received from 10.10.1.1.
dnisd: Server.SpeedLAN.local found in local database.
dnisd: Server.SpeedLAN.local resolved into 10.0.0.3.
=>(CTRL + S)
=>dns stats
DNS Statistics:
Corrupted packets recv      :    0
Local questions resolved   :    3
Local neg answers sent     :    0
Total DNS packets fwd      :    0
External answers recv      :    0
Fwd table full, discard    :    0
Spurious answers           :    0
Unknown query types        :    0
Total number of packets received :    3
=>
```

RELATED COMMANDS:

`dns clrstats`

Clear the DNS server/forwarder statistics.

dns status

Shows the current status of the DNS server/forwarder.

SYNTAX:

```
dns status
```

EXAMPLE:

```
=>dns status
DNS server status: stopped
DNS table size           : 73, in use: 4, free: 94 %
DNS forwarding table size : 10, in use: 0, free: 100 %
DNS forwarding dns servers table size : 25, in use: 4, free: 84 %
No dns cache.
Tracing: off
=>
```

dns stop

Stops the local DNS server/forwarder.

SYNTAX:

```
dns stop
```

EXAMPLE:

```
=>dns status
DNS server status: started
DNS table size           : 73, in use: 4, free: 94 %
DNS forwarding table size : 10, in use: 0, free: 100 %
DNS forwarding dns servers table size : 25, in use: 4, free: 84 %
No dns cache.
Tracing: off
=>dns stop
DNS server stopped.
=>dns status
DNS server status: stopped
DNS table size           : 73, in use: 4, free: 94 %
DNS forwarding table size : 10, in use: 0, free: 100 %
DNS forwarding dns servers table size : 25, in use: 4, free: 84 %
No dns cache.
Tracing: off
=>
```

RELATED COMMANDS:

- | | |
|-------------------------|---|
| <code>dns status</code> | Shows the current status of the DNS server/forwarder. |
| <code>dns start</code> | Starts the local DNS server and forwarder. |

dns toutfwd

Sets the DNS forwarding timeout.

SYNTAX:

```
dns toutfwd      timeout = <number>
```

timeout	A number (seconds). Represents the query forwarding timeout. This parameter determines how long the SpeedTouch™ DNS server should try to contact a (remote) DNS server before (temporarily) declaring the DNS requests unresolved. By default the timeout is 15 seconds.	REQUIRED
---------	---	----------

EXAMPLE:

```
=>dns fwdtable
Forwarding table:
Nr. Ip Address      (port#):id(hex)  (expiry)   dns server   tries
0   10.10.10.12     (54751):8331    (13 sec)   10.10.10.112 1
Timeout: 15 seconds
Table size: 10
amount of table used: 1 (10%)
=>dns toutfwd timeout=20
Current timeout: 15 seconds
Timeout set to: 20 seconds
=>dns fwdtable
Forwarding table:
Nr. Ip Address      (port#):id(hex)  (expiry)   dns server   tries
0   10.10.10.12     (54751):8331    (13 sec)   10.10.10.112 1
Timeout: 20 seconds
Table size: 10
amount of table used: 1 (10%)
=>
```

RELATED COMMANDS:

dns fwdadd	Adds a DNS forwarding entry.
dns fwddelete	Deletes a DNS forwarding entry.
dns fwdlist	Shows the current DNS forwarding entries.
dns fwdtable	Shows the DNS forwarding table.

dns troff

Disable verbose console messaging. No debug traces are generated.

SYNTAX:

```
dns troff
```

EXAMPLE:

```
=>dns status
DNS server status: started
DNS table size           : 73, in use: 4, free: 94 %
DNS forwarding table size : 10, in use: 0, free: 100 %
DNS forwarding dns servers table size : 25, in use: 4, free: 84 %
No dns cache.
Tracing: on
=>dns troff
=>dns status
DNS server status: started
DNS table size           : 73, in use: 4, free: 94 %
DNS forwarding table size : 10, in use: 0, free: 100 %
DNS forwarding dns servers table size : 25, in use: 4, free: 84 %
No dns cache.
Tracing: off
=>
```

RELATED COMMANDS:

dns fwdlist	Shows the current DNS forwarding entries.
dns fwdtable	Shows the DNS forwarding table.
dns status	Shows the current status of the DNS server/forwarder.
dns tron	Enable verbose console messaging.

dns tron

Enable verbose console messaging. Debug traces are generated.

SYNTAX:

```
dns tron
```

EXAMPLE:

```
=>dns status
DNS server status: started
DNS table size           : 73, in use: 4, free: 94 %
DNS forwarding table size : 10, in use: 0, free: 100 %
DNS forwarding dns servers table size : 25, in use: 4, free: 84 %
No dns cache.
Tracing: off
=>dns tron
Tracing on.
=>dns status
DNS server status: started
DNS table size           : 73, in use: 4, free: 94 %
DNS forwarding table size : 10, in use: 0, free: 100 %
DNS forwarding dns servers table size : 25, in use: 4, free: 84 %
No dns cache.
Tracing: on
=>(CTRL + Q)
dnisd: Internet class type A request received from 10.0.0.10.
dnisd: aa.aa.be is outside our domain: forward.
dnisd: forwarding request from 10.0.0.10 (1318,0x0001) to 138.203.68.61
      (try=1): 'reply to ant' mode.
dnisd: Internet class type A request received from 10.0.0.10.
dnisd: aa.aa.be is outside our domain: forward.
dnisd: forwarding request from 10.0.0.10 (1318,0x0001) to 138.203.68.11
      (try=2): 'reply to ant' mode.
dnisd: forward answer from 138.203.68.11 to 10.0.0.10 (1318,0001).
dnisd: Internet class type A request received from 10.0.0.10.
dnisd: aa.aa.be.lan unknown: return error.
.....
=>(CTRL + S)
```

RELATED COMMANDS:

dns fwdlist	Shows the current DNS forwarding entries.
dns fwdtable	Shows the DNS forwarding table.
dns status	Shows the current status of the DNS server/forwarder.
dns troff	Disable verbose console messaging.

Env Commands

Contents

This chapter covers the following commands

Topic	Page
env flush	152
env get	154
env list	156
env set	157
env unset	158

env flush

Flush all non-system environment variables.

SYNTAX:

```
env flush
```


EXAMPLE:

```

=>env list
_COMPANY_NAME=THOMSON multimedia
_COMPANY_URL=http://www.thomson-multimedia.com
_PROD_NAME=SpeedTouch
_PROD_URL=http://www.speedtouch.com
_PROD_FRIENDLY_NAME=SpeedTouch 510
_PROD_DESCRIPTION=DSL Internet Gateway
_PROD_NUMBER=510
_BOARD_SERIAL_NBR=0238J1035
_BUILD=4.0.0.9.0
_BUILDNAME=Sascha4.0090
_PRL=3EC36939AAAA
_FIA=ND
_BOARD_NAME=ADNT-Q
_COMPANY_ID=ALCL
_COPYRIGHT=Copyright (c) 1999-2002, THOMSON multimedia
_TPVERSION=1.1.0
_MACADDR=00-90-D0-03-02-01
_UDN=uuid:UPnP-SpeedTouch510-1_00-90-D0-03-02-01
CONF_REGION=world
CONF_PROVIDER=Generic ISP
CONF_DESCRIPTION=Relayed PPPoA Packet Service configuration
CONF_SERVICE=Relayed PPPoA
CONF_VERSION=4.1.1.0
HOST_SETUP=auto
UPGRADE_URL=http://www.speedtouch.com/upgrade500.htm
COLUMNS=80
ROWS=24
SESSIONTIMEOUT=0
DSL_ADDR=8*35
CONF_DATE=January 2003
=>env flush
=>env list
_COMPANY_NAME=THOMSON multimedia
_COMPANY_URL=http://www.thomson-multimedia.com
_PROD_NAME=SpeedTouch
_PROD_URL=http://www.speedtouch.com
_PROD_FRIENDLY_NAME=SpeedTouch 510
_PROD_DESCRIPTION=DSL Internet Gateway
_PROD_NUMBER=510
_BOARD_SERIAL_NBR=0238J1035
_BUILD=4.0.0.9.0
_BUILDNAME=LLT6AA4.0090
_PRL=3EC36939AAAA
_FIA=ND
_BOARD_NAME=ADNT-Q
_COMPANY_ID=ALCL
_COPYRIGHT=Copyright (c) 1999-2002, THOMSON multimedia
_TPVERSION=1.1.0
_MACADDR=00-90-D0-03-02-01
_UDN=uuid:UPnP-SpeedTouch510-1_00-90-D0-03-02-01
=>

```

RELATED COMMANDS:

`env list` List all environment variables.

env get

Get the current value of a environment variable.

SYNTAX:

```
env get          var = <string>
```

var

The name of the environment variable.
Execute env list to see a list of all environment variables.

REQUIRED

EXAMPLE:

```
=>env list
_COMPANY_NAME=THOMSON
_COMPANY_URL=http://www.thomson.net
_PROD_NAME=SpeedTouch
_PROD_URL=http://www.speedtouch.com
_PROD_FRIENDLY_NAME=SpeedTouch 610
_PROD_DESCRIPTION=Business DSL Router
_PROD_NUMBER=610
_BOARD_SERIAL_NBR=0238L1949
_BUILD=4.2.0.10
_MODEMLABEL=MOD_SACHEM_SEA_CPE_V2.11.27
_PHYSLAYERTYPE=POTS
_BUILDNAME=LLT5AA4.20A
_PRL=3EC18603GCAA
_FIA=NE
_BOARD_NAME=ADNT-R
_COMPANY_ID=ALCL
_COPYRIGHT=Copyright (c) 1999-2003, THOMSON
_TPVERSION=1.2.0
_ETHERNET=SWITCH
_MACADDR=00-90-D0-72-88-64
_UDN=uuid:UPnP-SpeedTouch610-1_00-90-D0-72-88-64
_IGDX_VERSION=1.1
CONF_REGION=world
CONF_PROVIDER=Any
CONF_DESCRIPTION=Routed PPPoA Packet Service configuration using always-on session connectivity.
CONF_SERVICE=Routed PPPoA - DHCP - NATP
CONF_DATE=Configuration set by Embedded wizard
CONF_TPVERSION=1.2.0
HOST_SETUP=auto
HOST_LANGUAGE=en
UPGRADE_URL=http://www.speedtouch.com/upgrade600.htm
ATM_addr=8.35
SNMP_public=public
SNMP_private=private
PPP_user=johndoe@ISP
CONF_TEMPLATE=pppoa.tpl
CONF_USR_COMMENT=Configuration created by Embedded wizard (profile:active/pppoe.tpl)
COLUMNS=80
ROWS=24
SESSIONTIMEOUT=0
=>env get var=ATM_addr
8*35
=>
```

RELATED COMMANDS:

- | | |
|-----------------------|---|
| <code>env list</code> | List all current environment variables. |
| <code>env set</code> | Create and set a non-system environment variable. |

env list

Show all currently available environment variables.

SYNTAX:

```
env list
```

EXAMPLE:

```
=>env list
_COMPANY_NAME=THOMSON
_COMPANY_URL=http://www.thomson.net
_PROD_NAME=SpeedTouch
_PROD_URL=http://www.speedtouch.com
_PROD_FRIENDLY_NAME=SpeedTouch 610
_PROD_DESCRIPTION=Business DSL Router
_PROD_NUMBER=610
_BOARD_SERIAL_NBR=0238L1949
_BUILD=4.2.0.10
_MODEMLABEL=MOD_SACHEM_SEA_CPE_V2.11.27
_PHYSLAYERTYPE=POTS
_BUILDNAME=LLT5AA4.20A
_PRL=3EC18603GCAA
_FIA=NE
_BOARD_NAME=ADNT-R
_COMPANY_ID=ALCL
_COPYRIGHT=Copyright (c) 1999-2003, THOMSON
_TPVERSION=1.2.0
_ETHERNET=SWITCH
_MACADDR=00-90-D0-72-88-64
_UDN=uuid:UPnP-SpeedTouch610-1_00-90-D0-72-88-64
_IGDX_VERSION=1.1
CONF_REGION=world
CONF_PROVIDER=Any
CONF_DESCRIPTION=Routed PPPoA Packet Service configuration using always-on session connectivity.
CONF_SERVICE=Routed PPPoA - DHCP - NATP
CONF_DATE=Configuration set by Embedded Wizard
CONF_TPVERSION=1.2.0
HOST_SETUP=auto
HOST_LANGUAGE=en
UPGRADE_URL=http://www.speedtouch.com/upgrade600.htm
ATM_addr=8.35
SNMP_public=public
SNMP_private=private
PPP_user=johndoe@ISP
CONF_TEMPLATE=pppoa.tpl
CONF_USR_COMMENT=Configuration created by Embedded wizard (profile:active/pppoe.tpl)
COLUMNS=80
ROWS=24
SESSIONTIMEOUT=0
=>
```

RELATED COMMANDS:

[env set](#)

Create and set a non-system environment variable.

[env unset](#)

Delete a non-system environment variable.

env set

Create and set a non-system environment variable or change the value of a non-system environment variable.

SYNTAX:

```
env set          var = <string>
                value = <translated string>
```

var	The name of the environment variable. When creating an environment variable, any name is allowed, however spaces are not allowed and the name may not start with “CONF”, “HOST”, an underscore “_” or the dollar sign “\$”.	REQUIRED
value	A quoted translated string which defines the value of the environment variable. The value of system variables (built-in variables with names starting with an underscore “_”, “CONF” or “HOST”) can not be changed.	REQUIRED

EXAMPLE:

```
=>env list
_COMPANY_NAME=THOMSON multimedia
_COMPANY_URL=http://www.speedtouch.com
_PROD_NAME=SpeedTouch
_PROD_FRIENDLY_NAME=SpeedTouch 610 Business DSL Router
.....
CONF_DESCRIPTION=Default Bridging configuration
CONF_SERVICE=Bridging on 0/35 and 8/35
CONF_DATE=May 2002
HOST_SETUP=user
=>env set ATM_addr value="8*35"
=>env get var=ATM_addr
8*35
=>env set var=ATM_addr value="11*35"
=>env get var=ATM_addr
11*35
=>
```

RELATED COMMANDS:

env get	Show the value of an environment variable.
env list	List all current environment variables.
env unset	Delete a non-system environment variable.

env unset

Delete a non-system environment variable.

SYNTAX:

```
env unset      var = <string>
```

var	The name of the environment variable to delete. System variables (built-in variables with names starting with an underscore “_”, “CONF” or “HOST”) can not be unset, nor changed or deleted.	REQUIRED
-----	---	----------

EXAMPLE:

```
=>env list
_COMPANY_NAME=THOMSON multimedia
_COMPANY_URL=http://www.speedtouch.com
_PROD_NAME=SpeedTouch
.....
CONF_DATE=May 2002
HOST_SETUP=user
ATM_addr=8*35
=>env unset var=ATM_addr
=>env list
_COMPANY_NAME=THOMSON multimedia
_COMPANY_URL=http://www.speedtouch.com
_PROD_NAME=SpeedTouch
.....
CONF_DATE=May 2002
HOST_SETUP=user
=>
```

RELATED COMMANDS:

- env list List all current environment variables.
- env set Create and set a non-system environment variable.

Eth Commands

Contents

This chapter covers the following topics:

Topic	Page
eth config	160
eth ifconfig	161
eth iflist	162

eth config

Enable/disable the Ethernet interface.

SYNTAX:

```
eth config          intf = <number {1-4}>
                   [state = <{enabled|disabled}>]
```

intf	The Ethernet interface to configure.	REQUIRED
state	Enable or disable the Ethernet interface. By default, all Ethernet interfaces are enabled.	OPTIONAL

EXAMPLE:

```
=>eth config
ETH Intf 1 port state = UP [forwarding]
ETH Intf 2 port state = UP [forwarding]
ETH Intf 3 port state = UP [forwarding]
ETH Intf 4 port state = UP [forwarding]
=>eth config intf=4 state=disabled
=>eth config
ETH Intf 1 port state = UP [forwarding]
ETH Intf 2 port state = UP [forwarding]
ETH Intf 3 port state = UP [forwarding]
ETH Intf 4 port state = DOWN [disabled]
=>
```

RELATED COMMANDS:

`eth iflist` Show Ethernet port configuration and current operating mode

eth ifconfig

Configure the Ethernet port.

SYNTAX:

```
eth ifconfig      intf = <number {1-4}>
                  type = <{auto|10BaseTHD|10BaseTFD|100BaseTHD|100BaseTFD}
                  or number>
```

intf	The Ethernet interface to configure.	REQUIRED
type	<p>The Ethernet type. Select either:</p> <ul style="list-style-type: none"> • auto Auto negotiation of Ethernet communication speed (10Mb/s or 100Mb/s) and Duplex mode (half duplex or full duplex). • 10BaseTHD 10Mb/s communication speed in half duplex mode. • 10BaseTFD 10Mb/s communication speed in full duplex mode. • 100BaseTHD 100Mb/s communication speed in half duplex mode. • 100BaseTFD 100Mb/s communication speed in full duplex mode. <p>or enter a number between 0 (auto) and 5 (100BaseTFD).</p> <p>Note By default, the Ethernet type is set to auto and should never be changed, except in case of communication problems.</p>	REQUIRED

RELATED COMMANDS:

[eth iflist](#) Show Ethernet port configuration and current operating mode

eth iflist

Show the Ethernet port configuration and current operating status.

SYNTAX:

```
eth iflist
```

EXAMPLE:

```
=>eth iflist
Intf  Type      Result Type
1     auto      100BaseTFD
2     auto      100BaseTFD
3     auto      100BaseTFD
4     auto      100BaseTFD
=>
```

DESCRIPTION:

Type	Indicates the configured Ethernet communication speed and duplex mode.
Result type	Indicates the effective operating status in case Type=auto. In other cases, when the Ethernet types do NOT match, Result type=unknown and no Ethernet connectivity will exist.

RELATED COMMANDS:

eth ifconfig Configure the Ethernet port.

ETHoA Commands

Contents

This chapter covers the following topics:

Topic	Page
ethoa flush	164
ethoa ifadd	165
ethoa ifattach	166
ethoa ifconfig	167
ethoa ifdelete	169
ethoa ifdetach	170
ethoa iflist	171

ethoa flush

Flushes ETHoA interfaces.

Note The flush command does not impact previously saved configurations.

SYNTAX:

```
ethoa flush
```

EXAMPLE:

```
=>ethoa iflist
Newethoa      : dest : Br3
                Retry : 10      QoS : default      Encaps : llc/snap      Fcs : off
                Connection State : connected
                RX bytes: 0      frames: 0
                TX bytes: 0      frames: 0      dropframes: 0
=>ethoa flush
=>ethoa iflist
=>
```

ethoa ifadd

Create a new ETHoA interface.

SYNTAX:

```
ethoa ifadd      [intf = <string>]
                 [dest = <phonebook entry>]
```

intf	The name for the new ETHoA interface. If not specified, the destination will double as interface name.	OPTIONAL
dest	The destination for the new ETHoA interface. Typically, a phonebook entry.	OPTIONAL

EXAMPLE:

```
=>ethoa iflist
Newethoa      : dest : Br3
                Retry : 10      QoS : default      Encaps : llc/snap      Fcs : off
                Connection State : connected
                RX bytes: 0      frames: 0
                TX bytes: 0      frames: 0          dropframes: 0
=>phonebook list
Name          Type      Use      Address
Br1           bridge    1        8.35
Br2           bridge    1        8.36
CIPPVC3      cip        1        8.82
CIPPVC4      cip        1        8.83
=>ethoa ifadd intf=Moreethoa dest=Br4
=>ethoa iflist
Newethoa      : dest : Br3
                Retry : 10      QoS : default      Encaps : llc/snap      Fcs : off
                Connection State : connected
                RX bytes: 0      frames: 0
                TX bytes: 0      frames: 0          dropframes: 0
Moreethoa     : dest : Br4
                Retry : 10      QoS : default      Encaps : llc/snap      Fcs : off
                Connection State : not-connected
=>
```

RELATED COMMANDS:

ethoa ifattach	Attach an ETHoA interface.
ethoa ifconfig	Configure an ETHoA interface.
ethoa ifdelete	Delete an ETHoA interface.
ethoa ifdetach	Detach an ETHoA interface.
ethoa iflist	Show current ETHoA interfaces.

ethoa ifattach

Attach (i.e. connect) an ETHoA interface.

SYNTAX:

```
ethoa ifattach   intf = <intfname>
```

intf	The name of the ETHoA interface to attach.	REQUIRED
------	--	----------

EXAMPLE:

```
=>ethoa iflist
Newethoa      : dest : Br3
                Retry : 10      QoS : default   Encaps : 11c/snap   Fcs : off
                Connection State : connected
                RX bytes: 0      frames: 0
                TX bytes: 0      frames: 0      dropframes: 0
Moreethoa     : dest : Br4
                Retry : 10      QoS : default   Encaps : 11c/snap   Fcs : off
                Connection State : not-connected
=>ethoa ifattach intf=Moreethoa
=>ethoa iflist
Newethoa      : dest : Br3
                Retry : 10      QoS : default   Encaps : 11c/snap   Fcs : off
                Connection State : connected
                RX bytes: 0      frames: 0
                TX bytes: 0      frames: 0      dropframes: 0
Moreethoa     : dest : Br4
                Retry : 10      QoS : default   Encaps : 11c/snap   Fcs : off
                Connection State : connected
                RX bytes: 0      frames: 0
                TX bytes: 0      frames: 0      dropframes: 0
=>
```

RELATED COMMANDS:

ethoa ifadd	Create a new ETHoA interface.
ethoa ifconfig	Configure an ETHoA interface.
ethoa ifdelete	Delete an ETHoA interface.
ethoa ifdetach	Detach an ETHoA interface.
ethoa iflist	Show current ETHoA interfaces.

ethoa ifconfig

Configure an ETHoA interface.

SYNTAX:

```
ethoa ifconfig  intf = <intfname>
                  [dest = <intfname>]
                  [qos = <string>]
                  [encaps = <{llc/snap|vcmux}>]
                  [retry = <number {0-65535}>]
```

intf	The name of the ETHoA interface to configure.	REQUIRED
dest	The destination for this interface. Typically a phonebook entry. This parameter needs only to be specified in case of an interface created without specified destination.	OPTIONAL
qos	The name of a qosbook entry defining the QoS parameters for the WAN link.	OPTIONAL
encaps	The type of encapsulation to be used for this ETHoA interface. Choose between: <ul style="list-style-type: none"> • llc/snap • vcmux 	OPTIONAL
retry	A number between 0 and 65535. Represents the number of WAN connection setup retries before giving up. By default the retry value is 10.	OPTIONAL

EXAMPLE:

```
=>ethoa iflist
Moreethoa   : dest : Br4
              Retry : 10    QoS : default    Encaps : vcmux    Fcs : off
              Connection State : connected
              RX bytes: 0    frames: 0
              TX bytes: 0    frames: 0        dropframes: 0
=>ethoa ifconfig intf=Moreethoa encaps=llc/snap retry=15
=>ethoa iflist
Moreethoa   : dest : Br4
              Retry : 15    QoS : default    Encaps : llc/snap Fcs : off
              Connection State : connected
              RX bytes: 0    frames: 0
              TX bytes: 0    frames: 0        dropframes: 0
=>
```

RELATED COMMANDS:

<code>ethoa ifadd</code>	Create a new ETHoA interface.
<code>ethoa ifattach</code>	Attach an ETHoA interface.
<code>ethoa ifdelete</code>	Delete an ETHoA interface.
<code>ethoa ifdetach</code>	Detach an ETHoA interface.
<code>ethoa iflist</code>	Show current ETHoA interfaces.

ethoa ifdelete

Delete an ETHoA interface.

SYNTAX:

```
ethoa ifdelete   intf = <intfname>
```

intf	The name of the ETHoA interface.	REQUIRED
------	----------------------------------	----------

EXAMPLE:

```
=>ethoa iflist
Newethoa      : dest : Br3
                Retry : 10      QoS : default   Encaps : 11c/snap   Fcs : off
                Connection State : connected
                RX bytes: 0   frames: 0
                TX bytes: 0   frames: 0           dropframes: 0
Moreethoa     : dest : Br4
                Retry : 10      QoS : default   Encaps : 11c/snap   Fcs : off
                Connection State : not-connected
=>ethoa ifdelete intf=Moreethoa
=>ethoa iflist
Newethoa      : dest : Br3
                Retry : 10      QoS : default   Encaps : 11c/snap   Fcs : off
                Connection State : connected
                RX bytes: 0   frames: 0
                TX bytes: 0   frames: 0           dropframes: 0
=>
```

RELATED COMMANDS:

ethoa ifadd	Create a new ETHoA interface.
ethoa ifattach	Attach an ETHoA interface.
ethoa ifconfig	Configure an ETHoA interface.
ethoa ifdetach	Detach an ETHoA interface.
ethoa iflist	Show current ETHoA interfaces.

ethoa ifdetach

Detach an ETHoA interface.

SYNTAX:

```
ethoa ifdetach intf = <intfname>
```

intf	The name of the ETHoA interface.	REQUIRED
------	----------------------------------	----------

EXAMPLE:

```
=>ethoa iflist
Newethoa      : dest : Br3
                Retry : 10      QoS : default  Encaps : llc/snap  Fcs : off
                Connection State : connected
                RX bytes: 0      frames: 0
                TX bytes: 0      frames: 0      dropframes: 0
Moreethoa     : dest : Br4
                Retry : 10      QoS : default  Encaps : llc/snap  Fcs : off
                Connection State : connected
                RX bytes: 0      frames: 0
                TX bytes: 0      frames: 0      dropframes: 0
=>ethoa ifdetach intf=Moreethoa
=>ethoa iflist
Newethoa      : dest : Br3
                Retry : 10      QoS : default  Encaps : llc/snap  Fcs : off
                Connection State : connected
                RX bytes: 0      frames: 0
                TX bytes: 0      frames: 0      dropframes: 0
Moreethoa     : dest : Br4
                Retry : 10      QoS : default  Encaps : llc/snap  Fcs : off
                Connection State : not-connected
=>
```

RELATED COMMANDS:

- [ethoa ifadd](#) Create a new ETHoA interface.
- [ethoa ifattach](#) Attach an ETHoA interface.
- [ethoa ifconfig](#) Configure an ETHoA interface.
- [ethoa ifdelete](#) Delete an ETHoA interface.
- [ethoa iflist](#) Show current ETHoA interfaces.

ethoa iflist

Show all or a specified ETHoA interface(s).

SYNTAX:

```
ethoa iflist [intf = <intfname>]
```

intf	The name of the ETHoA interface.	OPTIONAL
	Note If not specified, all ETHoA interfaces are listed.	

EXAMPLE:

```
=>ethoa iflist
Newethoa      : dest : Br3
               Retry : 10   QoS : default   Encaps : 11c/snap   Fcs : off
               Connection State : connected
               RX bytes: 0   frames: 0
               TX bytes: 0   frames: 0       dropframes: 0
Moreethoa     : dest : Br4
               Retry : 10   QoS : default   Encaps : 11c/snap   Fcs : off
               Connection State : connected
               RX bytes: 0   frames: 0
               TX bytes: 0   frames: 0       dropframes: 0
=>
```

RELATED COMMANDS:

ethoa ifadd	Create a new ETHoA interface.
ethoa ifattach	Attach an ETHoA interface.
ethoa ifconfig	Configure an ETHoA interface.
ethoa ifdelete	Delete an ETHoA interface.
ethoa ifdetach	Detach an ETHoA interface.

Firewall Commands

Contents

This chapter covers the following topics:

Topic	Page
firewall assign	174
firewall flush	176
firewall list	177
firewall troff	178
firewall tron	179
firewall unassign	180
firewall chain create	181
firewall chain delete	182
firewall chain flush	183
firewall chain list	184
firewall rule clear	185
firewall rule create	186
firewall rule delete	190
firewall rule flush	191
firewall rule list	192
firewall rule stats	193

firewall assign

Assign a chain to an entry point. An entry point, also referred to as hook or a Packet Interception Point (PIP), is the location where packets are intercepted to be compared against a chain of rules

SYNTAX:

```
firewall assign    hook = <{input|sink|forward|source|output}>
                  chain = <string>
```

hook	The name of the entry point to which a chain must be assigned. Choose between: <ul style="list-style-type: none"> • input: The point off all incoming traffic. At this point it can be determined whether the packet is allowed to reach the SpeedTouch™ IP router or local host. • sink: The point off all traffic destined to the SpeedTouch™ IP router itself. At this point it can be determined whether the packet is allowed to address the local host. • forward: The point off all traffic to be forwarded by the Speed-Touch™ IP router. At this point it can be determined whether the packet is allowed to be handled, i.e. routed. • source: The point off all traffic sourced by the SpeedTouch™ IP router. At this point it can be determined whether the packet is allowed to leave the local host. • output: The point off all outgoing traffic. At this point it can be determined whether the packet is allowed to leave the SpeedTouch™ IP router or local host. 	REQUIRED
chain	The name of the chain to use.	REQUIRED

EXAMPLE:

```
=>firewall list
assign    hook=sink      chain=sink
assign    hook=forward   chain=forward
assign    hook=source    chain=source
=>firewall chain create chain Telnet
=>firewall assign hook=sink chain=Telnet
=>firewall list
assign    hook=sink      chain=Telnet
assign    hook=forward   chain=forward
assign    hook=source    chain=source
=>
```

RELATED COMMANDS:

firewall flush	Clear all hooks.
firewall list	Show a list of all chain assignments.
firewall unassign	Clear a specific hook.

firewall flush

Clears all hooks, chains and rules.

Note The flush command does not impact previously saved configurations.

SYNTAX:

```
firewall flush
```

EXAMPLE:

```
=>firewall load
=>firewall list
assign    hook=sink      chain=sink
assign    hook=forward   chain=forward
assign    hook=source    chain=source
=>firewall flush hook=sink
=>firewall list
assign    hook=forward   chain=forward
assign    hook=source    chain=source
=>firewall flush
=>firewall list
=>
```

RELATED COMMANDS:

`firewall assign` Assign a chain to an entry point.

firewall list

Show the association(s) between all hooks or a specified hook and their chain(s).

SYNTAX:

```
firewall list [hook = <{input|sink|forward|source|output}>]
```

hook	The name of the hook for which the associations must be shown. Choose between:	OPTIONAL
	<ul style="list-style-type: none"> • input • sink • forward • source • output. 	
	Note If this parameter is not specified, the associations for all hooks are shown.	

EXAMPLE:

```
=>firewall list
firewall assign hook=input chain=None
firewall assign hook=sink chain=sink
firewall assign hook=forward chain=forward
firewall assign hook=source chain=source
firewall assign hook=output chain=None
=>firewall list hook=input
firewall assign hook=input chain=None
=>firewall list hook=forward
firewall assign hook=forward chain=forward
=>
```

RELATED COMMANDS:

firewall assign	Assign a chain to an entry point.
firewall flush	Clear associations for all or a selected entry point(s).

firewall troff

Disable verbose console messaging.

SYNTAX:

```
firewall troff
```

EXAMPLE:

```
=>firewall troff
```

RELATED COMMANDS:

firewall tron

Enable verbose console messaging.

firewall tron

Enable verbose console messaging.

SYNTAX:

```
firewall tron
```

EXAMPLE:

```
=>firewall tron
```

RELATED COMMANDS:

firewall troff

Disable verbose console messaging.

firewall unassign

Clear a specific hook.

SYNTAX:

```
firewall unassign hook = <{input|sink|forward|source|output}>
```

hook	The name of the hook to be cleared. Choose between: <ul style="list-style-type: none">• input• sink• forward• source• output.	REQUIRED
------	--	----------

RELATED COMMANDS:

- [firewall assign](#) Assign a chain to a hook.
- [firewall flush](#) Clear all hooks.
- [firewall list](#) Show a list of all chain assignments.

firewall chain create

Create a new chain.

SYNTAX:

```
firewall chain create chain = <string>
```

chain

The name of the chain to create.

REQUIRED

EXAMPLE:

```
=>firewall chain list
Tempo, source, forward, sink
=>firewall chain create chain=Telnet
=>firewall chain list
Telnet, Tempo, source, forward, sink
=>
```

RELATED COMMANDS:

[firewall chain delete](#)

Delete a chain.

[firewall chain list](#)

Show a list of all current chains.

firewall chain delete

Delete a chain.

SYNTAX:

```
firewall chain delete chain = <string>
```

chain	The name of the chain to be deleted.	REQUIRED
-------	--------------------------------------	----------

EXAMPLE:

```
=>firewall chain list
Telnet, Tempo, source, forward, sink
=>firewall chain list
Telnet, Tempo, source, forward, sink
=>firewall chain delete chain=Tempo
=>firewall chain list
Telnet, source, forward, sink
=>
```

RELATED COMMANDS:

firewall chain create	Create a chain.
firewall chain list	Show a list of all chains.

firewall chain flush

Flush all chains.

SYNTAX:

```
firewall chain flush
```

EXAMPLE:

```
=>firewall rule list chain=Telnet
:firewall rule create chain=Telnet index=0 srcintfgrp=lan src=10.0.0.0/8 |
dst=200.200.200.1/32 prot=tcp srcport=1024 srcportend=65535 dstport=telnet |
action=accept
:firewall rule create chain=Telnet index=1 srcintfgrp=wan |
src=200.200.200.1/32 dst=10.0.0.0/8 prot=tcp srcport=telnet dstport=1024 |
dstportend=65535 action=accept
:firewall rule create chain=Telnet index=2 action=drop
=>firewall rule flush chain=Telnet
=>firewall rule list chain=Telnet
=>
```

firewall chain list

Show a list of all current chains.

SYNTAX:

```
firewall chain list
```

EXAMPLE INPUT/OUTPUT:

```
=>firewall chain list
source, forward, sink
=>firewall chain create chain Telnet
=>firewall chain list
Telnet, source, forward, sink
=>firewall chain list
Telnet, source, forward, sink
=>
```

RELATED COMMANDS:

firewall assign	Assign a chain to an entry point.
firewall chain create	Create a chain.
firewall chain delete	Delete a chain.

firewall rule clear

Clear statistics for a given rule.

SYNTAX:

```
firewall rule clear [chain = <string>]
                   [index = <number>]
```

chain	The name of the chain in which the rule is to be found.	OPTIONAL
index	The index number (determined by the position) of the rule in the chain.	OPTIONAL

EXAMPLE:

```
=>firewall rule stats
Chain Telnet,  index 0,  packets 0,  bytes 0
Chain Telnet,  index 1,  packets 0,  bytes 0
Chain Telnet,  index 2,  packets 0,  bytes 0
Chain source,  index 0,  packets 203, bytes 15229
Chain source,  index 1,  packets 0,  bytes 0
Chain source,  index 2,  packets 0,  bytes 0
Chain forward, index 0,  packets 0,  bytes 0
Chain sink,   index 0,  packets 202, bytes 10159
Chain sink,   index 1,  packets 0,  bytes 0
Chain sink,   index 2,  packets 0,  bytes 0
=>firewall rule clear chain=source index=0
=>firewall rule stats
Chain Telnet,  index 0,  packets 0,  bytes 0
Chain Telnet,  index 1,  packets 0,  bytes 0
Chain Telnet,  index 2,  packets 0,  bytes 0
Chain source,  index 0,  packets 11,  bytes 559
Chain source,  index 1,  packets 0,  bytes 0
Chain source,  index 2,  packets 0,  bytes 0
Chain forward, index 0,  packets 0,  bytes 0
Chain sink,   index 0,  packets 409, bytes 21535
Chain sink,   index 1,  packets 0,  bytes 0
Chain sink,   index 2,  packets 0,  bytes 0
=>
```

RELATED COMMANDS:

firewall rule create	Create a rule.
firewall rule delete	Delete a specified rule in a chain.
firewall rule flush	Delete all rules in a chain.
firewall rule list	Show a list of all (or a specified) chains' rules.
firewall rule stats	Show statistics for all (or a specified) chains' rules.

firewall rule create

Create a rule.

SYNTAX:

```
firewall rule create chain = <string>
                    [index = <number>]
                    [srcintf [!]= <string>]
                    [srcintfgrp [!]= <{wan|local|lan}>]
                    [srcbridgeport [!]= <number>]
                    [src [!]= <ip-address>]
                    [dstintf [!]= <string>]
                    [dstintfgrp [!]= <{wan|local|lan}>]
                    [dst [!]= <ip-address>]
                    [tos [!]= <number{1-255}>]
                    [precedence [!]= <number{0-7}>]
                    [dscp [!]= <number{0-63}>]
                    [prot [!]= <{<supported IP protocol name>|<number>}>]
                    [syn = <yes|no>]
                    [urg = <yes|no>]
                    [ack = <yes|no>]
                    [srcport [!]= <{<supported TCP/UDP port name>|<number>}>]
                    [srcportend = <{<supported TCP/UDP port name>|<number>}>]
                    [dstport [!]= <{<supported TCP/UDP port name>|<number>}>]
                    [dstportend = <{<supported TCP/UDP port name>|<number>}>]
                    [icmptype [!]= <{<supported ICMP type name>|<number>}>]
                    [icmpcode [!]= <number{0-15}>]
                    [icmpcodeend = <number{0-15}>]
                    [clink = <string>]
                    [log = <{no|yes}>]
                    action = <{accept|deny|drop|count}>
```

chain	The name of the chain to insert the rule in.	REQUIRED
index	The number of the rule before which the new rule must be added.	OPTIONAL
srcintf	The name of the interface the packet should [or should NOT] arrive on to make this rule apply. (NOT applicable if used in a chain assigned to the output hook)	OPTIONAL
srcintfgrp	The interface group the packet should [or should NOT] arrive on. Choose between: <ul style="list-style-type: none"> • wan • local • lan (NOT applicable if used in a chain assigned to the output hook)	OPTIONAL

srcbridgeport	A number between 0 and 6. Represents the bridge port the virtual packet should [or should NOT] arrive on. Use <code>:bridge iflist</code> for a list of available bridge ports.	OPTIONAL
src	The source IP address (range) the packet should [or should NOT] come from. (Supports cidr notation).	OPTIONAL
dstintf	The name of the interface the packet should [or should NOT] be going to. (NOT applicable if used in a chain assigned to the input hook)	OPTIONAL
dstintfgrp	The interface group the packet should [or should NOT] be going to. Choose between: <ul style="list-style-type: none"> • wan • local • lan (NOT applicable if used in a chain assigned to the input hook)	OPTIONAL
dst	The destination IP address (range) the packet should [or should NOT] be going to. (supports cidr notation).	OPTIONAL
precedence	A number between 0 and 7. Represents the precedence in the IP packet (part of tos).	OPTIONAL
dscp	A number between 0 and 63 Represents the DSCP in the IP packet (part of tos).	OPTIONAL
tos	A number between 0 and 255. Represents the Type Of Service specification which should be expected [or NOT expected] in the IP packet. The Type of Service numbering specification is in accordance to the latest version of RFC1700: Assigned numbers.	OPTIONAL
prot	The protocol (name or number) expected [or NOT expected] in the IP packet. Select one of the supported protocol names (See “ Supported Internet Protocol (IP) Protocol Names ” on page 413 for a listing of protocol names supported by the SpeedTouch™). Alternatively, specify the protocol number.	OPTIONAL
syn	Expect TCP SYN flag set (yes) or not (no). In combination with TCP ACK this allows selection of incoming versus outgoing TCP connections.	OPTIONAL
urg	Expect TCP URG flag set (yes) or not (no).	OPTIONAL
ack	Expect TCP ACK flag set (yes) or not (no).	OPTIONAL
srcport	The TCP/UDP port (or beginning of range) the packet should [or should NOT] be from. Select one of the supported TCP/UDP port names (See “ Supported TCP/UDP Port Names ” on page 414 for a listing of TCP/UDP port names supported by the SpeedTouch™). Alternatively, specify the protocol number.	OPTIONAL

srcportend	<p>The source TCP/UDP port range end (inclusive). (Only applicable for ranges) Select one of the supported TCP/UDP port names (See “ Supported TCP/UDP Port Names” on page 414 for a listing of TCP/UDP port names supported by the SpeedTouch™). Alternatively, specify the protocol number.</p>	OPTIONAL
dstport	<p>The TCP/UDP port (or beginning of range) the packet should [or should NOT] be going to. Select one of the supported TCP/UDP port names (See “ Supported TCP/UDP Port Names” on page 414 for a listing of TCP/UDP port names supported by the SpeedTouch™). Alternatively, specify the protocol number.</p>	OPTIONAL
dstportend	<p>The destination TCP/UDP port range end (inclusive). (Only applicable for ranges) Select one of the supported TCP/UDP port names (See “ Supported TCP/UDP Port Names” on page 414 for a listing of TCP/UDP port names supported by the SpeedTouch™). Alternatively, specify the protocol number.</p>	OPTIONAL
icmpcode	<p>The expected [or NOT expected] ICMP type (name or number) of the packet. Select one of the supported ICMP type names (See “ Supported ICMP Type Names” on page 417 for a listing of ICMP type names supported by the SpeedTouch™). Alternatively, specify the protocol number.</p>	OPTIONAL
icmpcodeend	<p>A number between 0 and 15. Represents the expected [or NOT expected] ICMP code (or beginning of range) of the packet as specified in the latest version of RFC1700: Assigned number.</p>	OPTIONAL
icmpcodeend	<p>A number between 0 and 15. Represents the ICMP code range end. Only applicable for ranges.</p>	OPTIONAL
clink	<p>The name of the chain to be parsed when this rule applies. (action is ignored).</p>	OPTIONAL
log	<p>Logging is done when this rule applies.</p>	OPTIONAL
action	<p>Action to be taken when this rule applies. Choose between:</p> <ul style="list-style-type: none"> • accept: the packet may pass. • deny: ICMP error destination unreachable. An error message is sent back to the sender. • drop: packet disappears. It is silently dropped, that is, without sending an error message to the sender. • count: update of statistics. Has no influence on the packet. 	REQUIRED

RELATED COMMANDS:

firewall rule clear	Clear statistics for a given rule.
firewall rule delete	Delete a specified rule in a chain.
firewall rule flush	Delete all rules in a chain.
firewall rule list	Show a list of all (or a specified) chains' rules.
firewall rule stats	Show statistics for all (or a specified) chains' rules.

firewall rule delete

Delete a rule.

SYNTAX:

```
firewall rule delete chain = <string>
                    index = <number>
```

chain	The name of the chain in which to delete the rule.	REQUIRED
index	The index number of the rule in the chain.	REQUIRED
Note	Use :firewall rule list first to determine the index number of the applicable rule.	

EXAMPLE:

```
=>firewall rule list chain=Telnet
:firewall rule create chain=Telnet index=0 srcintfgrp=lan src=10.0.0.0/8
dst=200.200.200.1/32 prot=tcp srcport=1024 srcportend=65535 dstport=telnet
action=accept
:firewall rule create chain=Telnet index=1 srcintfgrp=wan
src=200.200.200.1/32 dst=10.0.0.0/8 prot=tcp srcport=telnet dstport=1024
dstportend=65535 action=accept
:firewall rule create chain=Telnet index=2 action=drop
=>firewall rule delete chain=Telnet index=1
=>firewall rule list chain=Telnet
:firewall rule create chain=Telnet index=0 srcintfgrp=lan src=10.0.0.0/8
dst=200.200.200.1/32 prot=tcp srcport=1024 srcportend=65535 dstport=telnet
action=accept
:firewall rule create chain=Telnet index=1 action=drop
=>
```

RELATED COMMANDS:

firewall rule clear	Clear statistics for a given rule.
firewall rule create	Create a rule.
firewall rule flush	Delete all rules in a chain.
firewall rule list	Show a list of all (or a specified) chains' rules.
firewall rule stats	Show statistics for all (or a specified) chains' rules.

firewall rule flush

Flush all rules created for a chain(s). The chain itself is not removed.

Note The flush command does not impact previously saved configurations.

SYNTAX:

```
firewall rule flush [chain = <string>]
```

chain The name of the chain to empty. OPTIONAL

Note If this parameter is not specified, all rules for all chains are deleted.

EXAMPLE:

```
=>firewall rule list chain=Telnet
:firewall rule create chain=Telnet index=0 srcintfgrp=lan src=10.0.0.0/8 |
dst=200.200.200.1/32 prot=tcp srcport=1024 srcportend=65535 dstport=telnet |
action=accept
:firewall rule create chain=Telnet index=1 srcintfgrp=wan |
src=200.200.200.1/32 dst=10.0.0.0/8 prot=tcp srcport=telnet dstport=1024 |
dstportend=65535 action=accept
:firewall rule create chain=Telnet index=2 action=drop
=>firewall rule flush chain=Telnet
=>firewall rule list chain=Telnet
=>
```

RELATED COMMANDS:

firewall rule clear	Clear statistics for a given rule.
firewall rule create	Create a rule.
firewall rule delete	Delete a specified rule in a chain.
firewall rule list	Show a list of all (or a specified) chains' rules.
firewall rule stats	Show statistics for all (or a specified) chains' rules.

firewall rule list

Show a list of rules.

SYNTAX:

```
firewall rule list [chain = <string>]
```

chain	The name of the chain to list the rules of.	OPTIONAL
	Note If this parameter is not specified, all rules for all chains are shown.	

EXAMPLE:

```
=>firewall rule list chain=Telnet
:firewall rule create chain=Telnet index=0 srcintfgrp=lan src=10.0.0.0/8 |
dst=200.200.200.1/32 prot=tcp srcport=1024 srcportend=65535 dstport=telnet |
action=accept
:firewall rule create chain=Telnet index=1 srcintfgrp=wan |
src=200.200.200.1/32 dst=10.0.0.0/8 prot=tcp srcport=telnet dstport=1024 |
dstportend=65535 action=accept
:firewall rule create chain=Telnet index=2 action=drop
=>firewall rule list
:firewall rule create chain=source index=0 dstintfgrp=!wan action=accept
:firewall rule create chain=source index=1 prot=udp dstport=dns |
action=accept
:firewall rule create chain=source index=2 prot=udp dstport=67 action=accept
:firewall rule create chain=source index=3 action=drop
:firewall rule create chain=forward index=0 srcintfgrp=wan dstintfgrp=wan |
action=drop
:firewall rule create chain=sink index=0 srcintf=eth0 srcbridgeport=1 |
action=accept
:firewall rule create chain=sink index=1 srcintfgrp=!wan action=accept
:firewall rule create chain=sink index=2 prot=udp dstport=dns action=accept
:firewall rule create chain=sink index=3 prot=udp dstport=68 action=accept
:firewall rule create chain=sink index=4 action=drop
:firewall rule create chain=Telnet index=0 srcintfgrp=lan src=10.0.0.0/8 |
dst=200.200.200.1/32 prot=tcp srcport=1024 srcportend=65535 dstport=telnet |
action=accept
:firewall rule create chain=Telnet index=1 srcintfgrp=wan |
src=200.200.200.1/32 dst=10.0.0.0/8 prot=tcp srcport=telnet dstport=1024 |
dstportend=65535 action=accept
:firewall rule create chain=Telnet index=2 action=drop
=>
```

RELATED COMMANDS:

firewall rule clear	Clear statistics for a given rule.
firewall rule create	Create a rule.
firewall rule delete	Delete a specified rule in a chain.
firewall rule flush	Delete all rules in a chain.
firewall rule stats	Show statistics for all (or a specified) chains' rules.

firewall rule stats

Show statistics, i.e. the number of packets and bytes which have passed the hooks.

SYNTAX:

```
firewall rule stats [chain = <string>]
                   [index = <number>]
```

chain	The name of the chain of which the statistics must be listed. In case this parameter is not specified the statistics for the rules applicable to all chains are shown.	OPTIONAL
index	The index number of the chain's rule of which the statistics must be listed. Execute firewall rule list first to determine the index number of the applicable rule. In case this parameter is not specified the statistics for all rules applicable to the specified chain are shown.	OPTIONAL

EXAMPLE:

```
=>firewall rule list chain=Test
:firewall rule create chain=Test index=0 srcintfgrp=lan src=200.200.0.1/32
dst=200.200.0.2/32 prot=udp srcport=0 srcportend=65535 dstport=telnet
action=deny
=>firewall rule clear
=>firewall rule stats
Chain sink, index 0, packets 43, bytes 1743
Chain sink, index 1, packets 0, bytes 0
Chain sink, index 2, packets 0, bytes 0
Chain sink, index 3, packets 0, bytes 0
Chain forward, index 0, packets 0, bytes 0
Chain source, index 0, packets 43, bytes 1977
Chain source, index 1, packets 0, bytes 0
Chain source, index 2, packets 0, bytes 0
Chain Test, index 0, packets 0, bytes 0
=>firewall rule stats
Chain sink, index 0, packets 104, bytes 6143
Chain sink, index 1, packets 0, bytes 0
Chain sink, index 2, packets 0, bytes 0
Chain sink, index 3, packets 0, bytes 0
Chain forward, index 0, packets 0, bytes 0
Chain source, index 0, packets 43, bytes 1977
Chain source, index 1, packets 0, bytes 0
Chain source, index 2, packets 0, bytes 0
Chain Test, index 0, packets 44, bytes 21032
=>
```

DESCRIPTION:

The statistics for the 'Test' chain are the result of sending udp packets to the SpeedTouch™. The chain 'Test' is assigned to the hook 'input' and prohibits the sending of udp packets from one host to another.

EXAMPLE INPUT/OUTPUT IN A NETWORKED ENVIRONMENT:

The SpeedTouch™ is configured as DHCP client on its Ethernet interface eth0.

```
=>firewall rule list chain=Sending
:firewall rule create chain=Sending index=0 srcintfgrp=lan src=10.0.0.3/32
dst=10.10.1.1/32 prot=icmp action=count
:firewall rule create chain=Sending index=1 srcintfgrp=lan src=10.10.1.1/32
dst=10.0.0.3/32 prot=icmp action=count
=>firewall rule stats
Chain source, index 0, packets 0, bytes 0
Chain source, index 1, packets 0, bytes 0
Chain source, index 2, packets 0, bytes 0
Chain source, index 3, packets 0, bytes 0
Chain forward, index 0, packets 0, bytes 0
Chain sink, index 0, packets 0, bytes 0
Chain sink, index 1, packets 144, bytes 5844
Chain sink, index 2, packets 0, bytes 0
Chain sink, index 3, packets 0, bytes 0
Chain sink, index 4, packets 0, bytes 0
Chain sink, index 5, packets 0, bytes 0
Chain sending, index 0, packets 0, bytes 0
Chain sending, index 1, packets 0, bytes 0
=>firewall rule clear
=>(Ping from server 10.10.1.1 to client 10.0.0.3)
=>firewall rule stats
Chain source, index 0, packets 0, bytes 0
Chain source, index 1, packets 0, bytes 0
Chain source, index 2, packets 0, bytes 0
Chain source, index 3, packets 0, bytes 0
Chain forward, index 0, packets 0, bytes 0
Chain sink, index 0, packets 0, bytes 0
Chain sink, index 1, packets 42, bytes 1782
Chain sink, index 2, packets 0, bytes 0
Chain sink, index 3, packets 0, bytes 0
Chain sink, index 4, packets 0, bytes 0
Chain sink, index 5, packets 0, bytes 0
Chain sending, index 0, packets 4, bytes 240
Chain sending, index 1, packets 4, bytes 240
=>
```

RELATED COMMANDS:

<code>firewall rule clear</code>	Clear statistics for a given rule.
<code>firewall rule create</code>	Create a rule.
<code>firewall rule delete</code>	Delete a specified rule in a chain.
<code>firewall rule flush</code>	Delete all rules in a chain.
<code>firewall rule list</code>	Show a list of all (or a specified) chains' rules.

IP Commands

Contents

This chapter covers the following commands:

Topic	Page
ip apadd	196
ip apdelete	198
ip aplist	199
ip arpadd	200
ip arpdelete	201
ip arplist	202
ip config	203
ip flush	206
ip ifconfig	207
ip iflist	208
ip ifwait	209
ip ping	210
ip rtadd	211
ip rtdelete	212
ip rtlist	213
ip sendto	214
ip traceroute	215
ip auto flush	217
ip auto ifadd	218
ip auto ifattach	219
ip auto ifconfig	220
ip auto ifdelete	221
ip auto ifdetach	222
ip auto iflist	223

ip apadd

Assign an Internet Protocol (IP) address to an interface.

SYNTAX:

```
ip apadd          addr = <ip-address>
                  [netmask = <ip-mask (dotted or cidr)>]
                  intf = <interface name>
                  [pointopoint = <ip-address>]
                  [addrtrans = <{none|pat}>]
                  [addroute = <{no|yes}>]
```

addr	The new IP address to add.	REQUIRED
netmask	The subnetmask associated with this address.	OPTIONAL
intf	The IP interface name.	REQUIRED
pointopoint	The remote IP address in case of a dedicated point-to-point link.	OPTIONAL
addrtrans	Indicates whether network address translation mode is allowed (pat) for this IP address or not (none).	OPTIONAL
addroute	Add typical net/subnet routes automatically according to the default (or specified) subnet mask (yes) or not (no).	OPTIONAL

EXAMPLE:

```

=>ip aplist
1 eth0 Type:Ethernet HWaddr 00:80:9f:24:ab:cf BRHWaddr ff:ff:ff:ff:ff:ff
  inet addr:10.10.10.147 Bcast:10.10.10.255 Mask:255.0.0.0
  UP RUNNING pat MTU:1500 ReasmMAX:65535 Group:2
  IPRX bytes:19791886 unicastpkts:11341 brcastpkts:290555
  IPTX bytes:839550 unicastpkts:11477 brcastpkts:0 droppkts:0
  HWRX bytes:0 unicastpkts:0 brcastpkts:0
  HWTX bytes:0 unicastpkts:0 brcastpkts:0 droppkts:0
0 loop Type:0
  inet addr:127.0.0.1 Bcast:127.255.255.255 Mask:255.0.0.0
  UP RUNNING MTU:1500 ReasmMAX:65535 Group:1
  IPRX bytes:116 unicastpkts:0 brcastpkts:2
  IPTX bytes:0 unicastpkts:0 brcastpkts:0 droppkts:0
  HWRX bytes:0 unicastpkts:0 brcastpkts:0
  HWTX bytes:0 unicastpkts:0 brcastpkts:0 droppkts:0
=>ip apadd addr=10.0.0.2 netmask=255.255.255.0 intf=eth0 addrtrans=pat
  addroute=yes
=>ip aplist
2 eth0 Type:Ethernet HWaddr 00:80:9f:24:ab:cf BRHWaddr ff:ff:ff:ff:ff:ff
  inet addr: 10.0.0.2 Bcast:10.0.0.255 Mask:255.255.255.0
  UP RUNNING pat MTU:1500 ReasmMAX:65535 Group:2
  IPRX bytes:0 unicastpkts:0 brcastpkts:0
  IPTX bytes:0 unicastpkts:0 brcastpkts:0 droppkts:0
  HWRX bytes:0 unicastpkts:0 brcastpkts:0
  HWTX bytes:0 unicastpkts:0 brcastpkts:0 droppkts:0
1 eth0 Type:Ethernet HWaddr 00:80:9f:24:ab:cf BRHWaddr ff:ff:ff:ff:ff:ff
  inet addr: 10.10.10.147 Bcast: 10.10.10.255 Mask: 255.0.0.0
  UP RUNNING pat MTU:1500 ReasmMAX:65535 Group:2
  IPRX bytes:19810763 unicastpkts:11515 brcastpkts:290669
  IPTX bytes:853114 unicastpkts:11662 brcastpkts:0 droppkts:0
  HWRX bytes:0 unicastpkts:0 brcastpkts:0
  HWTX bytes:0 unicastpkts:0 brcastpkts:0 droppkts:0
0 loop Type:0
  inet addr:127.0.0.1 Bcast:127.255.255.255 Mask:255.0.0.0
  UP RUNNING MTU:1500 ReasmMAX:65535 Group:1
  IPRX bytes:116 unicastpkts:0 brcastpkts:2
  IPTX bytes:0 unicastpkts:0 brcastpkts:0 droppkts:0
  HWRX bytes:0 unicastpkts:0 brcastpkts:0
  HWTX bytes:0 unicastpkts:0 brcastpkts:0 droppkts:0
=>

```

RELATED COMMANDS:

- [ip adelete](#) Remove an IP address from an interface.
- [ip aplist](#) Show current IP addresses.

ip apdelete

Remove an IP address from an interface.

SYNTAX:

```
ip apdelete      addr = <ip-address>
```

addr	The IP address to delete.	REQUIRED
------	---------------------------	----------

EXAMPLE:

```
=>ip aplist
2 eth0 Type:Ethernet HWaddr 00:80:9f:24:ab:cf BRHWaddr ff:ff:ff:ff:ff:ff
  inet addr:10.0.0.2      Bcast:10.0.0.255      Mask:255.255.255.0
  UP RUNNING pat      MTU:1500 ReasmMAX:65535 Group:2
  IPRX bytes:0          unicastpkts:0      brcastpkts:0
  IPTX bytes:0          unicastpkts:0      brcastpkts:0 dropkts:0
  HWRX bytes:0          unicastpkts:0      brcastpkts:0
  HWTX bytes:0          unicastpkts:0      brcastpkts:0 dropkts:0
1 eth0 Type:Ethernet HWaddr 00:80:9f:24:ab:cf BRHWaddr ff:ff:ff:ff:ff:ff
  inet addr:10.10.10.147 Bcast: 10.10.10.255 Mask: 255.0.0.0
  UP RUNNING pat      MTU:1500 ReasmMAX:65535 Group:2
  IPRX bytes:19791886  unicastpkts:11341  brcastpkts:290555
  IPTX bytes:839550    unicastpkts:11477  brcastpkts:0 dropkts:0
  HWRX bytes:0         unicastpkts:0      brcastpkts:0
  HWTX bytes:0         unicastpkts:0      brcastpkts:0 dropkts:0
0 loop Type:0
  inet addr:127.0.0.1   Bcast:127.255.255.255 Mask:255.0.0.0
  UP RUNNING          MTU:1500 ReasmMAX:65535 Group:1
  IPRX bytes:116       unicastpkts:0      brcastpkts:2
  IPTX bytes:0         unicastpkts:0      brcastpkts:0 dropkts:0
  HWRX bytes:0         unicastpkts:0      brcastpkts:0
  HWTX bytes:0         unicastpkts:0      brcastpkts:0 dropkts:0
=>ip apdelete addr=10.0.0.2
=>ip aplist
1 eth0 Type:Ethernet HWaddr 00:80:9f:24:ab:cf BRHWaddr ff:ff:ff:ff:ff:ff
  inet addr:10.10.10.147 Bcast: 10.10.10.255 Mask: 255.0.0.0
  UP RUNNING pat      MTU:1500 ReasmMAX:65535 Group:2
  IPRX bytes:19791886  unicastpkts:11341  brcastpkts:290555
  IPTX bytes:839550    unicastpkts:11477  brcastpkts:0 dropkts:0
  HWRX bytes:0         unicastpkts:0      brcastpkts:0
  HWTX bytes:0         unicastpkts:0      brcastpkts:0 dropkts:0
0 loop Type:0
  inet addr:127.0.0.1   Bcast:127.255.255.255 Mask:255.0.0.0
  UP RUNNING          MTU:1500 ReasmMAX:65535 Group:1
  IPRX bytes:116       unicastpkts:0      brcastpkts:2
  IPTX bytes:0         unicastpkts:0      brcastpkts:0 dropkts:0
  HWRX bytes:0         unicastpkts:0      brcastpkts:0
  HWTX bytes:0         unicastpkts:0      brcastpkts:0 dropkts:0
=>
```

RELATED COMMANDS:

ip apadd	Add an IP address to an interface.
ip aplist	Show current IP addresses.

ip aplist

Show a list of all configured IP addresses.

SYNTAX:

```
ip aplist
```

EXAMPLE:

```
=>ip aplist
2 eth0 Type:Ethernet Hwaddr 00:80:9f:24:ab:cf BRHWaddr ff:ff:ff:ff:ff:ff
inet addr: 10.0.0.2 Bcast: 10.0.0.255 Mask: 255.255.255.0
UP RUNNING pat MTU:1500 ReasmMAX:65535 Group:2
IPRX bytes:0 unicastpkts:0 bcastpkts:0
IPTX bytes:0 unicastpkts:0 bcastpkts:0 dropkts:0
HWRX bytes:0 unicastpkts:0 bcastpkts:0
HWTX bytes:0 unicastpkts:0 bcastpkts:0 dropkts:0
1 eth0 Type:Ethernet Hwaddr 00:80:9f:24:ab:cf BRHWaddr ff:ff:ff:ff:ff:ff
inet addr: 10.10.10.147 Bcast: 10.10.10.255 Mask: 255.0.0.0
UP RUNNING pat MTU:1500 ReasmMAX:65535 Group:2
IPRX bytes:19791886 unicastpkts:11341 bcastpkts:290555
IPTX bytes:839550 unicastpkts:11477 bcastpkts:0 dropkts:0
HWRX bytes:0 unicastpkts:0 bcastpkts:0
HWTX bytes:0 unicastpkts:0 bcastpkts:0 dropkts:0
0 loop Type:0
inet addr:127.0.0.1 Bcast:127.255.255.255 Mask:255.0.0.0
UP RUNNING MTU:1500 ReasmMAX:65535 Group:1
IPRX bytes:116 unicastpkts:0 bcastpkts:2
IPTX bytes:0 unicastpkts:0 bcastpkts:0 dropkts:0
HWRX bytes:0 unicastpkts:0 bcastpkts:0
HWTX bytes:0 unicastpkts:0 bcastpkts:0 dropkts:0
=>
```

RELATED COMMANDS:

- `ip apadd` Add an IP address to an interface.
- `ip apdelete` Remove an IP address from an interface.

ip arpadd

Adds an entry to the ARP cache of a broadcast IP interface.

SYNTAX:

```
ip arpadd          intf = <interface name>
                   ip = <ip-address>
                   [hwaddr = <hardware-address>]
```

intf	The interface name.	REQUIRED
ip	The IP address.	REQUIRED
hwaddr	The hardware address (e.g. the Ethernet MAC address).	OPTIONAL

EXAMPLE:

```
=>ip arplist
Intf      IP-address      HW-address      Type
eth0      10.0.0.1         00:01:42:5f:7d:81  DYNAMIC
eth0      10.0.0.8         00:a0:24:ae:66:e1  DYNAMIC
eth0      10.0.1.99        52:41:53:20:20:4d  STATIC
eth0      10.0.1.100       52:41:53:20:f0:90  STATIC
=>ip arpadd intf=eth0 ip=10.0.0.2 hwaddr=00:10:a4:d0:9a:db
=>ip arplist
Intf      IP-address      HW-address      Type
eth0      10.0.0.1         00:01:42:5f:7d:81  DYNAMIC
eth0      10.0.0.8         00:a0:24:ae:66:e1  DYNAMIC
eth0      10.0.1.99        52:41:53:20:20:4d  STATIC
eth0      10.0.1.100       52:41:53:20:f0:90  STATIC
eth0      10.0.0.2         00:10:a4:d0:9a:db  STATIC
=>
```

RELATED COMMANDS:

<code>ip arpdelete</code>	Delete an ARP entry.
<code>ip arplist</code>	Show current ARP cache.

ip arpdelete

Remove an entry from the SpeedTouch™ ARP cache.

SYNTAX:

```
ip arpdelete      intf = <interface name>
                  ip = <ip-address>
                  [hwaddr = <hardware-address>]
```

intf	The interface name.	REQUIRED
ip	The IP address.	REQUIRED
hwaddr	The hardware address (e.g. the Ethernet MAC address).	OPTIONAL

EXAMPLE:

```
=>ip arplist
Intf      IP-address      HW-address      Type
eth0      10.0.0.1         00:01:42:5f:7d:81  DYNAMIC
eth0      10.0.0.8         00:a0:24:ae:66:e1  DYNAMIC
eth0      10.0.1.99        52:41:53:20:20:4d  STATIC
eth0      10.0.1.100       52:41:53:20:f0:90  STATIC
eth0      10.0.0.2         00:10:a4:d0:9a:db  STATIC
=>ip arpdelete intf=eth0 ip=10.0.0.2 hwaddr=00:10:a4:d0:9a:db
=>ip arplist
Intf      IP-address      HW-address      Type
eth0      10.0.0.1         00:01:42:5f:7d:81  DYNAMIC
eth0      10.0.0.8         00:a0:24:ae:66:e1  DYNAMIC
eth0      10.0.1.99        52:41:53:20:20:4d  STATIC
eth0      10.0.1.100       52:41:53:20:f0:90  STATIC
eth0      10.0.0.2         00:10:a4:d0:9a:db  STATIC
=>
```

RELATED COMMANDS:

ip arppadd	Add a static ARP entry.
ip arplist	Show current ARP cache.

ip arplist

Show the SpeedTouch™ ARP cache.

SYNTAX:

```
ip arplist
```

EXAMPLE:

```
=>ip arplist
Intf          IP-address      HW-address      Type
eth0          10.0.0.1        00:01:42:5f:7d:81  DYNAMIC
eth0          10.0.0.8        00:a0:24:ae:66:e1  DYNAMIC
eth0          10.0.1.99       52:41:53:20:20:4d  STATIC
eth0          10.0.1.100      52:41:53:20:f0:90  STATIC
eth0          10.0.0.2        00:10:a4:d0:9a:db  STATIC
=>
```

RELATED COMMANDS:

- `ip arpadd` Add a static entry to the ARP cache.
- `ip arpdelete` Delete an entry from the ARP cache.

ip config

Show/set global IP stack configuration options.

SYNTAX:

```
ip config      [forwarding = <{off|on}>]
               [firewalling = <{off|on}>]
               [redirects = <{off|on}>]
               [sourcerouting = <{off|on}>]
               [netbroadcasts = <{off|on}>]
               [ttl = <number{0-255}>]
               [fraglimit = <number{1-1024}>]
               [defragmode = <{normal|always|nat}>]
               [addrcheck = <{off|own|static|dynamic}>]
               [msscclamping = <{off|on}>]
```

forwarding	Disable (off) or enable (on) the IP routing functionality.	OPTIONAL
firewalling	Enable (on) or disable (off) IP firewalling (master switch). If applicable the CLI firewall level allows configuration of the SpeedTouch™ firewall. For security reasons this parameter is enabled per default. It is strongly recommended never to disable the SpeedTouch™ firewall.	OPTIONAL
redirects	Disable (off) or enable (on) the sending of ICMP redirect messages. A router can send a redirect message in case a shorter path than the path followed is discovered. For security reasons this parameter is disabled per default.	OPTIONAL
sourcerouting	Disallow (off) or allow (on) IP source routed packets. IP source routed packets are packets with the route to follow specified in the header. For security reasons this parameter is disabled per default.	OPTIONAL
netbroadcasts	Disallow (off) or allow (on) net directed broadcasts. This parameter is per default disabled. In case netbroadcasts are allowed no traces of netbroadcasts are generated.	OPTIONAL
ttl	A number between 0 and 255. Represents the default time-to-live (ttl) for locally generated IP packets. This parameter determines the number of hop-counts the IP packet may pass before it is dropped. Generally the time-to-live is 64 hop-counts. By limiting the time-to-live continuous circulation of IP packets on the network without ever reaching a destination is avoided.	OPTIONAL

fraglimit	<p>A number between 1 and 1024. Represents the maximum number of IP packet fragments waiting for completion. Generally the fragmentation limit is 64. By limiting the fragmentation limit the depletion of the buffer is avoided.</p>	OPTIONAL
defragmode	<p>Define which packets are reassembled under which circumstances. Choose between:</p> <ul style="list-style-type: none">• normal Packets to be forwarded will not be reassembled. Packets with local destination, i.e. destined for the Speed-Touch™, are reassembled.• always Packets are always reassembled.• nat Same behaviour as normal except for packets to be forwarded through the Network Address Translation (NAT) engine. Packets on which address translation is performed are reassembled as the NAT engine requires the entire packet.	OPTIONAL
addrcheck	<p>Set the level of IP address checks. Choose between:</p> <ul style="list-style-type: none">• off No address checking is performed. For advanced users only; in normal circumstances there should always be some kind of address checking.• own Minimum level of checking. Only the address configuration on the SpeedTouch™ is checked.• static Checking of the address configuration of the SpeedTouch™ and also of traffic: addresses of incoming packets; this checking is related to constants (e.g. an address may not be entirely composed of one's or zero's).• dynamic Besides the address configuration of the SpeedTouch™ itself, and besides the checking of traffic on a constants level, additional checking is performed on the IP addresses that are determined by the configuration, more specifically by the network.	OPTIONAL
mssclamping	<p>Disable (off) or enable (on) mss clamping for low mtu interfaces. Mss clamping assures that the size of a TCP packet never exceeds the available mtu of the outgoing interface. It is recommended not to disable this parameter.</p>	OPTIONAL

EXAMPLE:

```
=>ip config
Forwarding on
Firewalling off
Sendredirects off
Sourcerouting on
NetBroadcasts off
Default TTL 128
Fraglimit 32 fragments
Fragcount currently 0 fragments
Defragment mode : always
Address checks : static
Mss clamping : on
=>ip config firewalling=on ttl=64 fraglimit=64 defragmode=nat
=>ip config
Forwarding on
Firewalling on
Sendredirects off
Sourcerouting on
NetBroadcasts off
Default TTL 64
Fraglimit 64 fragments
Fragcount currently 0 fragments
Defragment mode : nat
Address checks : static
Mss clamping : on
=>
```

RELATED COMMANDS:

[ip ifconfig](#)

Configure interface parameters.

ip flush

Flush complete IP configuration. Dynamic configurations (e.g. from PPP or CIP links) remain.

Note As an ip flush causes all local IP connectivity to be deleted, do not execute this command during an IP based local connection, e.g. a Telnet CLI session, or web based CLI access.

Note The flush command does not impact previously saved configurations.

SYNTAX:

```
ip flush
```

EXAMPLE:

```
=>ip aplist
3 cip1 Type:ATM
  inet addr:172.16.0.5 Bcast:172.16.0.255 Mask:255.255.255.0
  UP RUNNING pat MTU:9180 ReasmMAX:65535 Group:0
  IPRX bytes:0 unicastpkts:0 bcastpkts:0
  IPTX bytes:0 unicastpkts:0 bcastpkts:0 droppkts:0
  HWRX bytes:0 unicastpkts:0 bcastpkts:0
  HWTX bytes:0 unicastpkts:0 bcastpkts:0 droppkts:0
2 eth0 Type:EthernetWaddr 00:80:9f:24:ab:cf BRHWaddr ff:ff:ff:ff:ff:ff
  inet addr: 10.0.0.2 Bcast: 10.0.0.255 Mask: 255.255.255.0
  UP RUNNING pat MTU:1500 ReasmMAX:65535 Group:2
  IPRX bytes:0 unicastpkts:0 bcastpkts:0
  IPTX bytes:0 unicastpkts:0 bcastpkts:0 droppkts:0
  HWRX bytes:0 unicastpkts:0 bcastpkts:0
  HWTX bytes:0 unicastpkts:0 bcastpkts:0 droppkts:0
0 loop Type:0
  inet addr:127.0.0.1 Bcast:127.255.255.255 Mask:255.0.0.0
  UP RUNNING MTU:1500 ReasmMAX:65535 Group:1
  IPRX bytes:116 unicastpkts:0 bcastpkts:2
  IPTX bytes:0 unicastpkts:0 bcastpkts:0 droppkts:0
  HWRX bytes:0 unicastpkts:0 bcastpkts:0
  HWTX bytes:0 unicastpkts:0 bcastpkts:0 droppkts:0
=>ip flush
=>ip aplist
3 cip1 Type:ATM
  inet addr:172.16.0.5 Bcast:172.16.0.255 Mask:255.255.255.0
  UP RUNNING pat MTU:9180 ReasmMAX:65535 Group:0
  IPRX bytes:0 unicastpkts:0 bcastpkts:0
  IPTX bytes:0 unicastpkts:0 bcastpkts:0 droppkts:0
  HWRX bytes:0 unicastpkts:0 bcastpkts:0
  HWTX bytes:0 unicastpkts:0 bcastpkts:0 droppkts:0
0 loop Type:0
  inet addr: 127.0.0.1 Bcast:127.255.255.255 Mask:255.0.0.0
  UP RUNNING MTU:1500 ReasmMAX:65535 Group:1
  IPRX bytes:116 unicastpkts:0 bcastpkts:2
  IPTX bytes:0 unicastpkts:0 bcastpkts:0 droppkts:0
  HWRX bytes:0 unicastpkts:0 bcastpkts:0
  HWTX bytes:0 unicastpkts:0 bcastpkts:0 droppkts:0
=>
```

ip ifconfig

Configure IP interface parameters.

SYNTAX:

```
ip ifconfig      intf = <interface name>
                  [mtu = <number{68-20000}>]
                  [status = <{down|up}>]
                  [hwaddr = <hardware-address>]
                  [group = <{wan|local|lan}>]
                  [linksensing = <{off|on}>]
```

intf	The IP interface name.	REQUIRED
mtu	A number between 68 and 20000. Represents the maximum transmission unit, i.e. the maximum packet size (including IP header) to use on this interface. The default value depends on the connection and packet service for which the interface was created.	OPTIONAL
status	The administrative status of the interface. Choose between: <ul style="list-style-type: none"> down up 	OPTIONAL
hwaddr	The hardware address (e.g. the Ethernet MAC address) of this interface.	OPTIONAL
group	The group this interface belongs to (e.g. for oriented firewalling).	OPTIONAL
linksensing	The IP interface's awareness of link state transitions.	OPTIONAL

EXAMPLE:

```
=>ip iflist
Interface GRP  MTU  RX      TX      TX-DROP  STATUS  HWADDR
0  loop    1    1500  116     0        0        UP
1  eth0    2    3000  21045795  1019664  0        UP      00:80:9f:24:ab:cf
2  NewETHoA 0    1500  0        0        0        UP      00:80:9f:24:ab:cf
5  cip0    0    9180  0        0        0        UP
=>ip ifconfig intf=eth0 mtu=1500
=>ip iflist
Interface GRP  MTU  RX      TX      TX-DROP  STATUS  HWADDR
0  loop    1    1500  116     0        0        UP
1  eth0    2    1500  21054963  1025417  0        UP      00:80:9f:24:ab:cf
2  NewETHoA 0    1500  0        0        0        UP      00:80:9f:24:ab:cf
5  cip0    0    9180  0        0        0        UP
=>
```

RELATED COMMANDS:

`ip config` Show/set global IP stack configuration options.

ip iflist

Show all current interfaces.

SYNTAX:

```
ip iflist
```

EXAMPLE:

```
=>ip iflist
Interface  GRP  MTU  RX      TX      TX-DROP  STATUS  HWADDR
0  loop  1    1500  116     0        0       UP
1  eth0  2    3000  21045795  1019664  0       UP      00:80:9f:24:ab:cf
2  NewETHoA  0    1500  0        0        0       UP      00:80:9f:24:ab:cf
5  cip0  0    9180  0        0        0       UP
=>
```

RELATED COMMANDS:

[ip ifconfig](#) Configure interface parameters.

ip ifwait

Wait for a status change of an IP interface.

SYNTAX:

```
ip ifwait          intf = <interface name>
                   [timeout = <number{1-600000}>]
                   [adminstatus = <{down|up}>]
                   [operstatus = <{down|up}>]
                   [linkstatus = <{down|up}>]
```

intf	The IP interface name.	REQUIRED
timeout	A number between 1 and 600000. Represents the timeout in seconds.	OPTIONAL
adminstatus	The administrative state of the interface. Choose between: <ul style="list-style-type: none"> • down • up 	OPTIONAL
operstatus	The operational state of the interface. Choose between: <ul style="list-style-type: none"> • down • up 	OPTIONAL
linkstatus	The link state of the interface. Choose between: <ul style="list-style-type: none"> • down • up 	OPTIONAL

ip ping

Send ICMP ECHO_REQUEST packets.

SYNTAX:

```
ip ping          addr = <ip-address>
                  [count = <number{1-1000000}>]
                  [size = <number{1-20000}>]
                  [interval = <number{100-1000000}>]
                  [listen = <{off|on}>]
```

addr	The destination IP address.	REQUIRED
count	A number between 1 and 1000000. Represents the number of pings to send.	OPTIONAL
size	A number between 1 and 20000 (bytes). Represents the size of the ping packet(s).	OPTIONAL
interval	A number between 100 and 10000000 (milliseconds). Represents the intermediate interval between two sent ICMP packets.	OPTIONAL
listen	Listen for incoming ICMP packets (on) or only send ICMP packets (off).	OPTIONAL

EXAMPLE:

```
=>ip ping addr=10.0.0.148 listen=off
=>ip ping addr=10.0.0.148 listen=on
9 bytes from 10.0.0.148: Echo Request
=>ip ping addr=10.0.0.148 count=15 listen=on
9 bytes from 10.0.0.148: Echo Request
9 bytes from 10.0.0.148: Echo Request
9 bytes from 10.0.0.148: Echo Request
9 bytes from 10.0.0.148: Echo Request
9 bytes from 10.0.0.148: Echo Request
9 bytes from 10.0.0.148: Echo Request
9 bytes from 10.0.0.148: Echo Request
9 bytes from 10.0.0.148: Echo Request
9 bytes from 10.0.0.148: Echo Request
9 bytes from 10.0.0.148: Echo Request
9 bytes from 10.0.0.148: Echo Request
9 bytes from 10.0.0.148: Echo Request
9 bytes from 10.0.0.148: Echo Request
9 bytes from 10.0.0.148: Echo Request
9 bytes from 10.0.0.148: Echo Request
9 bytes from 10.0.0.148: Echo Request
=>
```

RELATED COMMANDS:

`ip sendto` Send UDP packets.

ip rtadd

Add a route to the SpeedTouch™ routing table.

SYNTAX:

```
ip rtadd          dst = <ip-address>
                  [dstmsk = <ip-mask(dotted or cidr)>]
                  [label = <{}>]
                  [gateway = <ip-address>]
                  [intf = <interface name>]
                  [metric = <number{0-100}>]
```

dst	The destination IP address(es) for this route. Supports cidr notation.	REQUIRED
dstmsk	The destination IP address mask.	OPTIONAL
label	The name of the label.	OPTIONAL
gateway	The IP address of the next hop. Must be directly connected. The parameters 'gateway' and 'intf' are mutually exclusive.	OPTIONAL
intf	Only for special interface routes: the outgoing IP interface name. The parameters 'gateway' and 'intf' are mutually exclusive.	OPTIONAL
metric	The metric for this route (weight factor).	OPTIONAL

EXAMPLE:

```
=>ip rtlist
  Destination Label      Gateway      Intf Mtrc Status
  10.0.0.0/24           10.0.0.140  eth0  0  [UP]
  10.0.0.140/32         10.0.0.140  eth0  0  [UP]
  127.0.0.1/32          127.0.0.1   loop  0  [UP]
=>ip rtadd dst=10.10.0.0/24 src=10.0.0.0/24 gateway=10.0.0.140
=>ip rtlist
  Destination Label      Gateway      Intf Mtrc Status
  10.0.0.0/24           10.0.0.140  eth0  0  [UP]
  10.10.0.0/24          10.0.0.140  eth0  0  [UP]
  10.0.0.140/32         10.0.0.140  eth0  0  [UP]
  127.0.0.1/32          127.0.0.1   loop  0  [UP]
=>
```

RELATED COMMANDS:

ip rtdelete	Remove a route from the routing table.
ip rtlist	Show current routing table.

ip rtdelete

Delete a route from the SpeedTouch™ routing table.

SYNTAX:

```
ip rtdelete      dst = <ip-address>
                  [dstmsk = <ip-mask(dotted or cidr)>]
                  [label = <{}>]
                  [gateway = <ip-address>]
                  [intf = <interface name>]
```

dst	The destination IP address(es) for this route. Supports cidr notation.	REQUIRED
dstmsk	The destination IP address mask.	OPTIONAL
label	The name of the label.	OPTIONAL
gateway	The IP address of the next hop. Must be directly connected. The parameters 'gateway' and 'intf' are mutually exclusive.	OPTIONAL
intf	Only for special interface routes: the outgoing IP interface name. The parameters 'gateway' and 'intf' are mutually exclusive.	OPTIONAL

EXAMPLE:

```
=>ip rtlist
  Destination Label      Gateway      Intf Mtrc Status
  10.0.0.0/24          10.0.0.140   eth0  0  [UP]
  10.10.0.0/24         10.0.0.140   eth0  0  [UP]
  10.0.0.140/32        10.0.0.140   eth0  0  [UP]
  127.0.0.1/32         127.0.0.1    loop  0  [UP]
=>ip rtdelete dst=10.10.0.0/24 src=10.0.0.0/24 gateway=10.0.0.140
=>ip rtlist
  Destination Label      Gateway      Intf Mtrc Status
  10.0.0.0/24          10.0.0.140   eth0  0  [UP]
  10.0.0.140/32        10.0.0.140   eth0  0  [UP]
  127.0.0.1/32         127.0.0.1    loop  0  [UP]
=>
```

RELATED COMMANDS:

- ip rtadd Add a route to the routing table.
- ip rtlist Show current routing table.

ip rtlist

Show current SpeedTouch™ routing table.

SYNTAX:

```
ip rtlist
```

EXAMPLE:

```
=>ip rtlist
  Destination Label      Gateway      Intf Mtrc Status
  10.0.0.0/24           10.0.0.140  eth0  0  [UP]
  10.0.0.140/32         10.0.0.140  eth0  0  [UP]
  127.0.0.1/32          127.0.0.1   loop  0  [UP]
=>
```

RELATED COMMANDS:

`ip rtadd`

Add a route to the routing table.

`ip rtdelete`

Remove a route from the routing table.

ip sendto

Send UDP packets.

SYNTAX:

```
ip sendto      addr = <ip-address>
                [count = <number{1-1000000}>]
                [size = <number{1-20000}>]
                [interval = <number{100-1000000}>]
                [listen = <{off|on}>]
                [srcport = <number{1-65535}>]
                dstport = <number{1-65535}>
```

addr	The destination IP address.	REQUIRED
count	A number between 1 (default) and 1000000. Represents the number of UDP packets to send.	OPTIONAL
size	A number between 1 (default) and 20000 (bytes). Represents the size of the ping packet(s).	OPTIONAL
interval	A number between 100 (default) and 10000000 (milliseconds). Represents the intermediate interval between two sent UDP packets.	OPTIONAL
listen	Listen for incoming UDP packets (on) or only send UDP packets (off). Default is off.	OPTIONAL
srcport	The UDP source port number to use.	OPTIONAL
dstport	The UDP destination port number to send to.	REQUIRED

EXAMPLE:

```
=>ip sendto addr=10.0.0.148 listen=on srcport=19 dstport=1025
=>ip sendto addr=10.0.0.148 listen=on srcport=19 dstport=1025
1 bytes from 10.0.0.148:1025
41                                     A
=>ip sendto addr=10.0.0.148 count=3 listen=on srcport=19 dstport=1025
1 bytes from 10.0.0.148:1025
41                                     A
1 bytes from 10.0.0.148:1025
41                                     A
1 bytes from 10.0.0.148:1025
41                                     A
=>
```

RELATED COMMANDS:

[ip ping](#)

Send ICMP ECHO_REQUEST packets.

ip traceroute

Sends ICMP/UDP packets to trace the ip path.

SYNTAX:

```
ip traceroute      addr = <ip-address>
                   [count = <number{1-10}>]
                   [size = <number{1-20000}>]
                   [interval = <number{1000-60000}>]
                   [maxhops = <number{1-255}>]
                   [dstport = <number{1-65535}>]
                   [maxfail = <number{0-255}>]
                   [type = <{icmp|udp}>]
                   [utime = <{no|yes}>]
```

addr	The destination IP address.	REQUIRED
count	A number between 1 and 10. Represents the number of times to reissue a traceroute request with the same time to live. Default is 3.	OPTIONAL
size	A number between 1 and 20000 (bytes). Represents the size of the traceroute packet(s). Default is 1.	OPTIONAL
interval	A number between 100 and 10000000 (milliseconds). Represents the intermediate interval between two packets. Default is 1000.	OPTIONAL
maxhops	A number between 1 and 255. Represents the upper limit on the number of routers through which a packet can pass. Default is 30.	OPTIONAL
dstport	A number between 1 and 65535. Represents the UDP destination port number to send to.	OPTIONAL
maxfail	A number between 0 and 255. Represents the max number of consecutive time-outs allowed before terminating a traceroute request. Default is 5	OPTIONAL
type	The type of traceroute packet(s). Choose between: <ul style="list-style-type: none"> • icmp (default) • udp 	OPTIONAL
utime	Display time in useconds (yes) or not (no). Default is yes.	OPTIONAL

EXAMPLE:

```
=>ip traceroute addr = 192.193.195.250 count=3 size=1 interval=1000 maxhops=30 dstport=33433  
maxfail=5 type=icmp utime=yes  
:ip traceroute addr=192.193.195.250  
ttl=1 192.193.195.250 676 us 1351 us 648 us  
  
=>
```

ip auto flush

Flush the autoIP interfaces.

SYNTAX:

```
ip auto flush
```

RELATED COMMANDS:

<code>ip auto ifadd</code>	Create a new autoIP interface.
<code>ip auto ifdelete</code>	Delete an existing autoIP interface.
<code>ip auto iflist</code>	Show the autoIP interfaces.

ip auto ifadd

Create a new autoIP interface.

SYNTAX:

```
ip auto ifadd      intf = <interface name>
                   [addr = <ip-address>]
```

intf	The name of the IP interface for which a link-local address has to be allocated.	REQUIRED
addr	The preferred link-local IP address.	OPTIONAL

RELATED COMMANDS:

ip auto flush	Flush the autoIP interfaces.
ip auto ifdelete	Delete an existing autoIP interface.
ip auto iflist	Show the autoIP interfaces.

ip auto ifattach

Select and assign a link-local address to an autoIP interface.

SYNTAX:

```
ip auto ifattach    intf = <interface name>
```

intf	The name of the IP interface for which a link-local address has to be attached.	REQUIRED
------	---	----------

RELATED COMMANDS:

ip auto ifdetach	Release the link-local address for the given autoIP interface.
----------------------------------	--

ip auto ifconfig

Configure an autoIP interface.

SYNTAX:

```
ip auto ifconfig    intf = <interface name>
                   [addr = <ip-address>]
                   [poolstart = <ip-address>]
                   [poolend = <ip-address>]
                   [netmask = <ip-mask(dotted or cidr)>]
                   [claim = <number{0-65535}>]
                   [defence = <number{0-65535}>]
                   [probe = <number{0-65535}>]
                   [interval = <number{1-65535}>]
```

intf	The name of the autoIP interface to configure.	REQUIRED
addr	The preferred link-local IP address.	OPTIONAL
poolstart	The start IP address of the link-local address pool.	OPTIONAL
poolend	The end IP address of the link-local address pool.	OPTIONAL
netmask	The netmask of the link-local IP address pool.	OPTIONAL
claim	A number between 0 and 65535. Represents the number of link-local address selection retries before giving up. Default is 10.	OPTIONAL
defence	A number between 0 and 65535. Represents the number of times the link-local address is defended before releasing the address. Default is 5.	OPTIONAL
probe	A number between 0 and 65535. Represents the number of ARP probes to be sent before accepting a link-local address. Default is 4.	OPTIONAL
interval	A number between 1 and 65535 (seconds). Represents the time interval between two ARP probe transmissions. Default is 2.	OPTIONAL

ip auto ifdelete

Delete an existing autoIP interface.

SYNTAX:

```
ip auto ifdelete    intf = <interface name>
```

intf	The name of the IP interface to be deleted.	REQUIRED
------	---	----------

RELATED COMMANDS:

ip auto flush	Flush the autoIP interfaces.
ip auto ifadd	Create a new autoIP interface.
ip auto iflist	Show the autoIP interfaces.

ip auto ifdetach

Release the link-local address for the given autoIP interface.

SYNTAX:

```
ip auto ifdetach    intf = <interface name>
```

intf	The name of the IP interface for which a link-local address has to be detached.	REQUIRED
------	---	----------

RELATED COMMANDS:

ip auto ifattach	Select and assign a link-local address to an autoIP interface.
----------------------------------	--

ip auto iflist

Shows the autoIP interfaces.

SYNTAX:

```
ip auto iflist [intf = <interface name>]
```

intf	The name of the interface to be listed. If no name is specified, all the autoIP interfaces are shown.	OPTIONAL
------	---	----------

EXAMPLE:

```
=>ip auto iflist
eth0      : [CLAIMED] 169.254.138.1
           poolstart = 169.254.1.1 poolend = 169.254.254.254 netmask = 255.255.0.0
           claim : 10 defence : 5 probe : 4 interval : 2 (sec)
           probes sent = 2
           collisions = 0

=>
```

RELATED COMMANDS:

ip auto ifadd	Create a new autoIP interface.
ip auto ifdelete	Delete an existing autoIP interface.

IPoA Commands

Contents

This chapter covers the following commands:

Topic	Page
ipoa flush	226
ipoa ifadd	227
ipoa ifattach	228
ipoa ifconfig	229
ipoa ifdelete	231
ipoa ifdetach	232
ipoa iflist	233

ipoa flush

Flush complete Routed IPoA configuration.

Note The flush command does not impact previously saved configurations.

SYNTAX:

```
ipoa flush
```

RELATED COMMANDS:

ipoa ifadd	Create a new Routed IPoA interface.
ipoa ifattach	Attach (i.e. connect) an Routed IPoA interface.
ipoa ifconfig	Configure a Routed IPoA interface.
ipoa ifdelete	Delete a Routed IPoA interface.
ipoa ifdetach	Detach a Routed IPoA interface.
ipoa iflist	Show current Routed IPoA interfaces.

ipoa ifadd

Create a new Routed IPoA interface.

SYNTAX:

```
ipoa ifadd      [intf = <string>]
                [dest = <phonebook entry>]
```

intf	The name for the new Routed IPoA interface. If not specified, the destination will double as interface name.	OPTIONAL
dest	The destination for the new Routed IPoA interface. Typically, a phonebook entry.	OPTIONAL

EXAMPLE:

```
=>ipoa iflist
IPoA_1      :  dest : IPoA_1
              Retry : 10   QoS : default   Encaps : llc/snap   Fcs : off
              Connection State : connected

=>phonebook list
Name      Type   Use  Address
IPoA_1    ipoa   1    8.35
IPoA_2    ipoa   0    8.36
=>ipoa ifadd dest=IPoA_2
=>ipoa iflist
IPoA_1      :  dest : IPoA_1
              Retry : 10   QoS : default   Encaps : llc/snap   Fcs : off
              Connection State : connected

IPoA_2      :  dest : IPoA_2
              Retry : 10   QoS : default   Encaps : llc/snap   Fcs : off
              Connection State : not-connected

=>
```

RELATED COMMANDS:

ipoa flush	Flush complete Routed IPoA configuration.
ipoa ifattach	Attach (i.e. connect) an Routed IPoA interface.
ipoa ifconfig	Configure a Routed IPoA interface.
ipoa ifdelete	Delete a Routed IPoA interface.
ipoa ifdetach	Detach a Routed IPoA interface.
ipoa iflist	Show current Routed IPoA interfaces.

ipoa ifattach

Attach (i.e. connect) an Routed IPoA interface.

SYNTAX:

```
ipoa ifattach    intf = <intfname>
```

intf	The name of the Routed IPoA interface to attach.	REQUIRED
------	--	----------

EXAMPLE:

```
=>ipoa iflist
IPoA_PVC1   :  dest : Br4
              Retry : 10   QoS : default   Encaps : llc/snap   Fcs : off
              Connection State : connected

Br3         :  dest : Br3
              Retry : 10   QoS : default   Encaps : llc/snap   Fcs : off
              Connection State : not-connected

=>ipoa ifattach intf=Br3
IPoA_PVC1   :  dest : Br4
              Retry : 10   QoS : default   Encaps : llc/snap   Fcs : off
              Connection State : connected

Br3         :  dest : Br3
              Retry : 10   QoS : default   Encaps : llc/snap   Fcs : off
              Connection State : connected

=>
```

RELATED COMMANDS:

ipoa ifadd	Create a new Routed IPoA interface.
ipoa ifconfig	Configure a Routed IPoA interface.
ipoa ifdelete	Delete a Routed IPoA interface.
ipoa ifdetach	Detach a Routed IPoA interface.
ipoa iflist	Show current Routed IPoA interfaces.

ipoa ifconfig

Configure an IPoA interface.

SYNTAX:

```
ipoa ifconfig    intf = <string>
                  [dest = <intfname>]
                  [qos = <string>]
                  [encaps = <{llc/snap|vcmux}>]
                  [retry = <number {0-65535}>]
```

intf	The name of the Routed IPoA interface to configure.	REQUIRED
dest	The destination for this interface. Typically a phonebook entry. This parameter needs only to be specified in case of an interface created without specified destination.	OPTIONAL
qos	The name of a configured Quality Of Service book entry. If not specified, the default Quality Of Service book entry will be used.	OPTIONAL
encaps	The type of encapsulation to be used for this bridge interface. Choose between: <ul style="list-style-type: none"> • llc/snap • vcmux 	OPTIONAL
retry	A number between 0 and 65535. Represents the number of Wide Area Network (WAN) connection setup retries before giving up. By default the retry value is 10.	OPTIONAL

EXAMPLE:

```
=>ipoa iflist
IPoA_PVC1   :  dest : Br4
                Retry : 10   QoS : default   Encaps : llc/snap   Fcs : off
                Connection State : connected

=>ipoa ifconfig intf=IPoA_PVC1 encaps=llc/snap retry=15
=>ipoa iflist
Moreipoa    :  dest : Br4
                Retry : 15   QoS : default   Encaps : llc/snap   Fcs : off
                Connection State : connected
                RX bytes: 0     frames: 0
                TX bytes: 0     frames: 0     dropframes: 0

=>
```

RELATED COMMANDS:

<code>ipoa ifadd</code>	Create a new Routed IPoA interface.
<code>ipoa ifattach</code>	Attach (i.e. connect) an Routed IPoA interface.
<code>ipoa ifdelete</code>	Delete a Routed IPoA interface.
<code>ipoa ifdetach</code>	Detach a Routed IPoA interface.
<code>ipoa iflist</code>	Show current Routed IPoA interfaces.

ipoa ifdelete

Delete a Routed IPoA interface.

SYNTAX:

```
ipoa ifdelete    intf = <intfname>
```

intf	The name of the Routed IPoA interface.	REQUIRED
------	--	----------

EXAMPLE:

```
=>ipoa iflist
Newipoa      :  dest : Br3
                Retry : 10    QoS : default    Encaps : llc/snap    Fcs : off
                Connection State : connected
                RX bytes: 0    frames: 0
                TX bytes: 0    frames: 0    dropframes: 0
Moreipoa     :  dest : Br4
                Retry : 10    QoS : default    Encaps : llc/snap    Fcs : off
                Connection State : not-connected
=>ipoa ifdelete intf=Moreipoa
=>ipoa iflist
Newipoa      :  dest : Br3
                Retry : 10    QoS : default    Encaps : llc/snap    Fcs : off
                Connection State : connected
                RX bytes: 0    frames: 0
                TX bytes: 0    frames: 0    dropframes: 0
=>
```

RELATED COMMANDS:

ipoa ifadd	Create a new Routed IPoA interface.
ipoa ifattach	Attach (i.e. connect) an Routed IPoA interface.
ipoa ifconfig	Configure a Routed IPoA interface.
ipoa ifdetach	Detach a Routed IPoA interface.
ipoa iflist	Show current Routed IPoA interfaces.

ipoa ifdetach

Detach a Routed IPoA interface.

SYNTAX:

```
ipoa ifdetach intf = <intfname>
```

intf	The name of the Routed IPoA interface.	REQUIRED
------	--	----------

EXAMPLE:

```
=>ipoa iflist
Newipoa      :  dest : Br3
                Retry : 10    QoS : default    Encaps : llc/snap    Fcs : off
                Connection State : connected
                RX bytes: 0    frames: 0
                TX bytes: 0    frames: 0    dropframes: 0
Moreipoa     :  dest : Br4
                Retry : 10    QoS : default    Encaps : llc/snap    Fcs : off
                Connection State : connected
                RX bytes: 0    frames: 0
                TX bytes: 0    frames: 0    dropframes: 0
=>ipoa ifdetach intf=Moreipoa
=>ipoa iflist
Newipoa      :  dest : Br3
                Retry : 10    QoS : default    Encaps : llc/snap    Fcs : off
                Connection State : connected
                RX bytes: 0    frames: 0
                TX bytes: 0    frames: 0    dropframes: 0
Moreipoa     :  dest : Br4
                Retry : 10    QoS : default    Encaps : llc/snap    Fcs : off
                Connection State : not-connected
=>
```

RELATED COMMANDS:

ipoa ifadd	Create a new Routed IPoA interface.
ipoa ifattach	Attach (i.e. connect) an Routed IPoA interface.
ipoa ifconfig	Configure a Routed IPoA interface.
ipoa ifdelete	Delete a Routed IPoA interface.
ipoa iflist	Show current Routed IPoA interfaces.

ipoa iflist

Show all or a specified Routed IPoA interface(s).

SYNTAX:

```
ipoa iflist [intf = <intfname>]
```

intf	The name of the Routed IPoA interface. If not specified all Routed IPoA interfaces are listed.	OPTIONAL
------	---	----------

EXAMPLE:

```
=>ipoa iflist
Newipoa      :  dest : Br3
                Retry : 10    QoS : default    Encaps : llc/snap    Fcs : off
                Connection State : connected
                RX bytes: 0    frames: 0
                TX bytes: 0    frames: 0    dropframes: 0
Moreipoa     :  dest : Br4
                Retry : 10    QoS : default    Encaps : llc/snap    Fcs : off
                Connection State : connected
                RX bytes: 0    frames: 0
                TX bytes: 0    frames: 0    dropframes: 0
=>
```

RELATED COMMANDS:

ipoa ifadd	Create a new Routed IPoA interface.
ipoa ifattach	Attach (i.e. connect) an Routed IPoA interface.
ipoa ifconfig	Configure a Routed IPoA interface.
ipoa ifdelete	Delete a Routed IPoA interface.
ipoa ifdetach	Detach a Routed IPoA interface.

Label Commands

Contents

This chapter covers the following commands:

Topic	Page
label add	236
label config	237
label delete	238
label flush	239
label list	240
label troff	241
label tron	242
label chain create	243
label chain delete	244
label chain flush	245
label chain list	246
label rule clear	247
label rule create	248
label rule delete	250
label rule flush	251
label rule list	252
label rule stats	253

label add

Create a new label.

SYNTAX:

```
label add          name = <string>
```

name	The name of the label to add.	REQUIRED
------	-------------------------------	----------

EXAMPLE:

```
=>label add name=Label1
=>label list
Name      Tosmark  Tos      Use      Trace
Label1    disabled 0       0        disabled
=>
```

RELATED COMMANDS:

label config Configure a label.

label delete Delete a label.

label config

Configure a label.

SYNTAX:

```
label config      name = <string>
                  [tosmarking = <{disabled|enabled}>]
                  [tos = <number{0-255}>]
                  [trace = <{disabled|enabled}>]
```

name	The name of the label to configure.	REQUIRED
tosmarking	Enable/disable tos marking. Choose between <ul style="list-style-type: none"> disabled (default). enabled. 	OPTIONAL
tos	A number between 0 and 255. Represents the Type of Service (ToS) specification in the IP packet (used for tosmarking). Default is 0.	OPTIONAL
trace	Enable/disable tracing for this label. Choose between <ul style="list-style-type: none"> disabled (default). enabled. 	OPTIONAL

EXAMPLE:

```
=>label list
Name      Tosmark  Tos      Use      Trace
Label1    disabled  0        0        disabled
=>label config name=fromconference tosmarking=enabled tos=184
=>label list
Name      Tosmark  Tos      Use      Trace
Label1    disabled  0        0        disabled
fromconference enabled  184     0        disabled
=>
```

Note A ToS value of 184 decimal is used for DiffServ Expedited Forwarding data.

RELATED COMMANDS:

label add	Add a label.
label delete	Delete a label.

label delete

Delete a label

SYNTAX:

```
label delete      name = <string>
                  [force = <{no|yes}>]
```

name	The name of the label to delete.	REQUIRED
force	Force delete and cleanup references even when the label is still in use. Choose between <ul style="list-style-type: none">no (default).yes.	OPTIONAL

RELATED COMMANDS:

label add	Add a label.
label config	Configure a label.

label flush

Flushes all labels.

Note The flush command does not impact previously saved configurations.

SYNTAX:

```
label flush
```

EXAMPLE:

```
=>label list
Name      Tosmark  Tos      Use      Trace
Label1    enabled  25       0        enabled
=>label flush
=>label list
Name      Tosmark  Tos      Use      Trace
=>
```

label list

Show the association(s) between all hooks or a specified hook and their chain(s).

SYNTAX:

```
label list [name = <string>]
```

name	The name of the label to be shown. If no name is specified, all labels will be shown	OPTIONAL
------	---	----------

EXAMPLE:

```
=>label list
Name    Tosmark Tos    Use    Trace
Label1  enabled 25    0      enabled
Label2  disabled 0      0      disabled
=>label list name=Label2
Name    Tosmark Tos    Use    Trace
Label2  disabled 0      0      disabled
=>
```

RELATED COMMANDS:

label flush	Flushes all labels.
-------------	---------------------

label troff

Disable verbose console messaging.

SYNTAX:

```
label troff
```

EXAMPLE:

```
=>label troff
```

RELATED COMMANDS:

label tron

Enable verbose console messaging.

label tron

Enable verbose console messaging.

SYNTAX:

```
label tron
```

EXAMPLE:

```
=>label tron
```

RELATED COMMANDS:

label troff

Disable verbose console messaging.

label chain create

Create a new chain.

SYNTAX:

```
label chain create    chain = <string>
```

chain

The name of the chain to create.

REQUIRED

RELATED COMMANDS:

[label chain delete](#)

Delete a chain.

[label chain list](#)

Show a list of all current chains.

label chain delete

Delete a chain.

SYNTAX:

```
label chain delete    chain = <string>
```

chain	The name of the chain to be deleted.	REQUIRED
-------	--------------------------------------	----------

RELATED COMMANDS:

- label chain create Create a chain.
- label chain list Show a list of all chains.

label chain flush

Flush all chains.

SYNTAX:

```
label chain flush
```

RELATED COMMANDS:

label chain create	Create a chain.
label chain delete	Delete a chain.
label chain list	Show a list of all chains.

label chain list

Show a list of all current chains.

SYNTAX:

```
label chain list
```

RELATED COMMANDS:

label chain create

Create a chain.

label chain delete

Delete a chain.

label rule clear

Clear statistics for a given rule.

SYNTAX:

```
label rule clear    [chain = <string>]
                   [index = <number>]
```

chain	The name of the chain in which the rule is to be found.	OPTIONAL
index	The index number (determined by the position) of the rule in the chain.	OPTIONAL

RELATED COMMANDS:

label rule create	Create a rule.
label rule delete	Delete a specified rule in a chain.
label rule flush	Delete all rules in a chain.
label rule list	Show a list of all (or a specified) chains' rules.
label rule stats	Show statistics for all (or a specified) chains' rules.

label rule create

Create a rule.

SYNTAX:

```
label rule create chain = <string>
                  [index = <number>]
                  [srcintf [!]= <string>]
                  [srcintfgrp [!]= <{wan|local|lan} or number>]
                  [srcbridgeport [!]= <number>]
                  [src [!]= <ip-range>]
                  [dst [!]= <ip-address>]
                  [tos [!]= <number{0-255}>]
                  [precedence [!]= <number{0-7}>]
                  [dscp [!]= <number{0-63}>]
                  [prot [!]= <{<supported IP protocol name>|<number>}>]
                  [srcport [!]= <{<supported TCP/UDP port name>|<number>}>]
                  [srcportend = <{<supported TCP/UDP port name>|<number>}>]
                  [dstport [!]= <{<supported TCP/UDP port name>|<number>}>]
                  [dstportend = <{<supported TCP/UDP port name>|<number>}>]
                  [clink = <string>]
                  [log = <{no|yes}>]
                  label = <{None|link|<labels>}>
```

chain	The name of the chain to insert the rule in.	REQUIRED
index	The number of the rule before which the new rule must be added.	OPTIONAL
srcintf	The name of the interface the packet should arrive on to make this rule apply. (NOT applicable if used in a chain assigned to the output hook)	OPTIONAL
srcintfgrp	The interface group the packet should arrive on.	OPTIONAL
srcbridgeport	A number between 0 and 6. Represents the bridge port the virtual packet should arrive on. Use :bridge iflist for a list of available bridge ports.	OPTIONAL
src	The source IP address (range) the packet should come from. (Supports ip/mask notation).	OPTIONAL
dst	The destination IP address (range) the packet should be going to. (Supports ip/mask notation).	OPTIONAL
tos	A number between 0 and 255. Represents the Type Of Service specification which should be expected in the IP packet. The Type of Service numbering specification is in accordance to the latest version of RFC1700: Assigned numbers.	OPTIONAL

precedence	A number between 0 and 7. Represents the precedence in the IP packet (is part of tos)	OPTIONAL
dscp	The DSCP in the IP packet (part of tos).	OPTIONAL
prot	The protocol (name or number) expected in the IP packet. Select one of the supported protocol names (See “Supported Internet Protocol (IP) Protocol Names” on page 413 for a listing of protocol names supported by the SpeedTouch™). Alternatively, specify the protocol number.	OPTIONAL
srcport	The TCP/UDP port (or beginning of range) the packet is coming from. Select one of the supported TCP/UDP port names (See “Supported TCP/UDP Port Names” on page 414 for a listing of TCP/UDP port names supported by the SpeedTouch™). Alternatively, specify the protocol number.	OPTIONAL
srcportend	The source TCP/UDP port range end (inclusive). (Only applicable for ranges) Select one of the supported TCP/UDP port names (See “Supported TCP/UDP Port Names” on page 414 for a listing of TCP/UDP port names supported by the SpeedTouch™). Alternatively, specify the protocol number.	OPTIONAL
dstport	The TCP/UDP port (or beginning of range) the packet is going to. Select one of the supported TCP/UDP port names (See “Supported TCP/UDP Port Names” on page 414 for a listing of TCP/UDP port names supported by the SpeedTouch™). Alternatively, specify the protocol number.	OPTIONAL
dstportend	The destination TCP/UDP port range end (inclusive). (Only applicable for ranges) Select one of the supported TCP/UDP port names (See “Supported TCP/UDP Port Names” on page 414 for a listing of TCP/UDP port names supported by the SpeedTouch™). Alternatively, specify the protocol number.	OPTIONAL
clink	The name of the chain to be parsed when this rule applies. (action is ignored).	OPTIONAL
log	Logging is done when this rule applies. Choose between: no or yes	OPTIONAL
label	None, link (when clink is used) or label name.	REQUIRED

RELATED COMMANDS:

label rule clear	Clear statistics for a given rule.
label rule delete	Delete a specified rule in a chain.
label rule flush	Delete all rules in a chain.
label rule list	Show a list of all (or a specified) chains' rules.
label rule stats	Show statistics for all (or a specified) chains' rules.

label rule delete

Delete a rule.

SYNTAX:

```
label rule delete    chain = <string>
                    index = <number>
```

chain	The name of the chain in which to delete the rule.	REQUIRED
index	The index number of the rule in the chain.	REQUIRED
Note	Use :label rule list first to determine the index number of the applicable rule.	

RELATED COMMANDS:

label rule clear	Clear statistics for a given rule.
label rule create	Create a rule.
label rule flush	Delete all rules in a chain.
label rule list	Show a list of all (or a specified) chains' rules.
label rule stats	Show statistics for all (or a specified) chains' rules.

label rule flush

Flush all rules created for a chain(s). The chain itself is not removed.

Note The flush command does not impact previously saved configurations.

SYNTAX:

```
label rule flush [chain = <string>]
```

chain

The name of the chain to empty.

OPTIONAL

Note If this parameter is not specified, all rules for all chains are deleted.

RELATED COMMANDS:

label rule clear	Clear statistics for a given rule.
label rule create	Create a rule.
label rule delete	Delete a specified rule in a chain.
label rule list	Show a list of all (or a specified) chains' rules.
label rule stats	Show statistics for all (or a specified) chains' rules.

label rule list

Show a list of rules.

SYNTAX:

```
label rule list [chain = <string>]
```

chain	The name of the chain to list the rules of.	OPTIONAL
	Note If this parameter is not specified, all rules for all chains are shown.	

EXAMPLE INPUT AND OUTPUT:

```
=>label rule list
label rule create chain=_auto_labels index=0 clink=user_labels label=link
=>
```

RELATED COMMANDS:

label rule clear	Clear statistics for a given rule.
label rule create	Create a rule.
label rule delete	Delete a specified rule in a chain.
label rule flush	Delete all rules in a chain.
label rule stats	Show statistics for all (or a specified) chains' rules.

label rule stats

Show statistics, i.e. the number of packets and bytes which have passed the hooks.

SYNTAX:

```
label rule stats    [chain = <string>]
                   [index = <number>]
```

chain	The name of the chain of which the statistics must be listed. In case this parameter is not specified the statistics for the rules applicable to all chains are shown.	OPTIONAL
index	The index number of the chain's rule of which the statistics must be listed. Execute label rule list first to determine the index number of the applicable rule. In case this parameter is not specified the statistics for all rules applicable to the specified chain are shown.	OPTIONAL

RELATED COMMANDS:

label rule clear	Clear statistics for a given rule.
label rule create	Create a rule.
label rule delete	Delete a specified rule in a chain.
label rule flush	Delete all rules in a chain.
label rule list	Show a list of all (or a specified) chains' rules.

Language Commands

Contents

This chapter covers the following commands:

Topic	Page
language config	256
language list	257
language remove	258

language config

Select a language.

SYNTAX:

```
language config [language = <string>]
```

language	Language code: OSI language code (2 chars) for language (en=english)	OPTIONAL
----------	--	----------

RELATED COMMANDS:

- [language list](#) List available languages archives.
- [language remove](#) Remove one or all language archives.

language list

List available languages archives.

SYNTAX:

```
language list
```

EXAMPLE:

```
=>language list
CODE LANGUAGE          VERSION FILENAME
en* english           4.2.0.1 <system>
```

RELATED COMMANDS:

language config	Select a language.
language remove	Remove one or all language archives.

language remove

Remove one or all language archives.

SYNTAX:

```
language remove [file = <string>]
                 [all <{yes|no}>]
```

file	Filename: name of language archive to remove	OPTIONAL
all	Removes all languages archives. Choose between: <ul style="list-style-type: none">• yes• no	OPTIONAL

RELATED COMMANDS:

language config	Select a language.
language remove	Remove one or all language archives.

NAT Commands

Contents

This chapter covers the following commands:

Topic	Page
nat applist	260
nat bind	261
nat bindlist	264
nat clear	265
nat config	266
nat create	267
nat defserver	269
nat delete	270
nat disable	272
nat enable	273
nat flush	274
nat list	275
nat multinatadd	276
nat multinatdelete	277
nat multinatlist	278
nat unbind	279

nat applist

List available Network Address Translation / Port Address Translation (NAT/PAT) protocol helpers.

Certain protocols are 'sensitive' to NAT/PAT in that they do not function properly when dealing with it. This list shows which 'NAT/PAT-sensitive' applications are supported on the SpeedTouch™, i.e. the inherent knowledge of the SpeedTouch™ on this matter.

SYNTAX:

```
nat applist
```

EXAMPLE OUTPUT:

```
=>nat applist
Application  Proto  DefaultPort
GRE          gre    1          INCOMING
PPTP        tcp    1723      OUTGOING  INCOMING
ESP         50     1          OUTGOING  INCOMING
IKE         udp    500       OUTGOING  INCOMING
SIP         udp    5060      OUTGOING  INCOMING
ILS         tcp    0          OUTGOING
H245        tcp    0          OUTGOING  INCOMING
H323        tcp    1720      OUTGOING  INCOMING
RAUDIO(PNA) tcp    7070      OUTGOING
RTSP        tcp    554       OUTGOING
IRC         tcp    6667      OUTGOING
FTP         tcp    21        OUTGOING
=>
```

RELATED COMMANDS:

- `nat bind` Create a new helper/port binding.
- `nat bindlist` List current NAT/PAT helper/port bindings.
- `nat unbind` Delete an existing helper/port binding.

nat bind

Create a new helper/port binding.

SYNTAX:

```
nat bind          application = <string>
                  port = <{<supported TCP/UDP port name>|<number>}>
                  [port_end = <{<supported TCP/UDP port name>|<number>}>]
```

application	The name of a NAT/PAT application helper. The name must be spelled exactly as listed in the application list (:nat applist).	REQUIRED
port	The TCP/UDP port this application handler should work on. Select one of the supported TCP/UDP port names (See “Supported TCP/UDP Port Names” on page 414 for a listing of TCP/UDP port names supported by the SpeedTouch™). Alternatively, specify the protocol number.	REQUIRED
port_end	The TCP/UDP end port of the range of ports this application handler should work on. Select one of the supported TCP/UDP port names (See “Supported TCP/UDP Port Names” on page 414 for a listing of TCP/UDP port names supported by the SpeedTouch™). Alternatively, specify the protocol number.	OPTIONAL

EXAMPLE:

```
=>nat bindlist
Application Proto Port
SIP         udp  5060
GRE         gre  1
PPTP        tcp  1723
ILS         tcp  1002
ILS         tcp  389
H323        tcp  1720
FTP         tcp  21
IRC         tcp  6667
RAUDIO(PNA) tcp  7070
=>nat bind application=RAUDIO(PNA) port=7071
=>nat bindlist
Application Proto Port
SIP         udp  5060
GRE         gre  1
PPTP        tcp  1723
ILS         tcp  1002
ILS         tcp  389
H323        tcp  1720
FTP         tcp  21
IRC         tcp  6667
RAUDIO(PNA) tcp  7070
RAUDIO(PNA) tcp  7071
=>
```

EXAMPLE:

```
=>nat bindlist
Application Proto Port
FTP         tcp  21
GRE         gre  1
H323        tcp  1720
ILS         tcp  389
ILS         tcp  1002
IRC         tcp  6667
PPTP        tcp  1723
RAUDIO(PNA) tcp  7070
RTSP        tcp  554
SIP         udp  5060
=>nat bind application=RAUDIO(PNA) port=7072 port_end=7075
=>nat bindlist
Application Proto Port
FTP         tcp  21
GRE         gre  1
H323        tcp  1720
ILS         tcp  389
ILS         tcp  1002
IRC         tcp  6667
PPTP        tcp  1723
RAUDIO(PNA) tcp  7070
RAUDIO(PNA) tcp  7072-7075
RTSP        tcp  554
SIP         udp  5060
=>
```

RELATED COMMANDS:

<code>nat applist</code>	List available NAT/PAT protocol helpers.
<code>nat bindlist</code>	List current NAT/PAT helper/port bindings.
<code>nat unbind</code>	Delete an existing helper/port binding

nat bindlist

List current NAT/PAT helper/port bindings.

SYNTAX:

```
nat bindlist
```

EXAMPLE:

```
=>nat bindlist
Application  Proto Port
SIP          udp  5060
GRE          gre   1
PPTP        tcp  1723
ILS          tcp  1002
ILS          tcp  389
H323        tcp  1720
FTP          tcp  21
RTSP        tcp  554
IRC          tcp  6667
RAUDIO(PNA) tcp  7070
=>
```

RELATED COMMANDS:

- | | |
|-----------------------------|--|
| nat applist | List available NAT/PAT protocol helpers. |
| nat bind | Create a new helper/port binding. |
| nat unbind | Delete an existing helper/port binding |

nat clear

Clear NAT/PAT connection database.

SYNTAX:

```
nat clear [addr = <ip-address>]
```

addr

An address-translation enabled IP address.
If no IP address is specified, all IP addresses are cleared.

OPTIONAL

nat config

Configure NAT parameters for IP session.

SYNTAX:

```
nat config      [timeout_ICMP = <string>]
                [timeout_UDP = <string>]
                [timeout_TCP = <string>]
                [timeout_TCP_nego = <string>]
                [timeout_other = <string>]
                [ike_port = <{fixed|floated}>]
```

timeout_ICMP	The interval of time for which an ICMP protocol session is allowed to remain valid without any activity. Syntax: <HH:MM> or <MM>.	OPTIONAL
timeout_UDP	The interval of time for which an UDP protocol session is allowed to remain valid without any activity. Syntax: <HH:MM> or <MM>.	OPTIONAL
timeout_TCP	The interval of time for which an TCP protocol session is allowed to remain valid without any activity. This time-out value applies to a TCP session during its data transfer phase. Syntax: <HH:MM> or <MM>.	OPTIONAL
timeout_TCP_nego	The interval of time for which an TCP protocol session is allowed to remain valid without any activity. This time-out value applies to a TCP session during its establishment and termination phases. Syntax:<HH:MM> or <MM>.	OPTIONAL
timeout_other	The interval of time for which an IP session for a protocol other than ICMP, UDP and TCP is allowed to remain valid without any activity. Syntax: <HH:MM> or <MM>.	OPTIONAL
ike_port	The IKE outside port number. This is the translation port number for IKEv1 (fixed to 500 or floated).	OPTIONAL

EXAMPLE:

```
=>nat config
Timeout ICMP idle      : 00h01m
Timeout UDP idle      : 00h05m
Timeout TCP idle      : 00h15m
Timeout TCP negotiation idle : 00h02m
Timeout other idle    : 00h01m
IKE outside port number : Floated
=>
```

nat create

Create a static NAT/PAT entry. Typically used to install specific servers behind the SpeedTouch™ NAT/PAT device.

SYNTAX:

```

nat create      protocol = <{<supported IP protocol name>|<number>}>
                inside_addr = <ip-address>
                [inside_port = <{<supported TCP/UDP port name>|<number>}>]
                outside_addr = <ip-address>
                [outside_port = <{<supported TCP/UDP port name>|<number>}>]
                [foreign_addr = <ip-address>]
                [foreign_port = <{<supported TCP/UDP port name>|<number>}>]

```

protocol	The IP protocol name (or number) of the incoming stream. Select one of the supported protocol names (See “ Supported Internet Protocol (IP) Protocol Names ” on page 413 for a listing of protocol names supported by the SpeedTouch™). Alternatively, specify the protocol number.	REQUIRED
inside_addr	The IP address of the local host (intended to receive the incoming traffic) behind the SpeedTouch™ 's NAT/PAT device. Typically, a private IP address.	REQUIRED
inside_port	The port of the application on the local host. Select one of the supported TCP/UDP port names (See “ Supported TCP/UDP Port Names ” on page 414 for a listing of TCP/UDP port names supported by the SpeedTouch™). Alternatively, specify the protocol number. Applicable for TCP and UDP protocols only. Other protocols do not need a port to be specified.	OPTIONAL
outside_addr	The apparent host IP address this application is running on, i.e. the NAT/PAT enabled WAN IP address of the SpeedTouch™. Use '0' to create a template. Such template will then be valid for any of SpeedTouch™ 's NAT/PAT enabled IP addresses, e.g. also dynamically assigned/negotiated IP addresses.	REQUIRED
outside_port	The apparent port number this application is running on. Select one of the supported TCP/UDP port names (See “ Supported TCP/UDP Port Names ” on page 414 for a listing of TCP/UDP port names supported by the SpeedTouch™). Alternatively, specify the protocol number. Applicable for TCP and UDP protocols only. Other protocols do not need a port to be specified.	OPTIONAL
foreign_addr	The IP address of the in-front-of-NAT/PAT routable address. Use '0' to match all foreign addresses.	OPTIONAL

foreign_port	The port of the routable host. Select one of the supported TCP/UDP port names (See “ Supported TCP/UDP Port Names” on page 414 for a listing of TCP/UDP port names supported by the SpeedTouch™). Alternatively, specify the protocol number. Do not use '0' in case a foreign IP address is specified. Applicable for TCP and UDP protocols only. Other protocols do not need a port to be specified.	OPTIONAL
--------------	--	----------

RELATED COMMANDS:

nat delete	Delete a static NAT/PAT entry.
nat disable	Disable NAT/PAT on a SpeedTouch™ IP address.
nat enable	Enable NAT/PAT on one of the SpeedTouch™ IP addresses.
nat list	List NAT/PAT connection database.

nat defserver

Define the default server behind the SpeedTouch™ NAT/PAT device that receives all (unknown) incoming packets.

In typical LAN configurations one local 'default' server will be responsible for all WAN-LAN mail, http, ftp, ... connectivity. This command allows to specify this server. For specific services, use :nat create.

SYNTAX:

```
nat defserver [addr = <ip-address>]
```

addr	The IP address of the server (on the 'inside') that will receive all (unknown) incoming packets. If not specified, the current default server is shown.	OPTIONAL
------	--	----------

EXAMPLE INPUT/OUTPUT:

```
=>nat defserver
Default server is undefined
=>nat defserver addr=10.0.0.1
=>nat defserver
Default server is 10.0.0.1
=>
```

nat delete

Delete a static NAT/PAT entry.

SYNTAX:

```
nat delete protocol = <{<supported IP protocol name>|<number>}>
inside_addr = <ip-address>
[inside_port = <{<supported TCP/UDP port name>|<number>}>]
outside_addr = <ip-address>
[outside_port = <{<supported TCP/UDP port name>|<number>}>]
[foreign_addr = <ip-address>]
[foreign_port = <{<supported TCP/UDP port name>|<number>}>]
```

protocol	The IP protocol name (or number) of the incoming stream. Select one of the supported protocol names (See “ Supported Internet Protocol (IP) Protocol Names ” on page 413 for a listing of protocol names supported by the SpeedTouch™). Alternatively, specify the protocol number.	REQUIRED
inside_addr	The IP address of the local host (intended to receive the incoming traffic) behind the SpeedTouch™ 's NAT/PAT device. Typically, a private IP address.	REQUIRED
inside_port	The port of the application on the local host. Select one of the supported TCP/UDP port names (See “ Supported TCP/UDP Port Names ” on page 414 for a listing of TCP/UDP port names supported by the SpeedTouch™). Alternatively, specify the protocol number. Applicable for TCP and UDP protocols only. Other protocols do not need a port to be specified.	OPTIONAL
outside_addr	The apparent host IP address this application is running on, i.e. the NAT/PAT enabled WAN IP address of the SpeedTouch™. Use '0' to delete an entry valid for any of SpeedTouch™ 's NAT/PAT enabled IP addresses, e.g. also dynamically assigned/negotiated IP addresses.	REQUIRED
outside_port	The apparent port number this application is running on. Select one of the supported TCP/UDP port names (See “ Supported TCP/UDP Port Names ” on page 414 for a listing of TCP/UDP port names supported by the SpeedTouch™). Alternatively, specify the protocol number. Applicable for TCP and UDP protocols only. Other protocols do not need a port to be specified.	OPTIONAL
foreign_addr	The IP address of the in-front-of-NAT/PAT routable address.	REQUIRED

foreign_port	The port of the routable host. Select one of the supported TCP/UDP port names (See “ Supported TCP/UDP Port Names” on page 414 for a listing of TCP/UDP port names supported by the SpeedTouch™). Alternatively, specify the protocol number. Applicable for TCP and UDP protocols only. Other protocols do not need a port to be specified.	OPTIONAL
--------------	---	----------

RELATED COMMANDS:

nat create	Create a static NAT/PAT entry.
nat disable	Disable NAT/PAT on a SpeedTouch™ IP address.
nat enable	Enable NAT/PAT on one of the SpeedTouch™ IP addresses.
nat list	List NAT/PAT connection database.

nat disable

Disable NAT or PAT on the specified SpeedTouch™ IP address.

SYNTAX:

```
nat disable      addr = <ip-address>
```

addr The SpeedTouch™ IP address for which NAT/PAT must be disabled. **REQUIRED**

EXAMPLE:

```
=>nat list
Indx Prot Inside-addr:Port Outside-addr:Port Foreign-addr:Port Flgs Expir State Control
1    6   10.0.0.138:80      172.16.0.5:1080   0.0.0.0:0         19  8    9
2    17  10.0.0.138:138    10.0.0.140:138   10.0.0.20:138    11  20   10
3    17  10.0.0.138:137    10.0.0.140:137   10.0.0.254:137   11  20   10
4    17  10.0.0.138:7938   10.0.0.140:7938  10.0.0.96:4756   11  20   10
5    17  10.0.0.138:513    10.0.0.140:513   10.0.0.109:513   11  20   10
6    17  10.0.0.138:111    10.0.0.140:111   10.0.0.96:4756   11  20   10
=>nat disable addr 172.16.0.5
=>nat list
Indx Prot Inside-addr:Port Outside-addr:Port Foreign-addr:Port Flgs Expir State Control
1    17  10.0.0.138:138    10.0.0.140:138   10.0.0.20:138    11  20   10
2    17  10.0.0.138:137    10.0.0.140:137   10.0.0.254:137   11  20   10
3    17  10.0.0.138:7938   10.0.0.140:7938  10.0.0.96:4756   11  20   10
4    17  10.0.0.138:513    10.0.0.140:513   10.0.0.109:513   11  20   10
5    17  10.0.0.138:111    10.0.0.140:111   10.0.0.96:4756   11  20   10
=>
```

RELATED COMMANDS:

- [nat create](#) Create a static NAT/PAT entry.
- [nat delete](#) Delete a static NAT/PAT entry.
- [nat enable](#) Enable NAT/PAT on one of the SpeedTouch™ IP addresses.
- [nat list](#) List NAT/PAT connection database.

nat enable

Enable NAT/PAT on a SpeedTouch™ IP address.

SYNTAX:

```
nat enable      addr = <ip-address>
                [type = <{none|pat}>]
```

addr	The IP address to use for outgoing address translation. Must be one of the own IP addresses of the SpeedTouch™ (see 'ip aplist').	REQUIRED
type	Enable port translation (pat) or not (none).	OPTIONAL

EXAMPLE:

```
=>ip aplist
1 eth0 Type:Ethernet HWaddr 00:80:9f:24:ab:cf BRHWaddr ff:ff:ff:ff:ff:ff
  inet  addr:10.10.10.147 Bcast:10.10.10.255 Mask:255.0.0.0
  UP RUNNING MTU:1500 ReasmMAX:65535 Group:2
  IPRX bytes:19791886 unicastpkts:11341 brcastpkts:290555
  IPTX bytes:839550 unicastpkts:11477 brcastpkts:0 droppkts:0
  HWRX bytes:0 unicastpkts:0 brcastpkts:0
  HWTX bytes:0 unicastpkts:0 brcastpkts:0 droppkts:0
0 loop Type:0
  inet  addr:127.0.0.1 Bcast:127.255.255.255 Mask:255.0.0.0
  UP RUNNING MTU:1500 ReasmMAX:65535 Group:1
  IPRX bytes:116 unicastpkts:0 brcastpkts:2
  IPTX bytes:0 unicastpkts:0 brcastpkts:0 droppkts:0
  HWRX bytes:0 unicastpkts:0 brcastpkts:0
  HWTX bytes:0 unicastpkts:0 brcastpkts:0 droppkts:0
=>nat enable addr=10.10.10.147 type=pat
=>ip aplist
1 eth0 Type:Ethernet HWaddr 00:80:9f:24:ab:cf BRHWaddr ff:ff:ff:ff:ff:ff
  inet  addr:10.10.10.147 Bcast:10.10.10.255 Mask:255.0.0.0
  UP RUNNING pat MTU:1500 ReasmMAX:65535 Group:2
  IPRX bytes:19791886 unicastpkts:11341 brcastpkts:290555
  IPTX bytes:839550 unicastpkts:11477 brcastpkts:0 droppkts:0
  HWRX bytes:0 unicastpkts:0 brcastpkts:0
  HWTX bytes:0 unicastpkts:0 brcastpkts:0 droppkts:0
0 loop Type:0
  inet  addr:127.0.0.1 Bcast:127.255.255.255 Mask:255.0.0.0
  UP RUNNING MTU:1500 ReasmMAX:65535 Group:1
  IPRX bytes:116 unicastpkts:0 brcastpkts:2
  IPTX bytes:0 unicastpkts:0 brcastpkts:0 droppkts:0
  HWRX bytes:0 unicastpkts:0 brcastpkts:0
  HWTX bytes:0 unicastpkts:0 brcastpkts:0 droppkts:0
=>
```

RELATED COMMANDS:

nat create	Create a static NAT/PAT entry.
nat delete	Delete a static NAT/PAT entry.
nat disable	Disable NAT/PAT on a SpeedTouch™ IP address.
nat list	List NAT/PAT connection database.

nat flush

Flush complete NAT/PAT configuration.

Note The flush command does not impact previously saved configurations.

SYNTAX:

```
nat flush
```

RELATED COMMANDS:

nat create	Create a static NAT/PAT entry.
nat delete	Delete a static NAT/PAT entry.
nat disable	Disable NAT/PAT on a SpeedTouch™ IP address.
nat enable	Enable NAT/PAT on one of the SpeedTouch™ IP addresses.

nat list

Lists NAT/PAT connection database.

SYNTAX:

```
nat list [addr = <ip-address>]
```

addr An address-translation enabled IP address. **OPTIONAL**
If no address is specified, all IP addresses are shown.

EXAMPLE:

```
=>nat list
Indx Prot Inside-addr:Port Outside-addr:Port Foreign-addr:Port Flgs Expir State Control
1 6 10.0.0.138:80 172.16.0.5:1080 0.0.0.0:0 19 8 9
2 17 10.0.0.138:138 10.0.0.140:138 10.0.0.20:138 11 20 10
3 17 10.0.0.138:137 10.0.0.140:137 10.0.0.254:137 11 20 10
4 17 10.0.0.138:7938 10.0.0.140:7938 10.0.0.96:4756 11 20 10
5 17 10.0.0.138:513 10.0.0.140:513 10.0.0.109:513 11 20 10
6 17 10.0.0.138:111 10.0.0.140:111 10.0.0.96:4756 11 20 10
=>
```

RELATED COMMANDS:

nat create Create a static NAT/PAT entry.

nat delete Delete a static NAT/PAT entry.

nat disable Disable NAT/PAT on a SpeedTouch™ IP address.

nat enable Enable NAT/PAT on one of the SpeedTouch™ IP addresses.

nat multinatadd

Add a MultiNAT configuration.

SYNTAX:

```
nat multinatadd  intf = <string>
                  inside_addr = <ip-range>
                  outside_addr = <ip-range>
```

intf	The IP interface name where MultiNAT has to be applied.	REQUIRED
inside_addr	The IP address of the host behind the address translation device that will receive the incoming traffic. Typically, a private IP address.	REQUIRED
outside_addr	The apparent host IP address this application will running on. Use '0' to create a template.	REQUIRED

RELATED COMMANDS:

nat multinatdelete	Delete a MultiNAT configuration.
nat multinatlist	List MultiNAT configurations.

nat multinatdelete

Delete a MultiNAT configuration.

SYNTAX:

```
nat multinatdelete  index = <number>
```

index

The MultiNAT index as listed by maplist.

REQUIRED

RELATED COMMANDS:

[nat multinatadd](#)

Add a MultiNAT configuration.

[nat multinatlist](#)

List MultiNAT configurations.

nat multinatlist

List MultiNAT configurations.

SYNTAX:

```
nat multinatlist [intf = <string>]
```

addr	The IP interface name for which the MultiNAT configuration must be listed. If no name is specified, all MultiNAT configurations are listed.	OPTIONAL
------	---	----------

RELATED COMMANDS:

nat multinatadd	Add a MultiNAT configuration.
nat multinatdelete	Delete a MultiNAT configuration.

nat unbind

Delete an existing helper/port binding.

SYNTAX:

```
nat unbind          application = <string>
                   port = <{<supported TCP/UDP port name>|<number>}>
                   [port_end = <{<supported TCP/UDP port name>|<number>}>]
```

application	The name of a NAT/PAT application helper. The name must be spelled exactly as listed in the application list (:nat applist).	REQUIRED
port	The TCP/UDP port this application handler is working on. Select one of the supported TCP/UDP port names (See “Supported TCP/UDP Port Names” on page 414 for a listing of TCP/UDP port names supported by the SpeedTouch™). Alternatively, specify the protocol number.	REQUIRED
port_end	The TCP/UDP end port of the range of ports this application handler should work on. Select one of the supported TCP/UDP port names (See “Supported TCP/UDP Port Names” on page 414 for a listing of TCP/UDP port names supported by the SpeedTouch™). Alternatively, specify the protocol number.	OPTIONAL

EXAMPLE:

```
=>nat bindlist
Application  Proto  Port
FTP          tcp    21
GRE          gre    1
H323        tcp    1720
ILS          tcp    389
ILS          tcp    1002
IRC          tcp    6667
PPTP        tcp    1723
RAUDIO(PNA) tcp    7070
RAUDIO(PNA) tcp    7072-7075
RTSP        tcp    554
SIP         udp    5060
=>nat unbind application=RAUDIO(PNA) port=7072 port_end=7075
=>nat bindlist
Application  Proto  Port
FTP          tcp    21
GRE          gre    1
H323        tcp    1720
ILS          tcp    389
ILS          tcp    1002
IRC          tcp    6667
PPTP        tcp    1723
RAUDIO(PNA) tcp    7070
RTSP        tcp    554
SIP         udp    5060
=>
```

RELATED COMMANDS:

- | | |
|---------------------------|--|
| <code>nat applist</code> | List available NAT/PAT protocol helpers. |
| <code>nat bind</code> | Create a new helper/port binding. |
| <code>nat bindlist</code> | List current NAT/PAT helper/port bindings. |

Phonebook Commands

Contents

This chapter covers the following commands:

Topic	Page
phonebook add	282
phonebook autolist	284
phonebook delete	285
phonebook flush	286
phonebook list	287

phonebook add

Add a phonebook entry.

SYNTAX:

```
phonebook add    name = <string>
                 addr = <atmchannel : PVC syntax is [port.]vpi.vci
                       port=ds10|ds11|...>
                 type = <{any|ethoa|pppoa|ipoa}>
```

name	<p>A free to choose phonebook name for the destination.</p> <p>Two limitations apply:</p> <ul style="list-style-type: none"> • The name of a phonebook entry intended for the Relayed PPPoA (PPPoA-to-PPTP Relaying) packet service may not start with capital P or capital T • The name of a phonebook entry intended for the PPP-to-DHCP spoofing packet service must start with DHCP, e.g. 'DHCP_Spoof01'. 	REQUIRED
addr	<p>The ATM address for this destination.</p> <p>It is composed of a Virtual Path Identifier (VPI) and a Virtual Channel Identifier (VCI) identifying ATM virtual channels.</p> <p>In most cases the values are provided by the Service Provider.</p> <p>Accepted VPI: a number between 0 and 15</p> <p>Accepted VCI: a number between 0 and 511.</p>	REQUIRED
type	<p>The Connection Service supported by the destination.</p> <p>Choose between:</p> <ul style="list-style-type: none"> • any (All Packet Services) • ethoa (Bridged Ethernet, Routed Ethernet, Bridged PPPoE, Routed PPPoE) • pppoa (Routed PPPoA and Relayed PPPoA) • ipoa (Classical IPoA and Routed IPoA). 	REQUIRED

EXAMPLE:

```
=>phonebook list
Name      Type    Use    Address
usb_port  any     1      usb.0.35
PVC1      any     1      8.35
PVC2      ethoa   0      8.36
Br4       ethoa   0      8.38
CIPPVC3   ipoa    1      8.82
=>phonebook add name=PVC_Test addr=8.68 type=pppoa
=>phonebook list
Name      Type    Use    Address
usb_port  any     1      usb.0.35
PVC1      any     1      8.35
PVC2      ethoa   0      8.36
Br4       ethoa   0      8.38
CIPPVC3   ipoa    1      8.82
PVC_Test  pppoa   0      8.68
=>
```

Note The usb_port phonebook entry is only applicable in case of a SpeedTouch™530 variant. Do not change this phonebook entry in any way, or delete it from the phonebook!

RELATED COMMANDS:

`phonebook delete` Remove a phonebook entry.

`phonebook list` Show current phonebook.

phonebook autolist

Show auto PVCs, if supported by the Central Office DSLAM.

SYNTAX:

```
phonebook autolist
```

EXAMPLE INPUT/OUTPUT:

```
=>phonebook autolist  
8.35  
=>
```

RELATED COMMANDS:

[phonebook list](#)

Show current phonebook.

phonebook delete

Remove an unused phonebook entry.

SYNTAX:

```
phonebook delete name = <string>
```

name	The name of the phonebook entry to delete. Only applicable for phonebook entries that are not used, i.e. not configured for any packet service. Execute phonebook list to check whether the entry is used (Use=1) or not (Use=0).	REQUIRED
------	---	-----------------

EXAMPLE:

```
=>phonebook list
Name      Type   Use   Address
usb_port  any    1     usb.0.35
PVC1      any    1     8.35
PVC2      ethoa  0     8.36
Br4       ethoa  0     8.38
CIPPVC3   ipoa   1     8.82
PVC_Test  pppoa  0     8.68
=>phonebook delete name=PVC_Test
=>phonebook list
Name      Type   Use   Address
usb_port  any    1     usb.0.35
PVC1      any    1     8.35
PVC2      ethoa  0     8.36
Br4       ethoa  0     8.38
CIPPVC3   ipoa   1     8.82
=>
```

Note In case of a SpeedTouch™530 variant, never delete the usb_port phonebook entry!

RELATED COMMANDS:

phonebook add	Add a phonebook entry.
phonebook list	Show current phonebook.

phonebook flush

Flush complete phonebook.

Note The flush command does not impact previously saved configurations.

SYNTAX:

```
phonebook flush
```

EXAMPLE:

```
=>phonebook list
Name      Type    Use    Address
usb_port  any     1      usb.0.35
PVC1      any     1      8.35
PVC2      ethoa   0      8.36
Br4       ethoa   0      8.38
CIPPVC3   ipoa    1      8.82
PVC_Test  pppoa   0      8.68
=>phonebook flush
=>phonebook list
Name      Type    Use    Address
=>
```

Note In case of a SpeedTouch™530 variant, flushing the Phonebook may cause the device to become inaccessible from its USB interface.

phonebook list

Show current phonebook.

SYNTAX:

```
phonebook list
```

EXAMPLE INPUT/OUTPUT:

```
=>phonebook list
Name      Type    Use    Address
usb_port  any     1      usb.0.35
PVC1      any     1      8.35
PVC2      ethoa   0      8.36
Br4       ethoa   0      8.38
CIPPVC3   ipoa    1      8.82
PVC_Test  pppoa   0      8.68
=>
```

Note The usb_port phonebook entry is only applicable in case of a SpeedTouch™530 variant. Do not change this phonebook entry in any way, or delete it from the phonebook!

RELATED COMMANDS:

phonebook add	Add a phonebook entry.
phonebook autolist	Show auto PVCs.
phonebook delete	Remove a phonebook entry.

PPPoA Commands

Contents

This chapter covers the following commands:

Topic	Page
pppoa flush	290
pppoa ifadd	291
pppoa ifattach	293
pppoa ifconfig	294
pppoa ifdelete	298
pppoa ifdetach	299
pppoa iflist	300
pppoa rtadd	301
pppoa rtdelete	303

pppoa flush

Flushes the current PPPoA configuration.

Note The flush command does not impact previously saved configurations.

SYNTAX:

```
pppoa flush
```

EXAMPLE:

```
=>pppoa iflist
pppoa1: dest : pppoa1
  Retry: 10 QoS default encaps VC-MUX
  mode = IP Routing
  flags = echo magic accomp mru addr route savepwd pppoa0A
  trans addr = pat mru = 1500
  route = 0.0.0.0/0 - 0.0.0.0/0 (metric 0)
  user name = guest password = *****
  admin state = down oper state = down link state = not-connected
  LCP: state = initial retransm = 10 term. reason =
  IPCP: state = initial retransm = 0 term. reason =
=>pppoa flush
=>pppoa iflist
=>
```

pppoa ifadd

Create a new PPPoA interface.

SYNTAX:

```
pppoa ifadd      [intf = <string>]
                 [dest = <phonebook entry>]
```

intf	The name for the new PPPoA interface. If not specified, the destination will double as interface name.	OPTIONAL
dest	The ATM channel to be used for this PPPoA interface. Typically, an phonebook entry.	OPTIONAL

EXAMPLE:

```
=>pppoa iflist
pppoa_pppoa: dest : pppoa      [00:00:00]
  Retry : 10  QoS default  encaps VC-MUX
  mode = IP routing
  flags = echo magic accomp restart mru addr route savepwd
  dns metric = 0  trans addr = pat  mru = 1500
  route : dst=0.0.0.0/0 - src=10.0.0.0/1 (metric 1)
  auth = auto  user = johndoe@ISP  password = *****
  admin state = up  oper state = down  link state = retrying
  LCP : state = starting  retransm = 1  term. reason =
  IPCP: state = initial  retransm = 0  term. reason =

=>pppoa ifadd intf=pppoa2 dest=pppoa2
=>pppoa iflist
pppoa_pppoa: dest : pppoa      [00:00:00]
  Retry : 10  QoS default  encaps VC-MUX
  mode = IP routing
  flags = echo magic accomp restart mru addr route savepwd
  dns metric = 0  trans addr = pat  mru = 1500
  route : dst=0.0.0.0/0 - src=10.0.0.0/1 (metric 1)
  auth = auto  user = johndoe@ISP  password = *****
  admin state = up  oper state = down  link state = retrying
  LCP : state = starting  retransm = 1  term. reason =
  IPCP: state = initial  retransm = 0  term. reason =

pppoa2: dest : pppoa2      [00:00:00]
  Retry : 10  QoS default  encaps VC-MUX
  mode = IP routing
  flags = echo magic accomp restart mru addr savepwd
  dns metric = 0  mru = 1500
  auth = auto  user =      password =
  admin state = down  oper state = down  link state = not-connected
  LCP : state = initial  retransm = 10  term. reason =
  IPCP: state = initial  retransm = 0  term. reason =

=>
```

RELATED COMMANDS:

<code>pppoa ifattach</code>	Attach a PPPoA interface.
<code>pppoa ifconfig</code>	Configure a PPPoA interface.
<code>pppoa ifdelete</code>	Delete a PPPoA interface.
<code>pppoa ifdetach</code>	Detach a PPPoA interface.
<code>pppoa iflist</code>	Show current PPPoA configuration.

pppoa ifattach

Attach (i.e. connect) a PPPoA interface.

SYNTAX:

```
pppoa ifattach intf = <intfname>
```

intf	The name of the PPPoA interface to attach.	REQUIRED
------	--	----------

EXAMPLE:

```
=>pppoa iflist
pppoa1: dest : pppoa1 [00:00:00]
  Retry : 10 QoS default encaps VC-MUX
  mode = IP routing
  flags = echo magic accomp restart mru addr savepwd
  dns metric = 0 mru = 1500
  auth = auto user = password =
  admin state = down oper state = down link state = not-connected
  LCP : state = initial retransm = 10 term. reason =
  IPCP: state = initial retransm = 0 term. reason =

=>pppoa ifattach intf=pppoa1
=>pppoa iflist
pppoa1: dest : pppoa1 [00:00:00]
  Retry : 10 QoS default encaps VC-MUX
  mode = IP routing
  flags = echo magic accomp restart mru addr savepwd
  dns metric = 0 mru = 1500
  auth = auto user = password =
  admin state = up oper state = up link state = connected
  LCP : state = starting retransm = 1 term. reason =
  IPCP: state = initial retransm = 0 term. reason =

=>
```

RELATED COMMANDS:

pppoa ifadd	Create a new PPPoA interface.
pppoa ifconfig	Configure a PPPoA interface.
pppoa ifdelete	Delete a PPPoA interface.
pppoa ifdetach	Detach a PPPoA interface.
pppoa iflist	Show current PPPoA configuration.

pppoa ifconfig

Configure a PPPoA interface.

Note The interface to be configured may not be connected at the time of configuration. Execute PPPoA ifdetach prior to executing the PPPoA ifconfig command.

SYNTAX:

```
pppoa ifconfig  intf = <intfname>
                 [dest = <phonebook entry>]
                 [user = <string>]
                 [password = <password>]
                 [qos = <{default}>]
                 [encaps = <{vcmux|llc}>]
                 [pcomp = <{off|on}>]
                 [accomp = <{on|off|negotiate}>]
                 [trace = <{off|on}>]
                 [auth = <{pap|chap|auto}>]
                 [restart = <{off|on}>]
                 [retryinterval = <number{0-65535}>]
                 [passive = <{off|on}>]
                 [silent = <{off|on}>]
                 [echo = <{off|on}>]
                 [mru = <number{293-8192}>]
                 [laddr = <ip-address>]
                 [raddr = <ip-address>]
                 [netmask = <ip-mask(dotted or cidr)>]
                 [format = <{cidr|dotted|none}>]
                 [pool = <{none}>]
                 [savepwd = <{off|on}>]
                 [demanddial = <{off|on}>]
                 [primdns = <ip-address>]
                 [secdns = <ip-address>]
                 [dnsmetric = <number{0-100}>]
                 [idle = <number{0-1000000}>]
                 [idletrigger = <{RxTx|Rx|Tx}>]
                 [addrtrans = <{none|pat}>]
                 [unnumbered = <{off|on}>]
```

intf	The name of the PPPoA interface to configure.	REQUIRED
dest	The destination for this PPPoA interface. Typically, a phonebook entry.	OPTIONAL
user	The user name for remote PAP/CHAP authentication.	OPTIONAL
password	The password for remote PAP/CHAP authentication.	OPTIONAL
qos	The name of a qosbook entry defining the QoS parameters for the WAN link.	OPTIONAL

encaps	The WAN protocol encapsulation to be used on this interface. Choose between: <ul style="list-style-type: none"> • vcmux • llc 	OPTIONAL
pcomp	Try (on) or do not try (off) to negotiate PPPoA protocol compression (LCP PCOMP). By default the negotiation is disabled (off).	OPTIONAL
accomp	Try (on), do never try (off) or negotiate (negotiate) to negotiate PPPoA address & control field compression (LCP ACCOMP). In most cases, LCP ACCOMP should not be disabled nor negotiated, i.e. the address field FF-03 should not be sent over ATM. Therefore by default this parameter is enabled (on). If the accomp parameter is set to 'negotiate', the local side of the PPPoA connection demands to do ACCOMP and adapts itself to the result of this negotiation.	OPTIONAL
trace	Enable (on) or disable (off) verbose console logging. By default tracing is disabled (off).	OPTIONAL
auth	Select the authentication protocol. Choose between: <ul style="list-style-type: none"> • pap: Password Authentication Protocol (PAP) authentication will be forced. • chap: Challenge Handshake Authentication Protocol (CHAP) authentication will be forced. • auto: CHAP authentication will be used. If CHAP authentication is not successful, PAP authentication will be used instead. Auto is the default setting. 	OPTIONAL
restart	Automatically restart the connection when Link Control Protocol (LCP) link goes down (on) or do not restart automatically (off). By default restart is disabled (off).	OPTIONAL
retryinterval	A number between 0 and 65535 (seconds). Represents the intermediate interval between two retries to establish the connection on PPP level (LCP) (default=10).	OPTIONAL
passive	Put the link in listening state in case LCP times out (on) or not (off). This parameter allows to determine whether the link should be left open to wait for incoming messages from the remote side after 10 unsuccessful tries to establish the connection or not. By default, the listening state is disabled.	OPTIONAL
silent	Do not send anything at startup and just listen for incoming LCP messages (on) or retry up to 10 times to establish the connection (off). By default, the silent state is disabled.	OPTIONAL
echo	Send LCP echo requests at regular intervals (on) or not (off). By default, the sending of LCP echo requests is enabled.	OPTIONAL
mru	A number between 293 and 8192. Represents the maximum packet size the SpeedTouch™ should negotiate to be able to receive (default=1492).	OPTIONAL

laddr	<p>The local IP address of the peer-to-peer connection. Specifying a local IP address forces the remote side of the PPPoA link (if it allows to) to accept this IP address as the SpeedTouch™ PPPoA session IP address. If not specified, the SpeedTouch™ will accept any IP address. Typically the local IP address parameter is not specified.</p>	OPTIONAL
raddr	<p>The remote IP address of the peer-to-peer connection. Specifying a remote IP address forces the remote side of the PPPoA link (if it allows to) to accept this IP address as its PPPoA session IP address. If not specified, the SpeedTouch™ will accept any IP address. Typically the remote IP address parameter is not specified.</p>	OPTIONAL
netmask	<p>The subnetmask associated with this address. Specifying a subnetmask forces the remote side (if it allows to) to accept this subnetmask as the PPPoA session subnetmask. If not specified, the SpeedTouch™ will accept any subnetmask. The SpeedTouch™ will only request/accept a subnetmask if a DHCP server pool is associated, i.e. if the [pool] parameter is specified.</p>	OPTIONAL
format	<p>The negotiated subnetmask specified in the netmask parameter is specified in the dotted format (dotted) or in Classes Inter Domain Routing (CIDR) format (cidr). By default, the format is CIDR.</p>	OPTIONAL
pool	<p>The name of the free DHCP server pool to which the acquired IP subnet must be assigned to.</p>	OPTIONAL
savepwd	<p>Save password (on), if supplied, or do not save the password (off). By default, the saving of the password is disabled.</p>	OPTIONAL
demanddial	<p>Enable (on) or disable (off) the dial-on-demand feature.</p>	OPTIONAL
primdns	<p>The IP address of the primary DNS server. In case a primary DNS server is specified the SpeedTouch™ will negotiate this IP address with the remote side. If not specified, the SpeedTouch™ will accept any IP address.</p>	OPTIONAL
secdns	<p>The IP address of the (optional) secondary DNS server. In case a secondary DNS server is specified the SpeedTouch™ will negotiate this IP address with the remote side. If not specified, the SpeedTouch™ will accept any IP address.</p>	OPTIONAL
dnsmetric	<p>A number between 1 and 100. Represents the DNS route metric to be used for the negotiated DNS servers (default=1).</p>	OPTIONAL
idle	<p>A number between 0 and 1000000 (seconds). Represents after how many seconds an idle link goes down (default is 0).</p>	OPTIONAL

idletrigger	Consider the link being idle if no traffic is sent and/or received during the idle time. Choose between: <ul style="list-style-type: none"> • RxTx The idle time period restarts when a packet is transmitted or received (default value). • Rx: The idle time period restarts when a packet is received. Transmitted packets are ignored. • Tx The idle time period restarts when a packet is transmitted. Received packets are ignored. 	OPTIONAL
addrtrans	Automatically enable address translation for the IP address of this link (pat) or do not use address translation (none) (default is none).	OPTIONAL
unnumbered	Takes the local IP address from 'laddr' field and remote IP address from the IP address pool assigned to the incoming PPPoE link. In case the unnumbered parameter is disabled, the same IP address is used for each connection on the server side, thus reducing the number of used IP addresses.	OPTIONAL

EXAMPLE:

```

=>pppoa iflist
pppoa1: dest : pppoa1
  Retry: 10   QoS   default   encaps   LLC
  mode = IP Routing
  flags = echo magic accomp restart mru addr route savepwd pppoa0A
  trans addr = pat   mru = 1492
  route = 0.0.0.0/0 - 0.0.0.0/0 (metric 0)
  user name = My_Connection@MY_ISP   password = *****
  admin state = down   oper state = down   link state = not-connected
  LCP: state = initial   retransm = 10   term. reason =
  IPCP: state = initial   retransm = 0   term. reason =
=>pppoa ifconfig intf=pppoa1 prot=pppoa encaps=vcmux
=>pppoa iflist
pppoa1: dest : pppoa1
  Retry: 10   QoS   default   encaps   VC-MUX
  mode = IP Routing
  flags = echo magic accomp restart mru addr route savepwd pppoa0A
  trans addr = pat   mru = 1492
  route = 0.0.0.0/0 - 0.0.0.0/0 (metric 0)
  user name = My_Connection@MY_ISP   password = *****
  admin state = down   oper state = down   link state = not-connected
  LCP: state = initial   retransm = 10   term. reason =
  IPCP: state = initial   retransm = 0   term. reason =
=>

```

RELATED COMMANDS:

pppoa ifadd	Create a new PPPoA interface.
pppoa ifattach	Attach a PPPoA interface.
pppoa ifdelete	Delete a PPPoA interface.
pppoa ifdetach	Detach a PPPoA interface.
pppoa iflist	Show current PPPoA configuration.

pppoa ifdelete

Delete a PPPoA interface.

SYNTAX:

```
pppoa ifdelete   intf = <intfname>
```

intf	The name of the PPPoA interface to delete.	REQUIRED
-------------	--	-----------------

EXAMPLE:

```
=>pppoa iflist
pppoa_pppoa: dest : pppoa   [00:00:00]
  Retry : 10  QoS default  encaps VC-MUX
  mode = IP routing
  flags = echo magic accomp restart mru addr route savepwd
  dns metric = 0  trans addr = pat  mru = 1500
  route : dst=0.0.0.0/0 - src=10.0.0.0/1 (metric 1)
  auth = auto  user = johndoe@ISP  password = *****
  admin state = up  oper state = down  link state = retrying
  LCP : state = starting  retransm = 1  term. reason =
  IPCP: state = initial  retransm = 0  term. reason =
pppoa2: dest : pppoa2   [00:00:00]
  Retry : 10  QoS default  encaps VC-MUX
  mode = IP routing
  flags = echo magic accomp restart mru addr savepwd
  dns metric = 0  mru = 1500
  auth = auto  user =  password =
  admin state = down  oper state = down  link state = not-connected
  LCP : state = initial  retransm = 10  term. reason =
  IPCP: state = initial  retransm = 0  term. reason =
=>pppoa ifdelete intf=pppoa2
=>pppoa iflist
pppoa_pppoa: dest : pppoa   [00:00:00]
  Retry : 10  QoS default  encaps VC-MUX
  mode = IP routing
  flags = echo magic accomp restart mru addr route savepwd
  dns metric = 0  trans addr = pat  mru = 1500
  route : dst=0.0.0.0/0 - src=10.0.0.0/1 (metric 1)
  auth = auto  user = johndoe@ISP  password = *****
  admin state = up  oper state = down  link state = retrying
  LCP : state = starting  retransm = 1  term. reason =
  IPCP: state = initial  retransm = 0  term. reason =
=>
```

RELATED COMMANDS:

pppoa ifadd	Create a new PPPoA interface.
pppoa ifattach	Attach a PPPoA interface.
pppoa ifconfig	Configure a PPPoA interface.
pppoa ifdetach	Detach a PPPoA interface.
pppoa iflist	Show current PPPoA configuration.

pppoa ifdetach

Detach a PPPoA interface.

SYNTAX:

```
pppoa ifdetach intf = <intfname>
```

intf	The name of the PPPoA interface.	REQUIRED
------	----------------------------------	----------

EXAMPLE:

```
=>pppoa iflist
pppoa1: dest : pppoa1    [00:00:00]
  Retry : 10 QoS default  encaps VC-MUX
  mode = IP routing
  flags = echo magic accomp restart mru addr savepwd
  dns metric = 0 mru = 1500
  auth = auto user = password =
  admin state = up oper state = up link state = connected
  LCP : state = starting retransm = 1 term. reason =
  IPCP: state = initial retransm = 0 term. reason =

=>pppoa ifdetach =intf=pppoa1
=>pppoa iflist
pppoa1: dest : pppoa1    [00:00:00]
  Retry : 10 QoS default  encaps VC-MUX
  mode = IP routing
  flags = echo magic accomp restart mru addr savepwd
  dns metric = 0 mru = 1500
  auth = auto user = password =
  admin state = down oper state = down link state = not-connected
  LCP : state = starting retransm = 1 term. reason =
  IPCP: state = initial retransm = 0 term. reason =

=>
```

RELATED COMMANDS:

pppoa ifadd	Create a new PPPoA interface.
pppoa ifattach	Attach a PPPoA interface.
pppoa ifconfig	Configure a PPPoA interface.
pppoa ifdelete	Delete a PPPoA interface.
pppoa iflist	Show current PPPoA configuration.

pppoa iflist

Show current configuration of all or a specified PPPoA interface(s).

SYNTAX:

```
pppoa iflist [intf = <intfname>]
```

intf	The name of the PPPoA interface. If this parameter is not specified, all PPPoA interfaces are shown.	OPTIONAL
------	---	----------

EXAMPLE INPUT/OUTPUT:

```
=>pppoa iflist
pppoa_pppoa: dest : pppoa [00:00:00]
  Retry : 10 QoS default encaps VC-MUX
  mode = IP routing
  flags = echo magic accomp restart mru addr route savepwd
  dns metric = 0 trans addr = pat mru = 1500
  route : dst=0.0.0.0/0 - src=10.0.0.0/1 (metric 1)
  auth = auto user = johndoe@ISP password = *****
  admin state = up oper state = down link state = retrying
  LCP : state = starting retransm = 1 term. reason =
  IPCP: state = initial retransm = 0 term. reason =

pppoa2: dest : pppoa2 [00:00:00]
  Retry : 10 QoS default encaps VC-MUX
  mode = IP routing
  flags = echo magic accomp restart mru addr savepwd
  dns metric = 0 mru = 1500
  auth = auto user = password =
  admin state = down oper state = down link state = not-connected
  LCP : state = initial retransm = 10 term. reason =
  IPCP: state = initial retransm = 0 term. reason =

=>
```

RELATED COMMANDS:

pppoa ifadd	Create a new PPPoA interface.
pppoa ifattach	Attach a PPPoA interface.
pppoa ifconfig	Configure a PPPoA interface.
pppoa ifdelete	Delete a PPPoA interface.
pppoa ifdetach	Detach a PPPoA interface.

pppoa rtadd

Automatically add a route configuration to the routing table in case the specified PPPoA interface link comes up. This route configuration will determine which local hosts are allowed to use this link and/or which remote destinations should be or should not be reachable.

Note Execute the PPPoA ifdetach command for this interface prior to configuring routes.

SYNTAX:

```
PPPoA rtadd      intf = <intfname>
                  dst = <ip-address>
                  [dstmsk = <ip-mask(dotted or cidr)>]
                  [label = <labelname>]
                  [src = <ip-address>]
                  [srcmsk = <ip-mask(dotted or cidr)>]
                  [metric = <number{0-100}>]
```

intf	The name of the PPPoA interface.	REQUIRED
dst	The destination IP address for the route to be added when the link comes up.	REQUIRED
dstmsk	The destination IP mask. Depending on the destination netmask: <ul style="list-style-type: none"> • Any remote destination is reachable, i.e. the PPPoA connection acts as default route (dstmsk=0) • Only the remote (sub)net is reachable (dstmsk=1) • The actual destination mask will be the default netmask applicable for destination IP address • Only the single remote host is reachable (dstmsk=32) • Any valid (contiguous) netmask in case of Variable Length Subnet Masking (VLSM). 	OPTIONAL
label	The name of the label.	OPTIONAL
src	The source IP address specification for the route to be added when the link comes up.	OPTIONAL

srcmsk	<p>The source IP mask. Depending on the source netmask:</p> <ul style="list-style-type: none"> • Everybody is allowed to use this PPPoA connection (dstmsk=0) • Only members of the same subnet as the host which opened the PPPoA connection are allowed to use the PPPoA connection (dstmsk=1) • The actual destination mask will be the netmask applicable for the IP address of the host which opened the PPPoA connection. • Only the host which opened the PPPoA connection is allowed to use the PPPoA connection. (dstmsk=32) • Any valid (contiguous) netmask in case of VLSM. 	OPTIONAL
metric	<p>The route metric, i.e. the cost factor of the route. Practically, the cost is determined by the hop count.</p>	OPTIONAL

EXAMPLE:

```
=>pppoa iflist
pppoa1: dest : pppoa   [00:00:00]
  Retry : 10  QoS default  encaps VC-MUX
  mode = IP routing
  flags = echo magic accomp restart mru addr route savepwd
  dns metric = 0  trans addr = pat  mru = 1500
  route : dst=0.0.0.0/0 - src=10.0.0.0/1 (metric 1)
  auth = auto  user = johndoe@ISP  password = *****
  admin state = up   oper state = down   link state = retrying
  LCP : state = starting  retransm = 1  term. reason =
  IPCP: state = initial  retransm = 0  term. reason =

=>pppoa rtadd intf=pppoa1 dst=172.16.0.5 dstmsk=24 src=10.0.0.2 srcmask=24
=>pppoa iflist
pppoa1: dest : pppoa   [00:00:00]
  Retry : 10  QoS default  encaps VC-MUX
  mode = IP routing
  flags = echo magic accomp restart mru addr route savepwd
  dns metric = 0  trans addr = pat  mru = 1500
  route = 10.0.0.2/24 - 172.16.0.5/24 (metric 1)
  auth = auto  user = johndoe@ISP  password = *****
  admin state = up   oper state = down   link state = retrying
  LCP : state = starting  retransm = 1  term. reason =
  IPCP: state = initial  retransm = 0  term. reason =

=>
```

RELATED COMMANDS:

`pppoa rtdelete` Delete the route specification for an upcoming PPPoA link.

pppoa rtdelete

Delete the route specification for a PPPoA link.

Note Execute the PPPoA ifdetach command for this interface prior to deleting route configurations.

SYNTAX:

```
pppoa rtdelete   intf = <intfname>
```

intf	The PPPoA interface name for which to delete the route settings.	REQUIRED
------	--	----------

EXAMPLE:

```
=>pppoa iflist
pppoa_pppoa: dest : pppoa   [00:00:00]
  Retry : 10  QoS default  encaps VC-MUX
  mode = IP routing
  flags = echo magic accomp restart mru addr route savepwd
  dns metric = 0  trans addr = pat  mru = 1500
  route = 10.0.0.2/24 - 172.16.0.5/24 (metric 1)
  auth = auto  user = johndoe@ISP  password = *****
  admin state = up  oper state = down  link state = retrying
  LCP : state = starting  retransm = 1  term. reason =
  IPCP: state = initial  retransm = 0  term. reason =

=>pppoa rtdelete intf=pppoa1
=>pppoa iflist
pppoa_pppoa: dest : pppoa   [00:00:00]
  Retry : 10  QoS default  encaps VC-MUX
  mode = IP routing
  flags = echo magic accomp restart mru addr route savepwd
  dns metric = 0  trans addr = pat  mru = 1500
  auth = auto  user = johndoe@ISP  password = *****
  admin state = up  oper state = down  link state = retrying
  LCP : state = starting  retransm = 1  term. reason =
  IPCP: state = initial  retransm = 0  term. reason =

=>
```

RELATED COMMANDS:

pppoa rtadd	Configure a route specification for an upcoming PPPoA link.
-----------------------------	---

PPPoE Commands

Contents

This chapter covers the following commands:

Topic	Page
pppoe flush	306
pppoe ifadd	307
pppoe ifattach	309
pppoe ifconfig	311
pppoe ifdelete	316
pppoe ifdetach	317
pppoe iflist	318
pppoe ifscan	319
pppoe relay add	320
pppoe relay delete	321
pppoe relay flush	322
pppoe relay portlist	323
pppoe relay sesslist	324
pppoe rtadd	325
pppoe rtdelete	327

pppoe flush

Flush the current PPPoE configuration.

Note The flush command does not impact previously saved configurations.

SYNTAX:

```
pppoe flush
```

EXAMPLE:

```
=>pppoe iflist
PPPoE1: dest : PPPoE1
  Retry: 10 QoS default encaps VC-MUX
  mode = IP Routing
  flags = echo magic accomp mru addr route savepwd PPPoEOA
  trans addr = pat    mru = 1500
  route = 0.0.0.0/0 - 0.0.0.0/0 (metric 0)
  user name = guest   password = *****
  admin state = down   oper state = down   link state = not-connected
  LCP:   state = initial   retransm = 10   term. reason =
  IPCP:  state = initial   retransm = 0    term. reason =
=>pppoe flush
=>pppoe iflist
=>
```

pppoe ifadd

Create a new PPPoE interface.

SYNTAX:

```
pppoe ifadd      [intf = <string>]
                  [dest = <RELAY|phonebook entry>]
```

intf	The name for the new PPPoE interface. If not specified, the destination parameter must be specified. In this case the name of the destination will double as interface name.	OPTIONAL
dest	The ethernet port to be used for this PPPoE interface (e.g. an ETHoA name).	OPTIONAL

EXAMPLE:

```
=>pppoe iflist
PPPoE1: dest : pppoe1      [00:00:00]
  Retry : 10
  mode = IP routing
  flags = echo magic accomp restart mru addr savepwd
  dns metric = 0 mru = 1492
  auth = auto user = password =
  admin state = down oper state = down link state = not-connected
  LCP : state = initial retransm = 10 term. reason =
  IPCP: state = initial retransm = 0 term. reason =
  acname : --- service : ---
=>pppoe ifadd intf=PPPoE2 dest=pppoe2
=>pppoe iflist
PPPoE1: dest : pppoe1      [00:00:00]
  Retry : 10
  mode = IP routing
  flags = echo magic accomp restart mru addr savepwd
  dns metric = 0 mru = 1492
  auth = auto user = password =
  admin state = down oper state = down link state = not-connected
  LCP : state = initial retransm = 10 term. reason =
  IPCP: state = initial retransm = 0 term. reason =
  acname : --- service : ---

PPPoE2: dest : pppoe2      [00:00:00]
  Retry : 10
  mode = IP routing
  flags = echo magic accomp restart mru addr savepwd
  dns metric = 0 mru = 1492
  auth = auto user = password =
  admin state = down oper state = down link state = not-connected
  LCP : state = initial retransm = 10 term. reason =
  IPCP: state = initial retransm = 0 term. reason =
  acname : --- service : ---
=>
```

RELATED COMMANDS:

<code>pppoe ifattach</code>	Attach a PPPoE interface.
<code>pppoe ifconfig</code>	Configure a PPPoE interface.
<code>pppoe ifdelete</code>	Delete a PPPoE interface.
<code>pppoe ifdetach</code>	Detach a PPPoE interface.
<code>pppoe iflist</code>	Show current PPPoE configuration.

pppoe ifattach

Attach (i.e. connect) a PPPoE interface.

SYNTAX:

```
pppoe ifattach intf = <intfname>
```

intf	The name of the PPPoE interface to attach.	REQUIRED
------	--	----------

EXAMPLE:

```
=>pppoe iflist
PPPoE1: dest : PPPoE1
  Retry: 10   QoS   default   encaps   LLC
  mode = IP Routing
  flags = echo magic accomp restart mru addr route savepwd PPPoEOA
  trans addr = pat   mru = 1492
  route = 0.0.0.0/0 - 0.0.0.0/0 (metric 0)
  user name = guest   password = *****
  admin state = down   oper state = down   link state = not-connected
  LCP: state = initial   retransm = 10   term. reason =
  IPCP: state = initial   retransm = 0   term. reason =
=>pppoe ifattach =intf=PPPoE1
=>pppoe iflist
PPPoE1: dest : PPPoE1
  Retry: 10   QoS   default   encaps   LLC
  mode = IP Routing
  flags = echo magic accomp restart mru addr route savepwd PPPoEOA
  trans addr = pat   mru = 1492
  route = 0.0.0.0/0 - 0.0.0.0/0 (metric 0)
  user name = guest   password = *****
  admin state = up     oper state = down   link state = connected
  LCP: state = initial   retransm = 10   term. reason =
  IPCP: state = initial   retransm = 0   term. reason =
=>pppoe iflist
PPPoE1: dest : PPPoE1
  Retry: 10   QoS   default   encaps   LLC
  mode = IP Routing
  flags = echo magic accomp restart mru addr route savepwd PPPoEOA
  trans addr = pat   mru = 1492
  route = 0.0.0.0/0 - 0.0.0.0/0 (metric 0)
  user name = guest   password = *****
  admin state = up     oper state = up     link state = connected
  LCP: state = initial   retransm = 10   term. reason =
  IPCP: state = initial   retransm = 0   term. reason =
=>
```

RELATED COMMANDS:

<code>pppoe ifadd</code>	Create a new PPPoE interface.
<code>pppoe ifconfig</code>	Configure a PPPoE interface.
<code>pppoe ifdelete</code>	Delete a PPPoE interface.
<code>pppoe ifdetach</code>	Detach a PPPoE interface.
<code>pppoe iflist</code>	Show current PPPoE configuration.

pppoe ifconfig

Configure a PPPoE interface.

Note The interface to be configured may not be connected at the time of configuration. Execute `pppoe ifdetach` prior to executing the `pppoe ifconfig` command.

SYNTAX:

```
pppoe ifconfig  intf = <>
                  [dest = <{RELAY|eth0}>]
                  [user = <string>]
                  [password = <password>]
                  [acname = <quoted string>]
                  [servicename = <quoted string>]
                  [pcomp = <{off|on}>]
                  [accomp = <{on|off|negotiate}>]
                  [trace = <{off|on}>]
                  [concentrator = <{off|on}>]
                  [auth = <{pap|chap|auto}>]
                  [restart = <{off|on}>]
                  [retryinterval = <number{0-65535}>]
                  [passive = <{off|on}>]
                  [silent = <{off|on}>]
                  [echo = <{off|on}>]
                  [mru = <number{293-8192}>]
                  [laddr = <ip-address>]
                  [raddr = <ip-address>]
                  [netmask = <ip-mask(dotted or cidr)>]
                  [format = <{cidr|dotted|none}>]
                  [pool = <{none}>]
                  [savepwd = <{off|on}>]
                  [demanddial = <{off|on}>]
                  [primdns = <ip-address>]
                  [secdns = <ip-address>]
                  [dnsmetric = <number{0-100}>]
                  [idle = <number{0-1000000}>]
                  [idletrigger = <{RxTx|Rx|Tx}>]
                  [addrtrans = <{none|pat}>]
                  [unnumbered = <{off|on}>]
```

intf	The name of the PPPoE interface to configure.	REQUIRED
dest	The ethernet port to be used for this PPPoE interface (e.g. an ETHoA name).	OPTIONAL
user	The user name for remote PAP/CHAP authentication.	OPTIONAL
password	The password for remote PAP/CHAP authentication.	OPTIONAL

acname	The Access Concentrator name for a PPPoE connection. Execute the pppoe ifscan command to see the names of available access concentrators, if any.	OPTIONAL
servicename	The Service Name for a PPPoE connection. Execute the pppoe ifscan command to see the available service names, if any.	OPTIONAL
pcomp	Try (on) or do not try (off) to negotiate PPPoE protocol compression (LCP PCOMP). By default the negotiation is disabled (off).	OPTIONAL
accomp	Try (on), do never try (off) or negotiate (negotiate) to negotiate PPPoE address & control field compression (LCP ACCOMP). In most cases, LCP ACCOMP should not be disabled nor negotiated, i.e. the address field FF-03 should not be sent over ATM. Therefore by default this parameter is enabled (on). If the accomp parameter is set to 'negotiate', the local side of the PPPoE connection demands to do ACCOMP and adapts itself to the result of this negotiation.	OPTIONAL
trace	Enable (on) or disable (off) verbose console logging. By default tracing is disabled (off).	OPTIONAL
concentrator	The access concentrator is on this side of the PPPoE connection. Choose between: <ul style="list-style-type: none"> on: the PPPoE connection is terminated on the Access Concentrator (here the SpeedTouch™ itself) off: the SpeedTouch™ is PPPoE client (default) 	OPTIONAL
auth	Select the authentication protocol. Choose between: <ul style="list-style-type: none"> pap: Password Authentication Protocol (PAP) authentication will be forced. chap: Challenge Handshake Authentication Protocol (CHAP) authentication will be forced. auto: CHAP authentication will be used. If CHAP authentication is not successful, PAP authentication will be used instead. Auto is the default setting. 	OPTIONAL
restart	Automatically restart the connection when Link Control Protocol (LCP) link goes down (on) or do not restart automatically (off). By default restart is disabled (off).	OPTIONAL
retryinterval	A number between 0 and 65535 (seconds). Represents the intermediate interval between two retries to establish the connection on PPP level (LCP) (default=10).	OPTIONAL
passive	Put the link in listening state in case LCP times out (on) or not (off). This parameter allows to determine whether the link should be left open to wait for incoming messages from the remote side after 10 unsuccessful tries to establish the connection or not. By default, the listening state is disabled.	OPTIONAL

silent	Do not send anything at startup and just listen for incoming LCP messages (on) or retry up to 10 times to establish the connection (off). By default, the silent state is disabled.	OPTIONAL
echo	Send LCP echo requests at regular intervals (on) or not (off). By default the sending of LCP echo requests is enabled.	OPTIONAL
mru	A number between 293 and 8192. Represents the maximum packet size the SpeedTouch™ should negotiate to be able to receive (default=1492).	OPTIONAL
laddr	The local IP address of the peer-to-peer connection. Specifying a local IP address forces the remote side of the PPPoE link (if it allows to) to accept this IP address as the SpeedTouch™ PPPoE session IP address. If not specified, the SpeedTouch™ will accept any IP address. Typically the local IP address parameter is not specified.	OPTIONAL
raddr	The remote IP address of the peer-to-peer connection. Specifying a remote IP address forces the remote side of the PPPoE link (if it allows to) to accept this IP address as its PPPoE session IP address. If not specified, the SpeedTouch™ will accept any IP address. Typically the remote IP address parameter is not specified.	OPTIONAL
netmask	The subnetmask associated with this address. Specifying a subnetmask forces the remote side (if it allows to) to accept this subnetmask as the PPPoE session subnetmask. If not specified, the SpeedTouch™ will accept any subnetmask. The SpeedTouch™ will only request/accept a subnetmask if a DHCP server pool is associated, i.e. if the [pool] parameter is specified.	OPTIONAL
format	The negotiated subnetmask specified in the netmask parameter is specified in the dotted format (dotted) or in Classless Inter Domain Routing (CIDR) format (cidr). By default, the format is CIDR.	OPTIONAL
pool	The name of the free DHCP server pool to which the acquired IP subnet must be assigned to.	OPTIONAL
savepwd	Save password (on), if supplied, or do not save the password (off). By default, the saving of the password is disabled.	OPTIONAL
demanddial	Enable (on) or disable (off) the dial-on-demand feature.	OPTIONAL
primdns	The IP address of the primary DNS server. In case a primary DNS server is specified the SpeedTouch™ will negotiate this IP address with the remote side. If not specified, the SpeedTouch™ will accept any IP address.	OPTIONAL
secdns	The IP address of the (optional) secondary DNS server. In case a secondary DNS server is specified the SpeedTouch™ will negotiate this IP address with the remote side. If not specified, the SpeedTouch™ will accept any IP address.	OPTIONAL
dnsmetric	A number between 1 and 100. Represents the DNS route metric to be used for the negotiated DNS servers (default=1).	OPTIONAL

idle	A number between 0 and 1000000 (seconds). Represents after how many seconds an idle link goes down (default is 0).	OPTIONAL
idletrigger	Consider the link being idle if no traffic is sent and/or received during the idle time. Choose between: <ul style="list-style-type: none"> • RxTx The idle time period restarts when a packet is transmitted or received (default value). • Rx: The idle time period restarts when a packet is received. Transmitted packets are ignored. • Tx The idle time period restarts when a packet is transmitted. Received packets are ignored. 	OPTIONAL
addrtrans	Automatically enable address translation for the IP address of this link (pat) or do not use address translation (none) (default is none).	OPTIONAL
unnumbered	Takes the local IP address from 'laddr' field and remote IP address from the IP address pool assigned to the incoming PPPoE link. In case the unnumbered parameter is disabled, the same IP address is used for each connection on the server side, thus reducing the number of used IP addresses.	OPTIONAL

EXAMPLE:

```

=>pppoe iflist
PPPoE1: dest : PPPoE1
  Retry: 10   QoS  default   encaps  LLC
  mode = IP Routing
  flags = echo magic accomp restart mru addr route savepwd PPPoEOA
  trans addr = pat   mru = 1492
  route = 0.0.0.0/0 - 0.0.0.0/0 (metric 1)
  user name = My_Connection@MY_ISP   password = *****
  admin state = down   oper state = down   link state = not-connected
  LCP: state = initial   retransm = 10   term. reason =
  IPCP: state = initial   retransm = 0   term. reason =
=>pppoe ifconfig intf=PPPoE1 prot=pppoeoa encaps=vcmux
=>pppoe iflist
PPPoE1: dest : PPPoE1
  Retry: 10   QoS  default   encaps  VC-MUX
  mode = IP Routing
  flags = echo magic accomp restart mru addr route savepwd PPPoEOA
  trans addr = pat   mru = 1492
  route = 0.0.0.0/0 - 0.0.0.0/0 (metric 1)
  user name = My_Connection@MY_ISP   password = *****
  admin state = down   oper state = down   link state = not-connected
  LCP: state = initial   retransm = 10   term. reason =
  IPCP: state = initial   retransm = 0   term. reason =
=>

```

RELATED COMMANDS:

<code>pppoe ifadd</code>	Create a new PPPoE interface.
<code>pppoe ifattach</code>	Attach a PPPoE interface.
<code>pppoe ifdelete</code>	Delete a PPPoE interface.
<code>pppoe ifdetach</code>	Detach a PPPoE interface.
<code>pppoe iflist</code>	Show current PPPoE configuration.

pppoe ifdelete

Delete a PPPoE interface.

SYNTAX:

```
pppoe ifdelete intf = <intfname>
```

intf The name of the PPPoE interface to delete.

EXAMPLE:

```
=>pppoe iflist
PPPoE1: dest : PPPoE1
  Retry: 10   QoS   default   encaps   VC-MUX
  mode = IP Routing
  flags = echo magic accomp restart mru addr route savepwd PPPoEOA
  trans addr = pat   mru = 1500
  route = 0.0.0.0/0 - 0.0.0.0/0 (metric 0)
  user name = guest   password = *****
  admin state = down   oper state = down   link state = not-connected
  LCP: state = initial   retransm = 10   term. reason =
  IPCP: state = initial   retransm = 0   term. reason =
PPPoE2: dest : PVC2
  Retry: 10   QoS   default   encaps   VC-MUX
  mode = IP Routing
  flags = echo magic accomp restart mru addr savepwd PPPoEOA
  mru = 1500
  user name =   password =
  admin state = down   oper state = down   link state = not-connected
  LCP: state = initial   retransm = 10   term. reason =
  IPCP: state = initial   retransm = 0   term. reason =
=>pppoe ifdelete intf=PPPoE2
=>pppoe iflist
PPPoE1: dest : PPPoE1
  Retry: 10   QoS   default   encaps   VC-MUX
  mode = IP Routing
  flags = echo magic accomp restart mru addr route savepwd PPPoEOA
  trans addr = pat   mru = 1500
  route = 0.0.0.0/0 - 0.0.0.0/0 (metric 0)
  user name = guest   password = *****
  admin state = down   oper state = down   link state = not-connected
  LCP: state = initial   retransm = 10   term. reason =
  IPCP: state = initial   retransm = 0   term. reason =
=>
```

RELATED COMMANDS:

- | | |
|--------------------------------|-----------------------------------|
| pppoe ifadd | Create a new PPPoE interface. |
| pppoe ifattach | Attach a PPPoE interface. |
| pppoe ifconfig | Configure a PPPoE interface. |
| pppoe ifdetach | Detach a PPPoE interface. |
| pppoe iflist | Show current PPPoE configuration. |

pppoe ifdetach

Detach a PPPoE interface.

SYNTAX:

```
pppoe ifdetach intf = <intfname>
```

intf	The name of the PPPoE interface.	REQUIRED
------	----------------------------------	----------

EXAMPLE:

```
=>pppoe iflist
PPPoE1: dest : PPPoE1
  Retry: 10   QoS   default   encaps   VC-MUX
  mode = IP Routing
  flags = echo magic accomp restart mru addr route savepwd PPPoEOA
  trans addr = pat   mru = 1492
  route = 0.0.0.0/0 - 0.0.0.0/0 (metric 0)
  user name = guest   password = *****
  admin state = up     oper state = up     link state = connected
  LCP: state = initial   retransm = 0     term. reason =
  IPCP: state = initial   retransm = 10    term. reason =
=>pppoe ifdetach =intf=PPPoE1
=>pppoe iflist
PPPoE1: dest : PPPoE1
  Retry: 10   QoS   default   encaps   VC-MUX
  mode = IP Routing
  flags = echo magic accomp restart mru addr route savepwd PPPoEOA
  trans addr = pat   mru = 1492
  route = 0.0.0.0/0 - 0.0.0.0/0 (metric 0)
  user name = guest   password = *****
  admin state = down   oper state = down   link state = not-connected
  LCP: state = initial   retransm = 10    term. reason =
  IPCP: state = initial   retransm = 0     term. reason =
=>
```

RELATED COMMANDS:

pppoe ifadd	Create a new PPPoE interface.
pppoe ifattach	Attach a PPPoE interface.
pppoe ifconfig	Configure a PPPoE interface.
pppoe ifdelete	Delete a PPPoE interface.
pppoe iflist	Show current PPPoE configuration.

pppoe iflist

Show current configuration of all or a specified PPPoE interface(s).

SYNTAX:

```
pppoe iflist [intf = <intfname>]
```

intf	The name of the PPPoE interface. If this parameter is not specified, all PPPoE interfaces are shown.	OPTIONAL
------	---	----------

EXAMPLE INPUT/OUTPUT:

```
=>pppoe iflist
PPPoE1: dest : PPPoE1
  Retry: 10   QoS   default   encaps   VC-MUX
  mode = IP Routing
  flags = echo magic accomp restart mru addr route savepwd PPPoEOA
  trans addr = pat   mru = 1500
  route = 0.0.0.0/0 - 0.0.0.0/0 (metric 0)
  user name = guest   password = *****
  admin state = down   oper state = down   link state = not-connected
  LCP: state = initial   retransm = 10   term. reason =
  IPCP: state = initial   retransm = 0   term. reason =

PPPoE2: dest : PVC2
  Retry: 10   QoS   default   encaps   VC-MUX
  mode = IP Routing
  flags = echo magic accomp restart mru addr savepwd PPPoEOA
  mru = 1500
  user name =   password =
  admin state = down   oper state = down   link state = not-connected
  LCP: state = initial   retransm = 10   term. reason =
  IPCP: state = initial   retransm = 0   term. reason =

=>
```

RELATED COMMANDS:

pppoe ifadd	Create a new PPPoE interface.
pppoe ifattach	Attach a PPPoE interface.
pppoe ifconfig	Configure a PPPoE interface.
pppoe ifdelete	Delete a PPPoE interface.
pppoe ifdetach	Detach a PPPoE interface.

pppoe ifscan

Scan a PPPoE interface (proto=pppoe) for available Access Concentrator names and Service Names.

Note Execute the pppoe ifdetach command for this interface before performing a scan on it.

SYNTAX:

```
pppoe ifscan      intf = <intfname>
                  [time = <number{0-36000}>]
```

intf	The name of the Routed PPPoE interface to scan.	REQUIRED
time	A number between 0 and 36000 (seconds). Represents the time to scan for services.	OPTIONAL

EXAMPLE:

```
=>pppoe iflist
PPPoE1: dest : PPPoE1
  Retry: 10   QoS   default   encaps   VC-MUX
  mode = IP Routing
  flags = echo magic accomp restart mru addr route savepwd PPPoEOA
  trans addr = pat   mru = 1492
  route = 0.0.0.0/0 - 0.0.0.0/0 (metric 0)
  user name = guest   password = *****
  admin state = down   oper state = down   link state = not-connected
  LCP: state = initial   retransm = 10   term. reason =
  IPCP: state = initial   retransm = 0   term. reason =
=>pppoe ifscan intf=PPPoE1 time=45
      Service Name                Access Concentrator

Done !
=>
```

RELATED COMMANDS:

[pppoe ifconfig](#) Configure a PPPoE interface.

pppoe relay add

Add an ethernet port to the PPPoE relay list.

SYNTAX:

```
pppoe relay add      port = <port name>
```

port	The ethernet port to be added to the PPPoE relay agent list.	REQUIRED
------	--	----------

RELATED COMMANDS:

- [pppoe relay delete](#) Delete an ethernet port from the PPPoE relay agent list.
- [pppoe relay flush](#) Remove all ethernet ports from the PPPoE relay agent list and terminate all sessions.
- [pppoe relay portlist](#) List all ethernet ports added to the PPPoE relay agent list.
- [pppoe relay sesslist](#) List all active PPPoE relay sessions.

pppoe relay delete

Delete an ethernet port from the PPPoE relay agent list.

SYNTAX:

```
pppoe relay delete    port = <port name>
```

port	The ethernet port to be deleted from the PPPoE relay agent list.	REQUIRED
------	--	----------

RELATED COMMANDS:

pppoe relay add	Add an ethernet port to the PPPoE relay list.
pppoe relay flush	Remove all ethernet ports from the PPPoE relay agent list and terminate all sessions.
pppoe relay portlist	List all ethernet ports added to the PPPoE relay agent list.
pppoe relay sesslist	List all active PPPoE relay sessions.

pppoe relay flush

Remove all ethernet ports from the PPPoE relay agent list and terminate all sessions.

SYNTAX:

```
pppoe relay flush
```

RELATED COMMANDS:

pppoe relay add	Add an ethernet port to the PPPoE relay list.
pppoe relay delete	Delete an ethernet port from the PPPoE relay agent list.
pppoe relay portlist	List all ethernet ports added to the PPPoE relay agent list.
pppoe relay sesslist	List all active PPPoE relay sessions.

pppoe relay portlist

List all ethernet ports added to the PPPoE relay agent list.

SYNTAX:

```
pppoe relay portlist
```

EXAMPLE:

```
=>pppoe relay portlist
# Port      HWaddr      Status
1 ETHoA_1    00-90-d0-72-88-64  UP
=>
```

RELATED COMMANDS:

pppoe relay add	Add an ethernet port to the PPPoE relay list.
pppoe relay delete	Delete an ethernet port from the PPPoE relay agent list.
pppoe relay flush	Remove all ethernet ports from the PPPoE relay agent list and terminate all sessions.
pppoe relay sesslist	List all active PPPoE relay sessions.

pppoe relay sesslist

List all active PPPoE relay sessions.

SYNTAX:

```
pppoe relay sesslist
```

RELATED COMMANDS:

pppoe relay add	Add an ethernet port to the PPPoE relay list.
pppoe relay delete	Delete an ethernet port from the PPPoE relay agent list.
pppoe relay flush	Remove all ethernet ports from the PPPoE relay agent list and terminate all sessions.
pppoe relay portlist	List all ethernet ports added to the PPPoE relay agent list.

pppoe rtadd

Automatically add a route configuration to the routing table in case the specified PPPoE interface link comes up. This route configuration will determine which local hosts are allowed to use this link and/or which remote destinations should be or should not be reachable.

Note Execute the `pppoe ifdetach` command for this interface prior to configuring routes.

SYNTAX:

```
pppoe rtadd      intf = <intfname>
                  dst = <ip-address>
                  [dstmsk = <ip-mask(dotted or cidr)>]
                  [label = <{}>]
                  [src = <ip-address>]
                  [srcmsk = <ip-mask(dotted or cidr)>]
                  [metric = <number{0-100}>]
```

intf	The name of the PPPoE interface.	REQUIRED
dst	The destination IP address for the route to be added when the link comes up.	REQUIRED
dstmsk	The destination IP mask. Depending on the destination netmask: <ul style="list-style-type: none"> • Any remote destination is reachable, i.e. the PPPoE connection acts as default route (dstmsk=0) • Only the remote (sub)net is reachable (dstmsk=1) • The actual destination mask will be the default netmask applicable for destination IP address • Only the single remote host is reachable (dstmsk=32) • Any valid (contiguous) netmask in case of Variable Length Subnet Masking (VLSM). 	OPTIONAL
label	The name of the label	OPTIONAL
src	The source IP address specification for the route to be added when the link comes up.	OPTIONAL

srcmsk	<p>The source IP mask. Depending on the source netmask:</p> <ul style="list-style-type: none"> • Everybody is allowed to use this PPPoE connection (dstmsk=0) • Only members of the same subnet as the host which opened the PPPoE connection are allowed to use the PPPoE connection (dstmsk=1) • The actual destination mask will be the netmask applicable for the IP address of the host which opened the PPPoE connection. • Only the host which opened the PPPoE connection is allowed to use the PPPoE connection. (dstmsk=32) • Any valid (contiguous) netmask in case of VLSM. 	OPTIONAL
metric	<p>The route metric, i.e. the cost factor of the route. Practically, the cost is determined by the hop count.</p>	OPTIONAL

EXAMPLE:

```
=>pppoe iflist
PPPoE1: dest : PVC3
  Retry: 10   QoS   default   encaps   LLC
  mode = IP Routing
  flags = echo magic accomp restart mru addr route savepwd PPPoE
  trans addr = pat   mru = 1492
  user name = guest   password = *****
  admin state = down   oper state = down   link state = not-connected
  LCP : state = initial   retransm = 10   term. reason =
  IPCP: state = initial   retransm = 0   term. reason =
=>pppoe rtadd intf=PPPoE1 dst=172.16.0.5 dstmsk=24 src=10.0.0.2 srcmask=24
=>pppoe iflist
PPPoE1: dest : PVC3
  Retry: 10   QoS   default   encaps   LLC
  mode = IP Routing
  flags = echo magic accomp restart mru addr route savepwd PPPoE
  trans addr = pat   mru = 1492
  route = 10.0.0.2/24 - 172.16.0.5/24 (metric 1)
  user name = guest   password = *****
  admin state = down   oper state = down   link state = not-connected
  LCP : state = initial   retransm = 10   term. reason =
  IPCP: state = initial   retransm = 0   term. reason =
=>
```

RELATED COMMANDS:

`pppoe rtdelete` Delete the route specification for an upcoming PPPoE link.

pppoe rtdelete

Delete the route specification for a PPPoE link.

Note Execute the `pppoe ifdetach` command for this interface prior to deleting route configurations.

SYNTAX:

```
pppoe rtdelete   intf = <intfname>
```

intf	The PPPoE interface name for which to delete the route settings.	REQUIRED
------	--	----------

EXAMPLE:

```
=>pppoe iflist
PPPoE1: dest : PVC3
  Retry: 10   QoS   default   encaps   LLC
  mode = IP Routing
  flags = echo magic accomp restart mru addr route savepwd PPPoE
  trans addr = pat   mru = 1492
  route = 10.0.0.2/24 - 172.16.0.5/24 (metric 1)
  user name = guest   password = *****
  admin state = down   oper state = down   link state = not-connected
  LCP : state = initial   retransm = 10   term. reason =
  IPCP: state = initial   retransm = 0   term. reason =
=>pppoe rtdelete intf=PPPoE1
=>pppoe iflist
PPPoE1: dest : PVC3
  Retry: 10   QoS   default   encaps   LLC
  mode = IP Routing
  flags = echo magic accomp restart mru addr route savepwd PPPoE
  trans addr = pat   mru = 1492
  user name = guest   password = *****
  admin state = down   oper state = down   link state = not-connected
  LCP : state = initial   retransm = 10   term. reason =
  IPCP: state = initial   retransm = 0   term. reason =
=>
```

RELATED COMMANDS:

<code>pppoe rtadd</code>	Configure a route specification for an upcoming PPPoE link.
--------------------------	---

PPTP Commands

Contents

This chapter covers the following commands:

Topic	Page
pptp ifadd	330
pptp flush	331
pptp list	332
pptp profadd	333
pptp profdelete	334
pptp proflist	335

pptp ifadd

Adds a Point-to-Point Tunneling Protocol (PPTP) profile (backwards compatible with previous release, use profiles instead).

SYNTAX:

```
pptp ifadd      dest = <string>
                 [rate = <number{10-10000}>]
                 [encaps = <{vcmux|nlpid}>]
                 [ac = <{never|always|keep}>]
```

dest	The WAN destination for this PPTP tunnel. Typically a phonebook entry.	REQUIRED
rate	A number between 10 and 10000. Represents the transmit speed in bits/s for the WAN link.	OPTIONAL
encaps	The type of WAN encapsulation. Choose between: <ul style="list-style-type: none"> • vcmux • nlpid - Network Layer Protocol IDentifiers (NLPID) 	OPTIONAL
ac	The High-level Data Link Control (HDLC) framing option applicable to PPTP interfaces using this PPTP profile. Before relaying the encapsulated PPP frames over the PPPoA link, make sure that the address and control field (0xFF03) is always in front of the frames (always), make sure the address and control field will never be found in front of the frames (never) or do not change the frames arriving via the PPTP tunnel (keep). By default the address and control field is never sent (compliant to RFC2364). It is recommended to keep this setting.	OPTIONAL

pptp flush

Flush complete PPTP configuration.

Note The flush command does not impact previously saved configurations.

SYNTAX:

```
pptp flush
```

EXAMPLE:

```
=>pptp profadd name=Relay_PPP1 encaps=nlpid ac=always
=>pptp proflist
Profile      QoS      Encaps      AC
Relay_PPP1  default  nlpid       always
=>pptp flush
=>pptp proflist
=>
```

pptp list

Show current PPTP configuration.

SYNTAX:

```
pptp list
```

EXAMPLE INPUT/OUTPUT:

```
=>pptp list
Dialstr      Destination  QoS      Encaps  AC      State      User
              DIALUP_PPP3 default  vcmux   never    CONNECTED  (10.0.0.2)
=>
```

pptp profadd

Define a new PPTP profile.

SYNTAX:

```
pptp profadd    name = <string>
                [qos = <string>]
                [encaps = <{vcmux|nlpid}>]
                [ac = <{never|always|keep}>]
```

name	The name for the PPTP profile.	REQUIRED
qos	The name of the qosbook entry, containing the settings for this profile. This parameter never needs to be specified.	OPTIONAL
encaps	The type of WAN encapsulation applicable to PPTP interfaces using this PPTP profile. Choose between: <ul style="list-style-type: none"> vcmux nlpid - Network Layer Protocol IDentifiers (NLPID) 	OPTIONAL
ac	The High-level Data Link Control (HDLC) framing option applicable to PPTP interfaces using this PPTP profile. Before relaying the encapsulated PPP frames over the PPPoA link, make sure that the address and control field (0xFF03) is always in front of the frames (always), make sure the address and control field will never be found in front of the frames (never) or do not change the frames arriving via the PPTP tunnel (keep). By default the address and control field is never sent (compliant to RFC2364). It is recommended to keep this setting.	OPTIONAL

EXAMPLE:

```
=>pptp proflist
Profile    QoS      Encaps    AC
Relay_PPP1 default  nlpid     always
=>pptp profadd name=PPTPLink encaps=vcmux ac=never
=>pptp proflist
Profile    QoS      Encaps    AC
Relay_PPP1 default  nlpid     always
PPTPLink   default  vcmux     never
=>
```

RELATED COMMANDS:

pptp profdelete	Delete a PPTP profile.
pptp proflist	Show current PPTP profiles.

pptp profdelete

Delete a PPTP profile.

SYNTAX:

```
pptp profdelete name <string>
```

name	The name for the PPTP profile.	REQUIRED
------	--------------------------------	----------

EXAMPLE:

```
=>pptp proflist
Profile      QoS      Encaps      AC
Relay_PPP1   default  nlpid       always
PPTPLink     default  vcmux       never
=>pptp profdelete name=PPTPLink
=>pptp proflist
Profile      QoS      Encaps      AC
Relay_PPP1   default  nlpid       always
=>
```

RELATED COMMANDS:

pptp profadd	Define a new PPTP profile.
pptp proflist	Show current PPTP profiles.

pptp proflist

Show all current PPTP profiles.

SYNTAX:

```
pptp proflist
```

EXAMPLE:

```
=>pptp proflist
Profile      QoS      Encaps      AC
Relay_PPP1   default  nlpid       always
PPTPLink     default  vcmux       never
=>
```

RELATED COMMANDS:

<code>pptp profadd</code>	Define a new PPTP profile.
<code>pptp profdelete</code>	Delete a PPTP profile.

QoSBook Commands

Contents

This chapter covers the following commands:

Topic	Page
qosbook add	338
qosbook config	340
qosbook delete	341
qosbook flush	342
qosbook list	343

qosbook add

Add a Quality of Service (QoS) book entry.

SYNTAX:

```
qosbook add      name = <string>
                  class = <{ubr|cbr|vbr-rt|vbr-nrt}>
                  [tx_peakrate = <number{0-27786}>]
                  [tx_sustrate = <number{0-27786}>]
                  [tx_maxburst = <number{0-12240}>]
                  [rx_peakrate = <number{0-27786}>]
                  [rx_sustrate = <number{0-27786}>]
                  [rx_maxburst = <number{0-12240}>]
                  [framediscard = <{enabled|disabled}>]
```

name	The name for the new QoS entry.	REQUIRED
class	The ATM service category. Choose between: <ul style="list-style-type: none"> • ubr: unspecified bit rate • cbr: constant bit rate • vbr-rt: variable bit rate - real time • vbr-nrt: variable bit rate - non real time 	REQUIRED
tx_peakrate	A number between 0 and 27786 (Kilobits per second). Indicates the peak rate (in kilobits per second) in the transmit (upstream) direction. Use '0' to indicate linerate for UBR.	OPTIONAL
tx_sustrate	A number between 0 and 27786 (Kilobits per second). Indicates the sustainable rate (in kilobits per second) in the transmit (upstream) direction. (VBR only)	OPTIONAL
tx_maxburst	A number between 0 and 12240 (bytes per second). Indicates the maximum burst size (in bytes) in the transmit (upstream) direction. (VBR only)	OPTIONAL
rx_peakrate	A number between 0 and 27786 (Kilobits per second). Indicates the peak rate (in kilobits per second) in the receive (downstream) direction. Use '0' to indicate linerate for UBR. If not specified, copy of the transmit peak rate. (ATMF only)	OPTIONAL
rx_sustrate	A number between 0 and 27786 (Kilobits per second). Indicates the sustainable rate (in kilobits per second) in the receive (downstream) direction. If not specified, copy of the transmit sustainable rate. (VBR ATMF only)	OPTIONAL
rx_maxburst	A number between 0 and 12240 (bytes per second). Indicates the maximum burst size (in bytes) in the receive (downstream) direction. If not specified, copy of the transmit maximum burst size. (VBR ATMF only)	OPTIONAL
framediscard	Enable/disable frame discard.	OPTIONAL

EXAMPLE:

```

=>qosbook list
Name    Ref Type    TX peak  sust    burst    RX peak  sust    burst    framediscard
      (Kbits) (Kbits) (bytes) (Kbits) (Kbits) (bytes)
default 3  ubr    linerate 0      0      linerate 0      0      disabled
=>qosbook add name=TestVBR class=vbr-nrt tx_peakrate=1500 tx_sustrate=1000 tx_maxburst=100
rx_sustrate=1000 rx_maxburst=100 framediscard=enabled
=>qosbook list
Name    Ref Type    TX peak  sust    burst    RX peak  sust    burst    framediscard
      (Kbits) (Kbits) (bytes) (Kbits) (Kbits) (bytes)
default 3  ubr    linerate 0      0      linerate 0      0      disabled
TestVBR 0  vbr-nrt 1500    1000    144     1500    1000    144     enabled
=>

```

IMPORTANT NOTE:

The SpeedTouch™ always rounds up specified burst sizes to a multiple of 48 bytes, i.e. a multiple of ATM cells.

Example

In the example above a burst size of 100 bytes is specified (`tx_maxburst=100`). The SpeedTouch™ will round up the burst size to the closest matching multiple of 48 bytes, as can be seen when displaying the profile via `:qosbook list` (`burst=144`).

RELATED COMMANDS:

`qosbook delete` Remove a QoS book entry.

`qosbook list` Show current QoS book.

qosbook config

Sets or changes qosbook config parameters.

SYNTAX:

```
qosbook config [format = <{bytes|cells}>]
```

format	The input, output format of the qosbook. Choose between:	OPTIONAL
	<ul style="list-style-type: none"> • bytes (default) • cells 	

EXAMPLE:

```
=>qosbook list
Name Ref Type TX peak sust burst RX peak sust burst framediscard
(Kbits) (Kbits) (bytes) (Kbits) (Kbits) (bytes)
default 3 ubr linate 0 0 0 linate 0 0 disabled
=>qosbook config
qosbook format in bytes
=>qosbook config format=cells
=>qosbook config
qosbook format in cells
=>qosbook list
Name Ref Type TX peak sust burst RX peak sust burst framediscard
(cps) (cps) (cells) (cps) (cps) (cells)
default 3 ubr linate 0 0 0 linate 0 0 disabled
=>
```

qosbook delete

Remove a Quality of Service book entry.

SYNTAX:

```
qosbook delete      name = <string>
                   [force = <{no|yes}>]
```

name	The name of the QoS book entry to delete.	REQUIRED
force	Force deletion of the entry even if it is still in use (yes) or do not force the deletion (no). By default forced deletion is disabled.	OPTIONAL

EXAMPLE:

```
=>qosbook list
Name   Ref Type   TX peak  sust    burst   RX peak  sust    burst   framediscard
      (Kbits) (Kbits) (bytes) (Kbits) (Kbits) (bytes)
default 3   ubr   linerate 0      0      linerate 0      0      disabled
TestVBR 0   vbr-nrt 1500  1000   144    1500    1000   144    enabled
=>qosbook delete name=TestVBR
=>qosbook list
Name   Ref Type   TX peak  sust    burst   RX peak  sust    burst   framediscard
      (Kbits) (Kbits) (bytes) (Kbits) (Kbits) (bytes)
default 3   ubr   linerate 0      0      linerate 0      0      disabled
=>
```

RELATED COMMANDS:

qosbook add	Add a QoS book entry.
qosbook list	Show current QoS book.

qosbook flush

Flush complete Quality of Service book.

Note The flush command does not impact previously saved configurations.

SYNTAX:

```
qosbook flush
```

qosbook list

Show current Quality of Service book.

SYNTAX:

```
qosbook list
```

EXAMPLE:

```
=>qosbook list
Name      Ref Type      TX peak  sust      burst     RX peak  sust      burst     framediscard
          (Kbits) (kbits)  (bytes)  (Kbits)  (kbits)  (bytes)
default 3   ubr        linerate 0         0         linerate 0         0         disabled
TestVBR 0   vbr-nrt   1500     1000     144      1500     1000     144      enabled
=>
```

RELATED COMMANDS:

`qosbook add` Add a QoS book entry.

`qosbook delete` Remove a QoS book entry.

Script Commands

WARNING

Scripting is not a general purpose mechanism but is only used in the autoPVC/ILMI mechanism.

Do not use any commands of the Script CLI command group.

SNMP Commands

Contents

This chapter covers the following commands:

Topic	Page
snmp config	348
snmp get	349
snmp getNext	350
snmp list	351
snmp walk	352

snmp config

Show/set global Simple Network Management Protocol (SNMP) parameters.

SYNTAX:

```
snmp config      [RWCommunity = <string>]
                  [ROCommunity = <string>]
                  [sysContact = <quoted string>]
                  [sysName = <quoted string>]
                  [sysLocation = <quoted string>]
```

RWCommunity	The read-write community name. By default the read-write community name is private.	OPTIONAL
ROCommunity	The read-only community name. By default the read-only community name is public.	OPTIONAL
sysContact	The SNMP system contact. By default the system contact is Service Provider.	OPTIONAL
sysName	The SNMP system name. By default the system name is SpeedTouch™ 610.	OPTIONAL
sysLocation	The SNMP system location. By default the system location is Customer Premises.	OPTIONAL

EXAMPLE (default configuration):

```
=>snmp config

Read-write SNMP community name : private
Read-only SNMP community name : public
SNMP System Contact      : Service Provider
SNMP System Name        : SpeedTouch 610
SNMP System Location    : Customer Premises
=>
```

snmp get

Get from the supplied SNMP Object Identifier (OID), e.g. get ObjectId=.1.3.6.1.2.1.1.1.0.

SYNTAX:

```
snmp get [ObjectId = <string>]
```

ObjectId	The Object Identifier. Object id to get from ... must include the instance which is 0 for scalar objects e.g. .1.3.6.1.2.1.1.1.0 sysDescription If not specified, the sysdescription OID .1.3.6.1.2.1.1.1.0 is assumed. Its value is SpeedTouch™.	OPTIONAL
----------	--	----------

EXAMPLE:

```
=>snmp get
VB_octetStr .1.3.6.1.2.1.1.1.0 SpeedTouch™ 610
=>
```

RELATED COMMANDS:

snmp getNext	GetNext from the supplied SNMP OID.
snmp walk	Walk from the supplied SNMP OID.

snmp getNext

GetNext from the supplied SNMP OID.

SYNTAX:

```
snmp get [ObjectId = <string>]
```

[ObjectId]	The Object Identifier. Object id to getNext from e.g. .1.3.6.1.2.1.1 system returns sysDescription	OPTIONAL
------------	---	----------

EXAMPLE:

```
=>snmp getNext ObjectId=.1.3.6.1.2.1.1.4.0  
VB_octetStr .1.3.6.1.2.1.1.5.0 Sascha  
=>
```

RELATED COMMANDS:

snmp get	Get from the supplied SNMP OID.
snmp walk	Walk from the supplied SNMP OID.

snmp list

List all SNMP global parameters.

SYNTAX:

```
snmp list
```

EXAMPLE:

```
=>snmp list
Read-write SNMP community name : private
Read-only SNMP community name : public
SNMP System Contact      : Service Provider
SNMP System Name        : Sascha
SNMP System Location    : Customer Premises
=>
```

snmp walk

Walk from the supplied SNMP Object Identifier (OID).

SYNTAX:

```
snmp get [ObjectId = <string>]
```

ObjectId	The Object Identifier. Object id to walk from e.g. .1.3.6.1.2.1.1 system walks the system group	OPTIONAL
----------	--	----------

EXAMPLE:

```
=>snmp walk ObjectId=.1.3.6.1.2.1.1
VB_octetStr .1.3.6.1.2.1.1.1.0      SpeedTouch™ 610
VB_objId    .1.3.6.1.2.1.1.2.0      .1.3.6.1.4.1.637.61.2
VB_timeTicks .1.3.6.1.2.1.1.3.0     2927636
VB_octetStr .1.3.6.1.2.1.1.4.0     Service Provider
VB_octetStr .1.3.6.1.2.1.1.5.0     Sascha
VB_octetStr .1.3.6.1.2.1.1.6.0     Customer Premises
VB_integer  .1.3.6.1.2.1.1.7.0     72
=>
```

RELATED COMMANDS:

snmp get	Get from the supplied SNMP OID.
snmp getNext	GetNext from the supplied SNMP OID.

Software Commands

Contents

This chapter covers the following commands:

Topic	Page
software version	354
software upgrade	355

software version

Show the software version.

SYNTAX:

```
software version
```

EXAMPLE:

```
=>software version  
Flash image : 4.2.1.15.0  
Build name  : LLT5AA4.20E  
=>
```

software upgrade

Reboots the modem to initiate the SW upgrade. New software available on a remote LAN host will be uploaded to the modem.

SYNTAX:

```
software upgrade
```

Switch Commands

Introduction

These commands are only applicable to the SpeedTouch™510 four port switch version.

Contents

This chapter covers the following commands:

Topic	Page
switch group flush	358
switch group list	359
switch group move	360
switch mirror capture	361
switch mirror egress	362
switch mirror ingress	363

switch group flush

To set all ports to the default settings, i.e. all ports in group 0.

SYNTAX:

```
switch group flush
```

EXAMPLE:

```
=>switch group list  
Group 0 Ports: 4  
Group 1 Ports: 1  
Group 2 Ports: 2 3  
=>switch group flush  
=>switch group list  
Group 0 Ports: 1 2 3 4  
=>
```

RELATED COMMANDS:

- | | |
|-------------------|---|
| switch group list | List all configured groups. |
| switch group move | Move a specified port to a specified group. |

switch group list

List all configured groups.

SYNTAX:

```
switch group list
```

EXAMPLE:

```
=>switch group list  
Group 0 Ports: 4  
Group 1 Ports: 1  
Group 2 Ports: 2 3
```

RELATED COMMANDS:

`switch group flush`

To set all ports to the default settings (all ports in group 0).

`switch group move`

Move a specified port to a specified group.

switch group move

Move a specified port to a specified group.

SYNTAX:

```
switch group move    group = <number{0-4}>
                    port = <number{1-4}>
```

group	The group id.	REQUIRED
port	The port.	REQUIRED

EXAMPLE:

```
=>switch group list
Group 0 Ports: 1 2 3 4
=>switch group move group=3 port=1
=>switch group list
Group 0 Ports: 2 3 4
Group 3 Ports: 1
```

RELATED COMMANDS:

switch group flush	To set all ports to the default settings (all ports in group 0).
switch group list	List all configured groups.

switch mirror capture

Define the specified port to be the Mirror Capture Port.

Note Only one port can be the Mirror Capture Port at any one time.

SYNTAX:

```
switch mirror capture port = <number{1-4}>
```

port	The port to be the Mirror Capture Port.	REQUIRED
------	---	----------

EXAMPLE:

```
=>switch mirror capture port=2  
=>switch mirror capture  
Mirror capture port = 2
```

RELATED COMMANDS:

switch mirror egress	Enable or disable the specified port to be a Transmitted Port Mirroring.
switch mirror ingress	Enable or disable the specified port to be a Received Port Mirroring.

switch mirror egress

Enable or disable the specified port to be the Mirror Egress Port.

Note Only one port can be the Mirror Egress Port at any one time.
But a port can be the Mirror Egress Port and the Mirror Ingress Port at the same time.

SYNTAX:

```
switch mirror egress port = <number{1-4}>
                        [state = <{enabled|disabled}>]
```

port	The port to be the Mirror Egress Port.	REQUIRED
state	This parameter permits to enable/disable a port as Mirror Egress Port.	OPTIONAL

EXAMPLE:

Here port 1 is enabled as Mirror Egress Port

```
=>switch mirror egress port=1
=>switch mirror egress
Egress mirror port = 1
```

Here port 1 is disabled as Mirror Egress Port

```
=>switch mirror egress
Egress mirror port = 1
=>switch mirror egress port=1 state=disabled
=>switch mirror egress
=>
```

RELATED COMMANDS:

- [switch mirror capture](#) Define the specified port to be the Mirror Capture Port.
- [switch mirror ingress](#) Enable or disable the specified port to be a Received Port Mirroring.

switch mirror ingress

Enable or disable the specified port to be the Mirror Ingress Port.

Note Only one port can be the Mirror Ingress Port at any one time.
But a port can be the Mirror Egress Port and the Mirror Ingress Port at the same time.

SYNTAX:

```
switch mirror egress port = <number{1-4}>
[state = <{enabled|disabled}>]
```

port	The port to be the Mirror Ingress Port.	REQUIRED
state	This parameter permits to enable/disable a port as Mirror Ingress Port.	OPTIONAL

EXAMPLE:

Here port 2 is enabled as Mirror Ingress Port

```
=>switch mirror ingress port=2
=>switch mirror ingress
Ingress mirror port = 2
```

Here port 2 is disabled as Mirror Ingress Port

```
=>switch mirror ingress
Ingress mirror port = 2
=>switch mirror ingress port=2 state=disabled
=>switch mirror ingress
=>
```

RELATED COMMANDS:

- `switch mirror capture` Define the specified port to be the Mirror Capture Port.
- `switch mirror egress` Enable or disable the specified port to be a Transmitted Port Mirroring.

System Commands

Contents

This chapter covers the following commands:

Topic	Page
system clearpassword	366
system config	367
system flush	368
system reboot	369
system reset	370
system setpassword	371
system stats	372

system clearpassword

Clear current SpeedTouch™ system password.

Note To avoid unrestricted and unauthorized access to the SpeedTouch™, it is highly recommended to make sure it is protected by a SpeedTouch™ system password (via :system setpassword) and to change the password on a regular basis.

SYNTAX:

```
system clearpassword
```

EXAMPLE:

```
=>system clearpassword  
Security notification: Password changed, use 'saveall' to make it permanent.  
=>
```

RELATED COMMANDS:

`system setpassword` Set/change current system password.

system config

Show/set SpeedTouch™ system configuration parameters.

Note For a good operation of Universal Plug and Play (UPnP) and the discovery mechanism, it is highly recommended not to change the System config settings.

SYNTAX:

```
system config      [upnp = <{disabled|enabled}>]
                  [mdap = <{disabled|enabled}>]
                  [drst = <{disabled|enabled}>]
                  [digestauth = <{disabled|enabled}>]
                  [strictusername = <{disabled|enabled}>]
                  [dcache = <{disabled|enabled}>]
```

upnp	Enable (enabled) or disable (disabled) Universal Plug and Play (UPnP) discovery. By default UPnP discovery is enabled.	OPTIONAL
mdap	Enable (enabled) or disable (disabled) proprietary discovery protocol. By default MDAP discovery is enabled.	OPTIONAL
drst	Enable / disable DRST access. By default DRST access is disabled.	OPTIONAL
digestauth	Enable / disable HTTP digest authentication. By default HTTP digest authentication is disabled.	OPTIONAL
strictusername	Enable / disable strict username check. By default strict username check is disabled.	OPTIONAL
dcache	Enable (enabled) or disable (disabled) data cache. By default data cache is enabled. For internal use only. Do not alter in any way.	OPTIONAL

EXAMPLE:

```
=>system config
upnp discovery      : enabled
mdap discovery      : enabled
drst support        : disabled
digest authentication : disabled
strict username     : disabled
dcache              : enabled
=>
```

system flush

Flush current SpeedTouch™ system configuration, i.e. the System password and the system config settings (dcache excluded).

Note To avoid unrestricted and unauthorized access to the SpeedTouch™, it is highly recommended to make sure it is protected by a SpeedTouch™ system password (via `:system setpassword`) and to change the password on a regular basis.

Note The flush command does not impact previously saved configurations.

SYNTAX:

```
system flush
```

EXAMPLE:

```
=>system flush  
Security notification: Password changed, use 'saveall' to make it permanent.  
=>
```

system reboot

Reboot the SpeedTouch™. Non-saved configuration settings are lost after reboot.

SYNTAX:

```
system reboot
```

EXAMPLE:

```
=>system reboot
.....
(lost session connectivity due to reboot)
.....
```

system reset

Reset the SpeedTouch™ to its factory default settings and reboot the device. All user and Service Provider specific settings and all saved configuration changes are lost after reboot.

SYNTAX:

```
system reset          factory yes/no = <{yes|no}>
                    proceed no/yes = <{no|yes}>
```

factory yes/no	Proceed with resetting the SpeedTouch™ device to its factory default settings (yes) or not (no). By default the system reboot command is discarded if no explicit positive confirmation is given.	REQUIRED
proceed no/yes	Confirmation for resetting the modem.	REQUIRED

EXAMPLE:

```
=>system reset
-----
!! WARNING !!
-----
The modem will be reset to (factory) defaults clearing all user (and ISP) settings.
Specifying <factory=yes> deletes user and ISP specific settings.
                        Connectivity with the ISP network might be lost.
        <factory=no>  deletes user specific settings only.
factory yes/no = no
proceed no/yes = no
:system reset factory yes/no=no proceed no/yes=no
=>
=>system reset
-----
!! WARNING !!
-----
The modem will be reset to (factory) defaults clearing all user (and ISP) settings.
Specifying <factory=yes> deletes user and ISP specific settings.
                        Connectivity with the ISP network might be lost.
        <factory=no>  deletes user specific settings only.
factory yes/no = yes
proceed no/yes = yes
:system reset factory yes/no=yes proceed no/yes=yes

.....
(lost session connectivity due to reboot)
.....
```

system setpassword

Set/change the current SpeedTouch™ system password.

Note To avoid unrestricted and unauthorized access to the SpeedTouch™, it is highly recommended to make sure it is protected by a SpeedTouch™ system password (via :system setpassword) and to change the password on a regular basis

SYNTAX:

```
system setpassword [userid = <string>]
                  password = <password>
```

userid	The new access userid.	OPTIONAL
password	The new access password.	REQUIRED

EXAMPLE:

```
=>system setpassword password=Sascha
Security notification: Password changed, use 'saveall' to make it permanent.
=>saveall
=>
```

RELATED COMMANDS:

`system clearpassword` Clear current system password.

system stats

Show/set the SpeedTouch™ cpu and memory statistics.

SYNTAX:

```
system stats [reset = <{no|yes}>]
```

reset	Reset cpu statistics.	OPTIONAL
-------	-----------------------	----------

EXAMPLE:

```
=>system stats
Cpu statistics:
-----
Maximum cpu load: 75%
Minimum cpu load: 4%
Average cpu load: 13%
Current cpu load: 10%

Memory statistics:
-----
CHIP memory      total/used/free (in KB): 891/750/141
Application memory total/used/free (in KB): 1519/1388/131
=>
```

Systemlog Commands

Contents

This chapter covers the following commands:

Topic	Page
systemlog show	374
systemlog send	375

systemlog show

Show syslog messages in the internal SpeedTouch™ Syslog message buffer.

SYNTAX:

```
systemlog show [fac = <supported facility name>]
               [sev = <supported severity name>]
               [hist = <{no|yes}>]
```

fac	Optionally, Specify the facility name of the syslog messages to show. Use one of the supported facility names (See “ Supported Syslog Facilities ” on page 418 for a listing of syslog facility names supported by the SpeedTouch™). Do not specify to show messages of any facility.	OPTIONAL
sev	Specify the lowest priority severity of the syslog messages to show. Specifying a severity actually means specifying to show the syslog messages with a severity as specified, and all messages with a higher severity. Use one of the supported severity names (See “ Supported Syslog Severities ” on page 419 for a listing of syslog facility names supported by the SpeedTouch™). Do not specify to show messages of any facility.	OPTIONAL
hist	The show messages over several SpeedTouch™ reboots (yes) or show only messages since latest startup (no). Do not specify to only show the recent messages.	OPTIONAL

EXAMPLE:

```
=>syslog msgbuf show fac=kern sev=emerg hist=yes
<0> SysUpTime: 14:45:43 KERNEL Controlled restart (after internal error or
explicit system reboot)
<0> SysUpTime: 02:58:18 KERNEL Controlled restart (after internal error or
explicit system reboot)
<0> SysUpTime: 04 days 04:52:37 KERNEL Controlled restart (after internal
error or explicit system reboot)
<0> SysUpTime: 00:00:41 KERNEL Controlled restart (after internal error or
explicit system reboot)
=>syslog msgbuf show fac=kern sev=warning hist=yes
<4> SysUpTime: 00:00:00 KERNEL Cold restart
<0> SysUpTime: 14:45:43 KERNEL Controlled restart (after internal error or
explicit system reboot)
<4> SysUpTime: 00:00:00 KERNEL Warm restart
<0> SysUpTime: 02:58:18 KERNEL Controlled restart (after internal error or
explicit system reboot)
<4> SysUpTime: 00:00:00 KERNEL Warm restart
<0> SysUpTime: 04 days 04:52:37 KERNEL Controlled restart (after internal
error or explicit system reboot)
<4> SysUpTime: 00:00:00 KERNEL Warm restart
<0> SysUpTime: 00:00:41 KERNEL Controlled restart (after internal error or
explicit system reboot)
=>
```

systemlog send

Send syslog messages from the internal SpeedTouch™ Syslog message buffer to a specified local or remote syslog server host.

SYNTAX:

```
systemlog send    [fac = <supported facility name>]
                  [sev = <supported severity name>]
                  [hist = <{no|yes}>]
                  dest = <ip-address>
```

fac	Optionally, Specify the facility name of the syslog messages to show. Use one of the supported facility names (See “ Supported Syslog Facilities ” on page 418 for a listing of syslog facility names supported by the SpeedTouch™). Do not specify to show messages of any facility.	OPTIONAL
sev	Specify the lowest priority severity of the syslog messages to show. Specifying a severity actually means specifying to show the syslog messages with a severity as specified, and all messages with a higher severity. Use one of the supported severity names (See “ Supported Syslog Severities ” on page 419 for a listing of syslog facility names supported by the SpeedTouch™).	OPTIONAL
hist	The show messages over several SpeedTouch™ reboots (yes) or show only messages since latest startup (no).	OPTIONAL
dest	The IP address of the remote host on the local or remote network, i.e. the collector's IP address, to send the syslog messages to.	REQUIRED

Note There will be no notification on whether the host has received the messages or not.

TD Commands

Contents

This chapter covers the following commands:

Topic	Page
td call	378

td call

Call a 'Trace & Debug' command.

Note For qualified personnel only.

SYNTAX:

td call	cmd = <string>
---------	----------------

cmd

Quoted 'Trace & Debug' command string.

REQUIRED

UPnP Commands

Contents

This chapter covers the following commands:

Topic	Page
upnp config	380
upnp flush	381
upnp list	382

upnp config

Configure UPnP parameter(s).

SYNTAX:

```
upnp config [maxage = <number{60-999999}>]
            [defcservice = <interface_name>]
            [writemode = <{full|natonly|readonly}>]
            [safenat = <{disabled|enabled}>]
            [preferredaddress = <ip-address>]
```

maxage	This parameter allows to configure how often the SpeedTouch™ sends a notification message to advertise its presence as an Internet Gateway Device (IGD) on the network. Setting this parameter to a low value will increase the number of packets sent over time on the network, but will make the state of the device more up to date. Default is 1800 seconds.	OPTIONAL
defcservice	This parameter allows to configure the connection service to be used by the DrSpeedTouch™ application during the troubleshooting process (only when several connection services are configured on the SpeedTouch™, else only the default connection service).	OPTIONAL
writemode	Choose the set of rules to limit remote access from UPnP. Choose between: <ul style="list-style-type: none"> • full: the host will accept all the UPnP SET and GET actions. • natonly: GET and NAT related SET actions will be accepted, all other actions will be ignored. • readonly: the UPnP control point will only be able to retrieve information, all the SET actions are ignored. 	OPTIONAL
safenat	Enable / disable check on safe NAT entries. If this check is enabled, all NAT create/delete requests for a LAN side IP address different from the source IP address of the UPnP message will be discarded.	OPTIONAL
preferredaddress	Preferred ip address for UPnP advertisements (enter 0.0.0.0 for none)	OPTIONAL

EXAMPLE:

```
=>upnp config
ssdp max-age          : 1800
default connection service : PPPoA_1
write mode           : full
safe nat entries      : disabled
preferred address     :
=>
```

upnp flush

Flushes the UPnP configuration (i.e. reset to default configuration).

SYNTAX:

```
upnp flush
```

EXAMPLE:

```
=>upnp config
ssdp max-age           : 4200
default connection service : pppoa
write mode              : natonly
safe nat entries        : enabled
preferred address       :
=>upnp flush
=>upnp config
ssdp max-age           : 1800
default connection service : pppoa
write mode              : full
safe nat entries        : disabled
preferred address       :
=>
```

upnp list

This command lists the devices and services currently offered by the SpeedTouch™.

Use this command to check whether a PPP connection is properly configured and thus advertised as a PPP service.

SYNTAX:

```
upnp list [verbose = <number{0-2}>]
```

verbose	Verbose level. Default is 1.	OPTIONAL
---------	------------------------------	----------

EXAMPLE:

```
=>upnp list
Advertising UPnP devices on address: 10.0.0.138
----- device: IGD.xml -----
++ Root Device: urn:schemas-upnp-org:device:InternetGatewayDevice:1
-- Service 1: urn:upnp-org:serviceId:layer3f
-- Service 2: urn:upnp-org:serviceId:lanhcm
-- Service 3: urn:upnp-org:serviceId:wancic
-- Service 4: urn:upnp-org:serviceId:wandsllc:PVC_1
-- Service 5: urn:upnp-org:serviceId:wanpppc:PPPoA_1
----- end -----
----- device: DRST.xml -----
++ Root Device: urn:schemas-upnp-org:device:DRST_Device:1
-- Service 1: urn:upnp-org:serviceId:layer3f_DRST
-- Service 2: urn:upnp-org:serviceId:devinfo_DRST
-- Service 3: urn:upnp-org:serviceId:lanhcm_DRST
-- Service 4: urn:upnp-org:serviceId:lane1c_DRST
-- Service 5: urn:upnp-org:serviceId:wancic_DRST
-- Service 6: urn:upnp-org:serviceId:wandsllc_DRST:PVC_1
-- Service 7: urn:upnp-org:serviceId:wanpppc_DRST:PPPoA_1
----- end -----
=>
```

USB Commands

Introduction

These commands are valid only for the SpeedTouch™ 530.

Contents

This chapter covers the following commands:

Topic	Page
usb add	384
usb config	385
usb delete	386
usb info	387
usb list	388

usb add

Adds a VP or VC cross-connection between the USB interface and the WAN interface.

SYNTAX:

```
usb add          wanvpi = <number{0-15}>
                 wanvci = <number{0-511}>
                 usbvpi = <number{0-15}>
                 usbvci = <number{0-511}>
                 [qos = <{default}>]
```

wanvpi	A number between 0 and 15. Represents the Virtual Path Identifier (VPI) of WAN interface.	REQUIRED
wanvci	A number between 0 and 511. Represents the Virtual Channel Identifier (VCI) of WAN interface.	REQUIRED
usbvpi	A number between 0 and 15. Represents the Virtual Path Identifier (VPI) of USB interface.	REQUIRED
usbvci	A number between 0 and 511. Represents the Virtual Channel Identifier (VCI) of USB interface.	REQUIRED
qos	The name of a qosbook entry.	OPTIONAL

RELATED COMMANDS:

usb delete	Deletes a cross-connection on the USB interface.
usb list	Shows all cross-connections on the USB interface.

usb config

Enable/disable the USB interface.

SYNTAX:

```
usb config [state = <{enabled|disabled}>]
```

state

Enable or disable the USB interface.

OPTIONAL

EXAMPLE:

```
=>usb config
USB port state = UP [configured]
=>
```

usb delete

Deletes a cross-connection on the USB interface.

SYNTAX:

```
usb delete wanvpi = <number{0-15}>
           [wanvci = <number{0-511}>]
```

wanvpi	A number between 0 and 15. Represents the Virtual Path Identifier (VPI) of WAN interface.	REQUIRED
wanvci	A number between 0 and 511. Represents the Virtual Channel Identifier (VCI) of WAN interface.	OPTIONAL

RELATED COMMANDS:

- [usb add](#) Adds a VP or VC cross-connection between the USB interface and the WAN interface
- [usb list](#) Shows all cross-connections on the USB interface.

usb info

Show USB properties/statistics

SYNTAX:

```
usb info
```

EXAMPLE:

```
=>usb info
MACaddr = 00-90-d0-02-8a-a3
Type = Device
Rate = FullSpeed
Power = SelfPowered
Properties
  DataTransferType = Bulk
  DataAlternateSetting = 1
  MajorRelease = 1
  MinorRelease = 10
  NumberOfConfigurations = 1
  DeviceClass = Vendor Specific
  DeviceSubclass = 0
  Protocol = 0
  Ep0MaxPacketSize = 64 Bytes
Statistics :
  ReservedBandwidth = Not Applicable
  Bytes tx = 406245
  Bytes rx = 238341
  Blocks tx = 6697
  Blocks rx = 4256
=>
```

usb list

Shows all cross-connections on the USB interface.

SYNTAX:

```
usb list
```

Wireless Commands

Introduction

The commands covered in this chapter can only be used with the SpeedTouch™570.

Contents

This chapter covers the following commands:

Topic	Page
wireless aclconfig	390
wireless aclentry	391
wireless aclflush	392
wireless acllist	393
wireless counters	394
wireless flush	395
wireless ifconfig	396
wireless scan	397
wireless status	398
wireless wepconfig	399

wireless aclconfig

Configures the Access Control List (ACL) mode.

SYNTAX:

```
wireless aclconfig [control = <{register|lock|unlock}>]
```

control	Select the Access Control List mode. Choose between: <ul style="list-style-type: none">• register button mode• lock/unlock mode Default setting is Register Button mode.	OPTIONAL
---------	--	----------

DESCRIPTION:

Register	A wireless client is only added to the ACL after using the Register Button.
Lock	The ACL is locked. No new entries are allowed in the ACL.
Unlock	Every wireless client is unconditionally added to the ACL.

EXAMPLE:

```
=>wireless aclconfig
Access control list : Register Button mode
=>wireless aclconfig control=lock
=>wireless aclconfig
Access control list : ACL locked (lock/unlock mode)
=>
```

wireless aclentry

Adds or deletes an entry in the ACL.

SYNTAX:

<code>wireless aclentry</code>	<code>hwaddr = <hardware-address></code> <code>action = <{allow deny delete}></code> <code>[name = <quoted string>]</code>	
<code>hwaddr</code>	The Ethernet MAC address of the ACL entry	REQUIRED
<code>action</code>	The action to be performed on ACL entry. Choose between: <ul style="list-style-type: none"> • allow • deny • delete 	REQUIRED
<code>name</code>	The name of the station	OPTIONAL

EXAMPLE:

```
=>wireless acllist
Station Name  MAC address      Allowed
Test          00:90:D0:01:47:DE : yes
=>wireless aclentry hwaddr=01:67:d0:23:56:cd action=allow name="Test2"
=>wireless acllist
Station Name  MAC address      Allowed
Test          00:90:D0:01:47:DE : yes
Test2        01:67:D0:23:56:CD : yes
=>wireless aclentry hwaddr=01:67:d0:23:56:cd action=deny name="Test2"
=>wireless acllist
Station Name  MAC address      Allowed
Test          00:90:D0:01:47:DE : yes
Test2        01:67:D0:23:56:CD : no
=>wireless aclentry hwaddr=01:67:d0:23:56:cd action=delete name="Test2"
=>wireless acllist
Station Name  MAC address      Allowed
Test          00:90:D0:01:47:DE : yes
=>
```

RELATED COMMANDS:

- `wireless aclflush` Flushes the ACL.
- `wireless acllist` Lists the ACL.

wireless aclflush

Flushes the ACL.

SYNTAX:

```
wireless aclflush
```

EXAMPLE:

```
=>wireless acllist
Station Name   MAC address    Allowed
Test           00:90:D0:01:47:DE : yes
Test2          01:67:D0:23:56:CD : yes
=>wireless aclflush
=>wireless acllist
The access control list is empty.
=>
```

RELATED COMMANDS:

wireless aclentry	Adds or deletes an entry in the ACL.
wireless acllist	Lists the ACL.

wireless acllist

Lists the ACL.

SYNTAX:

```
wireless acllist
```

EXAMPLE:

```
=>wireless acllist
Station Name   MAC address      Allowed
Test          00:02:2d:42:E5:2D : yes
Test2         01:67:D0:23:56:CD : yes
=>
```

RELATED COMMANDS:

`wireless aclentry` Adds or deletes an entry in the ACL.

`wireless aclflush` Flushes the ACL.

wireless counters

Displays the value of the SpeedTouch™570 WLAN counters.

SYNTAX:

```
wireless counters [clear <{no|yes}>]
```

clear	Clear the wireless counters	OPTIONAL
-------	-----------------------------	----------

EXAMPLE:

```
=>wireless counters
Tx Frames Count      : 497
Rx Frames Count      : 0
Station Not Allowed Count : 0
Station Unknown Count : 0
WEPICV Error Count   : 0
WEP Excluded count   : 0
=>wireless counters clear=yes
=>wireless counters
Tx Frames Count      : 0
Rx Frames Count      : 0
Station Not Allowed Count : 0
Station Unknown Count : 0
WEPICV Error Count   : 0
WEP Excluded count   : 0
=>
```

wireless flush

Flush the SpeedTouch™570 WLAN parameters, i.e. reset to the factory defaults.

SYNTAX:

```
wireless flush
```

EXAMPLE:

```
=>wireless ifconfig
Wireless SSID      : Schozach
Accept any SSID    : yes
Communication channel : 3 (selected)
RTS Threshold      : 1000
Short Retry Limit  : 4
Long Retry Limit   : 4
Beacon Period      : 100
=>wireless flush
=>wireless ifconfig
Wireless SSID      : Schozach
Accept any SSID    : no
Communication channel : 6 (auto)
RTS Threshold      : 2347
Short Retry Limit  : 4
Long Retry Limit   : 4
Beacon Period      : 100
=>
```

RELATED COMMANDS:

- | | |
|-----------------------------------|---|
| wireless ifconfig | Configures and shows the wireless interface parameters. |
| wireless scan | Scans for other APs in the area. |
| wireless status | Shows the status of the wireless interface. |

wireless ifconfig

Configures and shows the SpeedTouch™570 Wireless LAN (WLAN) interface parameters.

SYNTAX:

```
wireless ifconfig      [ssid = <quoted string>]
                       [any = <{off|on}>]
                       [channel = <{auto|1|2|3|4|5|6|7|8|9|10|11}>]
                       [rts = <number{0-2347}>]
```

ssid	The Service Set Identifier (SSID)	OPTIONAL
any	Accept stations with SSID 'any' or blank SSID. Default is no.	OPTIONAL
channel	The communication channel number. Choose between: <ul style="list-style-type: none"> • auto: the best communication channel is automatically selected by the SpeedTouch™ (default). • a number between 1 and 11. 	OPTIONAL
rts	A number between 0 and 2347. Represents the value of the RTS threshold. Setting this parameter to a small value causes RTS messages to be send more often, consuming more of the available bandwidth, therefore reducing the apparent throughput of other network packets. However, the more often RTS packets are sent, the quicker the system can recover from interference or collisions. Default is 2347.	OPTIONAL

EXAMPLE:

```
=>wireless ifconfig
Wireless SSID      : Schozach
Accept any SSID    : no
Communication channel : 6 (auto)
RTS Threshold      : 2347
Short Retry Limit  : 4
Long Retry Limit   : 4
Beacon Period      : 100
=>wireless ifconfig any=on channel=3 rts=1000
=>wireless ifconfig
Wireless SSID      : Schozach
Accept any SSID    : yes
Communication channel : 3 (selected)
RTS Threshold      : 1000
Short Retry Limit  : 4
Long Retry Limit   : 4
Beacon Period      : 100
=>
```

RELATED COMMANDS:

wireless scan	Scans for other APs in the area.
wireless status	Shows the status of the wireless interface.

wireless scan

Scans for other APs in the area. An overview of all the radio channels and their respective ratings is shown.

SYNTAX:

```
wireless scan
```

EXAMPLE:

```
=>wireless scan

Channel rating
1 : Good
2 : Good
3 : Good
4 : Good
5 : Good
6 : Recommended
7 : Good
8 : Good
9 : Good
10 : Good
11 : Good

Recommended channel : 6

=>
```

RELATED COMMANDS:

[wireless ifconfig](#)

Configures and shows the wireless interface parameters.

[wireless status](#)

Shows the status of the wireless interface.

wireless status

Shows the status of the SpeedTouch™570 WLAN interface.

SYNTAX:

```
wireless status
```

EXAMPLE:

```
=>wireless status  
wireless interface is running on channel 3 in clear-text  
=>wireless ifconfig channel=auto  
=>wireless status  
wireless interface is running on channel 6 in clear-text
```

RELATED COMMANDS:

- | | |
|-----------------------------------|---|
| wireless ifconfig | Configures and shows the wireless interface parameters. |
| wireless scan | Scans for other APs in the area. |

wireless wepconfig

Configures Wired Equivalent Privacy (WEP) secure access.

SYNTAX:

```
wireless wepconfig      [key = <password>]
                        [encryption = <{off|on}>]
```

key	The WEP key. In case of 64 bit encryption, the 40-bits WEPkey must consist of 10 hexadecimal digits. In case of 128 bit encryption, the 104-bits WEPkey must consist of 26 hexadecimal digits.	OPTIONAL
encryption	Encryption is turned on/off	OPTIONAL

EXAMPLE:

```
=>wireless wepconfig key=0123456789abcdef0123456789 encryption=on
=>wireless wepconfig
WEP encryption : enabled
WEP key       : 128-bit
=>
```


Abbreviations

The table below lists all the abbreviations used in the CLI Guide.

Abbreviation	Description
AAL5	ATM Adaption Layer 5
ACL	Access Control List
ADSL	Asymmetric Digital Subscriber Line
AH	Authentication Header
ARP	Address Resolution Protocol
ATM	Asynchronous Transfer Mode
ATMF	ATM Forum
BGP	Border Gateway Protocol
CA	Certificate Authority
CC	Continuity Check
CEP	Certificate Enrollment Protocol
CHAP	Challenge Handshake Authentication Protocol
CIDR	Classless Inter Domain Routing
CLI	Command Line Interface
CRL	Certificate Revocation List
DHCP	Dynamic Host Configuration Protocol
DN	Distinguished Name
DNS	Domain Name System
DSL	Digital Subscriber Line
EGP	Exterior Gateway Protocol
ESP	Encapsulating Security Payload
FTP	File Transfer Protocol
GRE	General Routing Encapsulation
GRP	Gateway Routing Protocol
HDLC	High-level Data Link Control
HTTP	HyperText Transfer Protocol
ICMP	Internet Control Message Protocol

Abbreviation	Description
IGD	Internet Gateway Device
IGMP	Internet Group Management Protocol
IKE	Internet Key Exchange
IMAP	Interim Mail Access Protocol
IMAP	Interactive Mail Access Protocol
IP	Internet Protocol
IPCP	Internet Protocol Control Protocol
IPCP	IP Payload Compression Protocol
IP oA	IP over ATM
IPSec	IP Security
IRC	Internet Relay Chat
ISDN	Integrated Services Digital Network
LAN	Local Area Network
LCP	Link Control Protocol
LDAP	Light-weight Directory Access Protocol
LIS	Logical IP Subnet
LLC	Logical Link Control
MAC	Medium Access Control
MD5	Message Digest 5
MER	MAC Encapsulated Routing
NAPT	Network Address and Port Translation
NAT	Network Address Translation
NBP	Name Binding Protocol
NLPID	Network Layer Protocol IDentifiers
NNTP	Network News Transfer Protocol
NTP	Network Time Protocol
OAM	Operation and Maintenance
OBC	On Board Controller
OID	Object IDentifier
PAP	Password Authentication Protocol
PBX	Private Branch Exchange

Abbreviation	Description
PIP	Packet Interception Point
PKCS	Public Key Cryptography Standard
PKI	Public Key Infrastructure
POP	Post Office Protocol
POTS	Plain Old Telephone Service
PPP	Point-to-Point Protocol
PPPoA	PPP over ATM
PPPoE	PPP over Ethernet
PPTP	Point-to-Point Tunneling Protocol
PSD	Power Spectral Density
PVC	Permanent Virtual Channel
QoS	Quality of Service
RIP	Routing Information Protocol
RTMP	RouTing Maintenance Protocol
RTSP	Real Time Stream Control Protocol
SAs	Security Associations
SHDSL	Symmetric High speed Digital Subscriber Line
SIP	Session Initiation Protocol
SMTP	Simple Mail Transfer Protocol
SNAP	Sub Network Access Protocol
SNMP	Simple Network Management Protocol
SNPP	Simple Network Paging Protocol
SNR	Signal-to-Noise Ratio
SNTP	Simple Network Time Protocol
SVC	Switched Virtual Channel
TCP	Transmission Control Protocol
TFTP	Trivial File Transfer Protocol
ToS	Type of Service
UDP	User Datagram Protocol
UPnP	Universal Plug and Play
URI	Uniform Resource Identifier

Abbreviations

Abbreviation	Description
URL	Uniform Resource Locator
VC	Virtual Channel
VCMUX	Virtual Channel MULTipleXing
VDSL	Very high speed Digital Subscriber Line
VLSM	Variable Length Subnet Masking
VP	Virtual Path
VPN	Virtual Private Networking
WAN	Wide Area Network
WCD	WAN Connection Device
WEP	Wired Equivalent Privacy
WINS	Windows Internet Naming Service
WLAN	Wireless LAN
WWW	World Wide Web
ZIS	Zone Information System

Syslog Messages

Introduction

This chapter lists the different Syslog messages.

Contents

This chapter covers the following topics:

Topic	Page
Auto-PVC Module	406
Configuration Module	406
DHCP Client Module	407
DHCP Relay Module	407
DHCP Server Module	408
Firewall Module	408
HTTP Module	408
Kernel Module	409
Linestate Module	409
Login Module	409
NAPT Module	409
PPP Module	410
PPTP Module	410
Routing Module	410
Software Module	411
UPnP Module	411

Auto-PVC Module

Facility	Severity	Contents
LOCAL5	WARNING	AUTOPVC script <script_name> failed
LOCAL5	WARNING	AUTOPVC script <script_name> failed
LOCAL5	WARNING	AUTOPVC script <script_name> (name1, qosb_name) failed
LOCAL5	WARNING	AUTOPVC script <script_name> (name1, qosb_name, name2) failed
LOCAL5	WARNING	AUTOPVC script 'autopvc_change_qos (itable[i].intf, name1, qosb_name) failed
LOCAL5	WARNING	AUTOPVC script <script_name> (name1, name2) failed

Configuration Module

Facility	Severity	Contents
USER	INFO	CONFIGURATION saved after running Embedded Setup Wizard
USER	INFO	CONFIGURATION saved by user <user_id>
USER	INFO	CONFIGURATION backup by user to file <filename>
USER	INFO	CONFIGURATION <conf_version> upgraded to version <version>
KERN	INFO	SYSTEM reset by user <user_id> to factory defaults: user settings deleted

DHCP CLient Module

Facility	Severity	Contents
LOCAL2	WARNING	DHCP lease ip-address <ip-address> bound to intf <intf_id>
LOCAL2	WARNING	DHCP intf <intf_id> renews lease ip-address <ip-address>
LOCAL2	WARNING	DHCP intf <intf_id> rebinds lease ip-address <ip-address> from server (<ip-address>)
LOCAL2	WARNING	DHCP offer received from <ip-address> (can be relay agent) for intf <intf_id>
LOCAL2	WARNING	DHCP server (<ip-address>) offers <ip-address> to intf <intf_id>
LOCAL2	WARNING	DHCP unable to configure ip address: <ip-address> (bootp-reply)
LOCAL2	WARNING	DHCP bootp lease ip-address <ip-address> bound to intf <intf_id> from server (<ip-address>)
LOCAL2	WARNING	DHCP <ip-address> already configured on intf <intf_id>: failure
LOCAL2	WARNING	DHCP <ip-address> (<ip-address>) set on intf <intf_id>: {failure ok}
LOCAL2	WARNING	DHCP <ip-address> deleted: {failure ok}

DHCP Relay Module

Facility	Severity	Contents
LOCAL2	WARNING	DHCP relay: Dropping boot rqs on interface <intf_id> due to invalid giaddr for server (<ip-address>)
LOCAL2	WARNING	DHCP relay: Dropping boot reply with invalid relay agent option from <intf_id>
LOCAL2	WARNING	DHCP relay: Dropping boot request containing the relay agent option from <intf_id>
LOCAL2	WARNING	DHCP relay: Dropping boot reply to unknown interface from <intf_id>
LOCAL2	WARNING	DHCP relay: Dropping boot reply to inactive interface <intf_id>
LOCAL2	WARNING	DHCP relay: Dropping boot reply to inactive interface <intf_id>
LOCAL2	WARNING	DHCP relay: Dropping boot request packet with spoofed giaddr field from <intf_id>
LOCAL2	WARNING	DHCP relay: Dropping boot request received on unknown interface from <intf_id>
LOCAL2	WARNING	DHCP relay: Dropping boot request on inactive interface <intf_id>
LOCAL2	WARNING	DHCP relay: Dropping boot request with invalid hops field on interface <intf_id>
LOCAL2	WARNING	DHCP relay: Dropping boot request with invalid giaddr on interface <intf_id>

DHCP Server Module

Facility	Severity	Contents
LOCAL2	WARNING	DHCP server: DHCP OFFER DHCPACK cannot be send due to invalid server identifier
LOCAL2	WARNING	DHCP server: DHCPACK cannot be send due to invalid server identifier
LOCAL2	WARNING	DHCP server: DHCPNAK cannot be send due to invalid server identifier
LOCAL2	WARNING	DHCP Auto DHCP: server detected on LAN, own dhcp server disabled
LOCAL2	WARNING	DHCP Auto DHCP: no server detected on LAN, SpeedTouch server started
LOCAL2	WARNING	DHCP Auto DHCP: search for DHCP server stopped
LOCAL2	WARNING	DHCP server up
LOCAL2	WARNING	DHCP server went down

Firewall Module

Facility	Severity	Contents
AUTH	WARNING	FIREWALL Hook: <hookname> Rule ID:<rule_id> Protocol: ICMP Src_ip: <ip_address> Dst_ip: <ip_address> ICMP message type: <message_type_name message_type_id > Action: <action>
AUTH	WARNING	FIREWALL Hook: <hookname> Rule ID:<rule_id> Protocol: <protocol_name> Src_ip_port: <ip-address:ip_port> Dst_ip_port: <ip-address:ip_port> Action: <action>

HTTP Module

Facility	Severity	Contents
AUTH	NOTICE	LOGOUT User <user_id> logged out on <connection_type> (<ip-address>)
AUTH	NOTICE	LOGOUT User <user_id> logged out on <connection_type>
AUTH	NOTICE	LOGOUT <session_name> session of user <user_id> killed (<ip-address>)
AUTH	NOTICE	LOGOUT <session_name> session of user <user_id> killed
AUTH	NOTICE	LOGIN User <user_id> tried to login on <connection_type> (from <ip-address>)
AUTH	NOTICE	LOGIN User <user_id> logged in on <connection_type> (from <ip-address>)
AUTH	NOTICE	LOGIN User logged in on <connection_type> (<ip-address>)
AUTH	NOTICE	LOGIN User <user_id> tried to log in on <connection_type>

Kernel Module

Facility	Severity	Contents
KERN	WARNING	KERNEL cold reset
KERN	WARNING	KERNEL warm reset
KERN	EMERG	KERNEL Controlled restart (after internal error or explicit system reboot)

Linestate Module

Facility	Severity	Contents
LOCAL5	NOTICE	xDSL linestate up (downstream: <bitrate_in> kbit/s, upstream: <bitrate_out> kbit/s)
LOCAL5	NOTICE	xDSL linestate down

Login Module

Facility	Severity	Contents
AUTH	NOTICE	LOGIN User <username> logged <in out> on telnet (<ip address>)
AUTH	NOTICE	LOGIN User <username> logged in on http (<ip-address>)

NAPT Module

Facility	Severity	Contents
LOCAL4	INFO	NAPT Protocol: <TCP UDP ICMP> Open port: <port> Helper: <app_name> => <""failed"" ""ok"">

PPP Module

Facility	Severity	Contents
LOCAL0	WARNING	PPP Link up (<intf name>)
LOCAL0	WARNING	PPP Link down (<intf name>)
AUTH	ERROR	PPP PAP authentication failed (<intf name>) [protocol reject]
AUTH	INFO	PPP PAP on intf <intf_id> no response to PAP authenticate-request
AUTH	NOTICE	PPP PAP remote user <remote_user_name> succesfull authenticated
AUTH	ERROR	PPP PAP authentication for remote user <remote_user> failed
AUTH	DEBUG	PPP PAP Authenticate Ack received
AUTH	DEBUG	PPP PAP Authenticate Nack received
AUTH	DEBUG	PPP PAP Authenticate Request sent
AUTH	ERROR	PPP CHAP authentication failed (<intf name>)
AUTH	ERROR	PPP CHAP authentication failed [protocol reject(server)]
AUTH	ERROR	PPP CHAP authentication failed [protocol reject(client)]
AUTH	DEBUG	PPP CHAP Receive challenge (rhost = <hostname>)
AUTH	INFO	PPP CHAP Chap receive success: authentication ok
AUTH	DEBUG	PPP CHAP Challenge Send (Id = <challenge_id>)
AUTH	DEBUG	PPP CHAP Send status response: {ack nack}

PPTP Module

Facility	Severity	Contents
LOCAL0	WARNING	PPTP tunnel (<Pbname>) up:(<ip addr>)
LOCAL0	WARNING	PPTP tunnel (<Pbname>) down:(<ip addr>)

Routing Module

Facility	Severity	Contents
SECURITY	INFO	LABEL Rule Id:<rule_id> Protocol: ICMP Src_ip: <ip_address> Dst_ip: <ip_address> ICMP message type: <message_type_name message_type_id > Label: <label_name>
SECURITY	INFO	LABEL Rule Id:<rule_id> Protocol: <protocol_name> Src_ip: <ip_address> Dst_ip: <ip_address> Label: <label_name>

Software Module

Facility	Severity	Contents
KERN	INFO	SOFTWARE Copying all data files from <file_location> to <file_location>
KERN	INFO	SOFTWARE Switchover going down in <number> seconds
KERN	INFO	SOFTWARE No passive software found, duplicating active software
KERN	INFO	SOFTWARE Duplication of active software failed

UPnP Module

Facility	Severity	Contents
WARNING	SECURITY	UPnP<ActionName> refused for ip=<ip_address>
NOTICE	SECURITY	UPnP <ActionName> (<Error_string>) for ip=<ip_address>

Supported Key Names

Contents

This chapter lists all the key names supported by the SpeedTouch™, that can be used for completing CLI command parameters.

Supported Internet Protocol (IP) Protocol Names

For more information on the listed IP protocols, see RFC 1340 or www.iana.org.

Protocol name	Number	Description
icmp	1	Internet Control Message Protocol (ICMP)
igmp	2	Internet Group Management Protocol (IGMP)
ipinip	4	IP in IP (encapsulation)
tcp	6	Transmission Control Protocol (TCP)
egp	8	Exterior Gateway Protocol (EGP)
udp	17	User Datagram Protocol (UDP)
rsvp	46	Reservation Protocol
gre	47	General Routing Encapsulation (GRE)
ah	51	Authentication Header (AH)
esp	50	Encapsulating Security Payload (ESP)
vines	83	Vines
ipcomp	108	IP Payload Compression Protocol (IPCP)

Supported TCP/UDP Port Names

For more information on the listed TCP/UDP port assignments, see RFC1340 or www.iana.org.

Port name	Number	TCP	UDP	Description
echo	7	Y	Y	Echo
discard	9	Y	Y	Discard
systat	11	Y	Y	Active Users
daytime	13	Y	Y	Daytime
qotd	17	Y	Y	Quote of the Day
chargen	19	Y	Y	Character Generator
ftp-data	20	Y	Y	File Transfer (Default data)
ftp	21	Y	Y	File Transfer (Control)
telnet	23	Y	Y	Telnet
smtp	25	Y	Y	Simple Mail Transfer Protocol (SMTP)
time	37	Y	Y	Time
nicname	43	Y	Y	Who Is
dns	53	Y	Y	Domain Name System (DNS)
domain	53	Y	Y	Domain Name System (DNS)
sql*net	66	Y	Y	Oracle SQL*NET
bootps	67	Y	Y	Bootstrap Protocol Server
bootpc	68	Y	Y	Bootstrap Protocol Client
tftp	69	Y	Y	Trivial File Transfer Protocol (TFTP)
gopher	70	Y	Y	Gopher
finger	79	Y	Y	Finger
www-http	80	Y	Y	World Wide Web (WWW) HTTP
kerberos	88	Y	Y	Kerberos
rtelnet	107	Y	Y	Remote Telnet Service
pop2	109	Y	Y	Post Office Protocol (POP) - Version 2
pop3	110	Y	Y	Post Office Protocol (POP) - Version 3
sunrpc	111	Y	Y	SUN Remote Procedure Call
auth	113	Y	Y	Authentication Service
sqlserver	118	Y	Y	SQL Services

Port name	Number	TCP	UDP	Description
nntp	119	Y	Y	Network News Transfer Protocol (NNTP)
sntp	123	Y	Y	Simple Network Time Protocol (SNTP)
ntp	123	Y	Y	Network Time Protocol (NTP)
ingres-net	134	Y	Y	INGRES-NET Service
netbios-ns	137	Y	Y	NETBIOS Naming System
netbios-dgm	138	Y	Y	NETBIOS Datagram Service
netbios-ssn	139	Y	Y	NETBIOS Session Service
imap2	143	Y	Y	Interim Mail Access Protocol (IMAP) v2
sql-net	150	Y	Y	SQL-NET
pcmail-srv	158	Y	Y	PCMail Server
snmp	161	Y	Y	Simple Network Management Protocol (SNMP)
snmptrap	162	Y	Y	SNMP Trap
bgp	179	Y	Y	Border Gateway Protocol (BGP)
irc-o	194	Y	Y	Internet Relay Chat (IRC) - o
at-rtmp	201	Y	Y	AppleTalk RouTing Maintenance Protocol (RTMP)
at-nbp	202	Y	Y	AppleTalk Name Binding Protocol (NBP)
at-echo	204	Y	Y	AppleTalk Echo
at-zis	206	Y	Y	AppleTalk Zone Information System (ZIS)
ipx	213	Y	Y	
imap3	220	Y	Y	Interactive Mail Access Protocol (IMAP) v3
clearcase	371	Y	Y	ClearCase
ulistserv	372	Y	Y	UNIX Listserv
ldap	389	Y	Y	Light-weight Directory Access Protocol (LDAP)
netware-ip	396	Y	Y	Novell Netware over IP
snpp	444	Y	Y	Simple Network Paging Protocol (SNPP)
ike	500	Y	Y	ISAKMP
exec	512	Y	-	Remote process execution
biff	512	-	Y	Used by mail system to notify users of new mail received
login	513	Y	-	Remote login a la telnet
who	513	-	Y	Maintains data bases showing who's logged in to machines on a local net and the load average of the machine

Supported Key Names

Port name	Number	TCP	UDP	Description
syslog	514	-	Y	Syslog
printer	515	Y	Y	Spooler
talk	517	Y	Y	Like Tenex link, but across machine
ntalk	518	Y	Y	NTalk
utime	519	Y	Y	UNIX Time
rip	520	-	Y	Local routing process (on site); uses variant of Xerox NS Routing Information Protocol (RIP)
timed	525	Y	Y	Timeserver
netwall	533	Y	Y	For emergency broadcasts
uucp	540	Y	Y	uucpd
uucp-rlogin	540	Y	Y	uucpd remote login
new-rwho	540	Y	Y	uucpd remote who is
rtsp	554	Y	Y	Real Time Stream Control Protocol (RTSP)

Supported ICMP Type Names

For more information on the listed ICMP type names, see RFC1340 or www.iana.org.

ICMP Type name	Number	Description
echo-reply	0	Echo Reply
destination-unreachable	3	Destination Unreachable
source-quench	4	Source Quench
redirect	5	Redirect
echo-request	8	Echo
router-advertisement	9	Router Advertisement
router-solicitation	10	Router Solicitation
time-exceeded	11	Time Exceeded
parameter-problems	12	Parameter problems
timestamp-request	13	Timestamp
timestamp-reply	14	Timestamp Reply
information-request	15	Information Request
information-reply	16	Information Reply
address-mask-request	17	Address Mask Request
address-mask-reply	18	Address Mask Reply

Supported Syslog Facilities

For more information on the listed Syslog facilities, see RFC3164.

Facility Name	Hierarchy Code	Syslog facility (listed according descending importance)
kern	0	Kernel messages
user	8	User-level messages
mail	16	Mail system
daemon	24	System daemons
auth	32	Authorization messages
syslog	40	Syslog daemon messages
lpr	48	Line printer subsystem
news	56	Network news subsystem
uucp	64	UUCP subsystem
cron	72	Clock daemon
security	80	Security messages
ftp	88	FTP daemon
ntp	96	NTP subsystem
audit	104	Log audit
logalert	112	Log alert
clock	120	Clock daemon
local0	128	Local use messages
local1	136	
local2	144	
local3	152	
local4	160	
local5	168	
local6	176	
local7	184	
all	-	All facilities (SpeedTouch™ specific facility parameter value.

Supported Syslog Severities

For more information on the listed Syslog severities, see RFC3164.

Severity Name	Hierarchy Code	Syslog severity (listed according descending importance)
emerg	0	Emergency conditions, system unusable
alert	1	Alert conditions, immediate action is required
crit	2	Critical conditions
err	3	Error conditions
warning	4	Warning conditions
notice	5	Normal but significant conditions
info	6	Informational messages
debug	7	Debug-level messages



www.speedtouch.com



Built for excellence

speedtouch™