

speedtouch™

SpeedTouch™ 706 WL/780 WL

Residential ADSL Gateway



CLI Reference Guide



A THOMSON BRAND

SpeedTouch™

706 WL

780 WL 780 WL

CLI Reference Guide

Copyright

Copyright ©1999-2006 THOMSON. All rights reserved.

Distribution 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.

Thomson Telecom Belgium
Prins Boudewijnlaan, 47
B-2650 Edegem
Belgium

www.speedtouch.com

Trademarks

The following trademarks are used in this document:

- ▶ SpeedTouch™ is a trademark of THOMSON.
- ▶ Bluetooth® word mark and logos are owned by the Bluetooth SIG, Inc.
- ▶ Ethernet™ is a trademark of Xerox Corporation.
- ▶ Wi-Fi® and the Wi-Fi logo are registered trademarks of the Wi-Fi Alliance. "Wi-Fi CERTIFIED", "Wi-Fi ZONE", "Wi-Fi Alliance", their respective logos and "Wi-Fi Protected Access" are trademarks of the Wi-Fi Alliance.
- ▶ UPnP™ is a certification mark of the UPnP™ Implementers Corporation.
- ▶ Microsoft®, MS-DOS®, Windows® and Windows NT® are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.
- ▶ Apple® and Mac OS® are registered trademarks of Apple Computer, Incorporated, registered in the United States and other countries.
- ▶ UNIX® is a registered trademark of UNIX System Laboratories, Incorporated.
- ▶ Adobe®, the Adobe logo, Acrobat and Acrobat Reader are trademarks or registered trademarks of Adobe Systems, Incorporated, registered in the United States and/or other countries.
- ▶ Netscape® and Netscape Navigator® are registered trademarks of Netscape Communications Corporation.

Other brand and product names may be trademarks or registered trademarks of their respective holders.

Document Information

Status: v1.0 (April 2006)

Reference:E-DOC-CTC-20060414-0005

Short Title: CLI Reference Guide ST706 WL/780 WL R6.1

Contents

About this Document	1
Root Commands	3
ping.....	4
traceroute.....	5
ADSL Commands.....	7
adsl config.....	8
adsl info.....	9
ATM Commands	11
atm flush	13
atm ifadd.....	14
atm ifattach.....	15
atm ifconfig.....	16
atm ifdelete.....	18
atm ifdetach.....	19
atm iflist	20
atm bundle add	21
atm bundle attach	22
atm bundle config	23
atm bundle clear.....	24
atm bundle delete	25
atm bundle detach	26
atm bundle flush	27
atm bundle ifadd	28
atm bundle ifconfig	29
atm bundle ifdelete	30
atm bundle list.....	31
atm cac config	32
atm cac list.....	33
atm cac overbooking.....	34
atm debug aal5stats.....	35
atm debug gstats.....	36
atm debug portstats.....	37
atm oam config	38
atm oam list.....	39
atm oam modify.....	40

atm oam ping	42
atm oam cc list	43
atm oam cc modify	44
atm oam cc send	45
atm oam vclb add.....	46
atm oam vclb del.....	47
atm oam vclb list.....	48
atm phonebook add.....	49
atm phonebook autolist.....	50
atm phonebook delete.....	51
atm phonebook flush.....	52
atm phonebook list	53
atm qosbook add.....	54
atm qosbook config	55
atm qosbook ctdadd	56
atm qosbook ctddelete	58
atm qosbook ctlist.....	59
atm qosbook delete.....	60
atm qosbook flush.....	61
atm qosbook list	62
AutoPVC Commands.....	63
autopvc config	64
autopvc list.....	65
Config Commands	67
config delete	68
config dump.....	69
config flush	70
config list.....	71
config load	72
config save.....	73
Connection Commands.....	75
connection appconfig	76
connection appinfo	78
connection applist.....	79
connection bind.....	80
connection bindlist.....	81
connection clean	82

connection clear	83
connection config.....	84
connection debug.....	85
connection describe.....	86
connection flush.....	87
connection info.....	88
connection list	89
connection refresh	90
connection stats	91
connection timerclear	92
connection timerconfig.....	93
connection unbind	94
CWMP Commands.....	95
cwmp config	96
cwmp debug traceconfig.....	98
cwmp server config.....	99
Debug Commands	101
debug exec.....	102
DHCP Commands.....	103
dhcp client flush	105
dhcp client ifadd.....	106
dhcp client ifattach.....	107
dhcp client ifconfig.....	108
dhcp client ifdelete	109
dhcp client ifdetach	110
dhcp client iflist	111
dhcp client ifrenew.....	113
dhcp client debug clear.....	115
dhcp client debug stats.....	116
dhcp client debug traceconfig.....	117
dhcp client roptions add.....	118
dhcp client roptions delete.....	119
dhcp client roptions list	120
dhcp client roptions optionlist	121
dhcp client toptions add	122
dhcp client toptions delete	123
dhcp client toptions list.....	124

dhcp client txoptions optionlist.....	125
dhcp relay add	126
dhcp relay debug stats.....	127
dhcp relay debug traceconfig.....	128
dhcp relay config.....	129
dhcp relay delete	130
dhcp relay flush	131
dhcp relay ifconfig.....	132
dhcp relay iflist	134
dhcp relay list	135
dhcp relay modify	136
dhcp relay ruleadd	137
dhcp relay ruledelete	138
dhcp rule add.....	139
dhcp rule debug traceconfig	140
dhcp rule delete.....	141
dhcp rule flush.....	142
dhcp rule list	143
dhcp server config.....	144
dhcp server flush.....	145
dhcp server policy	146
dhcp server debug clear	147
dhcp server debug stats	148
dhcp server debug traceconfig	150
dhcp server lease add	151
dhcp server lease delete	153
dhcp server lease flush	154
dhcp server lease list	155
dhcp server option flush	156
dhcp server option instadd	157
dhcp server option instdelete	158
dhcp server option instlist	159
dhcp server option tmpladd	160
dhcp server option tmpldelete	161
dhcp server option tmpllist	162
dhcp server pool add	163
dhcp server pool config	164
dhcp server pool delete	166
dhcp server pool flush	167
dhcp server pool list.....	168
dhcp server pool optadd.....	169

dhcp server pool optdelete.....	170
dhcp server pool rtadd.....	171
dhcp server pool rtdelete.....	172
dhcp server pool ruleadd.....	173
dhcp server pool ruledelete.....	174
DNS Commands.....	175
dns client config	176
dns client dnsadd	177
dns client dnsdelete	178
dns client dnslist.....	179
dns client flush	180
dns client nslookup	181
dns server config.....	182
dns server flush.....	183
dns server debug clear	184
dns server debug stats.....	185
dns server debug spoof clear.....	186
dns server debug spoof getaddress	187
dns server debug spoof getflags	188
dns server debug spoof list	189
dns server debug spoof update	190
dns server host add.....	191
dns server host delete.....	192
dns server host flush.....	193
dns server host list.....	194
dns server route add.....	195
dns server route delete.....	196
dns server route flush.....	197
dns server route list	198
DSD Commands.....	199
dsd config	200
dsd debug config.....	201
dsd debug connection list	202
dsd debug proxy	203
dsd debug recycling.....	204
dsd debug stats	205
dsd intercept config	206
dsd syslog config	207

dsd syslog list.....	208
dsd urlfilter config	209
dsd urlfilter rule add.....	210
dsd urlfilter rule delete.....	211
dsd urlfilter rule flush.....	212
dsd urlfilter rule list.....	213
dsd urlfilter rule modify.....	214
DynDNS Commands.....	215
dyndns add	216
dyndns delete	217
dyndns flush	218
dyndns list.....	219
dyndns modify.....	220
dyndns host add.....	222
dyndns host delete.....	223
dyndns host flush.....	224
dyndns host list	225
dyndns service list.....	226
dyndns service modify.....	227
Env Commands.....	229
env flush.....	230
env get.....	231
env list.....	232
env set.....	233
env unset.....	234
Eth Commands.....	235
eth ifadd	237
eth ifattach	238
eth ifconfig	239
eth ifdelete	240
eth ifdetach	241
eth iflist.....	242
eth flush	243
eth bridge clear.....	244
eth bridge config	245
eth bridge dynvlan actlist	246
eth bridge dynvlan add.....	247

eth bridge dynvlan config.....	248
eth bridge dynvlan delete.....	249
eth bridge dynvlan flush.....	250
eth bridge dynvlan list	251
eth bridge flush	252
eth bridge ifadd	253
eth bridge ifattach	254
eth bridge ifconfig	255
eth bridge ifdelete	257
eth bridge ifdetach	258
eth bridge iflist.....	259
eth bridge macadd	260
eth bridge macdelete	261
eth bridge maclist.....	262
eth bridge mcdadd	263
eth bridge mcddelete	264
eth bridge mcclist	265
eth bridge rule add.....	266
eth bridge rule delete.....	267
eth bridge rule flush.....	268
eth bridge rule list	269
eth bridge vlan ifadd.....	270
eth bridge vlan ifconfig.....	271
eth bridge vlan ifdelete	272
eth bridge vlan iflist	273
eth device ifconfig	274
eth device iflist.....	275
eth switch group flush	276
eth switch group list.....	277
eth switch group move	278
eth switch mirror capture	279
eth switch mirror egress	280
eth switch mirror ingress.....	281
eth switch share add	282
eth switch share delete	283
eth switch share list	284
eth vlan add	285
eth vlan delete	286
eth vlan flush	287
eth vlan list.....	288

Expr Commands.....	289
expr add	290
expr delete	293
expr flush	295
expr list.....	296
expr modify.....	302
Firewall Commands.....	305
firewall config	306
firewall clear	308
firewall list.....	309
firewall chain add.....	310
firewall chain delete.....	311
firewall chain flush.....	312
firewall chain list	313
firewall debug clear.....	314
firewall debug stats.....	315
firewall debug traceconfig.....	316
firewall level add	317
firewall level delete	318
firewall level flush	319
firewall level list.....	320
firewall level modify.....	321
firewall level set.....	322
firewall rule add.....	323
firewall rule delete.....	324
firewall rule flush.....	325
firewall rule list.....	326
firewall rule modify.....	328
firewall rule debug clear.....	329
firewall rule debug stats	331
firewall rule debug traceconfig	332
GRP Commands.....	333
grp config.....	334
grp flush.....	335
grp rtlist.....	336
grp rip config	337
grp rip flush	339
grp rip ifconfig	340

grp rip show.....	341
Hostmgr Commands	343
hostmgr add	344
hostmgr clear.....	346
hostmgr config	347
hostmgr delete	348
hostmgr flush	349
hostmgr list.....	350
IDS Commands	351
ids clear	352
ids config.....	353
ids parser list	354
ids parser modify	355
ids pattern clear.....	356
ids pattern list.....	357
ids pattern stats.....	358
ids signature list.....	359
ids signature modify	360
ids threshold clear.....	361
ids threshold list.....	362
ids threshold modify.....	363
IGMP Commands	365
igmp host config	366
igmp host flush.....	367
igmp host list.....	368
igmp host ifconfig	369
igmp host iflist.....	370
igmp host debug clear	371
igmp host debug stats.....	372
igmp proxy config	373
igmp proxy flush	375
igmp proxy grouplist	376
igmp proxy ifconfig	377
igmp proxy iflist	378
igmp proxy mbslist	379
igmp proxy debug clear.....	380
igmp proxy debug stats.....	381

igmp proxy debug traceconfig.....	382
Interface Commands	383
interface list	384
IP Commands	387
ip arpadd.....	389
ip arpdelete.....	390
ip arplist	391
ip config	392
ip flush.....	394
ip ifadd	395
ip ifattach	396
ip ifconfig	397
ip ifdelete	399
ip ifdetach	400
ip iflist.....	401
ip ifwait.....	402
ip ipadd	403
ip ipconfig	404
ip ipdelete	405
ip iplist.....	406
ip mcast rtadd.....	407
ip mcast rtdelete.....	408
ip mcast rtlist	409
ip mcast flush	410
ip rtadd	411
ip rtdelete.....	412
ip rtlist	413
ip auto flush.....	414
ip auto ifadd.....	415
ip auto ifattach.....	416
ip auto ifconfig.....	417
ip auto ifdelete.....	419
ip auto ifdetach.....	420
ip auto iflist	421
ip debug httpprobe	422
ip debug sendto.....	423
ip debug stats	425
ip debug traceconfig	426

IPQoS Commands	429
ipqos config	430
ipqos list.....	432
ipqos ef config	433
ipqos ef list.....	434
ipqos ef stats.....	435
ipqos queue clear	436
ipqos queue config.....	437
ipqos queue list	439
ipqos queue stats	440
Label Commands	441
label add.....	442
label delete.....	443
label flush.....	444
label list.....	445
label modify.....	446
label chain add	448
label chain delete	449
label chain flush	450
label chain list.....	451
label rule add	452
label rule delete	455
label rule flush	457
label rule list	458
label rule modify	460
label rule debug clear	461
label rule debug stats.....	462
label rule debug traceconfig.....	463
Language Commands.....	465
language config.....	466
language delete	467
language list	468
MBUS Commands	469
mbus config.....	470
mbus client config.....	471
mbus client exec.....	472
mbus client register	474

mbus client test	475
mbus debug stats.....	476
mbus debug traceconfig.....	477
mbus listobjects	478
mbus listtypes	479
mbus loadobjects.....	480
mbus unloadobjects.....	481
MEMM Commands	483
memm debug lock traceconfig	484
memm debug traceconfig	485
memm listobjects.....	486
memm stats.....	487
MLP Commands	489
mlp flush	490
mlp import	491
mlp debug export.....	492
mlp debug stats.....	493
mlp debug traceconfig.....	494
mlp privilege add	495
mlp privilege addzone	497
mlp privilege config	498
mlp privilege delete	499
mlp privilege list.....	500
mlp privilege removezone.....	502
mlp role add.....	503
mlp role addpriv.....	504
mlp role config	505
mlp role delete.....	506
mlp role list.....	507
mlp role removepriv	508
NAT Commands.....	509
nat config	510
nat flush	511
nat ifconfig	512
nat iflist.....	513
nat mapadd.....	514
nat mapdelete.....	517

nat maplist	518
nat tmpladd	519
nat tmpldelete	522
nat tmpllist	523
nat tmplinst	524
PPP Commands.....	525
ppp flush	526
ppp ifadd	527
ppp ifattach	528
ppp ifconfig	530
ppp ifdelete	534
ppp ifdetach	535
ppp iflist	537
ppp ifscan	538
ppp rtadd	539
ppp rtdelete	541
ppp relay flush	543
ppp relay ifadd	544
ppp relay ifconfig	545
ppp relay ifdelete	546
ppp relay iflist	547
ppp relay sesslist	548
PPTP Commands.....	549
pptp ifadd	550
pptp flush	551
pptp list	552
pptp profadd	553
pptp profdelete	554
pptp proflist	555
Script Commands	557
script add	558
script delete	559
script flush	560
script list	561
script run	562
Service Commands.....	563

service host assign.....	564
service host config	565
service host add	566
service host delete	567
service host disable.....	568
service host flush	569
service host list.....	570
service host stats.....	571
service host triggerlist	572
service host rule add.....	573
service host rule delete.....	574
service system ifadd	575
service system ifdelete	576
service system ipadd	577
service system ipdelete	578
service system list	579
service system mapadd	581
service system mapdelete	582
service system modify.....	583
SNMP Commands.....	585
snmp config	586
snmp get	587
snmp getnext.....	588
snmp walk.....	589
snmp community add.....	590
snmp community delete.....	591
snmp community list	592
snmp community modify	593
snmp ifadd	594
snmp ifdelete	595
snmp ifattach	596
snmp ifdetach	597
snmp ifconfig.....	598
snmp iflist	599
SNTP Commands.....	601
sntp add	602
sntp config	603
sntp delete	604

sntp flush	605
sntp list.....	606
Software Commands.....	607
software download	608
software upgrade	609
software version.....	610
System Commands	611
system config	612
system flush.....	614
system locale.....	615
system reboot.....	617
system reset.....	618
system timedreboot.....	619
system debug autosave.....	620
system debug stats	621
system ra config.....	622
system rtc synchronize	623
system rtc settime	624
Systemlog Commands	625
systemlog flush	626
systemlog show	627
systemlog send	628
Upgrade Commands.....	629
upgrade config	630
upgrade start	631
upgrade debug traceconfig	632
upgrade debug sesslist.....	633
upgrade profile add.....	634
upgrade profile modify	635
upgrade profile delete.....	636
upgrade profile list.....	637
UPnP Commands	639
upnp config.....	640
upnp flush.....	641
upnp list	642

User Commands	643
user add	644
user config	646
user delete	648
user flush	649
user list.....	650
user rights.....	652
Voice Commands.....	653
voice config.....	655
voice state.....	657
voice codec list.....	658
voice codec config.....	659
voice country config.....	660
voice country list	661
voice dialplan add	662
voice dialplan delete	664
voice dialplan modify.....	665
voice dialplan list.....	667
voice dialplan flush	668
voice fax config	669
voice fxoport config	670
voice fxsport config.....	671
voice mgcp config	672
voice pb add	674
voice phonebook add.....	675
voice phonebook delete.....	676
voice phonebook list.....	677
voice profile add.....	678
voice profile delete.....	680
voice profile modify	681
voice profile list	683
voice profile flush.....	684
voice qos list.....	685
voice qos config	686
voice services config.....	687
voice services list	689
voice services provision	690
voice services withdraw	691
voice services activate	692

voice services deactivate	693
voice services assign	694
voice services assign_pxsc.....	695
voice services flush	696
voice sip config.....	697
voice stats list	699
voice stats reset.....	700
Wireless Commands.....	701
wireless ifconfig	702
wireless reset.....	704
wireless debug stats	706
wireless macacl add.....	707
wireless macacl config.....	708
wireless macacl delete.....	709
wireless macacl flush.....	710
wireless macacl list	711
wireless macacl modify	712
wireless macacl register	713
wireless multiuni add.....	714
wireless multiuni delete.....	715
wireless multiuni flush.....	716
wireless multiuni list.....	717
wireless multiuni scanresults.....	718
wireless qos config	719
wireless qos apacconfig	720
wireless qos staacconfig.....	721
wireless secmode config	722
wireless secmode wep.....	723
wireless secmode wpa-psk.....	724
wireless secmode wpa.....	725
wireless stations list.....	726
wireless wds add.....	727
wireless wds config.....	728
wireless wds delete.....	729
wireless wds flush.....	730
wireless wds list	731
wireless wds scanresults	732
Abbreviations	733

System Logging Messages 737

Auto-PVC Module.....	738
Configuration Module.....	738
DHCP Client Module.....	738
DHCP Relay Module.....	739
DHCP Server Module	739
Dyndns Module	740
Firewall Module.....	740
LOGIN Module.....	741
Kernel Module	741
Linestate Module.....	741
NAPT Module	741
PPP Module.....	742
PPTP Module	742
RIP Module.....	743
Routing Module.....	744
Session Module.....	744
SNTP Module.....	744
Software Module.....	745
UPnP Module.....	745
VOIP Module.....	745
.....	745

Supported Key Names 747

Supported IP Protocols.....	748
Supported TCP/UDP Port Names.....	749
Supported ICMP Type Names.....	752
Supported Facilities	753
Supported Severities	754
IP Precedence	754
Differentiated Services Code Point (DSCP).....	755

About this Document

Introduction

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

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

For the fastidious user, this Reference Guide aims to be a handbook for advanced and detailed configuration and troubleshooting of the SpeedTouch™ via its character based CLI command set.

Applicability

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

- ▶ SpeedTouch™706 WL(i) (R6.1)
- ▶ SpeedTouch™780 WL(i) (R6.1)

For readability, all are referred to as SpeedTouch™ throughout this document unless a specific variant is concerned.

Contents

The CLI Reference Guide describes all the available CLI commands of the SpeedTouch™ 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 necessary completed by a description of the possible impact on the user and/or the SpeedTouch™.
- ▶ 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.

Other Information

For more information on the SpeedTouch™ CLI, various access methods, general manipulations to navigate through the CLI refer to the SpeedTouch™.

For a description on the functional use of the CLI commands see the corresponding SpeedTouch™ configuration guides and application notes.

About this Document

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.

Symbols

The following symbols are used in this CLI Reference Guide:

 A **note** provides additional information about a topic.

 A **tip** provides an alternative method or shortcut to perform an action.

 A **caution** warns you about potential problems or specific precautions that need to be taken.

Conventions

The following conventions are used to present the CLI commands:

- ▶ Vertical bars (|) separate alternative, mutually exclusive, elements.
- ▶ Square brackets [] indicate optional elements.
- ▶ Braces {} indicate a required choice.
- ▶ Braces within square brackets [{}] indicate a required choice within an optional element.
- ▶ <string> indicates that an alphanumerical string without blanks must be used.
- ▶ <quoted string> indicates that an alphanumerical string with blanks can be used. The string must be enclosed in quotes “ ”.

Documentation and software updates

THOMSON continuously develops new solutions, but is also committed to improve its existing products.

For suggestions regarding this document, please contact documentation.speedtouch@thomson.net.

For more information on THOMSON's latest technological innovations, documents and software releases, visit us at: www.speedtouch.com

Root Commands

From the root prompt, you can choose one of the following commands:

Command	Description
help	Displays the help information.
?	
menu	Displays the menu.
exit	Exits the shell.
..	Exits group selection.
saveall	Saves current configuration.
ping	Send ICMP ECHO_REQUEST packets.
traceroute	Send ICMP/UDP packets to trace the ip path.



To obtain help on a specific command group:

- ▶ type **help**, followed by the name of the command group, and then press ENTER, or
- ▶ type the name of the command group, press ENTER, and then type **help**.

EXAMPLE:

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

config      : Display/Modify firewall configuration.
list        : Display firewall configuration.
clear       : Clear firewall configuration.

Following command groups are available :

chain       debug       level       rule

<Administrator>=>
```

ping

Send ICMP ECHO_REQUEST packets.

SYNTAX:

```
ping          proto = <{ip|atm}>
             addr = <ip-address>
             dest = <string>
             [count = <number{0-1000000}>]
             [size = <number{28-20028}>]
             [interval(ms) = <number{100-1000000}>]
             [DF-bit = <{disabled|enabled}>] [srcaddr = <ip-address>]
```

where:

proto	The interface type to be used. Choose between: ▶ ip ▶ atm	REQUIRED
addr	The destination IP address.	REQUIRED
dest	The destination address for the request.	REQUIRED
count	A number between 1 and 1000000. Represents the number of pings to send. The default is 5 .	OPTIONAL
size	A number between 0 and 20000. Represents the size of the ping payload(s).	OPTIONAL
interval	A number between 100 and 1000000. Represents the interval in milliseconds between packets. The default is 100 .	OPTIONAL
DF-bit	Set the Don't Fragment bit (or leave unset) in the IP header of ping. The default is disabled .	OPTIONAL
scraddr	The IP source address to use.	OPTIONAL

EXAMPLE:

```
=>:ping proto=ip addr=192.168.1.70
Legend : Ping successful(!)
          Ping Timeout(..)
          Hit ctrl-g to abort...
!!!!!
--- ping statistics ---
5 packet(s) transmitted, 5 successful, 0% loss
rtt min/avg/max = 1 / 2 / 4 ms
=>
```

traceroute

Send ICMP/UDP packets to trace the IP path.

SYNTAX:

```
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 = <{disabled | enabled}>]
```

where:

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 TTL. The default is 3 .	OPTIONAL
size	A number between 1 and 20000 (bytes). Represents the size of the traceroute packet(s). The default is 1 .	OPTIONAL
interval	A number between 1000 and 60000 (milliseconds). Represents the intermediate interval between two packets. The default is 1000 .	OPTIONAL
maxhops	A number between 1 and 255. Represents the maximum number of routers through which a packet can pass. The 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 maximum number of consecutive timeouts allowed before terminating a traceroute request. The default is 5 .	OPTIONAL
type	The type of traceroute packet(s). Choose between: ▶ icmp ▶ udp . The default is icmp .	OPTIONAL
utime	Display time in useconds (enabled) or not (disabled). The default is enabled .	OPTIONAL

Root Commands

EXAMPLE:

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

ADSL Commands

Introduction

This chapter describes the commands of the **adsl** command group.

Contents

This chapter covers the following commands:

adsl config	Show/set the Asymmetric Digital Subscriber Line (ADSL) configuration.	8
adsl info	Display the ADSL statistics and information about the DSL line status.	9

adsl config

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

SYNTAX:

```
adsl config [opermode = <{multimode | multi_adsl2 | multi_readsl2
    | multi_adsl2plus}>]
    [trace = <{disabled | enabled}>]
```

where:

opermode	The operational mode of the SpeedTouch™. Choose between: <ul style="list-style-type: none">▶ multimode▶ multi_adsl2▶ multi_readsl2▶ multi_adsl2plus The default is <i>multi_adsl2plus</i> .	OPTIONAL
trace	Enable or disable ADSL tracing. The default is <i>enabled</i> .	OPTIONAL

EXAMPLE:

The example below shows the default configuration for a SpeedTouch™ ADSL/POTS variant:

```
=>adsl config
ADSL configuration:
    opermode = multi_adsl2plus
    trace = on
    modemoption = 00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00
=>
```

adsl info

Display the ADSL statistics and information about the DSL line status.

Although the same command is used 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™ ADSL/POTS variant):

```
=>adsl info
Modemstate      : up
Operation Mode   : G.992.1 Annex A
Channel Mode     : interleaved
Number of resets : 1

Vendor           Local          Remote
Country          : 0f            00
Vendor           : TMMB
VendorSpecific    : 0000          0000
StandardRevisionNr : 00            00

Margin [dB]       Downstream    Upstream
Attenuation [dB] : 26.0          18.0
OutputPower [dBm] : 0.0           2.0
                           4.0           11.5

Available Bandwidth Cells/s      Kbit/s
Downstream        : 14490         6144
Upstream         : 1509          640

Transfer statistics
Errors
  Received FEC   : 0
  Received CRC   : 0
  Received HEC   : 0
  Transmitted FEC: 0
  Transmitted CRC: 0
  Tranmsitted HEC: 0

Near end failures since reset
  Loss of frame: 0 failures
  Loss of signal: 0 failures
  Loss of power: 0 failures
  Errorred seconds: 0 seconds
Near end failures last 15 minutes
  Loss of frame: 0 seconds
  Loss of signal: 0 seconds
  Loss of power: 0 seconds
  Errorred seconds: 0 seconds
Near end failures current day
  Errorred seconds: 0 seconds
Near end failures previous day
  Errorred seconds: 0 seconds
=>
```


ATM Commands

Introduction

This chapter describes the commands of the **atm** command group.

Contents

This chapter covers the following commands:

atm flush	Flush all Asynchronous Transfer Mode (ATM) interfaces.	13
atm ifadd	Create a new ATM interface.	14
atm ifattach	Attach a ATM interface.	15
atm ifconfig	Configure an ATM interface.	16
atm ifdelete	Delete an ATM interface.	18
atm ifdetach	Detach an ATM interface.	19
atm iflist	Display the ATM interfaces.	20
atm bundle add	Add a new bundle of interfaces.	21
atm bundle attach	Attach a bundle of interfaces.	22
atm bundle config	Modify a bundle of interfaces.	23
atm bundle clear	Clear the bundle statistics.	24
atm bundle delete	Delete a bundle of interfaces.	25
atm bundle detach	Detach a bundle of interfaces.	26
atm bundle flush	Flush all bundles.	27
atm bundle ifadd	Add an interface on a bundle.	28
atm bundle ifconfig	Configure an interface from a bundle.	29
atm bundle ifdelete	Remove an interface from a bundle.	30
atm bundle list	Display the current bundles.	31
atm cac config	Configure the ATM connection admission control.	32
atm cac list	Display all the CAC parameters.	33
atm cac overbooking	Configure ATM overbooking parameters.	34
atm debug aal5stats	Display ATM Adaptation Layer 5 (AAL5) port specific ATM statistics.	35
atm debug gstats	Display ATM global statistics.	36
atm debug portstats	Display port specific ATM statistics.	37
atm oam config	Modify the ATM Operation and Maintenance (OAM) settings.	38
atm oam list	Display the ATM OAM settings.	39
atm oam modify	Modify the ATM OAM data blocking mode.	40
atm oam ping	Send ATM loopback cells.	42
atm oam cc list	Display Continuity Check (CC) configuration.	43
atm oam cc modify	Modify CC on the connection.	44
atm oam cc send	Send CC activate/deactivate to connection.	45

ATM Commands

atm oam vclb add	Create a loopback connection for VC.	46
atm oam vclb del	Delete a loopback connection for VC.	47
atm oam vclb list	List all VC loopback connections.	48
atm phonebook add	Add a new phonebook entry.	49
atm phonebook autolist	Show the auto PVCs.	50
atm phonebook delete	Delete an existing phonebook entry.	51
atm phonebook flush	Flush all the phonebook entries.	52
atm phonebook list	Display the current phonebook.	53
atm qosbook add	Add a new QoS book entry.	54
atm qosbook config	Modify the QoS book configuration.	55
atm qosbook ctdadd	Add a Connection Traffic Descriptor (CTD).	56
atm qosbook ctddelete	Delete a CTD.	58
atm qosbook ctdlist	Display all CTDs.	59
atm qosbook delete	Delete a QoS book entry.	60
atm qosbook flush	Flush all the QoS book entries.	61
atm qosbook list	Display the QoS book.	62

atm flush

Flush all Asynchronous Transfer Mode (ATM) interfaces.

SYNTAX:

```
atm flush
```

atm ifadd

Create a new ATM interface.

SYNTAX:

```
atm ifadd intf = <string>
```

where:

intf	The name for the new ATM interface.	REQUIRED
------	-------------------------------------	----------

Note If not specified, the destination parameter must be specified. In this case the name of the destination will double as interface name.

EXAMPLE:

```
=>atm iflist
atm_0_35 : dest : atm_pvc_0_35
            Retry : 10 QoS : default Encaps : vcmux Fcs : disabled Ulp : mac
            Connection State : connected

=>
=>atm ifadd intf = RtPPPoA_atm
=>
=>atm iflist
atm_0_35 : dest : atm_pvc_0_35
            Retry : 10 QoS : default Encaps : vcmux Fcs : disabled Ulp : mac
            Connection State : connected

RtPPPoA_atm: dest : (none)
            Retry : 10 QoS : default Encaps : llc Fcs : disabled Ulp : ip
            Connection State : not-connected

=>
```

RELATED COMMANDS:

atm ifdelete	Delete an ATM interface.
atm iflist	Display the ATM interfaces.

atm ifattach

Attach a ATM interface.

SYNTAX:

```
atm ifattach intf = <string>
```

where:

intf The name of the ATM interface to be attached.

REQUIRED

EXAMPLE:

```
=>atm iflist
atm_0_35 : dest : atm_pvc_0_35
    Retry : 10  QoS : default  Encaps : vcmux Fcs : off Ulp : mac
    Connection State : connected

RtPPPoA_atm: dest : RtPPPoA
    Retry : 10  QoS : default  Encaps : vcmux Fcs : off Ulp : ppp
    Connection State : not-connected

=>atm ifattach intf=RtPPPoA_atm
=>
=>atm iflist
atm_0_35 : dest : atm_pvc_0_35
    Retry : 10  QoS : default  Encaps : vcmux Fcs : off Ulp : mac
    Connection State : connected

RtPPPoA_atm: dest : RtPPPoA
    Retry : 10  QoS : default  Encaps : vcmux Fcs : off Ulp : ppp
    Connection State : connected

=>
```

RELATED COMMANDS:

atm ifdetach Detach an ATM interface.

atm ifconfig

Configure an ATM interface.



The interface to be configured must not be connected at the time of configuration. If this should be the case, use the command :**atm ifdetach** before using the command :**atm ifconfig**.

SYNTAX:

```
atm ifconfig  intf = <string>
              [dest = <string>]
              [qos = <string>]
              [encaps = <{llc | vcmux}>]
              [retry = <number{0-65535}>]
              [fcs = <{disabled | enabled | auto}>]
              [ulp = <{ip | mac | ppp}>]
```

where:

intf	The name of the ATM interface to be configured.	REQUIRED
dest	The WAN destination for this ATM interface. Typically, an ATM phonebook entry.	OPTIONAL
qos	The name of the Quality of Service (QoS) book entry to apply on this ATM interface. The default is default .	OPTIONAL
encaps	The type of encapsulation to be used for this ATM interface. Choose between: ▶ llc : Logical Link Control (LLC) / Sub Network Access Protocol (SNAP) ▶ vcmux : Virtual Channel MULTipleXing (VCMUX). The default is llc .	OPTIONAL
fcs	Enable or disable the inclusion of the Ethernet Frame Check Sequence (FCS) in the packet header on the WAN side (only used for llc encapsulation for mac). The default is disabled .	OPTIONAL
Note This parameter is normally left disabled.		
retry	A number between 0 and 65535. Represents the number of times the SpeedTouch™ retries to set up a WAN connection before giving up. The default is 10 .	OPTIONAL
ulp	Select the Upper Layer Protocol (ULP) for this interface. Choose between: ▶ ip (for a Routed IPoA interface). ▶ mac (for a Bridged Ethernet, Routed EThoA, Bridged PPP over Ethernet (PPPoE), Routed PPPoE or a PPPoE Relay interface). ▶ ppp (for a Routed PPP over ATM (PPPoA) interface). The default is ip .	OPTIONAL

EXAMPLE:

```
=>atm iflist
atm_0_35 : dest : atm_pvc_0_35
    Retry : 10 QoS : default Encaps : llc Fcs : disabled Ulp : mac
    Connection State : connected

atm_8_35 : dest : atm_pvc_8_35
    Retry : 10 QoS : default Encaps : llc Fcs : disabled Ulp : mac
    Connection State : not-connected
=>
=>atm ifconfig intf=atm_8_35 dest=atm_pvc_8_35 encaps=vcmux ulp=ppp
=>
=>atm iflist
atm_0_35 : dest : atm_pvc_0_35
    Retry : 10 QoS : default Encaps : llc Fcs : disabled Ulp : mac
    Connection State : connected

atm_8_35 : dest : RtPPPoA
    Retry : 10 QoS : default Encaps : vcmux Fcs : disabled Ulp : ppp
    Connection State : not-connected

=>
```

atm ifdelete

Delete an ATM interface.

SYNTAX:

```
atm ifdelete intf = <string>
```

where:

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

EXAMPLE:

```
=>atm iflist
atm_0_35 : dest : atm_pvc_0_35
    Retry : 10 QoS : default Encaps : vcmux Fcs : off Ulp : mac
    Connection State : connected

RtPPPoA_atm: dest : RtPPPoA
    Retry : 10 QoS : default Encaps : vcmux Fcs : off Ulp : ppp
    Connection State : connected

=>
=>atm ifdelete intf=RtPPPoA_atm
=>
=>atm iflist
atm_0_35 : dest : atm_pvc_0_35
    Retry : 10 QoS : default Encaps : vcmux Fcs : off Ulp : mac
    Connection State : connected

=>
```

RELATED COMMANDS:

atm ifadd	Create a new ATM interface.
atm iflist	Display the ATM interfaces.

atm ifdetach

Detach an ATM interface.

SYNTAX:

```
atm ifdetach intf = <string>
```

where:

intf The name of the ATM interface to be detached.

REQUIRED

EXAMPLE:

```
=>atm iflist
atm_0_35 : dest : atm_pvc_0_35
    Retry : 10 QoS : default Encaps : vcmux Fcs : off Ulp : mac
    Connection State : connected

RtPPPoA_atm: dest : RtPPPoA
    Retry : 10 QoS : default Encaps : vcmux Fcs : off Ulp : ppp
    Connection State : connected

=>
=>atm ifdetach intf=RtPPPoA_atm
=>
=>atm iflist
atm_0_35 : dest : atm_pvc_0_35
    Retry : 10 QoS : default Encaps : vcmux Fcs : off Ulp : mac
    Connection State : connected

RtPPPoA_atm: dest : RtPPPoA
    Retry : 10 QoS : default Encaps : vcmux Fcs : off Ulp : ppp
    Connection State : not-connected

=>
```

RELATED COMMANDS:

atm ifattach Attach a ATM interface.

atm iflist

Display the ATM interfaces.

SYNTAX:

```
atm iflist [intf = <string>]
```

where:

intf	The name of the ATM interface to be shown.	OPTIONAL
Note	If not specified, all the ATM interfaces are shown.	

EXAMPLE:

```
=>atm iflist
atm_0_35 : dest : atm_pvc_0_35
    Retry : 10  QoS : default  Encaps : vcmux Fcs : off Ulp : mac
    Connection State : connected

RtPPPoA_atm: dest : RtPPPoA
    Retry : 10  QoS : default  Encaps : vcmux Fcs : off Ulp : ppp
    Connection State : connected

=>atm iflist intf=RtPPPoA_atm
RtPPPoA_atm: dest : RtPPPoA
    Retry : 10  QoS : default  Encaps : vcmux Fcs : off Ulp : ppp
    Connection State : connected
=>
```

RELATED COMMANDS:

atm ifadd	Create a new ATM interface.
atm ifdelete	Delete an ATM interface.

atm bundle add

Add a new bundle of interfaces.

SYNTAX:

```
atm bundle add name = <string>
```

where:

name	The name of the new bundle.	REQUIRED
------	-----------------------------	----------

EXAMPLE:

```
=>atm bundle add name=myBundle
=>atm bundle list
myBundle: Connection state : not-connected
    Retry: 10          Policy: priority  Propagate: disabled
    TX requested bytes: 0      requested frames: 0
    TX bytes: 0          frames: 0
    RX bytes: 0          frames: 0
=>
```

RELATED COMMANDS:

atm bundle delete	Delete a bundle of interfaces.
atm bundle list	Display the current bundles.

atm bundle attach

Attach a bundle of interfaces.

SYNTAX:

```
atm bundle attach name = <string>
```

where:

name The name of the bundle to be attached.

REQUIRED

EXAMPLE:

```
=>atm bundle attach name=myBundle
=>atm bundle list
myBundle: Connection state : connected
    Retry: 10          Policy: priority Propagate: disabled
    TX requested bytes: 0  requested frames: 0
    TX bytes: 0        frames: 0
    RX bytes: 0        frames: 0
=>
```

RELATED COMMANDS:

atm bundle detach

Detach a bundle of interfaces.

atm bundle config

Modify a bundle of interfaces.



The bundle of interfaces to be configured must not be connected at the time of configuration. If this should be the case, use the command **:atm bundle detach** before using the command **:atm bundle config**.

SYNTAX:

```
atm bundle config name = <string>
    [policy = <{priority | connection}>]
    [propagate = <{disabled | enabled}>]
    [retry = <number{0-65535}>]
```

where:

name	The name of the bundle to be configured.	REQUIRED
policy	The traffic handling policy of the bundle. Choose between: <ul style="list-style-type: none"> ▶ priority: lower layer ATM interfaces can be configured for a certain priority range, marked packets will be sent on the corresponding interface. ▶ connection: all the packets of the same connection will be sent via the same interface. The default is priority .	OPTIONAL
propagate	Enable or disable propagation of packets. When propagation is enabled, packets are sent via the first interface configured for that traffic. If the interface goes down, the next interface in the bundle will be used. The default is disabled .	OPTIONAL
retry	A number between 0 and 65535. Represents the number of connection setup retries before giving up. The default is 10 .	OPTIONAL

EXAMPLE:

```
=>atm bundle list
myBundle: Connection state : not-connected
    Retry: 10          Policy: priority  Propagate: disabled
    TX requested bytes: 0  requested frames: 0
    TX bytes: 0        frames: 0
    RX bytes: 0        frames: 0
=>atm bundle config name=myBundle policy=connection retry=15
=>atm bundle list
myBundle: Connection state : not-connected
    Retry: 15          Policy: connection  Propagate: disabled
    TX requested bytes: 0  requested frames: 0
    TX bytes: 0        frames: 0
    RX bytes: 0        frames: 0
=>
```

atm bundle clear

Clear the bundle statistics.

SYNTAX:

```
atm bundle clear [name = <string>]
```

where:

name	The name of the bundle for which the statistics must be cleared.	OPTIONAL
Note	If not specified, the statistics for all the bundles will be cleared.	

atm bundle delete

Delete a bundle of interfaces.

SYNTAX:

```
atm bundle delete name = <string>
```

where:

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

RELATED COMMANDS:

atm bundle add	Add a new bundle of interfaces.
atm bundle list	Display the current bundles.

atm bundle detach

Detach a bundle of interfaces.

SYNTAX:

```
atm bundle detach intf = <string>
```

where:

intf	The name of the bundle to be detached.	REQUIRED
------	--	----------

RELATED COMMANDS:

atm bundle attach	Attach a bundle of interfaces.
-------------------	--------------------------------

atm bundle flush

Flush all bundles.

SYNTAX:

```
atm bundle flush
```

atm bundle ifadd

Add an interface on a bundle.



Up to 8 interfaces or (sub-)bundles can be added to one bundle.

SYNTAX:

```
atm bundle ifadd  name = <string>
                  intf = <string>
                  [index = <number{0-6}>]
```

where:

name	The name of the bundle to which an ATM interface must be added.	REQUIRED
intf	The name of the ATM interface to be added.	REQUIRED
index	A number between 0 and 6. Represents the index of the ATM interface.	OPTIONAL

EXAMPLE:

```
=>atm bundle ifadd name=myBundle intf=myATMintf
=>atm bundle list
myBundle: Connection state : not-connected
    Retry: 15          Policy: connection
    TX requested bytes: 0      requested frames: 0
    TX bytes: 0          frames: 0
    RX bytes: 0          frames: 0
    Interface: myATMintf  State: enabled  Selector: 0-15
        TX bytes: 0      frames: 0
        RX bytes: 0      frames: 0
        RX discarded bytes: 0      discarded frames: 0
        Disconnects: 0
=>
```

RELATED COMMANDS:

atm bundle ifdelete

Remove an interface from a bundle.

atm bundle ifconfig

Configure an interface from a bundle.



The interface to be configured must not be connected at the time of configuration. If this should be the case, use the command :**atm bundle ifdetach** before using the command :**atm bundle ifconfig**.

SYNTAX:

```
atm bundle ifconfig    name = <string>
                      intf = <string>
                      [state = <{disabled | enabled}>]
                      [low = <number{0-15}>]
                      [high = <number{0-15}>]
```

where:

name	The name of the bundle from which an ATM interface must be configured.	REQUIRED
intf	The name of the ATM interface to be configured.	REQUIRED
state	Enable or disable the ATM interface. The default is enabled .	OPTIONAL
low	A number between 0 and 15. Represents the low selector value.	OPTIONAL
Note This parameter must only be configured when the policy of the bundle is set to priority (see "atm bundle config" on page 23).		
high	A number between 0 and 15. Represents the high selector value.	OPTIONAL
Note This parameter must only be configured when the policy of the bundle is set to priority (see "atm bundle config" on page 23).		

atm bundle ifdelete

Remove an interface from a bundle.

SYNTAX:

```
atm bundle ifdelete name = <string>
                    intf = <string>
```

where:

name	The name of the bundle from which an ATM interface must be removed.	REQUIRED
intf	The name of the ATM interface to be removed.	REQUIRED

EXAMPLE:

```
=>atm bundle iflist
Test      : dest : (none)
          Retry : 10  QoS : default  Encaps : llc Fcs : off Ulp : ip
          Connection State : not-connected

=>atm bundle ifdelete intf=Test
=>atm bundle iflist

=>
```

RELATED COMMANDS:

atm bundle ifadd Add an interface on a bundle.

atm bundle list

Display the current bundles.

SYNTAX:

```
atm bundle list
```

EXAMPLE:

```
=>atm bundle list
Test      : Connection state : not-connected
          Retry: 10          Policy: priority
          TX requested bytes: 0    requested frames: 0
          TX bytes: 0           frames: 0
          RX bytes: 0           frames: 0
Interface: Test   State: enabled   Selector: 0-15
          TX bytes: 0           frames: 0
          RX bytes: 0           frames: 0
          RX discarded bytes: 0   discarded frames: 0
          Disconnects: 0
=>
```

RELATED COMMANDS:

atm bundle add	Add a new bundle of interfaces.
atm bundle delete	Delete a bundle of interfaces.

atm cac config

Configure the ATM connection admission control.

SYNTAX:

```
atm cac config      config port = <{ds10|ds11|atm2|atm3|aal5|atm5} or number>
                     state = <{disabled|enabled}>
```

where:

config port	The port for which CAC is configured.	REQUIRED
state	Enable/disable CAC for an ATM port.	REQUIRED

EXAMPLE:

```
{Administrator}>atm cac config
port = ds10
state = enabled
:atm cac config port=ds10 state=enabled
{Administrator}>
```

RELATED COMMANDS:

atm cac list	Display all the CAC parameters.
atm cac overbooking	Configure ATM overbooking parameters.

atm cac list

Display all the CAC parameters.

SYNTAX:

```
atm cac list
```

EXAMPLE:

```
{Administrator} [atm cac]=>list
CAC: port: ds10 state: enabled
CAC: port: ds11 state: enabled
CAC: port: atm2 state: enabled
CAC: port: aal5 state: disabled
CAC: port: atm5 state: disabled
Overbooking: realtime: 0% non-realtime: 0%.
{Administrator} [atm cac]=>
```

RELATED COMMANDS:

atm cac config	Configure the ATM connection admission control.
atm cac overbooking	Configure ATM overbooking parameters.

atm cac overbooking

Configure ATM overbooking parameters.

SYNTAX:

```
atm cac overbooking rt = <number{0-1000}>
                           nrt = <number{0-1000}>
```

where:

rt	A number between 0 and 1000. The realtime overbooking percentage.	REQUIRED
nrt	A number between 0 and 1000. The non-realtime overbooking percentage.	REQUIRED

EXAMPLE:

```
{Administrator} [atm cac]=>overbooking rt 20 nrt 500
{Administrator} [atm cac]=>list
CAC: port: ds10 state: enabled
CAC: port: ds11 state: enabled
CAC: port: atm2 state: enabled
CAC: port: aal5 state: disabled
CAC: port: atm5 state: disabled
Overbooking: realtime: 20% non-realtime: 500%.
{Administrator} [atm cac]=>
```

RELATED COMMANDS:

- | | |
|----------------|---|
| atm cac config | Configure the ATM connection admission control. |
| atm cac list | Display all the CAC parameters. |

atm debug aal5stats

Display ATM Adaptation Layer 5 (AAL5) port specific ATM statistics.

SYNTAX:

```
atm debug aal5stats port = <{ds10|ds11|atm2|atm3|aal5|atm5} or number>
                           vpi = <number{0-15}>
                           [vci = <number{0-511}>]
                           [clear = <{disabled | enabled}>]
```

where:

port	The port number 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 (ds10 has port number 0).	REQUIRED
vpi	A number between 0 and 15. Represents the Virtual Path Identifier (VPI) number for which statistics will be retrieved.	REQUIRED
vci	A number between 0 and 511. Represents the Virtual Channel Identifier (VCI) number for which statistics will be retrieved.	OPTIONAL
clear	Enable or disable clearing of the statistics after request.	OPTIONAL

RELATED COMMANDS:

atm debug gstats	Display ATM global statistics.
atm debug portstats	Display port specific ATM statistics.

atm debug gstats

Display ATM global statistics.

SYNTAX:

```
atm debug gstats [clear = <{disabled | enabled}>]
```

where:

clear	Enable or disable clearing of the statistics after request.	OPTIONAL
-------	---	----------

EXAMPLE:

```
{admin} [atm debug]=>gstats
      # of received octets = 1802.
      # of transmitted octets = 4346.
      # of received cells = 34.
      # of transmitted cells = 82.
      # of unknown cells = 0.
      # of errors on the input = 0.
      # of errors on output = 0.
{admin} [atm debug]=>
```

RELATED COMMANDS:

atm debug aal5stats	Display ATM Adaptation Layer 5 (AAL5) port specific ATM statistics.
atm debug portstats	Display port specific ATM statistics.

atm debug portstats

Display port specific ATM statistics.

SYNTAX:

```
atm debug portstats port = <{dsl0} or number>
    [clear = <{disabled | enabled}>]
```

where:

port	The port number for which statistics will be retrieved. Choose between: ▶ DSL0 Or specify a port number (dsl0 has port number 0).	REQUIRED
clear	Enable or disable clearing of the statistics after request.	OPTIONAL

EXAMPLE:

```
=>atm debug portstats port=dsl0
      # of received octets = 1961.
      # of transmitted octets = 4717.
      # of received cells = 37.
      # of transmitted cells = 89.
      # of unknown cells = 0.
      # of errors on the input = 0.
      # of errors on output = 0.
=>
```

RELATED COMMANDS:

atm debug aal5stats	Display ATM Adaptation Layer 5 (AAL5) port specific ATM statistics.
atm debug gstats	Display ATM global statistics.

atm oam config

Modify the ATM Operation and Maintenance (OAM) settings.

SYNTAX:

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

where:

RELATED COMMANDS:

atm oam list	Display the ATM OAM settings.
atm oam modify	Modify the ATM OAM data blocking mode.
atm oam ping	Send ATM loopback cells.

atm oam list

Display the ATM OAM settings.

SYNTAX:

```
atm oam list
```

EXAMPLE:

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

RELATED COMMANDS:

atm oam config	Modify the ATM Operation and Maintenance (OAM) settings.
atm oam modify	Modify the ATM OAM data blocking mode.
atm oam ping	Send ATM loopback cells.

atm oam modify

Modify the ATM OAM data blocking mode.

SYNTAX:

```
atm oam modify port = <{ds10} or number>
    blocking = <{disabled | enabled}>
```

where:

port	The port for which OAM blocking is configured. Choose between: ▶ DSL0 Or specify a port number (ds10 has port number 0).	REQUIRED
blocking	Enable or disable the OAM data blocking mode on this port. The default is enabled .	REQUIRED

EXAMPLE:

```
=>atm oam list
  OAM config dump
-----
      CLP bit value : 1
      Loopback id   : 6a 6a
      OAM data blocking mode
-----
      Port ds10: blocking
      Port ds11: blocking
      Port atm2: blocking
      Port atm3: blocking
      Port aal5: blocking
      Port atm5: blocking
=>atm oam modify port=ds11 blocking=disabled
=>:atm oam list
  OAM config dump
-----
      CLP bit value : 1
      Loopback id   : 6a 6a
      OAM data blocking mode
-----
      Port ds10: blocking
      Port ds11: non blocking
      Port atm2: blocking
      Port atm3: blocking
      Port aal5: blocking
      Port atm5: blocking
=>
```

RELATED COMMANDS:

- | | |
|----------------|--|
| atm oam config | Modify the ATM Operation and Maintenance (OAM) settings. |
| atm oam list | Display the ATM OAM settings. |

atm oam ping Send ATM loopback cells.

atm oam ping

Send ATM loopback cells.

SYNTAX:

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

where:

dest	The destination address for the request. This can be any ATM phonebook entry.	REQUIRED
count	A number between 1 and 1000000. Represents the number of pings to send.	OPTIONAL
interval	A number between 100 and 1000000 (milliseconds). Represents the interval between packets.	OPTIONAL

EXAMPLE:

```
=>atm oam ping dest=atm_ph_8_35 count=10 interval=1000
loopback: successful, sequence: 1 time: 37890 usec
loopback: successful, sequence: 2 time: 39118 usec
loopback: successful, sequence: 3 time: 39116 usec
loopback: successful, sequence: 4 time: 39187 usec
loopback: successful, sequence: 5 time: 38605 usec
loopback: successful, sequence: 6 time: 38764 usec
loopback: successful, sequence: 7 time: 38752 usec
loopback: successful, sequence: 8 time: 38813 usec
loopback: successful, sequence: 9 time: 38848 usec
loopback: successful, sequence: 10 time: 38941 usec

--- loopback statistics ---
10 loopbacks transmitted, 10 successful, 0% loss, time 1390 ms
rtt min/avg/max = 37890/38803/39187
=>
```

RELATED COMMANDS:

- | | |
|----------------|--|
| atm oam config | Modify the ATM Operation and Maintenance (OAM) settings. |
| atm oam list | Display the ATM OAM settings. |
| atm oam modify | Modify the ATM OAM data blocking mode. |

atm oam cc list

Display Continuity Check (CC) configuration.

SYNTAX:

```
atm oam cc list
```

EXAMPLE:

```
=>atm oam cc list
PORT = 0 VPI = 15 VCI = 16 End2End Mode = Auto Segment Mode = Auto
PORT = 0 VPI = 0 VCI = 35 End2End Mode = Auto Segment Mode = Auto
PORT = 0 VPI = 8 VCI = 35 End2End Mode = Auto Segment Mode = Auto
PORT = 0 VPI = 0 VCI = 16 End2End Mode = Auto Segment Mode = Auto
=>
```

RELATED COMMANDS:

- | | |
|--------------------------|--|
| atm oam cc modify | Modify CC on the connection. |
| atm oam cc send | Send CC activate/deactivate to connection. |

atm oam cc modify

Modify CC on the connection.

SYNTAX:

```
atm oam modify port = <{ds10|ds11|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}>]
```

where:

port	The ATM port number. Choose between: <ul style="list-style-type: none">▶ DSL0▶ DSL1▶ ATM2▶ ATM3▶ AAL5▶ ATM5 Or specify a port number (ds10 has port number 0).	REQUIRED
vpi	A number between 0 and 15. Represents the VPI.	REQUIRED
vci	A number between 0 and 511. Represents the VCI.	OPTIONAL
Note For a Virtual Path (VP) cross-connection, use 0 or do not specify.		
transmit	Enable or disable transmission of CC cells.	OPTIONAL
receive	Enable or disable loss of continuity.	OPTIONAL
auto	Enable or disable remote CC activation/deactivation.	OPTIONAL
span	Select the CC span. Choose between: <ul style="list-style-type: none">▶ end2end: monitoring occurs on the entire VC between two ATM end stations.▶ segment: monitoring occurs on a VC segment between the SpeedTouch™ and a first-hop ATM switch.	OPTIONAL

RELATED COMMANDS:

atm oam cc list	Display Continuity Check (CC) configuration.
atm oam cc send	Send CC activate/deactivate to connection.

atm oam cc send

Send CC activate/deactivate to connection.

SYNTAX:

```
atm oam cc send port = <{ds10} or number>
    vpi = <number{0-15}>
    [vci = <number{0-511}>]
    [span = <{segment | end2end}>]
    [action = <{activate | deactivate}>]
    [direction = <{source | sink | both}>]
```

where:

port	The ATM port number. Choose between: ▶ DSL0. Or specify a port number (ds10 has port number 0).	REQUIRED
vpi	A number between 0 and 15. Represents the VPI.	REQUIRED
vci	A number between 0 and 511. Represents the VCI. Note For a VP cross-connection, use 0 or do not specify.	OPTIONAL
span	Select the CC span. Choose between: ▶ end2end : monitoring occurs on the entire VC between two ATM end stations. ▶ segment : monitoring occurs on a VC segment between the SpeedTouch™ and a first-hop ATM switch.	OPTIONAL
action	Activate or deactivate CC. The default is deactivate .	OPTIONAL
direction	Indicates the direction of CC activity. Choose between: ▶ source ▶ sink ▶ both. The default is both .	OPTIONAL

RELATED COMMANDS:

atm oam cc list	Display Continuity Check (CC) configuration.
atm oam cc modify	Modify CC on the connection.

atm oam vclb add

Create a loopback connection for VC.

SYNTAX:

```
atm oam vclb add port = <{dsl0} or number>
    vpi = <number{0-15}>
    [vci = <number{0-511}>]
```

where:

port	The ATM port number. Choose between: ▶ DSL0 Or specify a port number (dsl0 has port number 0).	REQUIRED
vpi	A number between 0 and 15. Represents the VPI.	REQUIRED
vci	A number between 0 and 511. Represents the VCI. Note For a VP cross-connection, use 0 or do not specify.	OPTIONAL

EXAMPLE:

```
=>atm oam vclb add port=dsl0 vpi=0 vci=36
=>atm oam vclb list
PORT = 0 VPI = 0 VCI = 36
=>atm oam vclb add port=dsl1 vpi=0 vci=37
=>atm oam vclb list
PORT = 0 VPI = 0 VCI = 36
PORT = 1 VPI = 0 VCI = 37
=>
```

RELATED COMMANDS:

atm oam vclb del	Delete a loopback connection for VC.
atm oam vclb list	List all VC loopback connections.

atm oam vclb del

Delete a loopback connection for VC.

SYNTAX:

```
atm oam vclb del port = <{dsl0} or number>
                     vpi = <number{0-15}>
                     [vci = <number{0-511}>]
```

where:

port	The ATM port number. Choose between: ▶ DSL0 Or specify a port number (dsl0 has port number 0).	REQUIRED
vpi	A number between 0 and 15. Represents the VPI.	REQUIRED
vci	A number between 0 and 511. Represents the VCI. Note For a VP cross-connection, use 0 or do not specify.	OPTIONAL

EXAMPLE:

```
=>atm oam vclb list
PORT = 0 VPI = 0 VCI = 36
PORT = 1 VPI = 0 VCI = 37
=>atm oam vclb del port=dsl1 vpi=0 vci=37
=>atm oam vclb list
PORT = 0 VPI = 0 VCI = 36
=>
```

RELATED COMMANDS:

- | | |
|-------------------|--------------------------------------|
| atm oam vclb add | Create a loopback connection for VC. |
| atm oam vclb list | List all VC loopback connections. |

atm oam vclb list

List all VC loopback connections.

SYNTAX:

```
atm oam vclb list
```

EXAMPLE:

```
=>atm oam vclb list
PORT = 0 VPI = 0 VCI = 36
PORT = 1 VPI = 0 VCI = 37
=>
```

RELATED COMMANDS:

- | | |
|------------------|--------------------------------------|
| atm oam vclb add | Create a loopback connection for VC. |
| atm oam vclb del | Delete a loopback connection for VC. |

atm phonebook add

Add a new phonebook entry.

SYNTAX:

```
atm phonebook add name = <string>
                  addr = <atmchannel : PVC syntax is [port.]vpi.vci
                           port=dsl0 | ...>
```

where:

name	The name of the new phonebook entry. This name can be freely chosen, however two limitations apply:	REQUIRED
	<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 (for example DHCP_Spoof01). 	
addr	The ATM address for this destination. It is composed of a VPI and a VCI identifying ATM virtual channels. In most cases the values are provided by the Service Provider. Accepted VPI: a number between 0 and 15. Accepted VCI: a number between 0 and 511.	REQUIRED

EXAMPLE:

```
=>atm phonebook list
Name          Use   Address
atm_pvc_0_35 1    0.35
=>atm phonebook add name=RtPPPoA addr=8.35
=>atm phonebook list
Name          Use   Address
atm_pvc_0_35 1    0.35
RtPPPoA      0    8.35
=>
```

RELATED COMMANDS:

atm phonebook delete	Delete an existing phonebook entry.
atm phonebook list	Display the current phonebook.

atm phonebook autolist

Show the auto PVCs.



Auto PVCs are only shown if they are supported by the Central Office DSLAM.

SYNTAX:

```
atm phonebook autolist
```

EXAMPLE:

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

RELATED COMMANDS:

[atm phonebook list](#) Display the current phonebook.

atm phonebook delete

Delete an existing phonebook entry.



This command is only applicable for phonebook entries that are not used (in other words, not configured for any packet service).

SYNTAX:

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

where:

name	The name of the phonebook entry to be deleted.	REQUIRED
Tip	Use the command :atm phonebook list to check whether the entry is in use (<i>Use=1</i>) or not (<i>Use=0</i>).	

EXAMPLE:

```
=>atm phonebook list
Name      Use  Address
atm_pvc_0_35 1    0.35
RtPPPoA      0    8.35
=>atm phonebook delete name=RtPPPoA
=>atm phonebook list
Name      Use  Address
atm_pvc_0_35 1    0.35
=>
```

RELATED COMMANDS:

atm phonebook add	Add a new phonebook entry.
atm phonebook list	Display the current phonebook.

atm phonebook flush

Flush all the phonebook entries.



1. Phonebook entries that are in use, cannot be flushed.
2. This command does not impact previously saved configurations.

SYNTAX:

```
atm phonebook flush
```

EXAMPLE:

```
=>atm phonebook list
Name      Use   Address
atm_pvc_0_35 1    0.35
RtPPPoA      0    8.35
=>
=>atm phonebook flush
Some phonebook entries are still in use. Entries that are in use cannot be deleted.
=>
=>atm phonebook list
Name      Use   Address
atm_pvc_0_35 1    0.35
=>
```

atm phonebook list

Display the current phonebook.

SYNTAX:

```
atm phonebook list
```

EXAMPLE:

```
=>atm phonebook list
Name      Use   Address
atm_pvc_0_35 1    0.35
RtPPPoA      0    8.35
=>
```

RELATED COMMANDS:

atm phonebook add	Add a new phonebook entry.
atm phonebook autolist	Show the auto PVCs.
atm phonebook delete	Delete an existing phonebook entry.

atm qosbook add

Add a new QoS book entry.

SYNTAX:

```
atm qosbook add name = <string>
                  [txctd = <string>]
                  [rxctd = <string>]
```

where:

name	The name of the new QoS entry.	REQUIRED
txctd	The name of the Conformance Traffic Descriptor (CTD) for the transmit (upstream) direction.	OPTIONAL
rxctd	The name of the CTD for the receive (downstream) direction.	OPTIONAL

EXAMPLE:

```
=>atm qosbook list
Name          Ref Tx CTD          Rx CTD
default       3   default        default
=>
```

RELATED COMMANDS:

atm qosbook delete	Delete a QoS book entry.
atm qosbook list	Display the QoS book.

atm qosbook config

Modify the QoS book configuration.

SYNTAX:

```
atm qosbook config [format = <{bytes | cells}>]
```

where:

format	The input/output format of the QoS book. Choose between: <ul style="list-style-type: none">▶ bytes: the output is shown in Kbits or bytes.▶ cells: the output is shown in cps or cells. The default is bytes .	OPTIONAL
--------	---	----------

EXAMPLE:

```
=>atm qosbook ctdlist
Name    Ref Conf      Peak      Sust      Burst      Minrate     Frame      Cdvt      RT          FD
              (Kbits)   (Kbits)   (bytes)   (Kbits)   (bytes)
default 2    UBR      linerate 0        0         0         0         0       disabled  disabled
=>atm qosbook config format=cells
=>atm qosbook ctdlist
Name    Ref Conf      Peak      Sust      Burst      Minrate     Frame      Cdvt      RT          FD
              (cps)    (cps)    (cells)   (cps)    (cells)
default 2    UBR      linerate 0        0         0         0         0       disabled  disabled
=>
```

atm qosbook ctdadd

Add a Connection Traffic Descriptor (CTD).

SYNTAX:

```
atm qosbook ctdadd name = <string>
    conformance = <{UBR | CBR | VBR }>
    [peakrate = <number{0-27786}>]
    [sustrate = <number{0-27786}>]
    [maxburst = <number{0-12240}>]
    [realtime = <{enabled | disabled}>]
```

where:

name	The name of the new CTD.	REQUIRED
conformance	The ATM service conformance definition.	REQUIRED
peakrate	A number between 0 and 27786. Represents the peak rate (in kilobits per second). The default is 0 (indicates linerate for UBR).	OPTIONAL
sustrate	A number between 0 and 27786. Represents the sustainable rate (in kilobits per second) (VBR only). The default is 0 .	OPTIONAL
maxburst	A number between 48 and 12240. Represents the maximum burst size (in bytes) (VBR or GFR). The default is 0 .	OPTIONAL
realtime	Enable or disable realtime traffic (VBR only). The default is disabled .	OPTIONAL

EXAMPLE:

```
=>atm qosbook ctdadd name=High conformance=CBR peakrate=27786
=>atm qosbook ctdlist
Name      Ref Conf      Peak      Sust      Burst      Minrate     Frame      Cdvt      RT          FD
          (Kbits)   (Kbits)   (bytes)   (Kbits)   (bytes)
default   2     UBR      linerate  0          0          0          0          0        disabled  disabled
High      0     CBR      27786     0          0          0          0          0        disabled  disabled
=>
```

IMPORTANT NOTE:

The SpeedTouch™ always rounds up specified burst sizes to a multiple of 48 bytes (a multiple of ATM cells).

Example:

In the example below a burst size of 100 bytes is specified (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 the command :atm qosbook ctdlist (burst=144).

```
=>atm qosbook ctdadd name=Medium conformance=VBR peakrate=27786 sustrate=20000 maxburst=100
=>atm qosbook ctdlist
Name      Ref Conf      Peak      Sust      Burst      Minrate     Frame      Cdvt      RT      FD
          (Kbits)   (Kbits)   (bytes)   (Kbits)   (bytes)
default   2     UBR      linerate 0         0         0         0         0         disabled  disabled
Medium    0     VBR      27786   20000    144       0         0         0         disabled  disabled
=>
```

RELATED COMMANDS:

atm qosbook ctddelete	Delete a CTD.
atm qosbook ctdlist	Display all CTDs.

atm qosbook ctddelete

Delete a CTD.

SYNTAX:

```
atm qosbook ctddelete name = <string>
    [force = <{disabled | enabled}>]
```

where:

name	The name of the CTD entry to be deleted.	REQUIRED
force	Enable or disable to force delete the entry even when it is still in use. The default is disabled .	OPTIONAL

EXAMPLE:

```
=>atm qosbook ctdlist
Name      Ref Conf      Peak      Sust      Burst      Minrate     Frame      Cdvt      RT          FD
          (Kbits)   (Kbits)   (bytes)   (Kbits)   (bytes)
default   2     UBR       linerate  0          0          0          0          0          disabled   disabled
High      0     CBR       27786     0          0          0          0          0          disabled   disabled
=>atm qosbook ctddelete name=High
=>atm qosbook ctdlist
Name      Ref Conf      Peak      Sust      Burst      Minrate     Frame      Cdvt      RT          FD
          (Kbits)   (Kbits)   (bytes)   (Kbits)   (bytes)
default   2     UBR       linerate  0          0          0          0          0          disabled   disabled
=>
```

RELATED COMMANDS:

- | | |
|----------------------------|--|
| atm qosbook ctdadd | Add a Connection Traffic Descriptor (CTD). |
| atm qosbook ctdlist | Display all CTDs. |

atm qosbook ctdlist

Display all CTDs.

SYNTAX:

```
atm qosbook ctdlist
```

EXAMPLE:

```
=>atm qosbook ctdlist
Name      Ref Conf      Peak      Sust      Burst      Minrate     Frame      Cdvt      RT          FD
          (Kbits)   (Kbits)   (bytes)   (Kbits)   (bytes)
default   2    UBR      linerate  0         0         0         0         0       disabled   disabled
High      0    CBR      27786    0         0         0         0         0       disabled   disabled
=>
```

RELATED COMMANDS:

atm qosbook ctdadd Add a Connection Traffic Descriptor (CTD).

atm qosbook ctddelete Delete a CTD.

atm qosbook delete

Delete a QoS book entry.

SYNTAX:

```
atm qosbook delete name = <string>
[force = <{disabled | enabled}>]
```

where:

name	The name of the QoS book entry to be deleted.	REQUIRED
force	Enable or disable to force delete the entry even when it is still in use. The default is disabled .	OPTIONAL

EXAMPLE:

```
=>atm 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
=>atm qosbook delete name=TestVBR
=>atm 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:

- | | |
|-------------------------|---------------------------|
| atm qosbook add | Add a new QoS book entry. |
| atm qosbook list | Display the QoS book. |

atm qosbook flush

Flush all the QoS book entries.



The flush command does not impact previously saved configurations.

SYNTAX:

```
atm qosbook flush
```

atm qosbook list

Display the QoS book.

SYNTAX:

```
atm qosbook list
```

EXAMPLE:

```
=>atm 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:

atm qosbook add	Add a new QoS book entry.
atm qosbook delete	Delete a QoS book entry.

AutoPVC Commands

Introduction

This chapter describes the commands of the **autopvc** command group.

Contents

This chapter covers the following commands:

autopvc config	Configure autopvc.	64
autopvc list	Show the retrieved information.	65

autopvc config

Configure autopvc.

SYNTAX:

```
autopvc config [mode = <{pseudo | passive | active}>]
               [type = <{bridge | pppoerelay | ipoa | ethoa | pppoa | pppoe}>]
               [overwrite = <{disabled | enabled}>]
               [peakrate = <number{0-27786}>]
```

where:

mode	Select the autopvc mode:	OPTIONAL
	<ul style="list-style-type: none"> ▶ pseudo: only pseudo-ILMI (VP/VC 15/16) is enabled. When the connection parameters are written to the MIB, this information is displayed on CLI or web interface but these parameters are not used for configuration. ▶ passive: both ILMI (VP/VC 0/16) and pseudo-ILMI (VP/VC 15/16) are enabled. When the connection parameters are written to the MIB, this information is displayed on CLI or web interface but these parameters are not used for configuration. ▶ active: both ILMI (VP/VC 0/16) and pseudo-ILMI (VP/VC 15/16) are enabled. When the connection parameters are written to the MIB, these parameters are used to configure phonebook entries, qosbook profiles and bind bridge or PPPoE interfaces on top. 	
	The default is passive .	
type	Select the type of autopvc. Choose between:	OPTIONAL
	<ul style="list-style-type: none"> ▶ bridge ▶ pppoerelay: an ETHoA interface will be created, will be bound to the ILMI Permanent Virtual Channel (PVC) and will be added to the PPPoE relay as relay port. ▶ ipoa ▶ ethoa ▶ pppoa ▶ pppoe. 	
overwrite	Enable or disable UBR peak rate overwrite. The default is disabled .	OPTIONAL
peakrate	A number between 0 and 27786. Represents the UBR peak rate (in kilobits per second). The default is 0 (indicates the linerate).	OPTIONAL

EXAMPLE:

```
=>autopvc config
Autopvc mode      : passive
Autopvc type     :
Autopvc standard: unknown
Autopvc pseudo   : unknown
UBR overwrite   : disabled
UBR peak rate   : linerate
=>
```

autopvc list

Show the retrieved information.

SYNTAX:

```
autopvc list [table = <{Port | AtmLayer | Vpc | Vcc | Address
| AddressRegistrationAdmin | AtmServiceType
| AtmServiceConnectionInfo | AAL1Profile |
| AAL34Profile | AAL5Profile | AAL2CommonProfile
| AAL2TrunkingProfile | AAL2LESProfile
| AtmServiceConnInfoExtension |
| AtmServiceTypeExtension | AAL5ProfileExtension}>]
```

where:

table	Select the autopvc table for which the information must be shown. Choose between:	OPTIONAL
	<ul style="list-style-type: none"> ▶ Port ▶ AtmLayer ▶ Vpc ▶ Vcc ▶ Address ▶ AddressRegistrationAdmin ▶ AtmServiceType ▶ AtmServiceConnectionInfo ▶ AAL1Profile ▶ AAL34Profile ▶ AAL5Profile ▶ AAL2CommonProfile ▶ AAL2TrunkingProfile ▶ AAL2LESProfile ▶ AtmServiceConnInfoExtension ▶ AtmServiceTypeExtension ▶ AAL5ProfileExtension. 	

EXAMPLE:

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


Config Commands

Introduction

This chapter describes the commands of the **config** command group.

Contents

This chapter covers the following commands:

config delete	Delete a user configuration file.	68
config dump	Show the saved configuration file.	69
config flush	Flush the loaded configuration.	70
config list	Show the current configuration set.	71
config load	Load complete saved (backup) or default configuration file.	72
config save	Store the current configuration in a backup file.	73

config delete

Delete a user configuration file.

SYNTAX:

```
config delete [filename = <string>]
```

where:

filename	Name of the user configuration file to be deleted.	OPTIONAL
----------	--	----------

Note If not specified, all the user configuration files that were saved in the SpeedTouch™ permanent storage will be deleted.

RELATED COMMANDS:

config dump	Show the saved configuration file.
-------------	------------------------------------

config dump

Show the saved configuration file.

SYNTAX:

```
config dump
```

RELATED COMMANDS:

config delete Delete a user configuration file.

config flush

Flush the loaded configuration.



1. This flush command combines all the possible flush commands.
2. This command does not affect saved configurations.

SYNTAX:

```
config flush [flush_ip = <{enabled | disabled}>]
```

where:

flush_ip	Flush IP settings (enabled) or not (disabled). The default is enabled .	OPTIONAL
----------	---	----------

Note Not keeping the IP settings could cause lost IP connectivity in the LAN.

config list

Show the current configuration set.

SYNTAX:

```
config list [templates = <{disabled | enabled}>]
```

where:

templates	List the template files (disabled) or not (enabled). The default is disabled .	OPTIONAL
-----------	--	----------

RELATED COMMANDS:

config load	Load complete saved (backup) or default configuration file.
config save	Store the current configuration in a backup file.

config load

Load complete saved (backup) or default configuration file.



Use the command :**config flush** before loading a configuration file.

SYNTAX:

```
config load [load_ip = <{enabled | disabled}>]
            [defaults = <{enabled | disabled}>]
            [flush = <{enabled | disabled}>]
            [echo = <{disabled | enabled}>]
            [filename = <string>]
```

where:

load_ip	Load IP settings (enabled) or not (disabled).	OPTIONAL
	Note Not keeping the IP settings could cause lost IP connectivity in the LAN.	
defaults	Load default configuration (enabled) or saved configuration (disabled).	OPTIONAL
	Note If not specified, the saved configuration will be loaded.	
flush	Flush the current configuration before loading a new configuration (enabled) or not (disabled).	OPTIONAL
echo	Echo each command string when loaded (enabled) or not (disabled).	OPTIONAL
filename	Name of the configuration file to be loaded.	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   cipl     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
  172.16.0.0/24    0.0.0.0/0   172.16.0.5  cipl     1
=>config load 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
=>config load flush=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   cipl     0
  127.0.0.1/32     0.0.0.0/0   127.0.0.1    loop     0
  172.16.0.0/24    0.0.0.0/0   172.16.0.5  cipl     1
=>
```

RELATED COMMANDS:

- | | |
|--------------------|---|
| config list | Show the current configuration set. |
| config save | Store the current configuration in a backup file. |

config save

Store the current configuration in a backup file.

All the existing configurations and modifications entered by the user are saved.

The backup file is saved in the SpeedTouch™ permanent storage. This file can be downloaded via the SpeedTouch™ web pages or via an FTP session.

SYNTAX:

```
config save filename = <string>
```

where:

filename	The filename for the backup file of the current configuration.	REQUIRED
----------	--	----------

RELATED COMMANDS:

config list	Show the current configuration set.
config load	Load complete saved (backup) or default configuration file.

Connection Commands

Introduction

This chapter describes the commands of the **connection** command group.

Contents

This chapter covers the following commands:

connection appconfig	Configure the available CONN/NAT application helpers.	76
connection appinfo	Display CONN/NAT application specific info.	78
connection applist	List the available CONN/NAT application helpers.	79
connection bind	Create a new CONN/NAT application helper/port binding.	80
connection bindlist	List the current CONN/NAT application helper/port bindings.	81
connection clean	Clean the connection database by forcing timeouts.	82
connection clear	Kill all the connections.	83
connection config	Configure the connection handling.	84
connection debug	The connection debug commands	85
connection describe	Describe the streams of a connection.	86
connection flush	Flush the current connection configuration.	87
connection info	Show all the registered modules with some info.	88
connection list	Display the currently known connections.	89
connection refresh	Invalidate all the cached decisions.	90
connection stats	Display the connection and stream statistics.	91
connection timerclear	Clear the connection timeout to default.	92
connection timerconfig	Configure the connection timeout handling.	93
connection unbind	Delete an existing CONN/NAT application helper/port binding.	94

connection appconfig

Configure the available CONN/NAT application helpers.

SYNTAX:

```
connection appconfig application = <string>
    [trace = <{disabled | enabled}>]
    [timeout = <number{0-32000}>]
    [floating = <{disabled | enabled}>]
    [tracelevel = <number{1-4}>]
    [SIP_ALG = <{disabled|enabled}>]
    [RTP_predict_for_term_SIP_ALG = <{disabled|enabled}>]
```

where:

application	The name of a CONN/NAT application helper.	REQUIRED
	Tip Use the command :connection applist to obtain a list of CONN/NAT application helpers.	
trace	Enable or disable CONN/NAT application helper traces. The default is enabled .	OPTIONAL
timeout	A number between 0 and 32000 (seconds). Represents the maximum timeout to keep predicted child connections around.	OPTIONAL
floating	Enable / disabled floating port for IKE helper.	OPTIONAL
tracelevel	A number between 1 and 4. Represents the SIP trace level. Choose between: <ul style="list-style-type: none"> ▶ 1: feature errors ▶ 2: feature traces ▶ 3: all errors ▶ 4: all traces. The default is 1 .	OPTIONAL
SIP_ALG	SIP only: enable/disable traditional sip alg behaviour	OPTIONAL
RTP_predict_for_term_SIP_ALG	Enable/disable RTP connection prediction for terminated SIP	OPTIONAL

EXAMPLE:

```
=>connection applist
Application Proto DefaultPort Traces      Timeout
IP6TO4      6to4   0      enabled     unavailable
PPTP        tcp    1723    enabled     unavailable
ESP         esp    0      unavailable 15' 0"
IKE         udp    500    disabled   15' 0"      FLOATING
SIP         udp    5060   disabled   6 0"       trace level 1
...
LOOSE(UDP)  udp    0      enabled     5' 0"
FTP         tcp    21     enabled     unavailable
=>connection appconfig application=SIP trace=enabled
=>connection applist
Application Proto DefaultPort Traces      Timeout
IP6TO4      6to4   0      enabled     unavailable
PPTP        tcp    1723    enabled     unavailable
ESP         esp    0      unavailable 15' 0"
IKE         udp    500    disabled   15' 0"      FLOATING
SIP         udp    5060   enabled    6 0"       trace level 1
...
LOOSE(UDP)  udp    0      enabled     5' 0"
FTP         tcp    21     enabled     unavailable
=>
```

RELATED COMMANDS:

- | | |
|---------------------------|--|
| connection appinfo | Display CONN/NAT application specific info. |
| connection applist | List the available CONN/NAT application helpers. |

Connection Commands

connection appinfo

Display CONN/NAT application specific info.

SYNTAX:

```
connection appinfo application = <{string}>
```

where:

application	The name of a CONN/NAT application helper.	REQUIRED
Tip	Use the command :connection applist to obtain a list of CONN/NAT application helpers.	

EXAMPLE:

```
=>connection appinfo application=SIP  
SIP ALG session SIPALG: pool=138, in use=0, bottom=138  
=>
```

RELATED COMMANDS:

connection appconfig	Configure the available CONN/NAT application helpers.
connection applist	List the available CONN/NAT application helpers.

connection applist

List the available CONN/NAT application helpers.

SYNTAX:

```
connection applist
```

EXAMPLE:

```
=>connection applist
Application Proto DefaultPort Traces      Timeout
IP6TO4      6to4   0        enabled       unavailable
PPTP         tcp    1723    enabled       unavailable
ESP          esp    0        unavailable  15' 0"
IKE          udp    500     disabled     15' 0"      FLOATING
SIP          udp    5060    disabled     6 0"        trace level 1
JABBER       tcp    5222    disabled     2' 0"
CU/SeeMe     udp    7648    enabled       unavailable
RAUDIO(PNA)  tcp    7070    enabled       unavailable
RTSP         tcp    554     enabled       unavailable
ILS          tcp    389     unavailable  5' 0"
H245         tcp    0        unavailable  5' 0"
H323         tcp    1720    enabled       unavailable
IRC          tcp    6667    enabled       5' 0"
LOOSE(UDP)   udp    0        enabled       5' 0"
FTP          tcp    21      enabled       unavailable
=>
```



For some CONN/NAT application helpers, either **traces** or **timeout** are unavailable.

RELATED COMMANDS:

connection appconfig Configure the available CONN/NAT application helpers.

connection appinfo Display CONN/NAT application specific info.

Connection Commands

connection bind

Create a new CONN/NAT application helper/port binding.

SYNTAX:

```
connection bind  application = <string>
                  port = <port-range>
```

where:

application	The name of a CONN/NAT application helper.	REQUIRED
Tip	Use the command :connection applist to obtain a list of CONN/NAT application helpers.	
port	The port number or port range this application handler should work on.	REQUIRED

EXAMPLE:

```
=>connection bindlist
Application Proto Portrange Flags
JABBER      tcp   15222
JABBER      tcp   5222
FTP         tcp   21
IRC         tcp   6660
...
IP6TO4     6to4  0
=>connection bind application = IRC port = 6750
=>connection bindlist
Application Proto Portrange
IRC         tcp   6750
JABBER      tcp   15222
JABBER      tcp   5222
FTP         tcp   21
IRC         tcp   6660
...
IP6TO4     6to4  0
=>
```

RELATED COMMANDS:

connection bindlist	List the current CONN/NAT application helper/port bindings.
connection unbind	Delete an existing CONN/NAT application helper/port binding.

connection bindlist

List the current CONN/NAT application helper/port bindings.

SYNTAX:

```
connection bindlist
```

EXAMPLE:

```
=>connection bindlist
Application Proto Portrange Flags
LOOSE (UDP)    udp    67
JABBER        tcp    15222
JABBER        tcp    5222
FTP            tcp    21
IRC            tcp    6660-6669
H323           tcp    1720
ILS            tcp    1002
ILS            tcp    389
RTSP           tcp    554
RAUDIO (PNA)   tcp    7070
CU/SeeMe       udp    7648
SIP            udp    5060
IKE             udp    500
ESP             esp    0
PPTP           tcp    1723
IP6TO4         6to4   0
=>
```

RELATED COMMANDS:

- | | |
|--------------------------|--|
| connection bind | Create a new CONN/NAT application helper/port binding. |
| connection unbind | Delete an existing CONN/NAT application helper/port binding. |

connection clean

Clean the connection database by forcing timeouts.

SYNTAX:

```
connection clean [level = <number{0-9}>]
```

where:

level A number between 0 and 9. Represents the desired scrubbing level.	OPTIONAL
--	----------

EXAMPLE:

```
=>connection list
ID proto state      substate    flags   timeout
--  ----- -----
8   tcp   ACTIVE     [TCPS_ESTABLISHED-TCPS_ESTABLISHED] [.....] 15' 7"
  INIT: 16 192.168. 1. 64: 1377 192.168. 1.254: 23 [...] LocalNetwork 1390 tcp      0
  RESP: 17 192.168. 1.254: 23 192.168. 1. 64: 1377 [R..] loop      951 tcp      0
34  tcp   ACTIVE     [TCPS_CLOSE_WAIT-TCPS_FIN_WAIT_1] [I.....] 57' 16"
  INIT: 68 192.168. 1. 64: 1417 192.168. 1.254: 21 [...] LocalNetwork 11 tcp      0
  RESP: 69 192.168. 1.254: 21 192.168. 1. 64: 1417 [R..] loop      10 tcp      0
=>connection clean
=>connection list
ID proto state      substate    flags   timeout
--  ----- -----
8   tcp   ACTIVE     [TCPS_ESTABLISHED-TCPS_ESTABLISHED] [.....] 14' 59"
  INIT: 16 192.168. 1. 64: 1377 192.168. 1.254: 23 [...] LocalNetwork 1417 tcp      0
  RESP: 17 192.168. 1.254: 23 192.168. 1. 64: 1377 [R..] loop      967 tcp      0
=>
```

connection clear

Kill all the connections.

SYNTAX:

```
connection clear
```

EXAMPLE:

```
=>connection clear
```

```
Connection to host lost.
```

Connection Commands

connection config

Configure the connection handling.

SYNTAX:

```
connection config [configchangemode = <{immediate | delayed}>]
                  [probes = <{disabled | enabled}>]
                  [udptrackmode = <{strict | loose}>]
```

where:

configchangemode	Select how configuration changes are handled. Choose between: ▶ immediate ▶ delayed. The default is immediate .	OPTIONAL
probes	Enable or disable live probes on idle connections. The default is disabled .	OPTIONAL
udptrackmode	Select the User Datagram Protocol (UDP) connection tracking mode. Choose between: ▶ strict : replies to a request from a client must be in a specific window to the client. ▶ loose : inbound packets are allowed on the port that was first used to start the communication with the server (for example to allow a client of an online game to obtain peer-to-peer information from other clients of that same online game). The default is strict .	OPTIONAL

EXAMPLE:

```
=>connection config
config change mode : immediate
alive probes      : disabled
udp tracking mode : loose
=>
```

connection debug

The connection debug commands

SYNTAX:

```
connection debug [trace = <{disabled|enabled}>]
```

where:

trace	Enable or disable traces. The default is disabled .	OPTIONAL
-------	---	----------

EXAMPLE:

```
=>connection debug
connection traces : disabled
=>
=>connection debug trace enabled
=>
=>connection debug
connection traces : enabled
=>
```

connection describe

Describe the streams of a connection.

SYNTAX:

```
connection describe [id = <number{0-2048}>]
```

where:

id	A number between 0 and 2048.	OPTIONAL
Represents the ID of the connection to be described.		
Note	If not specified, the connection with ID 0 will be described.	
Tip	Use the command :connection list to obtain the IDs of the different connections.	

EXAMPLE:

```
=>connection list
ID proto state      substate    flags   timeout
--  ----- -----
62  tcp   ACTIVE     [TCPS_ESTABLISHED-TCPS_ESTABLISHED] [...]
INIT: 124 192.168. 1. 64: 1979 192.168. 1.254: 23 [...] LocalNetwork 548 tcp      0
RESP: 125 192.168. 1.254: 23 192.168. 1. 64: 1979 [R..] loop      396 tcp      0
=>connection describe id=62
ID proto state      substate    flags   timeout
--  ----- -----
62  tcp   ACTIVE     [TCPS_ESTABLISHED-TCPS_ESTABLISHED] [...]
FW          : cache = valid; FP
IDS         : ...
NAT         : cache = valid; No translation
INIT: 124 192.168. 1. 64: 1979 192.168. 1.254: 23 [...] LocalNetwork 576 tcp      0
ROUTING     : cache = valid; FP (gateway 127.0.0.1)
LABEL       : cache = valid; FP (no route label); FP (QoS label Interactive)
IPQOS      : cache = valid; FP (label <no meter>, intf <no meter>)
TRIGGER    : cache = valid; FP (no trigger)
RESP: 125 192.168. 1.254: 23 192.168. 1. 64: 1979 [R..] loop      412 tcp      0
ROUTING     : cache = valid; FP (gateway 192.168.1.254)
LABEL       : cache = valid; FP (no route label); FP (QoS label default)
IPQOS      : cache = valid; FP (label <no meter>, intf <no meter>)
TRIGGER    : cache = valid; FP (no trigger)
=>
```

connection flush

Flush the current connection configuration.

SYNTAX:

```
connection flush
```

connection info

Show all the registered modules with some info.

SYNTAX:

```
connection info
```

EXAMPLE:

```
=>connection info
Registered connection modules :
  - Module : FW, holds private data (F:10264 S:6592).
  - Module : IDS, holds private data (F:0 S:0).
  - Module : NAT, holds private data (F:0 S:0).
Registered stream modules :
  - Module : ROUTING, holds private data (F:10199 S:6657).
  - Module : LABEL, holds private data (F:22546 S:19870).
  - Module : IPQOS, holds private data (F:10202 S:6653).
  - Module : TRIGGER, holds private data (F:10202 S:6659).
=>
```

connection list

Display the currently known connections.

SYNTAX:

```
connection list [nr = <number{1-2048}>]  
[history = <{disabled|enabled}>]
```

where:

nr	A number between 1 and 2048. Represents the number of connections to be displayed.	OPTIONAL
	Note If not specified, all the connections will be displayed.	
history	Enable or disable history display. The default is disabled .	OPTIONAL

EXAMPLE:

```
=>connection list  
ID proto state      substate    flags   timeout  
-- ----- --      -----  
58  tcp  ACTIVE      [TCPS_ESTABLISHED-TCPS_ESTABLISHED] [.....] 15' 7"  
    INIT: 116 10. 0. 0. 1: 1106 10. 0. 0.138: 23 [.]. eth0 331 tcp 0  
    RESP: 117 10. 0. 0.138: 23 10. 0. 0. 1: 1106 [R] loop 229 tcp 0  
=>
```

connection refresh

Invalidate all the cached decisions.

SYNTAX:

```
connection refresh
```

connection stats

Display the connection and stream statistics.

SYNTAX:

```
connection stats
```

EXAMPLE:

```
=>connection stats
Connection statistics:
-----
Maximum number of connections      : 1024
Maximum number of halfopen connections : 1024
-----
Number of active connections        : 3
Number of halfopen connections     : 0
Number of expected connections    : 0
Number of closing connections     : 0
Number of idle connections        : 1
-----
Number of TCP connections          : 2
Number of UDP connections         : 1
Number of ICMP connections        : 0
Number of non TCP/UDP/ICMP connections : 0
-----
Number of TCP open connections    : 0
Number of TCP established connections : 1
Number of TCP closing connections : 1
-----
Stream cache statistics:
-----
Maximum number of hash collisions   : 0
% of hash entries with collisions  : 0.00
% of hash entries unused           : 0.00
-----
CONN/NAT application helper statistics:
-----
Maximum number of helper bindings   : 24
Maximum number of connections with helper : 128
-----
Number of helper bindings          : 16
Number of connections with active helper : 0
=>
```

Connection Commands

connection timerclear

Clear the connection timeout to default.

SYNTAX:

```
connection timerclear [timer = <{tcpidle|tcpneg|tcpkill|udpidle|udpkill|
                           icmpkill|ipidle|ipkill}>]
```

where:

timer	The name of the connection idle timer to be reset.	REQUIRED
-------	--	----------

Note If not specified, all the timers will be reset to their default values.

EXAMPLE:

```
=>connection timerconfig
tcpidle      : 10' 30"
tcpneg       : 3' 0"
udp          : 1' 19"
icmp         : 2' 0"
ip           : 1' 0"
=>connection timerclear
=>connection timerconfig
tcpidle      : 15' 0"
tcpneg       : 2' 0"
udp          : 1' 0"
icmp         : 1' 0"
ip           : 1' 0"
=>
```

RELATED COMMANDS:

connection timerconfig Configure the connection timeout handling.

connection timerconfig

Configure the connection timeout handling.

SYNTAX:

```
connection timerconfig [timer = <{tcpidle|tcpneg|tcpkill|udpidle|udppkill|
                           icmpkill|ipidle|ipkill}>]
                        [value = <number{0-86400}>]
```

where:

timer	The name of the connection idle timer to be configured. Choose between: <ul style="list-style-type: none">▶ tcpidle▶ tcpneg▶ tcpkill▶ udpidle▶ udppkill▶ icmpkill▶ ipidle▶ ipkill	OPTIONAL
value	A number between 0 and 86400 (seconds). Represents the timer expire value.	OPTIONAL

EXAMPLE:

```
=>connection timerconfig
tcpidle      : 15' 0"
tcpneg       : 2' 0"
udp          : 1' 0"
icmp         : 1' 0"
ip           : 1' 0"
=>connection timerconfig timer=tcpidle value=360
=>connection timerconfig
tcpidle      : 6' 0"
tcpneg       : 2' 0"
udp          : 1' 0"
icmp         : 1' 0"
ip           : 1' 0"
=>
```

RELATED COMMANDS:

connection timerclear Clear the connection timeout to default.

connection unbind

Delete an existing CONN/NAT application helper/port binding.

SYNTAX:

```
connection unbind application = <string>
                  port = <port-range>
```

where:

application	The name of a CONN/NAT application helper.	REQUIRED
Tip	Use the command :connection applist to obtain a list of CONN/NAT application helpers.	
port	The port number or port range this application handler should work on.	REQUIRED

EXAMPLE:

```
=>connection bindlist
Application Proto Portrange
IRC          tcp   6750
JABBER       tcp   15222
JABBER       tcp   5222
FTP          tcp   21
IRC          tcp   6660
...
IP6TO4      6to4  0
=>connection unbind application=IRC port=6750
=>connection bindlist
Application Proto Portrange
JABBER       tcp   15222
JABBER       tcp   5222
FTP          tcp   21
IRC          tcp   6660
...
IP6TO4      6to4  0
=>
```

RELATED COMMANDS:

connection bind	Create a new CONN/NAT application helper/port binding.
connection bindlist	List the current CONN/NAT application helper/port bindings.

CWMP Commands

Introduction

This chapter describes the commands of the **cwmp** (CPE WAN Management Protocol) command group.

Contents

This chapter covers the following commands:

cwmp config	Configure the cwmpd as seen from the ACS.	96
cwmp debug traceconfig	Enable or disable tracing.	98
cwmp server config	Configure the cwmpd towards the ACS.	99

cwmp config

Configure the cwmpd as seen from the ACS.

SYNTAX:

```
cwmp config [state = <{disabled|enabled}>]
            [mode = <{readonly|full}>]
            [periodicInform = <{disabled|enabled}>]
            [periodicInflnt = <number>]
            [sessionTimeout = <number>]
            [noIpTimeout = <number>]
            [maxEnvelopes = <number>]
            [connectionRequest = <{disabled|enabled}>]
            [connectionReqPath = <string>]
            [connectionReqUserName = <string>]
            [connectionReqPsswd = <string>]
            [connectionReqAuth = <{none|basic|digest}>]
            [qos-class = <number>]
            [bootdelayrange = <number>]
```

where:

state	The state of the cwmp daemon. Choose between: ▶ enabled ▶ disabled. The default is disabled .	OPTIONAL
mode	Set the operation mode of the cwmp daemon. Choose between: ▶ readonly ▶ full. The default is readonly .	OPTIONAL
periodicInform	Set the periodicInform flag of the cwmp daemon. Choose between: ▶ enabled ▶ disabled. The default is enabled .	OPTIONAL
periodicInflnt	A number (of seconds). Represents the interval between two periodicInform messages. The default is 3600 .	OPTIONAL
sessionTimeout	Set HTTP session-timeout in seconds. The default is 60 .	OPTIONAL
noIpTimeout	Set time (in seconds) ip may be 0 after uploading new config file. The default is 10 .	OPTIONAL
maxEnvelopes	Set the maximum number of SOAP envelopes sent within one HTTP message. The default is 2 .	OPTIONAL

connectionRequest	Set the connection request flag of the cwmp daemon. Choose between: ▶ enabled ▶ disabled. The default is <i>enabled</i> .	OPTIONAL
connectionReqPath	Set the path where the cwmp daemon can be reached.	OPTIONAL
connectionReqUserName	Set the username the ACS must use to log in.	OPTIONAL
connectionReqPsswd	Set the password the ACS must use to log in.	OPTIONAL
connectionReqAuth	Set the digest authentication flag of the cwmp daemon. Choose between: ▶ enabled ▶ disabled. The default is <i>enabled</i> .	OPTIONAL
qos-class	Set the quality of service class for outgoing CWMP data. The default is <i>12</i> .	OPTIONAL
bootdelayrange	Set the delay on boot (in seconds) before inform is sent. The default is <i>0</i> .	OPTIONAL

EXAMPLE:

```
=>cwmp config
State : disabled
Mode : readonly
Max Envelopes : 2
Session Timeout : 60
No Ip Timeout : 10
Connection Request Port : 80
Periodic Inform : enabled
Periodic Inform Interval : 20000 ms
Connection Request : enabled
Connection Request UserName :
Connection Request Password :
Connection Request Path :
Connection Request Authentication : enabled
Qos class : 12
Boot delay range between 0 and : 0s
=>
```

cwmp debug traceconfig

Enable or disable tracing.

SYNTAX:

```
cwmp debug traceconfig [level = <number>]
```

where:

level	Set the tracelevel. Choose a higher level for more detailed tracing.	OPTIONAL
-------	--	----------

cwmp server config

Configure the cwmpd towards the ACS.

SYNTAX:

```
cwmp server config [url = <string>]  
                  [username = <string>]  
                  [password = <string>]
```

where:

url	Set the HTTP URL used to contact the ACS server.	OPTIONAL
username	Set the username for ACS Digest Authentication.	OPTIONAL
password	Set the password for ACS Digest Authentication.	OPTIONAL

EXAMPLE:

```
=>cwmp server config  
ACS url      : 10.11.10.248  
ACS username  :  
ACS password  :  
=>
```


Debug Commands

Introduction

This chapter describes the commands of the **debug** command group.

Contents

This chapter covers the following commands:

debug exec	Execute a 'Trace & Debug' command.	102
------------	------------------------------------	-----

debug exec

Execute a 'Trace & Debug' command.



This command is for qualified personnel only.

SYNTAX:

```
debug exec cmd = <quoted string>
```

where:

cmd A quoted 'Trace & Debug' command string.

REQUIRED

DHCP Commands

Introduction

This chapter describes the commands of the Dynamic Host Configuration Protocol (DHCP) command group.

Contents

This chapter covers the following commands:

dhcp client flush	Delete all the DHCP leases.	105
dhcp client ifadd	Create a DHCP client.	106
dhcp client ifattach	Activate a DHCP client.	107
dhcp client ifconfig	Configure a DHCP client.	108
dhcp client ifdelete	Delete a DHCP client.	109
dhcp client ifdetach	De-activate a DHCP client and releases its lease.	110
dhcp client iflist	List all the DHCP leases attached to dynamic interfaces.	111
dhcp client ifrenew	Renew the DHCP lease.	113
dhcp client debug clear	Clear the DHCP client statistics.	115
dhcp client debug stats	Print the DHCP client statistics.	116
dhcp client debug traceconfig	Modify the DHCP client trace configuration.	117
dhcp client roptions add	Add a DHCP Option Code to the Parameter Request List.	118
dhcp client roptions delete	Delete a DHCP Option Code from the Parameter Request List.	119
dhcp client roptions list	List all DHCP Option Codes in the Parameter Request List.	120
dhcp client roptions optionlist	List all DHCP Option Codes that can be used in the Parameter Request List.	121
dhcp client txoptions add	Add an option.	122
dhcp client txoptions delete	Delete an option.	123
dhcp client txoptions list	List all options.	124
dhcp client txoptions optionlist	Lists all DHCP Option Codes that can be used.	125
dhcp relay add	Add an entry to the DHCP forward list.	126
dhcp relay debug stats	Show the DHCP relay statistics.	127
dhcp relay debug traceconfig	Modify the DHCP relay trace configuration.	128
dhcp relay config	Set the DHCP relay configuration settings.	129
dhcp relay delete	Delete an entry from the DHCP forward list.	130
dhcp relay flush	Flush the DHCP relay settings.	131
dhcp relay ifconfig	Configure a DHCP relay interface.	132
dhcp relay iflist	Show the configuration of the relay interfaces.	134
dhcp relay list	List the DHCP forward list.	135
dhcp relay modify	Modify an entry from the DHCP forward list.	136
dhcp relay ruleadd	Add a selection rule to a DHCP forward entry.	137

DHCP Commands

dhcp relay ruledelete	Delete a selection rule from a DHCP forward entry.	138
dhcp rule add	Add a rule for DHCP conditional selection.	139
dhcp rule debug traceconfig	Modify DHCP rule trace configuration.	140
dhcp rule delete	Delete a DHCP rule.	141
dhcp rule flush	Flush all DHCP rules.	142
dhcp rule list	List all DHCP rules.	143
dhcp server config	Print the DHCP server configuration settings.	144
dhcp server flush	Flush all DHCP server pool and lease entries.	145
dhcp server policy	Print the DHCP server policy settings.	146
dhcp server debug clear	Clear the DHCP server statistics.	147
dhcp server debug stats	Show the DHCP server statistics.	148
dhcp server debug traceconfig	Modify the DHCP server trace configuration.	150
dhcp server lease add	Add a DHCP server lease.	151
dhcp server lease delete	Delete a DHCP server lease.	153
dhcp server lease flush	Flush all the DHCP server leases.	154
dhcp server lease list	List all the DHCP server leases.	155
dhcp server option flush	Flush all DHCP server option templates and instances.	156
dhcp server option instadd	Add a DHCP server option instance.	157
dhcp server option instdelete	Delete a DHCP server option instance.	158
dhcp server option instlist	List all the DHCP server option instances.	159
dhcp server option tmpladd	Add a DHCP server option template.	160
dhcp server option tmpldelete	Delete a DHCP server option template.	161
dhcp server option tmpllist	List all the DHCP server option templates.	162
dhcp server pool add	Add a DHCP server pool.	163
dhcp server pool config	Configure a DHCP server pool.	164
dhcp server pool delete	Delete a DHCP server pool.	166
dhcp server pool flush	Flush all DHCP server pools.	167
dhcp server pool list	List all DHCP server pools.	168
dhcp server pool optadd	Add an option instance to the DHCP server pool.	169
dhcp server pool optdelete	Delete an option instance from the DHCP server pool.	170
dhcp server pool rtadd	Add a route to the DHCP server pool.	171
dhcp server pool rtdelete	Delete a route from the DHCP server pool.	172
dhcp server pool ruleadd	Add a selection rule to the DHCP server pool.	173
dhcp server pool ruledelete	Delete a selection rule from the DHCP server pool.	174

dhcp client flush

Delete all the DHCP leases.



The flush command does not impact previously saved configurations.

SYNTAX:

```
dhcp client flush
```

dhcp client ifadd

Create a DHCP client.

SYNTAX:

```
dhcp client ifadd intf = <string>
[defrqloptions = <{disabled|enabled}>]
```

where:

intf	The name of the dynamic interface for which a DHCP lease must be created.	REQUIRED
defrqloptions	Enable/Disable the creation of the default Parameter Request List.	OPTIONAL

EXAMPLE:

```
=>dhcp client iflist
No dynamic interfaces defined.
=>dhcp client ifadd intf=myPPP_ppp
[dhcp client]=>iflist
DHCP Client Info :

      Interface      : myPPP_ppp
      DHCP Client State : [INIT]
      HW address     : 00:0e:50:91:26:5a
      Client ID      :
      IP Address      : 0.0.0.0
      p-t-p IP        : 0.0.0.0
      Default Gateway : 

Number of leases: 1
Total size of table: 68, in use: 1, free: 98 %=>
```

RELATED COMMANDS:

dhcp client ifdelete	Delete a DHCP client.
dhcp client iflist	List all the DHCP leases attached to dynamic interfaces.
interface list	Display interfaces.

dhcp client ifattach

Activate a DHCP client.

SYNTAX:

```
dhcp client ifattach intf = <string>
```

where:

intf	The name of the dynamic interface for which a DHCP client was created.	REQUIRED
------	--	----------

EXAMPLE:

```
=>dhcp client iflist
myPPP_ppp : [INIT]
    flags = bc dns rt
    IP address : 0.0.0.0
    HW address [SpeedTouch] : 00
    DHCP server: 255.255.255.255
    metric : rt = 1, DNS = 1

Number of leases: 1
Total size of table: 36, in use: 1, free: 97 %
=>dhcp client ifattach intf=myPPP_ppp
=>dhcp client iflist
myPPP_ppp : [SELECTING]
    flags = bc dns rt
    IP address : 0.0.0.0
    HW address [SpeedTouch] : 00
    DHCP server: 255.255.255.255
    metric : rt = 1, DNS = 1
    trying to get a lease for 1 sec
    transmission of DISCOVER in 0 sec
    retransmission timeout: 4
    nbr of retransmissions: 2

Number of leases: 1
Total size of table: 36, in use: 1, free: 97 %
=>
```

RELATED COMMANDS:

dhcp client ifadd	Create a DHCP client.
dhcp client ifdetach	De-activate a DHCP client and releases its lease.
dhcp client ifrenew	Renew the DHCP lease.
interface list	Display interfaces.

dhcp client ifconfig

Configure a DHCP client.

 Use the command :**dhcp client ifrelease** before configuring the dhcp client.

SYNTAX:

```
dhcp client ifconfig intf = <string>
    [label = <label name>]
    [metric = <number{0-255}>]
    [dnsmetric = <number{0-100}>]
    [broadcast = <{disabled|enabled}>]
```

where:

intf	The name of the dynamic interface to be configured.	REQUIRED
label	Label for default gateway and static routes.	OPTIONAL
metric	A number between 0 and 255. Represents the route metric for default gateway and static routes. The default is 1 .	OPTIONAL
dnsmetric	A number between 0 and 100. Represents the DNS route metric. The default is 1 .	OPTIONAL
broadcast	Operate client in unicast/broadcast mode. The default is enabled .	OPTIONAL

RELATED COMMANDS

label add Create a new label.

dhcp client ifdelete

Delete a DHCP client.

SYNTAX:

```
dhcp client ifdelete intf = <string>
```

where:

intf	The name of the dynamic interface for which the DHCP lease must be deleted.	REQUIRED
------	---	----------

EXAMPLE:

```
=>dhcp client iflist
myPPP_ppp : [INIT]
    flags = bc dns rt
    IP address : 10.0.0.1
    HW address [SpeedTouch] : 00:90:d0:01:47:f1
    DHCP server: 10.10.1.1
    hostname : myHostName
    client identifier : [00] myClientID
    user class identifier : myUserID
    metric : rt = 1, DNS = 1

Number of leases: 1
Total size of table: 36, in use: 1, free: 97 %
=>dhcp client ifdelete intf=myPPP_ppp
=>dhcp client iflist
No dynamic interfaces defined.
=>
```

RELATED COMMANDS:

dhcp client ifadd

Create a DHCP client.

dhcp client iflist

List all the DHCP leases attached to dynamic interfaces.

dhcp client ifdetach

De-activate a DHCP client and releases its lease.

SYNTAX:

```
ifdetach           intf = <string>
```

where:

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

RELATED COMMANDS:

dhcp client ifattach	Activate a DHCP client.
dhcp client ifrenew	Renew the DHCP lease.
dhcp client iflist	List all the DHCP leases attached to dynamic interfaces.

dhcp client iflist

List all the DHCP leases attached to dynamic interfaces.

SYNTAX:

```
dhcp client iflist      [intf = <string>]
                      [expand = <{disabled|enabled}>]
```

where:

intf	The name of the dynamic interface.	OPTIONAL
expand	Expand enabled/disabled.	OPTIONAL

EXAMPLE:

```
=>dhcp client iflist
myPPP_ppp : [INIT]
  flags = bc dns rt
  IP address : 10.0.0.1
  HW address [SpeedTouch] : 00:90:d0:01:47:f1
  DHCP server: 10.10.1.1
  hostname : myHostName
  client identifier : [00] myClientID
  user class identifier : myUserID
  metric : rt = 1, DNS = 1

Number of leases: 1
Total size of table: 36, in use: 1, free: 97 %
=>
```

EXAMPLE INPUT/OUTPUT IN A NETWORKED ENVIRONMENT:

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

```
=>dhcp client iflist
myPPP_ppp : [INIT]
  flags = bc dns rt
  IP address : 10.0.0.1
  HW address [SpeedTouch] : 00:90:d0:01:47:f1
  DHCP server: 10.10.1.1
  hostname : myHostName
  client identifier : [00] myClientID
  user class identifier : myUserID
  metric : rt = 1, DNS = 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: 36, in use: 1, free: 97 %
=>
```

RELATED COMMANDS:

dhcp client ifadd Create a DHCP client.

DHCP Commands

dhcp client ifdelete Delete a DHCP client.

dhcp client ifrenew

Renew the DHCP lease.

SYNTAX:

```
dhcp client ifrenew intf = <string>
```

where:

intf	The name of the dynamic interface for which the DHCP lease must be renewed.	REQUIRED
------	---	----------

EXAMPLE 1:

```
=>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.leasetime   = 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.leasetime   = 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 %
=>
```

DHCP Commands

EXAMPLE 2:

The SpeedTouch™ is configured as DHCP client disabled 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 ifattach Activate a DHCP client.

dhcp client debug clear

Clear the DHCP client statistics.

SYNTAX:

```
dhcp client debug clear
```

EXAMPLE:

```
=>dhcp client debug 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 debug clear
=>dhcp client debug 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 debug stats Print the DHCP client statistics.

dhcp client debug stats

Print the DHCP client statistics.

SYNTAX:

```
dhcp client debug stats
```

EXAMPLE:

```
=>dhcp client debug 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           :        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 debug clear Clear the DHCP client statistics.

dhcp client debug traceconfig

Modify the DHCP client trace configuration.

SYNTAX:

```
dhcp client debug traceconfig [state = <{disabled | enabled}>]
```

where:

state Enable or disable tracing. The default is disabled .	OPTIONAL
---	----------

EXAMPLE:

```
=>dhcp client debug traceconfig  
tracing: disabled  
=>dhcp client debug traceconfig trace=enabled  
=>dhcp client debug traceconfig  
tracing: enabled  
=>
```

dhcp client rqoptions add

Add a DHCP Option Code to the Parameter Request List.

SYNTAX:

```
dhcp client rqoptions add intf = <string>
                           option = <string or number>
                           [index = <number{0-255}>]
```

where:

intf	The name of the dynamic interface.	REQUIRED
option	The name or number of the option.	OPTIONAL
index	The index of the option.	OPTIONAL

RELATED COMMANDS:

dhcp client iflist	List all the DHCP leases attached to dynamic interfaces.
dhcp client rqoptions optionlist	List all DHCP Option Codes that can be used in the Parameter Request List.

dhcp client roptions delete

Delete a DHCP Option Code from the Parameter Request List.

SYNTAX:

```
dhcp client roptions delete    intf = <string>
                                option = <string or number>
```

where:

intf	The name of the dynamic interface.	REQUIRED
option	The name or number of the option.	OPTIONAL

RELATED COMMANDS:

dhcp client roptions list List all DHCP Option Codes in the Parameter Request List.

dhcp client rqoptions list

List all DHCP Option Codes in the Parameter Request List.

SYNTAX:

```
dhcp client rqoptions list [intf = <string>]
```

where:

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

RELATED COMMANDS:

dhcp client rqoptions delete Delete a DHCP Option Code from the Parameter Request List.

dhcp client roptions optionlist

List all DHCP Option Codes that can be used in the Parameter Request List.

SYNTAX:

```
dhcp client roptions optionlist
```

RELATED COMMANDS:

dhcp client roptions add	Add a DHCP Option Code to the Parameter Request List.
dhcp client roptions list	List all DHCP Option Codes in the Parameter Request List.

dhcp client txoptions add

Add an option.

SYNTAX:

```
dhcp client txoptions add      intf = <string>
                               option = <string or number>
                               value = <Value : (type)value; type being 8-bit,
                                         16-bit, 32-bit, addr, ascii, byte_array,
                                         clientid>
                               [index = <number{0-255}>]
```

where:

intf	The name of the dynamic interface.	REQUIRED
option	The name or number of the option.	REQUIRED
value	The value of the option.	REQUIRED
index	The index of the option.	OPTIONAL

EXAMPLE:

```
=>:dhcp client txoptions add intf=wan1 option=ien116-name-servers value=(8-bit)5
```

RELATED COMMANDS:

dhcp client iflist	List all the DHCP leases attached to dynamic interfaces.
dhcp client txoptions optionlist	Lists all DHCP Option Codes that can be used.

dhcp client txoptions delete

Delete an option.

SYNTAX:

```
dhcp client txoptions delete    intf = <string>
                                option = <string or number>
```

where:

intf	The name of the dynamic interface.	REQUIRED
option	The name or number of the option.	REQUIRED

RELATED COMMANDS:

dhcp client txoptions list List all options.

dhcp client txoptions list

List all options.

SYNTAX:

```
dhcp client txoptions list
```

RELATED COMMANDS:

dhcp client txoptions delete Delete an option.

dhcp client txoptions optionlist

Lists all DHCP Option Codes that can be used.

SYNTAX:

```
dhcp client txoptions optionlist
```

RELATED COMMANDS:

dhcp client txoptions add	Add an option.
dhcp client txoptions list	List all options.

dhcp relay add

Add an entry to the DHCP forward list.

SYNTAX:

```
dhcp relay add name = <string>
```

where:

name	The forward entry name.	REQUIRED
------	-------------------------	----------

RELATED COMMANDS:

dhcp relay delete	Delete an entry from the DHCP forward list.
dhcp relay list	List the DHCP forward list.
dhcp relay modify	Modify an entry from the DHCP forward list.

dhcp relay debug stats

Show the DHCP relay statistics.

SYNTAX:

```
dhcp relay debug stats
```

EXAMPLE:

```
=>dhcp relay debug 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
=>
```

RELATED COMMANDS:

dhcp relay debug traceconfig	Modify the DHCP relay trace configuration.
---	--

dhcp relay debug traceconfig

Modify the DHCP relay trace configuration.

SYNTAX:

```
dhcp relay debug traceconfig [state = <{disabled | enabled}>]
```

where:

state Enable or disable tracing. The default is disabled .	OPTIONAL
---	----------

EXAMPLE:

```
=>dhcp relay debug traceconfig  
Tracing: disabled  
=>
```

RELATED COMMANDS:

dhcp relay debug stats Show the DHCP relay statistics.

dhcp relay config

Set the DHCP relay configuration settings.

SYNTAX:

```
dhcp relay config [agentinfo = <{disabled | enabled}>]  
[agentmismatch = <{disabled | enabled}>]
```

where:

agentinfo	Set the relay agent info status (RFC3046) enabled or disabled. The default is disabled .	OPTIONAL
agentmismatch	Forward/drop DHCP reply packet when a relay agent info mismatch is detected (RFC3046) (enabled) or not (disabled). The default is disabled .	OPTIONAL

EXAMPLE:

```
=>dhcp relay config  
Agent info status : disabled  
Drop agent info mismatch status : disabled  
=>dhcp relay config agentinfo=enabled  
=>dhcp relay config  
Agent info status : enabled  
Drop agent info mismatch status : disabled  
=>
```

dhcp relay delete

Delete an entry from the DHCP forward list.

SYNTAX:

```
dhcp relay delete name = <string>
```

where:

name The forward entry name.	REQUIRED
---------------------------------	----------

EXAMPLE:

```
=>dhcp relay delete  
name = lan1_to_127.0.0.1  
:dhcp relay delete name=lan1_to_127.0.0.1  
=>
```

RELATED COMMANDS:

dhcp relay add	Add an entry to the DHCP forward list.
dhcp relay list	List the DHCP forward list.
dhcp relay modify	Modify an entry from the DHCP forward list.

dhcp relay flush

Flush the DHCP relay settings.



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        eth0          192.168.1.254
=>dhcp relay flush
=>dhcp relay list
No dynamic interfaces defined.
=>
```

dhcp relay ifconfig

Configure a DHCP relay interface.

SYNTAX:

```
dhcp relay ifconfig intf = <string>
    [relay = <{disabled | enabled}>]
    [maxhops = <number{0-16}>]
    [remoteid = <password>]
    [trusted = <{disabled | enabled}>]
```

where:

intf	The name of the dynamic interface to be configured.	REQUIRED
relay	Set the relay status to enabled or disabled. The default is disabled .	OPTIONAL
maxhops	A number between 0 and 16. Represents the maximum number of hops allowed in the DHCP packet. The default is 4 .	OPTIONAL
remoteid	Set 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=enabled) and the giaddr field is 0 (RFC3046) (enabled) or not (disabled). The default is disabled .	OPTIONAL

EXAMPLE:

```
=>dhcp relay iflist
eth0 :
    admin state = up      oper state = up
    max hops = 4
    trusted = disabled   remote id =
Internet_trigger :
    admin state = down    oper state = down
    max hops = 4
    trusted = disabled   remote id =
Internet :
    admin state = down    oper state = down
    max hops = 4
    trusted = disabled   remote id =
=>dhcp relay ifconfig intf=Internet_trigger relay=enabled trusted=enabled
=>dhcp relay iflist
eth0 :
    admin state = up      oper state = up
    max hops = 4
    trusted = disabled   remote id =
Internet_trigger :
    admin state = up      oper state = up
    max hops = 4
    trusted = enabled    remote id =
Internet :
    admin state = down    oper state = down
    max hops = 4
    trusted = disabled   remote id =
=>
```

dhcp relay iflist

Show the configuration of the relay interfaces.

SYNTAX:

```
dhcp relay iflist [intf = <string>]
```

where:

intf	The name of the dynamic interface to be listed.	OPTIONAL
Note	If not specified, all the relay interfaces will be shown.	

EXAMPLE:

```
=>dhcp relay iflist
eth0 :
    admin state = up      oper state = up
    max hops = 4
    trusted = disabled    remote id =
Internet_trigger :
    admin state = up      oper state = up
    max hops = 4
    trusted = enabled     remote id =
Internet :
    admin state = down    oper state = down
    max hops = 4
    trusted = disabled    remote id =
=>
```

dhcp relay list

List the DHCP forward list.

SYNTAX:

```
dhcp relay list [name = <string>]
```

where:

name	The forward entry name.	OPTIONAL
------	-------------------------	----------

EXAMPLE:

```
=>dhcp relay list
  DHCP server      Interface      giaddr
  -----
  127.0.0.1        eth0          10.0.0.138
=>
```

RELATED COMMANDS:

- | | |
|-------------------|---|
| dhcp relay add | Add an entry to the DHCP forward list. |
| dhcp relay delete | Delete an entry from the DHCP forward list. |
| dhcp relay modify | Modify an entry from the DHCP forward list. |

dhcp relay modify

Modify an entry from the DHCP forward list.

SYNTAX:

```
dhcp relay modify    name = <string>
                     [addr = <ip-address>]
                     [intf = <{None | guest1 | dmz1 | wan1 | lan1 | Internet}>]
                     [giaddr = <ip-address>]
                     [script = <>]
```

where:

name	The forward entry name.	REQUIRED
addr	The DHCP server IP address.	OPTIONAL
intf	The name of the relay interface, 'None' to indicate no interface is specified. The standard is None .	OPTIONAL
giaddr	The giaddr field to be used in relayed DHCP packets.	OPTIONAL
script	Script to be run when the forward entry is hit.	OPTIONAL

RELATED COMMANDS:

dhcp relay add	Add an entry to the DHCP forward list.
dhcp relay delete	Delete an entry from the DHCP forward list.
dhcp relay list	List the DHCP forward list.

dhcp relay ruleadd

Add a selection rule to a DHCP forward entry.

SYNTAX:

```
dhcp relay ruleadd    name = <string>
                      [key = <{or | and}>]
                      rulename = <string>
```

where:

name	The name of the forward entry.	REQUIRED
key	The logical key of the selection rule. The default is or .	OPTIONAL
rulename	The name of the DHCP selection rule.	REQUIRED

RELATED COMMANDS:

dhcp relay ruledelete Delete a selection rule from a DHCP forward entry.

dhcp relay ruledelete

Delete a selection rule from a DHCP forward entry.

SYNTAX:

```
dhcp relay ruledelete name = <string>
                           rulename = <string>
```

where:

name	The name of the forward entry.	REQUIRED
rulename	The name of the DHCP selection rule.	REQUIRED

RELATED COMMANDS:

dhcp relay ruleadd Add a selection rule to a DHCP forward entry.

dhcp rule add

Add a rule for DHCP conditional selection.

SYNTAX:

```
dhcp rule add name = <string>
               type = <{vci | uci | mac}>
               vci [!] = <quoted string>
               uci [!] = <quoted string>
               [match = <{exactly|as_substring}>]
               mac [!] = <hardware-address with wildcard
                           | ex: '00:9f:aa:*:*:*'>
```



If a value is preceded by a "!", it means NOT.

For example "mac!=00:9f:aa:bb:cc:dd" means "for MAC address different from 00:9f:aa:bb:cc:dd".

where:

name	The name of the new DHCP rule.	REQUIRED
type	Specify the DHCP rule type. Choose between: <ul style="list-style-type: none">▶ vci: vendor class identifier▶ uci: user class identifier▶ mac: MAC address.	REQUIRED
vci	The vendor class identifier string. Note Only required when type=vci.	REQUIRED
uci	The user class identifier string. Note Only required when type=uci.	REQUIRED
match	The vendor or user class identifier string matching.	OPTIONAL
mac	The MAC address. Note Only required when type=mac.	REQUIRED

RELATED COMMANDS:

dhcp rule delete	Delete a DHCP rule.
dhcp rule flush	Flush all DHCP rules.
dhcp rule list	List all DHCP rules.

dhcp rule debug traceconfig

Modify DHCP rule trace configuration.

SYNTAX:

```
dhcp rule debug traceconfig [state = {disabled | enabled}]
```

where:

state	Set tracing to disabled or enabled. The default is disabled .	OPTIONAL
-------	---	----------

dhcp rule delete

Delete a DHCP rule.

SYNTAX:

```
dhcp rule delete      name = <string>
```

where:

name	The name of the DHCP rule.	REQUIRED
------	----------------------------	----------

EXAMPLE:

```
=>dhcp rule delete
name = new
:dhcp rule delete name=new
=>
```

RELATED COMMANDS:

dhcp rule add	Add a rule for DHCP conditional selection.
dhcp rule flush	Flush all DHCP rules.
dhcp rule list	List all DHCP rules.

dhcp rule flush

Flush all DHCP rules.

SYNTAX:

```
dhcp rule flush
```

EXAMPLE:

```
=>dhcp rule list
Name           Use  Value
new            0    vci=test
=>dhcp rule flush
=>dhcp rule list
Name           Use  Value
=>
```

RELATED COMMANDS:

- | | |
|-------------------------|--|
| dhcp rule add | Add a rule for DHCP conditional selection. |
| dhcp rule delete | Delete a DHCP rule. |
| dhcp rule list | List all DHCP rules. |

dhcp rule list

List all DHCP rules.

SYNTAX:

```
dhcp rule list
```

EXAMPLE:

```
=>dhcp rule list
Name           Use  Value
new            0    vci=test
=>
```

RELATED COMMANDS:

- | | |
|------------------|--|
| dhcp rule add | Add a rule for DHCP conditional selection. |
| dhcp rule delete | Delete a DHCP rule. |
| dhcp rule flush | Flush all DHCP rules. |

dhcp server config

Print the DHCP server configuration settings.

SYNTAX:

```
dhcp server config [state = <{disabled | enabled}>]
```

where:

dhcp server flush

Flush all DHCP server pool and lease entries.



The flush command does not impact previously saved configurations.

SYNTAX:

```
dhcp server flush
```

dhcp server policy

Print the DHCP server policy settings.

SYNTAX:

```
dhcp server policy [verifyfirst = <disabled | enabled>]  
[trustclient = <disabled | enabled>]
```

where:

dhcp server debug clear

Clear the DHCP server statistics.

SYNTAX:

```
dhcp server debug clear
```

EXAMPLE:

```
=>dhcp server debug stats
DHCP server state: Running
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: 0 free: 100 %
=>dhcp server debug clear
=>dhcp server debug stats
DHCP server state: Running
DHCP server statistics:
Corrupted packet recv      :      0
DISCOVER                   :      0
REQUEST                    :      0
DECLINE                    :      0
RELEASE                    :      0
INFORM                     :      0
Pure BOOTP REQUESTS       :      0
Other message types        :      0
OFFERs sent                :      0
ACKs sent                  :      0
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: 0 free: 100 %
=>
```

RELATED COMMANDS:

dhcp server debug stats Show the DHCP server statistics.

dhcp server debug stats

Show the DHCP server statistics.

SYNTAX:

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

- ▶ **Second DHCP server seen:** whether a concurrent DHCP server was found disabled the LAN or not.

RELATED COMMANDS:

dhcp server debug clear Clear the DHCP server statistics.

dhcp server debug traceconfig

Modify the DHCP server trace configuration.

SYNTAX:

```
dhcp server debug traceconfig [state = <{disabled | enabled}>]
```

where:

state Enable or disable tracing. The default is disabled .	OPTIONAL
---	----------

EXAMPLE:

```
=>dhcp server debug traceconfig  
Tracing: disabled  
=>
```

dhcp server lease add

Add a DHCP server lease.

SYNTAX:

```
dhcp server lease add clientid = <client-id>
                  pool = <string>
                  [addr = <ip-address>]
                  [offset = <number>]
                  [leasetime = <number>]
                  [expirytime = <number>]
                  [gateway = <ip-address>]
                  [macaddr = <hardware-address>]
```

where:

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.	REQUIRED
Tip	Use the command :dhcp server pool list to obtain a list of available DHCP server pools.	
addr	The favoured IP address for this DHCP host. This IP address, if specified, must be in the range of the specified DHCP server pool.	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.	OPTIONAL
Note	Not specifying this parameter does not preserve an IP address for the host.	
leasetime	A number (of seconds). Represents the time the host is allowed to use this address.	OPTIONAL
Note	0 means infinite leasetime.	
expiretime	The time in seconds the DHCP server keeps the lease reserved	
Tip	0 means infinite expirytime.	
gateway	The IP address of the default router for this client.	OPTIONAL
macaddr	The MAC address of the host.	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
=>
```

DHCP Commands

RELATED COMMANDS:

- | | |
|---------------------------------|----------------------------------|
| dhcp server lease delete | Delete a DHCP server lease. |
| dhcp server lease list | List all the DHCP server leases. |

dhcp server lease delete

Delete a DHCP server lease.

SYNTAX:

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

where:

clientid The DHCP client identification string of the DHCP lease. OPTIONAL

Note If not specified, all DHCP clients are deleted.

index The DHCP server lease table index. OPTIONAL

Tip Use the command :**dhcp server lease list** to obtain a list of the index numbers of all current DHCP leases.

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 server lease.

dhcp server lease list List all the DHCP server leases.

dhcp server lease flush

Flush all the DHCP server leases.



The flush command does not impact previously saved configurations.

SYNTAX:

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

where:

pool	The name of the DHCP server pool to be flushed. Only the leases belonging to this pool will be deleted.	OPTIONAL
Note If not specified, all the DHCP server leases will be flushed.		

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
=>
```

dhcp server lease list

List all the DHCP server leases.

SYNTAX:

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

where:

clientid	The DHCP client identification string of the DHCP lease.	OPTIONAL
Note If not specified, the DSHCP server leases for all the DHCP clients are listed.		
index	The DHCP server lease table index.	OPTIONAL
Note If not specified, the complete DHCP server lease table will be shown.		

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 server lease.
dhcp server lease delete Delete a DHCP server lease.

dhcp server option flush

Flush all DHCP server option templates and instances.



The flush command does not impact previously saved configurations.

SYNTAX:

```
dhcp server option flush
```

dhcp server option instadd

Add a DHCP server option instance.

SYNTAX:

```
dhcp server option instadd name = <string>
                           tmplname = <string>
                           value = <Value : (type)value; type being 8-bit,
                                   16-bit, 32-bit, addr, ascii, byte_array>
```

where:

name	The name of the DHCP server option instance.	REQUIRED
tmplname	The name of the DHCP server option template.	REQUIRED
Tip	Use the command : dhcp server option tmplist to obtain a list of DHCP server option templates.	
value	The value of the DHCP server option instance. Format is (type)value where type is 8-bit, 16-bit, 32-bit, addr, ascii or byte_array.	REQUIRED

Note The type must be identical to the type of the DHCP server option template. Use the command :**dhcp server option tmplist** to obtain a list of DHCP server option templates.

EXAMPLE:

```
=>dhcp server option instlist
myInstance
    Tmpl name : myTmpl          (1)
    Use       : 0
    Value     : (32-bit)64

=>dhcp server option instadd name=yourInstance tmplname=yourTmpl value=(ascii)#
{root} [dhcp server option]=>instlist
yourInstance
    Tmpl name : yourTmpl        (2)
    Use       : 0
    Value     : (ascii)#

myInstance
    Tmpl name : myTmpl          (1)
    Use       : 0
    Value     : (32-bit)64

=>
```

RELATED COMMANDS:

- | | |
|--------------------------------------|--|
| dhcp server option instdelete | Delete a DHCP server option instance. |
| dhcp server option instlist | List all the DHCP server option instances. |

dhcp server option instdelete

Delete a DHCP server option instance.

SYNTAX:

```
dhcp server option instdelete name = <string>
```

where:

name	The name of the DHCP server option instance to be deleted.	REQUIRED
------	--	----------

EXAMPLE:

```
=>dhcp server option instlist
yourInstance
    Tmpl name : yourTmpl          (2)
    Use       : 0
    Value     : (ascii)#
myInstance
    Tmpl name : myTmpl           (1)
    Use       : 0
    Value     : (32-bit)64

=>dhcp server option instdelete name=yourInstance
=>dhcp server option instlist
myInstance
    Tmpl name : myTmpl           (1)
    Use       : 0
    Value     : (32-bit)64

=>
```

RELATED COMMANDS:

dhcp server option instadd Add a DHCP server option instance.

dhcp server option instlist List all the DHCP server option instances.

dhcp server option instlist

List all the DHCP server option instances.

SYNTAX:

```
dhcp server option instlist
```

EXAMPLE:

```
=>dhcp server option instlist
yourInstance
    Tmpl name : yourTmpl          (2)
    Use       : 0
    Value     : (ascii) '#33

myInstance
    Tmpl name : myTmpl           (1)
    Use       : 0
    Value     : (32-bit) 64

=>
```

RELATED COMMANDS:

dhcp server option instadd Add a DHCP server option instance.

dhcp server option instdelete Delete a DHCP server option instance.

dhcp server option tmpladd

Add a DHCP server option template.

SYNTAX:

```
dhcp server option tmpladd name = <string>
                  optionid = <number{1-254}>
                  type = <{8-bit | 16-bit | 32-bit | addr | ascii
                           | byte_array}>
```

where:

name	The name of the DHCP server option template.	REQUIRED
optionid	A number between 1 and 254. Specifies the DHCP server option code.	REQUIRED
type	Specifies the DHCP server option type. Choose between: <ul style="list-style-type: none">▶ 8-bit▶ 16-bit▶ 32-bit▶ addr▶ ascii▶ byte_array.	REQUIRED

EXAMPLE:

```
=>dhcp server option tmpllist
Name          Option  Type      Use
myTmpl        1       32-bit    0
=>dhcp server option tmpladd name=yourTmpl optionid=2 type=ascii
=>dhcp server option tmpllist
Name          Option  Type      Use
yourTmpl     2       ascii     0
myTmpl        1       32-bit    0
=>
```

RELATED COMMANDS:

- | | |
|--------------------------------------|--|
| dhcp server option tmpldelete | Delete a DHCP server option template. |
| dhcp server option tmpllist | List all the DHCP server option templates. |

dhcp server option tmpldelete

Delete a DHCP server option template.

SYNTAX:

```
dhcp server option tmpldelete name = <string>
```

where:

name	The name of the DHCP server option template to be deleted.	REQUIRED
------	--	----------

EXAMPLE:

```
=>dhcp server option tmpllist
Name          Option  Type      Use
yourTmpl      2       ascii     0
myTmpl        1       32-bit   0
=>dhcp server option tmpldelete name=yourTmpl
=>dhcp server option tmpllist
Name          Option  Type      Use
myTmpl        1       32-bit   0
=>
```

RELATED COMMANDS:

dhcp server option tmpladd	Add a DHCP server option template.
dhcp server option tmpllist	List all the DHCP server option templates.

dhcp server option tmpllist

List all the DHCP server option templates.

SYNTAX:

```
dhcp server option tmpllist
```

EXAMPLE:

```
=>dhcp server option tmpllist
Name          Option  Type      Use
yourTmpl      2       ascii     0
myTmpl        1       32-bit   0
=>
```

RELATED COMMANDS:

- | | |
|--------------------------------------|---------------------------------------|
| dhcp server option tmpladd | Add a DHCP server option template. |
| dhcp server option tmpldelete | Delete a DHCP server option template. |

dhcp server pool add

Add a DHCP server pool.

SYNTAX:

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

where:

name	The name of the DHCP server pool.	REQUIRED
Note	If not specified, the name is "dhcp_pool_x", where x is a subsequent number.	
index	The number of the pool before which you want the new pool to be added.	OPTIONAL
Note	If not specified, the DHCP server pool will be added at the bottom of the DHCP server lease table.	

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 server pool. |
| dhcp server pool list | List all DHCP server pools. |

dhcp server pool config

Configure a DHCP server pool.

SYNTAX:

```
dhcp server pool config name = <string>
    intf = <string>
    [index = <number>]
    [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 = <{disabled | enabled}>]
    [localgw = <{disabled | enabled}>]
```

where:

name	The name of the DHCP server pool to configure.	REQUIRED
intf	The interface for which the pool is allowed to lease IP addresses.	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
Tip Use the command :dhcp server pool list to obtain a list of the index numbers of all current DHCP server pools.		
poolstart	The lowest IP address in the DHCP address range to use for leasing. The 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. The 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. The 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. The 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. The 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 Windows Internet Naming Service (WINS) server for DHCP clients.	OPTIONAL
secwins	The IP address of the secondary WINS server for DHCP clients.	OPTIONAL
leasetime	A number (of seconds). Represents the time in seconds a client is allowed to use an address. Note Specifying 0 makes the lease permanent.	OPTIONAL
unnumbered	Assign an IP address from this pool to the DHCP server (enabled) or not (disabled).	OPTIONAL
localgw	Note For dynamic pools only.	
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
=>
```

dhcp server pool delete

Delete a DHCP server pool.

SYNTAX:

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

where:

name	The name of the DHCP server pool to be deleted.	REQUIRED
Tip	Use the command : dhcp server pool list to obtain a list of all current DHCP leases.	

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 server pool.
dhcp server pool list	List all DHCP server pools.

dhcp server pool flush

Flush all DHCP server pools.



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
=>
```

dhcp server pool list

List all DHCP server pools.

SYNTAX:

```
dhcp server pool list [name = <string>]
```

where:

name	The name of the DHCP server pool to be shown.	OPTIONAL
Note	If not specified, all the DHCP server pools are shown.	

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 server pool.
dhcp server pool delete Delete a DHCP server pool.

dhcp server pool optadd

Add an option instance to the DHCP server pool.

SYNTAX:

```
dhcp server pool optadd name = <string>
                      instname = <string>
```

where:

name	The name of the DHCP server pool to which an option instance must be added.	REQUIRED
instname	The name of the DHCP server option instance.	REQUIRED
Tip	Use the command : dhcp server option instlist to obtain a list of DHCP server option instances.	

EXAMPLE:

```
=>dhcp server pool optadd name=dhcp_pool_1 instname=yourInstance
=>dhcp server pool list name=dhcp_pool_1
Pool          Start            End           Intf       State
1  dhcp_pool_1    0.0.0.0      0.0.0.0     lan1      FREE

DHCP server   = 0.0.0.0 [unnumbered]
Netmask       = 0.0.0.0
Leasetime     = infinite
Gateway       = 0.0.0.0
DNS domain   = lan
DNS metric   = 0

DNS address list:
0.0.0.0 (local DNS)

Option instance list:
Name          Option
yourInstance   2      (yourTmpl)
=>
```

RELATED COMMANDS:

dhcp server pool optdelete Delete an option instance from the DHCP server pool.

dhcp server pool optdelete

Delete an option instance from the DHCP server pool.

SYNTAX:

```
dhcp server pool optdelete name = <string>
                      instname = <string>
```

where:

name	The name of the DHCP server pool from which an option instance must be deleted.	REQUIRED
instname	The name of the DHCP server option instance to be deleted.	REQUIRED
Tip	Use the command : dhcp server option list to obtain a list of DHCP server option instances.	

EXAMPLE:

```
=>dhcp server pool list name=dhcp_pool_1
Pool          Start          End           Intf      State
1  dhcp_pool_1    0.0.0.0     0.0.0.0       lan1     FREE

DHCP server   = 0.0.0.0 [unnumbered]
Netmask        = 0.0.0.0
Leasetime      = infinite
Gateway        = 0.0.0.0
DNS domain    = lan
DNS metric    = 0

DNS address list:
0.0.0.0 (local DNS)

Option instance list:
Name          Option
yourInstance   2  (yourTmpl)
=>dhcp server pool optdelete name=dhcp_pool_1 instname=yourInstance
=>dhcp server pool list name=dhcp_pool_1
Pool          Start          End           Intf      State
1  dhcp_pool_1    0.0.0.0     0.0.0.0       lan1     FREE

DHCP server   = 0.0.0.0 [unnumbered]
Netmask        = 0.0.0.0
Leasetime      = infinite
Gateway        = 0.0.0.0
DNS domain    = lan
DNS metric    = 0

DNS address list:
0.0.0.0 (local DNS)
=>
```

RELATED COMMANDS:

dhcp server pool optadd Add an option instance to the DHCP server pool.

dhcp server pool rtadd

Add a route to the DHCP server pool.

SYNTAX:

```
dhcp server pool rtadd name = <string>
                      dst = <ip-address>
                      [dstmsk = <ip-mask(dotted or cidr)>]
                      [gateway = <ipaddress | 0>]
```

where:

name	The name of the DHCP server pool to which a route must be added.	REQUIRED
dst	The IP destination address of the route for DHCP clients.	REQUIRED
dstmsk	The destination IP address mask.	OPTIONAL
gateway	The IP address of the next hop. Must be directly connected to the DHCP client	OPTIONAL

RELATED COMMANDS:

dhcp server pool rtdelete Delete a route from the DHCP server pool.

dhcp server pool rtdelete

Delete a route from the DHCP server pool.

SYNTAX:

```
dhcp server pool rtdelete name = <string>
                           dst = <ip-address>
                           [dstmsk = <ip-mask(dotted or cidr)>]
                           [gateway = <ipaddress | 0>]
```

where:

name	The name of the DHCP server pool from which a route must be deleted.	REQUIRED
dst	The IP destination address of the route for DHCP clients.	REQUIRED
dstmsk	The destination IP address mask.	OPTIONAL
gateway	The IP address of the next hop. Must be directly connected to the DHCP client	OPTIONAL

RELATED COMMANDS:

dhcp server pool rtadd Add a route to the DHCP server pool.

dhcp server pool ruleadd

Add a selection rule to the DHCP server pool.

SYNTAX:

```
dhcp server pool ruleadd name = <string>
                           [key = {or | and}]
                           rulename = <string>
```

where:

name	The name of the DHCP server pool to which a selection rule must be added.	REQUIRED
key	The logical key of the selection rule. The default is or .	OPTIONAL
rulename	The name of the DHCP selection rule.	REQUIRED
Tip	Use the command : dhcp server rule list to obtain a list of DHCP server rules.	

RELATED COMMANDS:

dhcp server pool ruledelete Delete a selection rule from the DHCP server pool.

dhcp server pool ruledelete

Delete a selection rule from the DHCP server pool.

SYNTAX:

```
dhcp server pool ruleadd    name = <string>
                           rulename = <string>
```

where:

name	The name of the DHCP server pool from which a selection rule must be deleted.	REQUIRED
rulename	The name of the DHCP selection rule to be deleted.	REQUIRED
Tip	Use the command : dhcp server rule list to obtain a list of DHCP server rules.	

RELATED COMMANDS:

dhcp server pool ruleadd Add a selection rule to the DHCP server pool.

DNS Commands

Introduction

This chapter describes the commands of the **dns** command group.

Contents

This chapter covers the following commands:

dns client config	Modify the Domain Name System (DNS) resolver configuration.	176
dns client dnsadd	Add a DNS server.	177
dns client dnsdelete	Delete a DNS server.	178
dns client dnslist	List all DNS servers.	179
dns client flush	Remove all DNS servers.	180
dns client nslookup	DNS lookup for a domain name or an address.	181
dns server config	Modify the DNS resolver configuration.	182
dns server flush	Flush all local DNS hosts and routes.	183
dns server debug clear	Clear the DNS server/forwarder statistics.	184
dns server debug stats	Print the DNS server/forwarder statistics.	185
dns server debug spoof clear	Clear the intercept cache table	186
dns server debug spoof getaddress	Get the real ip for the given spoofed ip	187
dns server debug spoof getflags	Get the error flags for the given spoofed ip	188
dns server debug spoof list	List the intercept cache table.	189
dns server debug spoof update	Update the intercept cache table.	190
dns server host add	Add a local DNS host.	191
dns server host delete	Delete a local DNS host.	192
dns server host flush	Flush all the local DNS hosts.	193
dns server host list	List all the local DNS hosts.	194
dns server route add	Adds a DNS forwarding route.	195
dns server route delete	Delete a DNS forwarding route.	196
dns server route flush	Flush all DNS forwarding routes.	197
dns server route list	List all the DNS forwarding routes.	198

dns client config

Modify the Domain Name System (DNS) resolver configuration.

SYNTAX:

```
dns client config [timeout = <number{1-900}>]
                  [retry = <number{0-10}>]
                  [search = <{disabled | enabled}>]
                  [list = <string>]
                  [trace = <{disabled | enabled}>]
```

where:

timeout	A number between 1 and 900 (seconds). Represents the query timeout. The default is 5 .	OPTIONAL
retry	A number between 0 and 10. Represents the number of query retries before giving up. The default is 3 .	OPTIONAL
search	Use the search list to construct fully qualified domain names. The default is disabled .	OPTIONAL
list	Specify a search list. This is a slash separated list of domain name suffixes.	OPTIONAL
trace	Enable or disable verbose logging. The default is disabled .	OPTIONAL

EXAMPLE:

```
=>dns client config
timeout : 5s
retry   : 3
search   : on
srchlist: <empty>
trace    : off
=>
```

dns client dnsadd

Add a DNS server.

SYNTAX:

```
dns client dnsadd    addr = <string>
                      [port = <number>]
```

where:

addr	The IP address of the DNS server to be added.	REQUIRED
port	The DNS server port number. The default is 53 .	OPTIONAL

EXAMPLE:

```
=>dns client dnsadd addr=150.150.150.150
=>dns client dnslist

Entry      State       Family     Server
 1      CONNECTED     IP        [port] 53 - [addr] 127.0.0.1
 2      CONNECTED     IP        [port] 53 - [addr] 150.150.150.150

=>
```

RELATED COMMANDS:

- | | |
|----------------------|-----------------------|
| dns client dnsdelete | Delete a DNS server. |
| dns client dnslist | List all DNS servers. |

dns client dnsdelete

Delete a DNS server.

SYNTAX:

```
dns client dnsdelete index = <number{1-99}>
```

where:

index	A number between 1 and 99. Represents the index number of the DNS server to be deleted.	REQUIRED
-------	--	----------

Tip Use the command :dns client dnslist to obtain a list of DNS servers.

EXAMPLE:

```
=>dns client dnslist
Entry      State      Family      Server
 1    CONNECTED      IP      [port] 53 - [addr] 127.0.0.1
 2    CONNECTED      IP      [port] 53 - [addr] 150.150.150.150

=>dns client dnsdelete index=2
=>dns client dnslist
Entry      State      Family      Server
 1    CONNECTED      IP      [port] 53 - [addr] 127.0.0.1
=>
```

RELATED COMMANDS:

dns client dnsadd	Add a DNS server.
dns client dnslist	List all DNS servers.

dns client dnslist

List all DNS servers.

SYNTAX:

```
dns client dnslist
```

EXAMPLE:

```
=>dns client dnslist
Entry      State       Family     Server
 1      CONNECTED    IP        [port] 53 - [addr] 127.0.0.1
 2      CONNECTED    IP        [port] 53 - [addr] 150.150.150.150
=>
```

RELATED COMMANDS:

dns client dnsadd	Add a DNS server.
dns client dnsdelete	Delete a DNS server.

dns client flush

Remove all DNS servers.

SYNTAX:

```
dns client flush
```

dns client nslookup

DNS lookup for a domain name or an address.

SYNTAX:

```
dns client nslookup host = <string>
```

where:

host The DNS domain name string for which to query.

REQUIRED

dns server config

Modify the DNS resolver configuration.

SYNTAX:

```
dns server config [domain = <string>]
                  [timeout = <number{0-2147483647}>]
                  [suppress = <number>]
                  [state = <{disabled | enabled}>]
                  [trace = <{disabled | enabled}>]
                  [WANDownSpoofing = <{disabled | enabled}>]
                  [WDspoofedIP = <ip-address>]
```

where:

domain	The DNS server domain name.	OPTIONAL
timeout	A number between 0 and 2147483647 (seconds). Represents the forwarded DNS query timeout. The default is 15 .	OPTIONAL
suppress	Suppress not more than the specified amount of remote DNS server errors. The default is 0 .	OPTIONAL
state	Enable or disable the local DNS server/forwarder. The default is enabled .	OPTIONAL
trace	Enable or disable verbose logging. The default is disabled .	OPTIONAL
WANDownSpoofing	Enable or disable DNS spoofing when no applicable forwarding route present. The default is disabled .	OPTIONAL
WDspoofedIP	The IP address to be used for spoofing when WANDownSpoofing is enabled.	OPTIONAL

EXAMPLE:

```
=>dns server config
domain : lan
timeout : 15s
suppress : 0
state : enabled
trace : off
spoofing : off
spooft ip : 0.0.0.0
=>
```

dns server flush

Flush all local DNS hosts and routes.

SYNTAX:

```
dns server flush
```

dns server debug clear

Clear the DNS server/forwarder statistics.

SYNTAX:

```
dns server debug clear
```

EXAMPLE:

```
=>dns server debug stats
Corrupted packets received      :      100
Local questions resolved       :      3
Local negative answers sent    :      1
Total DNS packets forwarded    :      0
External answers received      :      0
Spoofed responses              :      0
Forward table full, discard   :      0
Spurious answers              :      0
Unknown query types           :      0
=>dns server debug clear
=>dns server debug stats
Corrupted packets received      :      0
Local questions resolved       :      0
Local negative answers sent    :      0
Total DNS packets forwarded    :      0
External answers received      :      0
Spoofed responses              :      0
Forward table full, discard   :      0
Spurious answers              :      0
Unknown query types           :      0
=>
```

RELATED COMMANDS:

dns server debug stats Print the DNS server/forwarder statistics.

dns server debug stats

Print the DNS server/forwarder statistics.

SYNTAX:

```
dns server debug stats
```

EXAMPLE:

```
=>dns server debug stats
Corrupted packets received      :      100
Local questions resolved       :        3
Local negative answers sent   :        1
Total DNS packets forwarded   :        0
External answers received     :        0
Spoofed responses             :        0
Forward table full, discard  :        0
Spurious answers              :        0
Unknown query types          :        0
=>
```

RELATED COMMANDS:

dns server debug clear Clear the DNS server/forwarder statistics.

dns server debug spoof clear

Clear the intercept cache table

SYNTAX:

```
dns server debug spoof clear
```

RELATED COMMANDS:

dns server debug spoof getaddress	Get the real ip for the given spoofed ip
dns server debug spoof getflags	Get the error flags for the given spoofed ip
dns server debug spoof list	List the intercept cache table.
dns server debug spoof update	Update the intercept cache table.

dns server debug spoof getaddress

Get the real ip for the given spoofed ip

SYNTAX:

```
dns server debug      addr = <ip-address>
spoof getaddress
```

where:

addr	The IP address of the spoofed server.	REQUIRED
------	---------------------------------------	----------

EXAMPLE:

```
{Administrator}=>dns server debug spoof getaddress addr=198.18.1.1
:dns server debug spoof getaddress addr=198.18.1.1
Resolved ip = 0.0.0.0.
{Administrator}=>
```

RELATED COMMANDS:

dns server debug spoof clear	Clear the intercept cache table
dns server debug spoof getflags	Get the error flags for the given spoofed ip
dns server debug spoof list	List the intercept cache table.
dns server debug spoof update	Update the intercept cache table.

dns server debug spoof getflags

Get the error flags for the given spoofed ip

SYNTAX:

```
dns server debug      addr = <ip-address>
spoof getflags
```

where:

addr	The IP address of the spoofed server.	REQUIRED
------	---------------------------------------	----------

EXAMPLE:

```
{Administrator}[dns server debug spoof]=>:dns server debug spoof getflags
addr = 192.168.1.254
:dns server debug spoof getflags addr=192.168.1.254
Invalid spoofed ip.
{Administrator}[dns server debug spoof]=>
```

RELATED COMMANDS:

dns server debug spoof clear	Clear the intercept cache table
dns server debug spoof getaddress	Get the real ip for the given spoofed ip
dns server debug spoof list	List the intercept cache table.
dns server debug spoof update	Update the intercept cache table.

dns server debug spoof list

List the intercept cache table.

SYNTAX:

```
dns server debug  
spoof list
```

EXAMPLE:

```
{Administrator} [dns server debug spoof]=>list  
Spoof IP          FQDN           Real IP      Flags  
198.18.1.1        eu.thmulti.com 0.0.0.0      Not resolved  
198.18.1.2        thmulti.com   0.0.0.0      Not resolved  
198.18.1.3        com           0.0.0.0      Not resolved  
198.18.1.4        edgmd588.eu.thmulti.com 0.0.0.0      Not resolved  
198.18.1.5        edgmsus01.eu.thmulti.com 0.0.0.0      Not resolved  
198.18.1.6        BOULSDCEU02.eu.thmulti.com 0.0.0.0      Not resolved  
198.18.1.7        juleke.nit    0.0.0.0      Not resolved
```

RELATED COMMANDS:

dns server debug spoof clear	Clear the intercept cache table
dns server debug spoof getaddress	Get the real ip for the given spoofed ip
dns server debug spoof getflags	Get the error flags for the given spoofed ip
dns server debug spoof update	Update the intercept cache table.

dns server debug spoof update

Update the intercept cache table.

SYNTAX:

```
dns server debug  
spoof update
```

EXAMPLE:

```
{Administrator}=>dns server debug spoof update  
{Administrator}=>
```

RELATED COMMANDS:

dns server debug spoof clear	Clear the intercept cache table
dns server debug spoof getaddress	Get the real ip for the given spoofed ip
dns server debug spoof getflags	Get the error flags for the given spoofed ip
dns server debug spoof list	List the intercept cache table.

dns server host add

Add a local DNS host.

SYNTAX:

```
dns server host add name = <string>
                     [addr = <ip-address>]
                     [ttl = <number{0-2147483647}>]
```

where:

name	The name of the IP host to be added.	REQUIRED
addr	The IP address of the host.	OPTIONAL
ttl	A number between 0 and 2147483647 (seconds). Represents the lifetime of the host. The default is 0 (in other words, no limit on the lifetime).	OPTIONAL

EXAMPLE:

```
=>dns server host add name=myDNS addr=150.150.150.150 ttl=3600
=>dns server host list
Address           Hostname          TTL (s)
150.150.150.150  myDNS            3600
<local>          speedtouch       0
<local>          dsldevice        0
=>
```

RELATED COMMANDS:

- | | |
|------------------------|-------------------------------|
| dns server host delete | Delete a local DNS host. |
| dns server host list | List all the local DNS hosts. |

dns server host delete

Delete a local DNS host.

SYNTAX:

```
dns server host delete name = <string>
```

where:

name	The name of the DNS host to be deleted.	REQUIRED
------	---	----------

EXAMPLE:

```
=>dns server host list
Address           Hostname          TTL (s)
150.150.150.150  myDNS            3600
<local>          speedtouch       0
<local>          dsldevice        0
=>dns server host delete name=myDNS
=>dns server host list
Address           Hostname          TTL (s)
<local>          speedtouch       0
<local>          dsldevice        0
=>
```

RELATED COMMANDS:

dns server host add	Add a local DNS host.
dns server host list	List all the local DNS hosts.

dns server host flush

Flush all the local DNS hosts.

SYNTAX:

```
dns server host flush
```

EXAMPLE:

```
=>dns server host list
Address          Hostname          TTL (s)
192.168.1.64    * Unknown-00-10-a4-ad-32-cf      60
<local>          dsldevice          1200
<local>          speedtouch        1200
=>dns server host flush
=>dns server host list
=>
```

dns server host list

List all the local DNS hosts.

SYNTAX:

```
dns server host list
```

EXAMPLE:

```
=>dns server host list
Address          Hostname           TTL (s)
192.168.1.64    * Unknown-00-10-a4-ad-32-cf      60
<local>         dsldevice            1200
<local>         speedtouch          1200
=>
```

RELATED COMMANDS:

- | | |
|-------------------------------|--------------------------|
| dns server host add | Add a local DNS host. |
| dns server host delete | Delete a local DNS host. |

dns server route add

Adds a DNS forwarding route.

SYNTAX:

```
dns server route add dns = <ip-address>
                    [src = <ip-address>]
                    [srcmask = <ip-mask(dotted or cidr)>]
                    [domain = <string>]
                    [metric = <number{0-100}>]
                    [intf = <string>]
```

where:

dns	The IP address of a DNS server.	REQUIRED
src	The source IP address(es) using this remote DNS server.	OPTIONAL
Note	Supports IP/mask notation.	
srcmask	The source IP address mask.	OPTIONAL
domain	The DNS domain label.	OPTIONAL
metric	A number between 0 and 100. Represents the metric (weight factor) for this DNS route. The lower the metric, the higher the weight.	OPTIONAL
intf	An interface name. DNS queries will only be forwarded on the specified interface.	OPTIONAL
Note	In no interface is specified, DNS queries will be forwarded on all interfaces.	

EXAMPLE:

```
=>dns server route add dns=150.150.150.150 src=10.0.0.0/8 domain=myDNS intf=eth0
=>dns server route list
DNS Server      Source          Domain        Metric   Intf     State
150.150.150.150  10.0.0.0/8    myDNS         0        eth0    UP
=>
```

RELATED COMMANDS:

- | | |
|--------------------------------|-------------------------------------|
| dns server route delete | Delete a DNS forwarding route. |
| dns server route list | List all the DNS forwarding routes. |

dns server route delete

Delete a DNS forwarding route.

SYNTAX:

```
dns server route delete dns = <ip-address>
```

where:

dns	The IP address of the DNS server for which the forwarding route must be deleted.	REQUIRED
-----	--	----------

EXAMPLE:

```
=>dns server route list
DNS Server      Source          Domain        Metric  Intf   State
150.150.150.150 10.0.0.0/8    myDNS         0       eth0   UP
=>dns server route delete dns=150.150.150.150
=>dns server route list
=>
```

RELATED COMMANDS:

dns server route add	Adds a DNS forwarding route.
dns server route list	List all the DNS forwarding routes.

dns server route flush

Flush all DNS forwarding routes.

SYNTAX:

```
dns server route flush
```

EXAMPLE:

```
=>dns server route list
DNS Server      Source          Domain        Metric  Intf   State
150.150.150.150 10.0.0.0/8    myDNS         0       eth0   UP
=>dns server route flush
=>dns server route list
=>
```

dns server route list

List all the DNS forwarding routes.

SYNTAX:

```
dns server route list
```

EXAMPLE:

```
=>dns server route list
DNS Server      Source          Domain        Metric  Intf   State
150.150.150.150 10.0.0.0/8    myDNS         0       eth0   UP
=>
```

RELATED COMMANDS:

dns server route add Adds a DNS forwarding route.

dns server route delete Delete a DNS forwarding route.

DSD Commands

Introduction

This chapter describes the commands of the **dsd** command group.

Contents

This chapter covers the following commands:

dsd config	Display/modify the Differentiated Service Delivery (DSD) framework configuration.	200
dsd debug config	Display/modify DSD debug settings	201
dsd debug connection list	Display the internal connection pool usage.	202
dsd debug proxy	Display/modify the HyperText Transfer Protocol (HTTP) Intercept fixed proxy configuration.	203
dsd debug recycling	Display/modify HTTPPI recycling settings	204
dsd debug stats	Display/clear DSD framework and module statistics	205
dsd intercept config	Display/modify the HTTP Intercept configuration.	206
dsd syslog config	Display/modify the HTTP Intercept logging configuration.	207
dsd syslog list	Display the HTTP Intercept log file.	208

dsd config

Display/modify the Differentiated Service Delivery (DSD) framework configuration.

SYNTAX:

```
dsd config [state = <{disabled | enabled | automatic}>]
```

where:

state	Choose between:	OPTIONAL
	▶ automatic (default): By default (DSD config state is automatic), the DSD framework is not active, for example no URL filtering or web site filtering is applied. Only in case the state of the default WAN IP interface goes down, DSD framework will be activated, for example in case the PPP link goes down (implying HTTP intercept active).	
	▶ enabled: If DSD config state is enabled the DSD framework is always active.	
	▶ disabled	



If you do not provide any parameters, the current configuration will be shown.

EXAMPLE:

```
=>dsd config
State    : automatic
=>
```

dsd debug config

Display/modify DSD debug settings

SYNTAX:

config	[turbomode = <{disabled enabled}>]
--------	------------------------------------

where:

turbomode	Enable/disable turbomode	OPTIONAL
-----------	--------------------------	----------

dsd debug connection list

Display the internal connection pool usage.

SYNTAX:

```
dsd debug connection list      [expand = <{disabled|enabled}>]
```

where:

expand	Enable or disable the expanded listing. The default is disabled .	optional
--------	---	----------

dsd debug proxy

Display/modify the HyperText Transfer Protocol (HTTP) Intercept fixed proxy configuration.

SYNTAX:

```
dsd debug proxy      [state = <{disabled | enabled}>]  
                     [dest = <ip-address>]  
                     [port = <{supported TCP/UDP port} or number>]
```

where:

state	Enable or disable fixed proxy redirecting. The default is disabled .	OPTIONAL
dest	The destination IP address to which requests will be forwarded.	OPTIONAL
port	The port to be used for connecting to proxy. Select one of the supported Transmission Control Protocol (TCP)/UDP port names (see “ Supported TCP/UDP Port Names ” on page 749) or, alternatively, specify the port number.	OPTIONAL

EXAMPLE:

```
=>dsd debug proxy  
State   : off  
Dest.IP : 0.0.0.0  
Port    : 0  
=>
```

dsd debug recycling

Display/modify HTTP1 recycling settings

SYNTAX:

```
dsd debug recycling state = <disabled>|<enabled>
                           interval = <number>
                           httpidle = <number>
                           otheridle = <number>
```

where:

state	Enable/disable stream recycling	OPTIONAL
interval	Time between successive activity checks	OPTIONAL
httpidle	Minimal idle count for recycling (filtered) http streams	OPTIONAL
otheridle	Minimal idle count for recycling other streams	OPTIONAL

dsd debug stats

Display/clear DSD framework and module statistics

SYNTAX:

```
dsd debug stats      name = <{intercept|urlfilter|recycling|syslog|all}
                      clear = <{no|yes}>
```

where:

name	Specify the name of a module	REQUIRED
clear	Clear the specified statistics	REQUIRED

dsd intercept config

Display/modify the HTTP Intercept configuration.

SYNTAX:

```
dsd intercept config [WDSpoofedIP = <ip-address>]
                     [servertimeout = <number>]
                     [servererrorurl = <string>]
                     [categoryerrorurl = <string>]
                     [monitorintercepturl = <string>]
                     [urlblockedurl = <string>]
                     [imageredirect = <{disabled | enabled}>]
                     [imageredirecturl = <string>]
                     [alwaysuseip = <{disabled | enabled}>]
```

where:

WDSpoofedIP	The IP address indicating unavailable WAN connection. The default is 198.18.1.1 .	OPTIONAL
servertimeout	A number of seconds. Represents the server timeout for redirect action. The default is 10 .	OPTIONAL
servererrorurl	The destination URL when the connection to the server failed.	OPTIONAL
categoryerrorurl	The destination URL when the connection to the category server failed.	OPTIONAL
monitorintercepturl	The destination URL when the request is intercepted by the monitor thread.	OPTIONAL
urlblockedurl	The destination URL when the requested URL is blocked.	OPTIONAL
imageredirect	Enable or disable substitution of blocked images. The default is enabled .	OPTIONAL
imageredirecturl	The URL of the image used when substituting blocked images.	OPTIONAL
alwaysuseip	Always use IP address when redirecting to a local page (enabled) or not (disabled). The default is enabled .	OPTIONAL

EXAMPLE:

```
=>dsd intercept config
WAN down spoofed IP   : 198.18.1.1
Servertimeout         : 10 sec
Connection failure    : /cgi/b/ic/connect/
Category server error : /cgi/b/ic/connect/
Monitor intercept     : /cgi/b/ic/connect/
Unauthorized request  : /cgi/b/sfltr/blocked/
Image redirecting      : enabled
Image redirect url    : /images/spacer.gif
Always use IP          : enabled
=>
```

dsd syslog config

Display/modify the HTTP Intercept logging configuration.

SYNTAX:

```
dsd syslog config [syslog = <{none | unauthorized | errors | intercepted  
| all}>]
```

where:

syslog	Define the type of events to log. Choose between: <ul style="list-style-type: none">▶ none: nothing is logged to syslog.▶ unauthorized: only the HTTP requests that are blocked because of a LocalRule, BlockCategory, BlockIPAddress or BlockObscure event, are logged.▶ errors: only the HTTP requests that are blocked because of Server errors, Category errors, Monitor intercept, loop detection, bad requests or redirects by local policy rules.▶ intercepted: only the HTTP requests that are blocked because of URLBlocked, Server errors, Category errors, Monitor intercept, loop detection, bad requests or redirects by local policy rules.▶ all: every received request is logged to syslog. The default is intercepted .	OPTIONAL
--------	---	----------

EXAMPLE:

```
=>dsd syslog config  
syslog    : errors  
=>
```

RELATED COMMANDS:

dsd syslog list Display the HTTP Intercept log file.

dsd syslog list

Display the HTTP Intercept log file.

SYNTAX:

```
dsd syslog list
```

EXAMPLE:

```
=>dsd syslog list
<86> SysUpTime: 00:22:37 [HTTPPI] src=10.0.0.1 src_port=1965 dst=141.11.196.35 dst_port=80 event=Se
rverConnect dst_name=aWebsite.com/icons/Button_Document.gif
<86> SysUpTime: 00:22:59 [HTTPPI] src=10.0.0.1 src_port=1968 dst=141.11.234.60 dst_port=80 event=Se
rverConnect dst_name=anotherWebsite.com/rawgen.asp
=>
```

RELATED COMMANDS:

dsd syslog config Display/modify the HTTP Intercept logging configuration.

dsd urlfilter config

Display/modify the URL filtering configuration.

SYNTAX:

```
dsd urlfilter config [state = <{disabled | enabled}>]
                      [blockproxy = <{disabled | enabled}>]
                      [blockipaddress = <{disabled | enabled}>]
                      [blockobscuredip = <{disabled | enabled}>]
                      [defaultaction = <{block | accept}>]
```

where:

state	Enable or disable URL filtering. The default is enabled .	OPTIONAL
blockproxy	Enable or disable blocking of HTTP requests via a proxy server. The default is disabled .	OPTIONAL
blockipaddress	Enable or disable blocking of HTTP requests if host name is IP address. The default is disabled .	OPTIONAL
blockobscuredip	Enable or disable blocking of HTTP requests if host name is believed to be an IP address. The default is disabled .	OPTIONAL
defaultaction	Select the action to be performed when no filter is applicable. Choose between: <ul style="list-style-type: none">▶ block▶ accept. The default is accept .	OPTIONAL

EXAMPLE:

```
=>dsd urlfilter config
Filter state      : enabled
Block proxy traffic : disabled
Block IP addr    : disabled
Block obscured IP addr : disabled
Default action   : accept
=>
```

dsd urlfilter rule add

Add a rule.

SYNTAX:

```
dsd urlfilter rule add url = <string>
                           action = <{block | accept | redirect}>
                           redirect = <string>
```

where:

url	Definition of the URL filter.	REQUIRED
action	The action to perform when URL matches URL. Choose between: <ul style="list-style-type: none">▶ block▶ accept▶ redirect.	REQUIRED
redirect	The redirection URL.	REQUIRED

Note This parameter is only required when the action is **redirect**.

EXAMPLE:

```
=>dsd urlfilter rule add url=www.speedtouch.com action=redirect redirect=www.speedtouch.be
=>dsd urlfilter rule list
Index Action      Url                                Redirect
----- -----  ---
0       redirect  www.speedtouch.com                www.speedtouch.be

Rule list statistics : 1 present, 128 total (0.0% used)
=>
```

RELATED COMMANDS:

- | | |
|---------------------------|------------------------------------|
| dsd urlfilter rule delete | Delete a rule. |
| dsd urlfilter rule list | Display the list of current rules. |

dsd urlfilter rule delete

Delete a rule.

SYNTAX:

```
dsd urlfilter rule delete index = <number>
                           [url = <string>]
```

where:

index	Insertion position in the list.	REQUIRED
-------	---------------------------------	----------

Tip Use the command :dsd urlfilter rule list to obtain the index number of the rule to be deleted.

url	Definition of the URL filter.	OPTIONAL
-----	-------------------------------	----------

EXAMPLE:

```
=>dsd urlfilter rule list
Index Action Url                                Redirect
----- ----- ---
0      redirect www.speedtouch.com               www.speedtouch.be
1      accept   www.baen.com

Rule list statistics : 2 present, 128 total (0.0% used)
=>dsd urlfilter rule delete index=1
=>dsd urlfilter rule list
Index Action Url                                Redirect
----- ----- ---
0      redirect www.speedtouch.com               www.speedtouch.be

Rule list statistics : 1 present, 128 total (0.0% used)
=>
```

RELATED COMMANDS:

dsd urlfilter rule add Add a rule.

dsd urlfilter rule list Display the list of current rules.

dsd urlfilter rule flush

Remove all current rules.

SYNTAX:

```
dsd urlfilter rule flush
```

EXAMPLE:

```
=>dsd urlfilter rule list
Index Action Url Redirect
----- ---
0 redirect www.speedtouch.com www.speedtouch.be
1 accept www.baen.com

Rule list statistics : 2 present, 128 total (0.0% used)
=>dsd urlfilter rule flush
=>dsd urlfilter rule list
Index Action Url Redirect
----- ---

Rule list statistics : 0 present, 128 total (0.0% used)
=>
```

dsd urlfilter rule list

Display the list of current rules.

SYNTAX:

```
dsd urlfilter rule list
```

EXAMPLE:

```
=>dsd urlfilter rule list
Index Action Url Redirect
----- --- -----
0     redirect www.speedtouch.com www.speedtouch.be
1     accept   www.baen.com

Rule list statistics : 2 present, 128 total (0.0% used)
=>
```

RELATED COMMANDS:

dsd urlfilter rule add Add a rule.
dsd urlfilter rule delete Delete a rule.

dsd urlfilter rule modify

Modify an existing rule.

SYNTAX:

```
dsd urlfilter rule modify index = <number>
                           [newurl = <string>]
                           action = <{block | accept | redirect}>
                           redirect = <string>
```

where:

index	Insertion position in the list.	REQUIRED
newurl	Definition of the new URL filter.	OPTIONAL
action	Action to perform when URL matches URL. Choose between: <ul style="list-style-type: none">▶ block▶ accept▶ redirect.	REQUIRED
redirect	The redirection URL.	REQUIRED

Note This parameter is only required when the action is **redirect**.

EXAMPLE:

```
=>dsd urlfilter rule list
Index Action Url Redirect
----- ---
0 redirect www.speedtouch.com www.speedtouch.be
1 accept www.baen.com

Rule list statistics : 2 present, 128 total (0.0% used)
=>dsd urlfilter rule modify index=0 newurl=www.speed*.* action=redirect redirect=www.speedtouch.be
=>dsd urlfilter rule list
Index Action Url Redirect
----- ---
0 redirect www.speed*.*
1 accept www.baen.com

Rule list statistics : 2 present, 128 total (0.0% used)
=>
```

RELATED COMMANDS:

- | | |
|---------------------------|------------------------------------|
| dsd urlfilter rule add | Add a rule. |
| dsd urlfilter rule delete | Delete a rule. |
| dsd urlfilter rule list | Display the list of current rules. |

DynDNS Commands

Introduction

Dynamic DNS is a service that allows to bind a host name to an IP address. In contrast to classical DNS, this service allows regular updates of the IP address related to a host name. In this way, hosts of which the IP address regularly changes (for example due to a limited DHCP lease time) can be identified by a static host name that is unique throughout the Internet.

This chapter describes the commands of the **dyndns** command group.

Contents

This chapter covers the following commands:

dyndns add	Add a dynamic DNS client.	216
dyndns delete	Delete a dynamic DNS client.	217
dyndns flush	Delete all the dynamic DNS clients.	218
dyndns list	List all the dynamic DNS clients.	219
dyndns modify	Modify a dynamic DNS client.	220
dyndns host add	Add a fully qualified host name.	222
dyndns host delete	Delete a host name.	223
dyndns host flush	Delete all hosts.	224
dyndns host list	List all host names.	225
dyndns service list	List all dynamic DNS services.	226
dyndns service modify	Modify specific dynamic DNS service settings.	227

dyndns add

Add a dynamic DNS client.

SYNTAX:

```
dyndns add name = <string>
```

where:

name	The name of the new dynamic DNS client.	REQUIRED
Note	A maximum of 5 clients can be configured.	

EXAMPLE:

```
=>dyndns add name=WAN
=>dyndns list
WAN      : [INIT]

    user =     password =
    addr = 0.0.0.0

=>
```

RELATED COMMANDS:

dyndns delete	Delete a dynamic DNS client.
dyndns list	List all dynamic DNS clients.

dyndns delete

Delete a dynamic DNS client.

SYNTAX:

```
dyndns delete name = <string>
```

where:

name	The name of the dynamic DNS client to be deleted.	REQUIRED
------	---	----------

EXAMPLE:

```
=>dyndns list
WAN      : [INIT]

user =    password =
addr = 0.0.0.0

=>dyndns delete name=WAN
=>dyndns list
=>
```

RELATED COMMANDS:

dyndns add	Add a dynamic DNS client.
dyndns list	List all dynamic DNS clients.

dyndns flush

Delete all the dynamic DNS clients.

SYNTAX:

```
dyndns flush
```

dyndns list

List all the dynamic DNS clients.

SYNTAX:

```
dyndns list
```

EXAMPLE:

```
=>dyndns list
WAN      : [INIT]

  user =   password =
  addr = 0.0.0.0

=>
```

RELATED COMMANDS:

dyndns add	Add a dynamic DNS client.
dyndns delete	Delete a dynamic DNS client.

dyndns modify

Modify a dynamic DNS client.

SYNTAX:

```
dyndns modify name = <string>
    [intf = <string>]
    [user = <string>]
    [password = <password>]
    [group = <string>]
    [mx = <string>]
    [backmx = <{disabled | enabled}>]
    [wildcard = <{disabled | enabled}>]
    [offline = <{disabled | enabled}>]
    [service = <{dyndns | statdns | custom | No-IP | DtDNS
        | gnudip}>]
    [status = <{disabled | enabled}>]
    [trace = <{disabled | enabled}>]
```

where:

name	The name of the dynamic DNS client to be modified.	REQUIRED
intf	The dynamic DNS client interface.	OPTIONAL
	Note This parameter must be defined to enable the dynamic DNS client.	
user	The username for dynamic DNS authentication.	OPTIONAL
password	The password for dynamic DNS authentication.	OPTIONAL
group	The dynamic DNS host group.	OPTIONAL
mx	The mail exchanger.	OPTIONAL
backmx	Set up the mail exchanger as a backup mail exchanger (enabled) or not (disabled). The default is disabled .	OPTIONAL
wildcard	Allow the use of hostname wildcards (enabled) or not (disabled). The default is disabled .	OPTIONAL
offline	Set the host to offline mode (enabled) or not (disabled). The default is disabled .	OPTIONAL
service	The dynamic DNS service. Choose between: <ul style="list-style-type: none"> ▶ dyndns ▶ statdns ▶ custom ▶ No-IP ▶ DtDNS ▶ gnudip. 	OPTIONAL
status	Enable or disable the dynamic DNS client. The default is disabled .	OPTIONAL
trace	Enable or disable the verbose console logging for the dynamic DNS client. The default is disabled .	OPTIONAL

EXAMPLE:

```
=>dyndns list
WAN      : [INIT]

user =     password =
addr = 0.0.0.0

=>dyndns modify name=WAN intf=RtPPPoE user=DNSuser password=_DEV_BA8C0C963BD84130 service=gnudip
=>dyndns list
WAN      : RtPPPoE_ppp [INIT]
options = gnudip
user = my_DNS_user password = *****
addr = 0.0.0.0

=>
```

dyndns host add

Add a fully qualified host name.

SYNTAX:

```
dyndns host add group = <string>
                name = <string>
```

where:

group	The dynamic DNS host group.	REQUIRED
Note A maximum of 4 different groups can be created.		
name	The name of an IP host to add.	REQUIRED
Note A maximum of 20 hosts can be configured.		

EXAMPLE:

```
=>dyndns host add group=local name=localhost.com
=>dyndns host list
local :
[ ] localhost.com

legend :
[ ] not initialised           [o] update in progress
[v] update successful         [x] error
=>
```

RELATED COMMANDS:

dyndns host delete	Delete a host name.
dyndns host list	List all host names.

dyndns host delete

Delete a host name.

SYNTAX:

```
dyndns host delete name = <string>
```

where:

name	The name of the IP host to be deleted.	REQUIRED
------	--	----------

EXAMPLE:

```
=>dyndns host list
local :
[ ] localhost.com

public :
[ ] publichost.com

legend :
[ ] not initialised [o] update in progress
[v] update successful [x] error
=>dyndns host delete name=publichost.com
=>dyndns host list
local :
[ ] localhost.com

legend :
[ ] not initialised [o] update in progress
[v] update successful [x] error
=>
```

RELATED COMMANDS:

dyndns host add	Add a fully qualified host name.
dyndns host list	List all host names.

dyndns host flush

Delete all hosts.



The hosts cannot be flushed, if there is still a group referenced to the hosts.

SYNTAX:

```
dyndns host flush
```

dyndns host list

List all host names.

SYNTAX:

```
dyndns host list
```

EXAMPLE:

```
=>dyndns host list
local :
[ ] localhost.com

public :
[ ] publichost.com

legend :
[ ] not initialised [o] update in progress
[v] update successful [x] error
=>
```

RELATED COMMANDS:

dyndns host add	Add a fully qualified host name.
dyndns host delete	Delete a host name.

dyndns service list

List all dynamic DNS services.

SYNTAX:

```
dyndns service list
```

EXAMPLE:

```
=>dyndns service list
dyndns  :
    server      = members.dyndns.org
    port        = 80
    request     = /nic/update
    update interval = 2097120
    retry interval = 30
    max retry    = 3

statdns  :
    server      = members.dyndns.org
    port        = 80
    request     = /nic/update
    update interval = 0
    retry interval = 30
    max retry    = 3

custom   :
    server      = members.dyndns.org
    port        = 80
    request     = /nic/update
    update interval = 0
    retry interval = 30
    max retry    = 3

No-IP    :
    server      = dynupdate.no-ip.com
    port        = 80
    request     = /ducupdate.php
    update interval = 86400
    retry interval = 30
    max retry    = 3

DtDNS   :
    server      = dtdns.com
    port        = 80
    request     = /api/autodns.cfm
    update interval = 86400
    retry interval = 30
    max retry    = 3

=>
```

RELATED COMMANDS:

dyndns service modify Modify specific dynamic DNS service settings.

dyndns service modify

Modify specific dynamic DNS service settings.

SYNTAX:

```
dyndns service modify name = <string>
    [server = <string>]
    [port = <{supported TCP/UDP port name} or number>]
    [request = <string>]
    [updateinterval = <number{0-2097120}>]
    [retryinterval = <number{0-600}>]
    [max_retry = <number{1-5}>]
```

where:

name	The name of the dynamic DNS service. Choose between: <ul style="list-style-type: none">▶ dyndns▶ statdns▶ custom▶ No-IP▶ DtDNS▶ gnudip.	REQUIRED
server	The hostname of the dynamic DNS server.	OPTIONAL
port	The port of the dynamic DNS server. Select one of the supported TCP/UDP port names (see "Supported TCP/UDP Port Names" on page 749) or, alternatively, specify the port number.	OPTIONAL
request	The dynamic DNS request string.	OPTIONAL
updateinterval	A number between 0 and 2097120 (seconds). Represents the time interval before a dynamic update is sent to the dynamic DNS server. Note The value 0 means disabled.	OPTIONAL
retryinterval	A number between 0 and 600 (seconds). Represents the interval between retries if communication with the dynamic DNS server fails. The default is 30 . Note The value 0 means disabled.	OPTIONAL
max_retry	A number between 1 and 5. Represents the maximum number of retries if communication with the dynamic DNS server fails. The default is 3 .	OPTIONAL

DynDNS Commands

EXAMPLE:

```
=>dyndns service modify name=custom server=mydyndns.org port=www-http request=hereiam
| updateinterval=2000000 retryinterval=15 max_retry=5
=>dyndns service list
dyndns :
    server          = members.dyndns.org
    port            = 80
    request         = /nic/update
    update interval = 2097120
    retry interval  = 30
    max retry       = 3

statdns :
    server          = members.dyndns.org
    port            = 80
    request         = /nic/update
    update interval = 0
    retry interval  = 30
    max retry       = 3

custom :
    server          = mydyndns.org
    port            = 80
    request         = hereiam
    update interval = 2000000
    retry interval  = 15
    max retry       = 5

No-IP :
    server          = dynupdate.no-ip.com
    port            = 80
    request         = /ducupdate.php
    update interval = 86400
    retry interval  = 30
    max retry       = 3

DtDNS :
    server          = dtdns.com
    port            = 80
    request         = /api/autodns.cfm
    update interval = 86400
    retry interval  = 30
    max retry       = 3

=>
```

RELATED COMMANDS:

dyndns service list List all dynamic DNS services.

Env Commands

Introduction

This chapter describes the commands of the **env** command group.

Contents

This chapter covers the following commands:

env flush	Flush all the non-system environment variables.	230
env get	Get the current value of an environment variable.	231
env list	Show all the currently available environment variables.	232
env set	Set an environment variable.	233
env unset	Delete a environment variable.	234

env flush

Flush all the non-system environment variables.

SYNTAX:

```
env flush
```

env get

Get the current value of an environment variable.

SYNTAX:

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

where:

var	The name of the environment variable.	REQUIRED
-----	---------------------------------------	----------

Tip Use the command :env list to obtain a list of all environment variables.

EXAMPLE:

```
=>env get var=_PROD_FRIENDLY_NAME  
SpeedTouch v1.0  
=>
```

RELATED COMMANDS:

env list	List all current environment variables.
----------	---

env list

Show all the currently available environment variables.

SYNTAX:

```
env list
```

RELATED COMMANDS:

env get	Get the current value of an environment variable.
----------------	---

env set

Set an environment variable.

SYNTAX:

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

where:

var	The name of the environment variable.	REQUIRED
Tip When creating an environment variable, any name is allowed. However spaces are NOT allowed and the name may NOT start with: <ul style="list-style-type: none">▶ "CONF"▶ "HOST"▶ an underscore "_"▶ the dollar sign "\$".		
value	A quoted translated string which defines the value of the environment variable.	REQUIRED
Note The value of system variables (built-in variables with names starting with an underscore "_", "CONF" or "HOST") cannot be changed.		

EXAMPLE:

For infinite TELNET time out, set the value of the variable **SESSIONTIMEOUT** to "0":

```
=>env set var=SESSIONTIMEOUT value=0
=>
```

RELATED COMMANDS:

env unset Delete a non-system environment variable.

Env Commands

env unset

Delete a environment variable.

SYNTAX:

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

where:

var	The name of the environment variable to be deleted.	REQUIRED
-----	---	----------

Note System variables (built-in variables with names starting with an underscore “_”, “CONF” or “HOST”) cannot be unset, changed or deleted.

EXAMPLE:

```
=>env list
_COMPANY_NAME=THOMSON multimedia
_COMPANY_URL=http://www.speedtouch.com
_PROD_NAME=SpeedTouch
....
CONF_DATE=March 2004
CONF_REGION=World
HOST_SETUP=user
=>env unset var=CONF_REGION
=>env list
_COMPANY_NAME=THOMSON multimedia
_COMPANY_URL=http://www.speedtouch.com
_PROD_NAME=SpeedTouch
....
CONF_DATE=March 2004
HOST_SETUP=user
=>
```

RELATED COMMANDS:

env set	Create and set a non-system environment variable.
---------	---

Eth Commands

Introduction

This chapter describes the commands of the **eth** command group.

Contents

This chapter covers the following commands:

eth ifadd	Create a new ETH interface.	237
eth ifattach	Attach an ETH interface.	238
eth ifconfig	Modify an ETH interface.	239
eth ifdelete	Delete an ETH interface.	240
eth ifdetach	Detach an ETH interface.	241
eth iflist	Display the ETH interfaces.	242
eth flush	Flush all the ETH interfaces.	243
eth bridge clear	Clear the bridge statistics.	244
eth bridge config	Modify/display the bridge configuration settings.	245
eth bridge dynvlan actlist	Display the active MAC entries for the dynamic VLAN membership.	246
eth bridge dynvlan add	Add a dynamic VLAN membership entry.	247
eth bridge dynvlan config	Modify the dynamic VLAN membership configuration.	248
eth bridge dynvlan delete	Delete a dynamic VLAN membership entry.	249
eth bridge dynvlan flush	Flush all dynamic VLAN membership entries.	250
eth bridge dynvlan list	Delete a dynamic VLAN membership entry.	249
eth bridge flush	Flush bridge interfaces and parameters.	252
eth bridge ifadd	Create a new bridge interface.	253
eth bridge ifattach	Attach a bridge interface.	254
eth bridge ifconfig	Modify a bridge interface configuration.	255
eth bridge ifdelete	Delete a bridge interface.	257
eth bridge ifdetach	Detach a bridge interface.	258
eth bridge iflist	Display the current bridge interfaces.	259
eth bridge macadd	Add a static MAC address to a bridge interface.	260
eth bridge macdelete	Remove a MAC address from the filtering database.	261
eth bridge maclist	Display the MAC address database.	262
eth bridge rule add	Add a new constraint to the VLAN learning system.	266
eth bridge rule delete	Delete a constraint from the VLAN learning system.	267
eth bridge rule flush	Flush all constraints from the VLAN learning system.	268
eth bridge rule list	Display all constraints from the VLAN learning system.	269
eth bridge vlan ifadd	Add a bridge interface to a VLAN.	270
eth bridge vlan ifconfig	Modify a bridge interface from a VLAN.	271

Eth Commands

eth bridge vlan ifdelete	Delete a bridge interface from a VLAN.	272
eth bridge vlan iflist	Display all VLANs.	273
eth device ifconfig	Configure an Ethernet interface.	274
eth device iflist	Show status of Ethernet interfaces.	275
eth switch group flush	Set all the ports to the default settings.	276
eth switch group list	List all the configured switch groups.	277
eth switch group move	Move a specified port to a specified switch group.	278
eth switch mirror capture	Define the specified port to be the Mirror Capture Port.	279
eth switch mirror egress	Enable or disable the specified port to be the Mirror Egress Port.	280
eth switch mirror ingress	Enable or disable the specified port to be the Mirror Ingress Port.	281
eth switch share add	Add a port to be shared.	282
eth switch share delete	Delete a shared port.	283
eth switch share list	Display shared ports.	284
eth vlan add	Add a new VLAN.	285
eth vlan delete	Delete a VLAN.	286
eth vlan flush	Flush all VLANs.	287
eth vlan list	Display all VLANs.	288

eth ifadd

Create a new ETH interface.

SYNTAX:

```
eth ifadd  intf = <string>
```

where:

intf The name of the new ETH interface.

REQUIRED

RELATED COMMANDS:

eth ifdelete Delete an ETH interface.

eth iflist Display the ETH interfaces.

eth ifattach

Attach an ETH interface.

SYNTAX:

```
eth ifattach intf = <string>
```

where:

intf	The name of the ETH interface to be attached.	REQUIRED
------	---	----------

RELATED COMMANDS:

eth ifdetach	Detach an ETH interface.
--------------	--------------------------

eth ifconfig

Modify an ETH interface.

SYNTAX:

```
eth ifconfig intf = <string>
    [dest = <string>]
    [retry = <number{0-65535}>]
    [vlan = <string>]
```

where:

intf	The name of the ETH interface to be configured.	REQUIRED
dest	The destination interface for this ETH interface.	OPTIONAL
retry	A number between 0 and 65535. Represents the number of times the ETH connection setup should retry before giving up. The default is 10 .	OPTIONAL
vlan	The Virtual Local Area Network (VLAN) for this ETH interface.	OPTIONAL

eth ifdelete

Delete an ETH interface.

SYNTAX:

```
eth ifdelete intf = <string>
```

where:

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

RELATED COMMANDS:

eth ifadd	Create a new ETH interface.
eth iflist	Display the ETH interfaces.

eth ifdetach

Detach an ETH interface.

SYNTAX:

```
eth ifdetach intf = <string>
```

where:

intf The name of the ETH interface to be detached.

REQUIRED

RELATED COMMANDS:

eth ifattach Attach an ETH interface.

eth iflist

Display the ETH interfaces.

SYNTAX:

```
eth iflist [intf = <string>]
```

where:

intf The name of the ETH interface to be displayed.

OPTIONAL

Note If not specified, all the ETH interfaces will be displayed.

RELATED COMMANDS:

eth ifadd Create a new ETH interface.

eth ifdelete Delete an ETH interface.

eth flush

Flush all the ETH interfaces.



The flush command does not impact previously saved configurations.

SYNTAX:

```
eth flush
```

eth bridge clear

Clear the bridge statistics.

SYNTAX:

```
eth bridge clear
```

eth bridge config

Modify/display the bridge configuration settings.

SYNTAX:

```
eth bridge config [age = <number{10-100000}>]  
                  [filter = <{no_WAN_broadcast | none}>]  
                  [vlan = <{disabled | enabled}>]
```

where:

age	A number between 10 and 100000 (seconds). Represents the lifetime of a dynamically learned Medium Access Control (MAC) address. The default is 300 .	OPTIONAL
filter	The bridge filter to be applied for all Wide Area Network (WAN) bridge ports. Choose between: <ul style="list-style-type: none">▶ no_WAN_broadcast: broadcasts from the SpeedTouch™ itself to the WAN are filtered out, broadcasts from the LAN to the WAN are still passed through▶ none: no broadcasts are filtered out. The default is no_WAN_broadcast .	OPTIONAL
vlan	Enable or disable the use of the VLAN ID of the received VLAN packets. The default is enabled .	OPTIONAL

eth bridge dynvlan actlist

Display the active MAC entries for the dynamic VLAN membership.

SYNTAX:

```
eth bridge dynvlan actlist
```

RELATED COMMANDS:

eth bridge dynvlan add	Add a dynamic VLAN membership entry.
eth bridge dynvlan config	Modify the dynamic VLAN membership configuration.
eth bridge dynvlan delete	Delete a dynamic VLAN membership entry.
eth bridge dynvlan list	Display a dynamic VLAN membership entry.

eth bridge dynvlan add

Add a dynamic VLAN membership entry.

SYNTAX:

```
eth bridge dynvlan add      [id = <number{0-100000}>]
                            hwaddr = <masked-hardware-address>
                            vlan = <string>
                            [remvlan = <{default}>]
```

where:

id	A number between 0 and 100000. Represents the id of the dynamic VLAN membership entry.	OPTIONAL
hwaddr	The (masked) ethernet MAC address of the dynamic VLAN membership entry.	REQUIRED
vlan	The VLAN for the dynamic VLAN membership entry.	REQUIRED
remvlan	The VLAN that will be removed from the bridge interface.	OPTIONAL

RELATED COMMANDS:

eth bridge dynvlan actlist	Display the active MAC entries for the dynamic VLAN membership.
eth bridge dynvlan config	Modify the dynamic VLAN membership configuration.
eth bridge dynvlan delete	Delete a dynamic VLAN membership entry.
eth bridge dynvlan list	Display a dynamic VLAN membership entry.

eth bridge dynvlan config

Modify the dynamic VLAN membership configuration.

SYNTAX:

```
eth bridge dynvlan config [timeout = <number{0-100000}>]
```

where:

timeout	A number between 0 and 100000 (seconds). Represents timeout for the dynamic entries.	OPTIONAL
---------	---	----------

RELATED COMMANDS:

eth bridge dynvlan actlist	Display the active MAC entries for the dynamic VLAN membership.
eth bridge dynvlan add	Add a dynamic VLAN membership entry.
eth bridge dynvlan delete	Delete a dynamic VLAN membership entry.
eth bridge dynvlan list	Display a dynamic VLAN membership entry.

eth bridge dynvlan delete

Delete a dynamic VLAN membership entry.

SYNTAX:

```
eth bridge dynvlan delete    id = <number{0-100000}>
```

where:

id	A number between 0 and 100000. Represents the id of a dynamic VLAN membership.	REQUIRED
----	---	----------

RELATED COMMANDS:

eth bridge dynvlan actlist	Display the active MAC entries for the dynamic VLAN membership.
eth bridge dynvlan add	Add a dynamic VLAN membership entry.
eth bridge dynvlan config	Modify the dynamic VLAN membership configuration.
eth bridge dynvlan list	Display a dynamic VLAN membership entry.

eth bridge dynvlan flush

Flush all dynamic VLAN membership entries.

SYNTAX:

```
eth bridge dynvlan flush
```

eth bridge dynvlan list

Display a dynamic VLAN membership entry.

SYNTAX:

```
eth bridge dynvlan list
```

RELATED COMMANDS:

<code>eth bridge dynvlan actlist</code>	Display the active MAC entries for the dynamic VLAN membership.
<code>eth bridge dynvlan add</code>	Add a dynamic VLAN membership entry.
<code>eth bridge dynvlan config</code>	Modify the dynamic VLAN membership configuration.
<code>eth bridge dynvlan delete</code>	Delete a dynamic VLAN membership entry.

eth bridge flush

Flush bridge interfaces and parameters.



The flush command does not impact previously saved configurations.

SYNTAX:

```
eth bridge flush
```

eth bridge ifadd

Create a new bridge interface.

SYNTAX:

```
eth bridge ifadd intf = <string>
```

where:

intf	The name of the new bridge interface.	REQUIRED
------	---------------------------------------	----------

RELATED COMMANDS:

eth bridge ifdelete	Delete a bridge interface.
eth bridge iflist	Display the current bridge interfaces.

eth bridge ifattach

Attach a bridge interface.

SYNTAX:

```
eth bridge ifattach  intf = <string>
```

where:

intf The name of the bridge interface to be attached.

REQUIRED

RELATED COMMANDS:

eth bridge ifdetach Detach a bridge interface.

eth bridge ifconfig

Modify a bridge interface configuration.

SYNTAX:

```
eth bridge ifconfig intf = <string>
    [dest = <string>]
    [portstate = <{disabled | learning | forwarding}>]
    [retry = <number{0-65535}>]
    [prioconfig = <{disabled|overwrite|increase}>]
    [vlan = <string>]
    [ipprec = <{disabled|precedence|dscp}>]
    [priority = <number{0-7}>]
    [regenprio = <string>]
    [ingressfiltering = <{disabled | enabled}>]
    [acceptvlanonly = <{disabled | enabled}>]
    [mcastfilter = <{disabled|enabled}>]
    [dynvlan = <{disabled|enabled}>]
    [igmpsnooping = <{disabled|enabled}>]
```

where:

intf	The name of the bridge interface to be configured.	REQUIRED
dest	The destination for this interface. Typically an ATM or a physical interface name.	OPTIONAL
portstate	The bridge portstate for this interface. Choose between: ▶ disabled ▶ learning ▶ forwarding. The default is 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. The default is 10 .	OPTIONAL
vlan	Select the default VLAN.	OPTIONAL
prioconfig	The priority configuration for this interface.	OPTIONAL
ipprec	The IP precedence for this interface. Choose between: ▶ disabled ▶ precedence ▶ dscp. The default is disabled .	OPTIONAL
priority	A number between 0 and 7. Represents the default priority for tagging egress packets. The default is 0 .	OPTIONAL

Eth Commands

regenprio	The priority regeneration table for tagged ingress packets. The default is 01234567 .	OPTIONAL
ingressfiltering	Enable/disable discard of tagged ingress packets if the interface is not part of the VLAN. The default is disabled .	OPTIONAL
acceptvlanonly	Enable or disable receipt of tagged ingress packets. The default is disabled .	OPTIONAL
mcastfilter	Enable or disable the discard of multicast packets on this port The default is disabled .	OPTIONAL
dynvlan	Enable or disable dynamic VLAN membership checking for this interface. The default is disabled .	OPTIONAL
igmpsnooping	Enable or disable IGMP snooping for this interface. The default is disabled .	OPTIONAL

EXAMPLE:

```
=>eth bridge iflist
OBC      : dest : Internal
...
...
BrEthoA_br: dest : (none)
    Connection State: not-connected    Retry: 10
    Port: (Unassigned)    PortNr: (Unknown)    PortState: forwarding

=>eth bridge ifconfig intf=BrEthoA_br dest=BrEthhoA_atm
=>eth bridge iflist
OBC      : dest : Internal
...
...
BrEthoA_br: dest : BrEthhoA_atm
    Connection State: not-connected    Retry: 10
    Port: (Unassigned)    PortNr: (Unknown)    PortState: forwarding
=>
```

eth bridge ifdelete

Delete a bridge interface.

SYNTAX:

```
eth bridge ifdelete intf = <string>
```

where:

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

RELATED COMMANDS:

eth bridge ifadd	Create a new bridge interface.
eth bridge iflist	Display the current bridge interfaces.

eth bridge ifdetach

Detach a bridge interface.

SYNTAX:

```
eth bridge ifdetach    intf = <string>
```

where:

intf The name of the bridge interface to be detached.

REQUIRED

RELATED COMMANDS:

eth bridge ifattach Attach a bridge interface.

eth bridge iflist

Display the current bridge interfaces.

SYNTAX:

```
eth bridge iflist [intf = <string>]
```

where:

intf	The name of the bridge interface to be displayed.	OPTIONAL
Note	If not specified, all bridge interfaces are shown.	

RELATED COMMANDS:

eth bridge ifadd	Create a new bridge interface.
eth bridge ifdelete	Delete a bridge interface.

Eth Commands

eth bridge macadd

Add a static MAC address to a bridge interface.

This command allows to manually add static MAC addresses, which should normally be dynamically discovered by the bridge itself.

SYNTAX:

```
eth bridge macadd    intf = <string>
                     hwaddr = <hardware-address>
                     [vlan = <string>]
```

where:

intf	The name of the bridge interface to which the MAC address must be added.	REQUIRED
hwaddr	The Ethernet MAC address of the new entry.	REQUIRED
vlan	The VLAN.	OPTIONAL

EXAMPLE:

```
=>eth bridge maclist
00:10:a4:ad:32:cf -- dynamic, ethport1, 300 seconds
00:90:d0:8b:fc:2c -- permanent
01:00:5e:00:00:67 -- static
01:00:5e:7f:ff:fa -- static
01:80:c2:00:00:00 -- permanent
01:80:c2:00:00:01 -- permanent
01:80:c2:00:00:02 -- permanent
01:80:c2:00:00:03 -- permanent
...
01:80:c2:00:00:0e -- permanent
01:80:c2:00:00:0f -- permanent
01:80:c2:00:00:10 -- permanent
ff:ff:ff:ff:ff:ff -- permanent
=>eth bridge macadd intf=ethport2 hwaddr=00:80:9f:01:23:45
=>eth bridge maclist
00:0d:9d:47:dd:aa -- dynamic, ethport1, 300 seconds
00:80:9f:01:23:45 -- static
00:90:d0:72:88:64 -- permanent
01:00:5e:00:00:67 -- static
01:00:5e:7f:ff:fa -- static
01:80:c2:00:00:00 -- permanent
01:80:c2:00:00:01 -- permanent
01:80:c2:00:00:02 -- permanent
01:80:c2:00:00:03 -- permanent
01:80:c2:00:00:0e -- permanent
01:80:c2:00:00:0f -- permanent
01:80:c2:00:00:10 -- permanent
ff:ff:ff:ff:ff:ff -- permanent
=>
```

RELATED COMMANDS:

- | | |
|----------------------|---|
| eth bridge macdelete | Remove a MAC address from the filtering database. |
| eth bridge maclist | Display the MAC address database. |

eth bridge macdelete

Remove a MAC address from the filtering database.

SYNTAX:

```
eth bridge macdelete hwaddr = <hardware-address>
                           [vlan = <string>]
```

where:

hwaddr	The Ethernet MAC address of the entry to be deleted.	REQUIRED
vlan	The VLAN.	OPTIONAL

EXAMPLE:

```
=>eth bridge maclist
00:0d:9d:47:dd:aa -- dynamic, ethport1
00:80:9f:01:23:45 -- static
00:90:d0:72:88:64 -- permanent
01:00:5e:00:00:67 -- static
01:00:5e:7f:ff:fa -- static
01:80:c2:00:00:00 -- permanent
01:80:c2:00:00:01 -- permanent
01:80:c2:00:00:02 -- permanent
01:80:c2:00:00:03 -- permanent
...
01:80:c2:00:00:0e -- permanent
01:80:c2:00:00:0f -- permanent
01:80:c2:00:00:10 -- permanent
ff:ff:ff:ff:ff:ff -- permanent
=>eth bridge macdelete hwaddr=00:80:9f:01:23:45
=>eth bridge maclist
00:0d:9d:47:dd:aa -- dynamic, ethport1
00:90:d0:72:88:64 -- permanent
01:00:5e:00:00:67 -- static
01:00:5e:7f:ff:fa -- static
01:80:c2:00:00:00 -- permanent
01:80:c2:00:00:01 -- permanent
01:80:c2:00:00:02 -- permanent
01:80:c2:00:00:03 -- permanent
01:80:c2:00:00:0e -- permanent
01:80:c2:00:00:0f -- permanent
01:80:c2:00:00:10 -- permanent
ff:ff:ff:ff:ff:ff -- permanent
=>
```

RELATED COMMANDS:

- | | |
|--------------------|---|
| eth bridge macadd | Add a static MAC address to a bridge interface. |
| eth bridge maclist | Display the MAC address database. |

Eth Commands

eth bridge maclist

Display the MAC address database.

SYNTAX:

```
eth bridge maclist
```

EXAMPLE:

```
=>eth bridge maclist
00:0d:9d:47:dd:aa -- dynamic, ethport1, 300 seconds
00:90:d0:72:88:64 -- permanent
01:00:5e:00:00:67 -- static
01:00:5e:7f:ff:fa -- static
01:80:c2:00:00:00 -- permanent
01:80:c2:00:00:01 -- permanent
01:80:c2:00:00:02 -- permanent
01:80:c2:00:00:03 -- permanent
01:80:c2:00:00:04 -- permanent
01:80:c2:00:00:05 -- permanent
01:80:c2:00:00:06 -- permanent
01:80:c2:00:00:07 -- permanent
01:80:c2:00:00:08 -- permanent
01:80:c2:00:00:09 -- permanent
01:80:c2:00:00:0a -- permanent
01:80:c2:00:00:0b -- permanent
01:80:c2:00:00:0c -- permanent
01:80:c2:00:00:0d -- permanent
01:80:c2:00:00:0e -- permanent
01:80:c2:00:00:0f -- permanent
01:80:c2:00:00:10 -- permanent
ff:ff:ff:ff:ff:ff -- permanent
=>
```

RELATED COMMANDS:

eth bridge macadd	Add a static MAC address to a bridge interface.
eth bridge macdelete	Remove a MAC address from the filtering database.

eth bridge mcdadd

Add a multicast group to be dropped to the database.

SYNTAX:

```
eth bridge mcdadd    srcintfs = <string or combination of strings separated by  
                           '+'>  
                      dstintfs = <string or combination of strings separated by  
                           '+'>
```

where:

srcports	The source bridge interfaces from which multicast packets must be dropped.	REQUIRED
dstports	The destination bridge interfaces for which the multicast packets must be dropped.	REQUIRED

RELATED COMMANDS:

eth bridge mcddelete	Delete a multicast group to be dropped from the database.
eth bridge mcclist	Display the multicast group to be dropped from the database.

eth bridge mcddelete

Delete a multicast group to be dropped from the database.

SYNTAX:

```
eth bridge mcddelete
```

RELATED COMMANDS:

eth bridge mcdadd Add a multicast group to be dropped to the database.

eth bridge mcclist Display the multicast group to be dropped from the database.

eth bridge mcdlist

Display the multicast group to be dropped from the database.

SYNTAX:

```
eth bridge mcdlist
```

RELATED COMMANDS:

- | | |
|----------------------|---|
| eth bridge mcdadd | Add a multicast group to be dropped to the database. |
| eth bridge mcddelete | Delete a multicast group to be dropped from the database. |

eth bridge rule add

Add a new constraint to the VLAN learning system.

SYNTAX:

```
eth bridge rule add    type = <{shared | independent}>
                      vlan = <string>
                      [vlan2 = <string>]
                      [isi = <number{0-32}>]
```

where:

type	Type of constraint. Choose between: ▶ shared ▶ independent.	REQUIRED
vlan	The VLAN to which the constraint belongs.	REQUIRED
vlan2	The second VLAN for a shared constraint.	OPTIONAL
	Note This parameter is required when type is shared .	
isi	A number between 0 and 32. Represents the Independent Set ID (ISI) for an independent constraint.	OPTIONAL
	Note This parameter is required when type is independent .	

EXAMPLE:

```
=>eth bridge rule add type=independant vlan=default isi=1
=>eth bridge rule list
Index      Type          VLAN                  Parameter
-----      ----          ----                  -----
1          independant   default               Independent set identifier: 1
=>
```

RELATED COMMANDS:

- | | |
|-------------------------------|--|
| eth bridge rule delete | Delete a constraint from the VLAN learning system. |
| eth bridge rule list | Display all constraints from the VLAN learning system. |

eth bridge rule delete

Delete a constraint from the VLAN learning system.

SYNTAX:

```
eth bridge rule delete index = <number{0-32}>
```

where:

index	Index of the constraint to be deleted.	REQUIRED
-------	--	----------

Tip Use the command :**eth bridge rule list** to obtain a list of the indexes of the constraints.

EXAMPLE:

```
=>eth bridge rule list
Index      Type          VLAN
-----    ----
1         independant   default
=>eth bridge rule delete index=1
=>eth bridge rule list
Index      Type          VLAN
-----    ----
=>
```

RELATED COMMANDS:

eth bridge rule add Add a new constraint to the VLAN learning system.

eth bridge rule list Display all constraints from the VLAN learning system.

eth bridge rule flush

Flush all constraints from the VLAN learning system.

SYNTAX:

```
eth bridge rule flush
```

eth bridge rule list

Display all constraints from the VLAN learning system.

SYNTAX:

```
eth bridge rule list
```

EXAMPLE:

```
=>eth bridge rule list
Index      Type          VLAN
-----      ----
1           independant   default
=>                                         Parameter
                                         -----
                                         Independent set identifier: 1
```

RELATED COMMANDS:

eth bridge rule add Add a new constraint to the VLAN learning system.

eth bridge rule delete Delete a constraint from the VLAN learning system.

eth bridge vlan ifadd

Add a bridge interface to a VLAN.

SYNTAX:

```
eth bridge vlan ifadd name = <string>
                      intf = <string>
                      [untagged = <{disabled | enabled}>]
```

where:

name	The VLAN name to which a bridge interface must be added.	REQUIRED
intf	The name of the bridge interface to be added to the VLAN.	REQUIRED
untagged	Enable or disable the interface as untagged for this VLAN. The default is disabled .	OPTIONAL

RELATED COMMANDS:

eth bridge vlan ifconfig	Modify a bridge interface from a VLAN.
eth bridge vlan ifdelete	Delete a bridge interface from a VLAN.
eth bridge vlan iflist	Display all VLANs.

eth bridge vlan ifconfig

Modify a bridge interface from a VLAN.

SYNTAX:

```
eth bridge vlan ifconfig name = <string>
                           intf = <string>
                           untagged = <{disabled | enabled}>
```

where:

name	The VLAN name for which a bridge interface must be modified.	REQUIRED
intf	The name of the bridge interface to be modified.	REQUIRED
untagged	Enable or disable the interface as untagged for this VLAN.	REQUIRED

RELATED COMMANDS:

- eth bridge vlan ifadd** Add a bridge interface to a VLAN.
- eth bridge vlan ifdelete** Delete a bridge interface from a VLAN.
- eth bridge vlan iflist** Display all VLANs.

eth bridge vlan ifdelete

Delete a bridge interface from a VLAN.

SYNTAX:

```
eth bridge vlan ifdelete name = <string>
                           intf = <string>
```

where:

name	The VLAN name for which a bridge interface must be deleted.	REQUIRED
intf	The name of the bridge interface to be deleted.	REQUIRED

RELATED COMMANDS:

eth bridge vlan ifadd	Add a bridge interface to a VLAN.
eth bridge vlan ifconfig	Modify a bridge interface from a VLAN.
eth bridge vlan iflist	Display all VLANs.

eth bridge vlan iflist

Display all VLANs.

SYNTAX:

```
eth bridge vlan iflist
```

RELATED COMMANDS:

- | | |
|--------------------------|--|
| eth bridge vlan ifadd | Add a bridge interface to a VLAN. |
| eth bridge vlan ifconfig | Modify a bridge interface from a VLAN. |
| eth bridge vlan ifdelete | Delete a bridge interface from a VLAN. |

eth device ifconfig

Configure an Ethernet interface.

SYNTAX:

```
eth device ifconfig  intf = <string>
                    [type = <{auto | 10BaseTHD | 10BaseTFD | 100BaseTHD
                                | 100BaseTFD} or number>]
                    [state = <{enabled | disabled}>]
```

where:

intf	The name of a physical interface.	REQUIRED
type	The Ethernet type. Select either: <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. or enter a number between 0 (auto) and 5 (100BaseTFD).	OPTIONAL
state	Enable or disable the interface. The default is enabled .	OPTIONAL

EXAMPLE:

```
=>eth device iflist
Interface      Type          Result        State
ethif1        auto          100BaseTFD   UP [forwarding]
ethif2        auto          100BaseTFD   UP [forwarding]
ethif3        auto          100BaseTFD   UP [forwarding]
ethif4        auto          100BaseTFD   UP [forwarding]
usbif1
=>eth device ifconfig intf=ethif4 state=disabled
=>eth device iflist
Interface      Type          Result        State
ethif1        auto          100BaseTFD   UP [forwarding]
ethif2        auto          100BaseTFD   UP [forwarding]
ethif3        auto          100BaseTFD   UP [forwarding]
ethif4        auto          100BaseTFD   DOWN [disabled]
usbif1
=>
```

RELATED COMMANDS:

eth device iflist Show status of Ethernet interfaces.

eth device iflist

Show status of Ethernet interfaces.

SYNTAX:

```
eth device iflist
```

EXAMPLE:

```
=>eth device iflist
Interface      Type      Result      State
ethif1        auto     100BaseTFD    UP [forwarding]
ethif2        auto     100BaseTFD    UP [forwarding]
ethif3        auto     100BaseTFD    UP [forwarding]
ethif4        auto     100BaseTFD    UP [forwarding]
usbif1
wlif1
wlif2
wlif3
wlif4
wlif5
=>
```

DESCRIPTION:

- ▶ **Type**: Indicates the configured Ethernet communication speed and duplex mode.
- ▶ **Result** : Indicates the effective operating status if Type equals “auto”. In other cases, when the Ethernet types do NOT match, Result Type will equal “unknown” and no Ethernet connectivity will exist.

RELATED COMMANDS:

eth device ifconfig Configure an Ethernet interface.

eth switch group flush

Set all the ports to the default settings.

All the ports are placed in group 0.

SYNTAX:

```
eth switch group flush
```

EXAMPLE:

```
=>eth switch group list
Group 0 Ports: 4
Group 1 Ports: 1
Group 2 Ports: 2 3
=>eth switch group flush
=>eth switch group list
Group 0 Ports: 1 2 3 4
=>
```

eth switch group list

List all the configured switch groups.

SYNTAX:

```
eth switch group list
```

EXAMPLE:

```
=>eth switch group list
Group 0 Ports: 4
Group 1 Ports: 1
Group 2 Ports: 2 3
=>
```

RELATED COMMANDS:

[eth switch group move](#)

Move a specified port to a specified switch group.

eth switch group move

Move a specified port to a specified switch group.

SYNTAX:

```
eth switch group move  group = <number{0-3}>
                      port = <number{1-4}>
```

where:

group	A number between 0 and 4. Represents the group ID to which the port must be moved.	REQUIRED
port	A number between 01 and 4. Represents the port to be moved.	REQUIRED

EXAMPLE:

```
=>eth switch group list
Group 0 Ports: 1 2 3 4
=>eth switch group move group=3 port=1
=>eth switch group list
Group 0 Ports: 2 3 4
Group 3 Ports: 1
=>
```

RELATED COMMANDS:

eth switch group list List all the configured switch groups.

eth switch mirror capture

Define the specified port to be the Mirror Capture Port.



Only one port can be the Mirror Capture Port at any one time.

SYNTAX:

```
eth switch mirror capture    port = <number{1-4}>
```

where:

port	The port to be the Mirror Capture Port.	REQUIRED
Note	If no port number is specified, then the port number of the Mirror Capture Port is shown.	

EXAMPLE:

```
=>eth switch mirror capture port=2
=>eth switch mirror capture
Mirror capture port = 2
=>
```

RELATED COMMANDS:

eth switch mirror egress	Enable or disable the specified port to be the Mirror Egress Port.
eth switch mirror ingress	Enable or disable the specified port to be the Mirror Ingress Port.

eth switch mirror egress

Enable or disable the specified port to be the Mirror Egress Port.



- 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:

```
eth switch mirror egress    port = <number{1-4}>
                            [state = <{enabled | disabled}>]
```

where:

port	The port to be the Mirror Egress Port.	REQUIRED
------	--	----------

Note If no port number is specified, then the port number of the current Mirror Egress Port is shown.

state	Enable or disable the port as Mirror Egress Port. The default is enabled .	OPTIONAL
-------	--	----------

EXAMPLE:

Here port 1 is enabled as Mirror Egress Port

```
=>eth switch mirror egress port=1
=>eth switch mirror egress
Egress mirror port = 1
=>
```

Here port 1 is disabled as Mirror Egress Port

```
=>eth switch mirror egress
Egress mirror port = 1
=>eth switch mirror egress port=1 state=disabled
=>eth switch mirror egress
=>
```

RELATED COMMANDS:

eth switch mirror capture Define the specified port to be the Mirror Capture Port.

eth switch mirror ingress Enable or disable the specified port to be the Mirror Ingress Port.

eth switch mirror ingress

Enable or disable the specified port to be the Mirror Ingress Port.



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:

```
eth switch mirror ingress    port = <number{1-4}>
                                [state = <{enabled | disabled}>]
```

where:

port The port to be the Mirror Ingress Port. REQUIRED

Note If no port number is specified, then the port number of the current Mirror Ingress Port is shown.

state Enable or disable the port as Mirror Ingress Port. OPTIONAL
The default is **enabled**.

EXAMPLE:

Here port 2 is enabled as Mirror Ingress Port

```
=>eth switch mirror ingress port=2
=>eth switch mirror ingress
Ingress mirror port = 2
=>
```

Here port 2 is disabled as Mirror Ingress Port

```
=>eth switch mirror ingress
Ingress mirror port = 2
=>eth switch mirror ingress port=2 state=disabled
=>eth switch mirror ingress
=>
```

RELATED COMMANDS:

eth switch mirror capture Define the specified port to be the Mirror Capture Port.

eth switch mirror egress Enable or disable the specified port to be the Mirror Egress Port.

eth switch share add

Add a port to be shared.

SYNTAX:

```
eth switch share add    port = <number{1-4}>
                           shared = <number{1-4}>
```

where:

port	A number between 1 and 4. Represents the port.	REQUIRED
shared	A number between 1 and 4. Represents the shared port.	REQUIRED

RELATED COMMANDS:

- | | |
|-------------------------|-----------------------|
| eth switch share delete | Delete a shared port. |
| eth switch share list | Display shared ports. |

eth switch share delete

Delete a shared port.

SYNTAX:

```
eth switch share delete    port = <number{1-4}>
                           shared = <number{1-4}>
```

where:

port	A number between 1 and 4. Represents the port.	REQUIRED
shared	A number between 1 and 4. Represents the shared port.	REQUIRED

RELATED COMMANDS:

- | | |
|-----------------------|--------------------------|
| eth switch share add | Add a port to be shared. |
| eth switch share list | Display shared ports. |

eth switch share list

Display shared ports.

SYNTAX:

```
eth switch share list
```

RELATED COMMANDS:

eth switch share add Add a port to be shared.

eth switch share delete Delete a shared port.

eth vlan add

Add a new VLAN.

SYNTAX:

```
eth vlan add  name = <string>
              vid = <number{2-4094}>
              [addrule = <{disabled | enabled}>]
```

where:

name	The new VLAN name.	REQUIRED
vid	A number representing the new VLAN ID.	REQUIRED
Note	The ID 1 is reserved for the default VLAN.	
addrule	Add (enabled) the default bridge constraint or not (disabled). The default is enabled .	OPTIONAL

EXAMPLE:

```
=>eth vlan list
Vid      Name
---      ---
1        default
=>eth vlan add name=myVLAN vid=2
=>eth vlan list
Vid      Name
---      ---
1        default
2        myVLAN
=>
```

RELATED COMMANDS:

eth vlan delete	Delete a VLAN.
eth vlan list	Display all VLANs.

eth vlan delete

Delete a VLAN.

SYNTAX:

```
eth vlan delete name = <string>
```

where:

name The name of the VLAN to be deleted.

REQUIRED

EXAMPLE:

```
=>eth vlan list
Vid      Name
---      ---
1        default
2        myVLAN
=>eth vlan delete name=myVLAN vid=2
=>eth vlan list
Vid      Name
---      ---
1        default
=>
```

RELATED COMMANDS:

eth vlan add	Add a new VLAN.
eth vlan list	Display all VLANs.

eth vlan flush

Flush all VLANs.



All the VLANs, except the default VLAN will be flushed.

SYNTAX:

```
eth vlan flush
```

EXAMPLE:

```
=>eth vlan list
Vid      Name
---      ---
1        default
2        myVLAN
3        yourVLAN
=>eth vlan flush
=>eth vlan list
Vid      Name
---      ---
1        default
=>
```

eth vlan list

Display all VLANs.

SYNTAX:

```
eth vlan list
```

EXAMPLE:

```
=>eth vlan list
Vid      Name
---      ---
1        default
2        myVLAN
=>
```

RELATED COMMANDS:

- | | |
|------------------------|-----------------|
| eth vlan add | Add a new VLAN. |
| eth vlan delete | Delete a VLAN. |

Expr Commands

Introduction

This chapter describes the commands of the **expr** command group.

Contents

This chapter covers the following commands:

expr add	Add a subexpression to an expression.	290
expr delete	Delete an expression.	293
expr flush	Flush all the expressions.	295
expr list	List the expressions.	296
expr modify	Modify an expression.	302

Expr Commands

expr add

Add a subexpression to an expression.

SYNTAX:

```
expr add name = <string>
        type = <{intf | ip | serv}>
        addr [!] = <ip-range>
        [intf [!] = <string>]
        [intfgroup [!] = <{wan|local|lan|tunnel|dmz|guest} or number>]
        [tos [!] = <number{0-255}>]
        [precedence [!] = <IP precedence type> or <number>]
        [dscp [!] = <DSCP name> or <number>]
        [proto = <supported IP protocol name> or <number>]
        [srcport [!] = <supported TCP/UDP port name> or <number>]
        [srcportend = <supported TCP/UDP port name> or <number>]
        [dstport [!] = <supported TCP/UDP port name> or <number>]
        [dstportend = <supported TCP/UDP port name> or <number>]
        [icmptype [!] = <supported ICMP type name> or <number>]
        [icmpcode [!] = <number{0-15}>]
        [icmpcodeend = <number{0-15}>]
```



If a value is preceded by a "!", it means NOT.

For example "intfgroup!=wan" means "if the interface group is different from WAN".

where:

name	The name of an existing expression.	REQUIRED
type	The expression type. Choose between: ▶ intf ▶ ip ▶ serv.	REQUIRED
addr	The IP address (or range). Supports ip/mask notation. Note Only for expression type <i>ip</i> .	REQUIRED
intf	The IP interface name. Note Only for expression type <i>intf</i> .	OPTIONAL
intfgroup	The IP interface group. Choose between: ▶ wan ▶ local ▶ lan ▶ tunnel ▶ dmz ▶ guest. Note Only for expression type <i>intf</i> .	OPTIONAL

tos	A number between 0 and 255. Represents the Type of Service (ToS) specification in the IP packet.	OPTIONAL
	Note The parameters <i>tos</i>, <i>precedence</i> and <i>dscp</i> are mutually exclusive.	
precedence	The precedence in the IP packet (part of tos). Select an IP precedence (see " IP Precedence " on page 754) or, alternatively, specify the number.	OPTIONAL
	Note The parameters <i>tos</i>, <i>precedence</i> and <i>dscp</i> are mutually exclusive.	
dscp	The Differentiated Services Code Point (DSCP) in the IP packet (part of tos). Select a DSCP (see " Differentiated Services Code Point (DSCP) " on page 755) or, alternatively, specify the number.	OPTIONAL
	Note The parameters <i>tos</i>, <i>precedence</i> and <i>dscp</i> are mutually exclusive.	
proto	The protocol (name or number) expected in the IP packet. Select one of the following protocols: <i>icmp</i>, <i>igmp</i>, <i>ipinip</i>, <i>tcp</i>, <i>udp</i>, <i>ah</i>, <i>esp</i>, <i>ipcomp</i> or, 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 749) or, alternatively, specify the port number.	OPTIONAL
srcportend	The source TCP/UDP port range end (inclusive). Select one of the supported TCP/UDP port names (see " Supported TCP/UDP Port Names " on page 749) or, alternatively, specify the port 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 749) or, alternatively, specify the port number.	OPTIONAL
dstportend	The destination TCP/UDP port range end. (inclusive). Select one of the supported TCP/UDP port names (see " Supported TCP/UDP Port Names " on page 749) or, alternatively, specify the port number.	OPTIONAL
icmptype	The Internet Control Message Protocol (ICMP) type (name or number) of the packet. Select one of the supported ICMP types (see " Supported ICMP Type Names " on page 752) or, alternatively, specify the type number.	OPTIONAL
icmpcode	A number between 0 and 15. Represents the ICMP code (or beginning of range) of the packet.	OPTIONAL
icmpcodeend	A number between 0 and 15. Represents the ICMP code range end (inclusive).	OPTIONAL

EXAMPLE:

```
=>expr list
name          type    use flags expression

-----
_myPPP_ppp      intf    0 D    1. intf=myPPP_ppp
_`eth0          intf    0 D    1. intf=eth0
wan            intf    0      1. intfgroup=0
lan             intf   28     1. intfgroup=2
local           intf    0      1. intfgroup=1
notwan          intf    1      1. intfgroup=2
                           2. intfgroup=1
_10.0.0.138     ip     0 D    1. addr=10.0.0.138
_192.168.1.254 ip     0 D    1. addr=192.168.1.254

...
DiffServ        serv    0      1. dscp!=0
sip             serv    1      1. proto=17 dst-prt=5060
                           2. proto=6 dst-prt=5060
h323            serv    1      1. proto=6 dst-prt=1720
                           2. proto=17 dst-prt=1718
                           3. proto=17 dst-prt=1719

=>expr add name=myEXPR type=intf intf=Internet intfgroup=lan
=>expr list
name          type    use flags expression

-----
_myPPP_ppp      intf    0 D    1. intf=myPPP_ppp
_`eth0          intf    0 D    1. intf=eth0
wan            intf    0      1. intfgroup=0
lan             intf   28     1. intfgroup=2
local           intf    0      1. intfgroup=1
notwan          intf    1      1. intfgroup=2
                           2. intfgroup=1
myEXPR          intf    0      1. intf=Internet intfgroup=2
_10.0.0.138     ip     0 D    1. addr=10.0.0.138
_192.168.1.254 ip     0 D    1. addr=192.168.1.254

...
DiffServ        serv    0      1. dscp!=0
sip             serv    1      1. proto=17 dst-prt=5060
                           2. proto=6 dst-prt=5060
h323            serv    1      1. proto=6 dst-prt=1720
                           2. proto=17 dst-prt=1718
                           3. proto=17 dst-prt=1719
=>
```

RELATED COMMANDS:

- | | |
|--------------------|-----------------------|
| expr delete | Delete an expression. |
| expr list | List the expressions. |

expr delete

Delete an expression.

SYNTAX:

```
expr delete name = <string>
              [index = <number>]
```

where:

name	The name of the expression to be deleted.	REQUIRED
index	The index of the subexpression to be deleted.	OPTIONAL
Note If not specified, all the subexpressions wil be deleted.		

EXAMPLE:

```
=>expr list
name          type    use flags expression

-----
_myPPP_ppp      intf    0 D    1. intf=myPPP_ppp
_ether0         intf    0 D    1. intf=ether0
wan             intf    0      1. intfgroup=0
lan              lan    28     1. intfgroup=2
local            local   0      1. intfgroup=1
notwan          notwan  1      1. intfgroup=2
                           2. intfgroup=1
myEXPR          myEXPR  0      1. intf=Internet intfgroup=2
_10.0.0.138     ip      0 D    1. addr=10.0.0.138
_192.168.1.254 ip      0 D    1. addr=192.168.1.254

...
DiffServ        serv    0      1. dscp!=0
sip              serv    1      1. proto=17 dst-prt=5060
                           2. proto=6 dst-prt=5060
h323            serv    1      1. proto=6 dst-prt=1720
                           2. proto=17 dst-prt=1718
                           3. proto=17 dst-prt=1719

=>expr delete name=myEXPR
=>expr list
name          type    use flags expression

-----
_myPPP_ppp      intf    0 D    1. intf=myPPP_ppp
_ether0         intf    0 D    1. intf=ether0
wan             intf    0      1. intfgroup=0
lan              lan    28     1. intfgroup=2
local            local   0      1. intfgroup=1
notwan          notwan  1      1. intfgroup=2
                           2. intfgroup=1
_10.0.0.138     ip      0 D    1. addr=10.0.0.138
_192.168.1.254 ip      0 D    1. addr=192.168.1.254

...
DiffServ        serv    0      1. dscp!=0
sip              serv    1      1. proto=17 dst-prt=5060
                           2. proto=6 dst-prt=5060
h323            serv    1      1. proto=6 dst-prt=1720
                           2. proto=17 dst-prt=1718
                           3. proto=17 dst-prt=1719

=>
```

RELATED COMMANDS:

- | | |
|------------------|---------------------------------------|
| expr add | Add a subexpression to an expression. |
| expr list | List the expressions. |

expr flush

Flush all the expressions.

SYNTAX:

```
expr flush
```

expr list

List the expressions.

SYNTAX:

```
expr list [name = <string>]
          [type = <{intf | ip | serv}>]
          [format = <{pretty | cli}>]
```

where:

name	The name of an existing expression.	OPTIONAL
	Note If not specified, all the expressions will be listed.	
type	The expression type. Choose between: ▶ intf ▶ ip ▶ serv.	OPTIONAL
	Note If not specified, the expressions of all types will be shown.	
format	Select the output format of the list. Choose between: ▶ pretty : the expressions are shown as intuitive output in clear text. ▶ cli : the expressions are shown via the CLI commands configuration. The default is pretty .	OPTIONAL

EXAMPLE 1:

```
=>expr list
name          type   use flags expression
-----
_Internet      intf    0 D   1. intf=Internet
_LocalNetwork  intf    0 D   1. intf=LocalNetwork
_HTTPI_if_0    intf    1 D   1. intf=LocalNetwork
                           2. intf=LocalNetwork
                           3. intf=LocalNetwork
_HTTP_if_0     intf    1 D   1. intfgroup=2
_HTTPs_if_0    intf    1 D   1. intfgroup=2
_FTP_if_0      intf    1 D   1. intfgroup=2
_TELNET_if_0   intf    1 D   1. intfgroup=2
_DNS-S_if_0    intf    1 D   1. intfgroup=2
_SNMP_AGENT_if_0  intf    1 D   1. intfgroup=2
_PING_RESPONDER_if_0  intf    1 D   1. intfgroup=2
wan            intf    1     1. intfgroup=0
lan             intf    2     1. intfgroup=2
local           intf    0     1. intfgroup=1
_10.0.0.138    ip      0 D   1. addr=10.0.0.138
_192.168.1.254 ip      0 D   1. addr=192.168.1.254
private         ip      0     1. addr=10.0.0.0/8
                           2. addr=172.[16-31].*.*
                           3. addr=192.168.1.0/24
_ssdp_ip       ip      1     1. addr=239.255.255.250
_mdap_ip       ip      1     1. addr=224.0.0.103
_HTTP_sv_0     serv    1 D   1. proto=6 dst-prt=80
_HTTPs_sv_0    serv    1 D   1. proto=6 dst-prt=443
_FTP_sv_0      serv    1 D   1. proto=6 dst-prt=21
_TELNET_sv_0   serv    1 D   1. proto=6 dst-prt=23
_RIP_sv_0      serv    1 D   1. proto=17 src-prt=520 dst-prt=520
_RIP-Query_sv_0 serv    1 D   1. proto=17 dst-prt=520
_DNS-S_sv_0    serv    1 D   1. proto=17 dst-prt=53
_SNMP_AGENT_sv_0 serv    1 D   1. proto=17 dst-prt=161
_RAS_sv_0      serv    1 D   1. proto=6 dst-prt=80
_SRAS_sv_0     serv    1 D   1. proto=6 dst-prt=443
_ICMP_LISTEN_sv_0 serv    1 D   1. proto=1
_SENDDTO_LISTEN_sv_0 serv    1 D   1. proto=17
_PING_RESPONDER_sv_0 serv    1 D   1. proto=1 icmp-type=8
_HTTPI_sv_0    serv    1 D   1. proto=6 dst-prt=8080
_icmp          serv    1     1. proto=1
_igmp          serv    2     1. proto=2
_ftp            serv    0     1. proto=6 dst-prt=21
_telnet         serv    1     1. proto=6 dst-prt=23
_http           serv    1     1. proto=6 dst-prt=80
_httpproxy      serv    1     1. proto=6 dst-prt=8080
_https          serv    1     1. proto=6 dst-prt=443
_RPC            serv    0     1. proto=6 dst-prt=135
_NBT            serv    0     1. proto=17 dst-prt=137
                           2. proto=17 dst-prt=138
                           3. proto=6 dst-prt=139
_SMB            serv    0     1. proto=6 dst-prt=445
_imap           serv    1     1. proto=6 dst-prt=143
_imap3          serv    1     1. proto=6 dst-prt=220
_imap4-ssl      serv    1     1. proto=6 dst-prt=585
_imaps          serv    1     1. proto=6 dst-prt=993
_pop2           serv    1     1. proto=6 dst-prt=109
_pop3           serv    1     1. proto=6 dst-prt=110
_pop3s          serv    1     1. proto=6 dst-prt=995
_smtp           serv    1     1. proto=6 dst-prt=25
_ssh            serv    0     1. proto=6 dst-prt=22
_dns            serv    1     1. proto=6 dst-prt=53
                           2. proto=17 dst-prt=53
_nntp           serv    0     1. proto=6 dst-prt=119
_ipsec          serv    0     1. proto=51
                           2. proto=50
                           3. proto=17 src-prt=500 dst-prt=500
_esp            serv    1     1. proto=50
_ah             serv    1     1. proto=51
_ike            serv    1     1. proto=17 dst-prt=500
_DiffServ       serv    0     1. dscp!=0
_sip            serv    1     1. proto=17 dst-prt=5060
                           2. proto=6 dst-prt=5060
_h323           serv    1     1. proto=6 dst-prt=1720
                           2. proto=17 dst-prt=1718
                           3. proto=17 dst-prt=1719
_dhcp           serv    1     1. proto=17 dst-prt=68
                           2. proto=17 dst-prt=67
                           1. proto=17 dst-prt=554
                           2. proto=6 dst-prt=554
_ssdp_serv      serv    1     1. proto=17 dst-prt=1900
_mdap_serv      serv    1     1. proto=17 dst-prt=3235
=>
```

EXAMPLE 2:

```
=>expr list format=cli
:expr add name=_Internet type=intf intf=Internet
:expr add name=_LocalNetwork type=intf intf=LocalNetwork
:expr add name=HTTP1_if_0 type=intf intf=LocalNetwork
:expr add name=HTTP1_if_0 type=intf intf=LocalNetwork
:expr add name=HTTP1_if_0 type=intf intf=LocalNetwork
:expr add name=HTTP_if_0 type=intf intfgroup=lan
:expr add name=HTTPs_if_0 type=intf intfgroup=lan
:expr add name=FTP_if_0 type=intf intfgroup=lan
:expr add name=TELNET_if_0 type=intf intfgroup=lan
:expr add name=DNS-S_if_0 type=intf intfgroup=lan
:expr add name=SNMP_AGENT_if_0 type=intf intfgroup=lan
:expr add name=PING_RESPONDER_if_0 type=intf intfgroup=lan
:expr add name=wan type=intf intfgroup=wan
:expr add name=lan type=intf intfgroup=lan
:expr add name=local type=intf intfgroup=local
:expr add name=_10.0.0.138 type=ip addr=10.0.0.138
:expr add name=_192.168.1.254 type=ip addr=192.168.1.254
:expr add name=private type=ip addr=10.0.0.0/8
:expr add name=private type=ip addr=172.[16-31].*.*
:expr add name=private type=ip addr=192.168.1.0/24
:expr add name=ssdp_ip type=ip addr=239.255.255.250
:expr add name=mdap_ip type=ip addr=224.0.0.103
:expr add name=HTTP_sv_0 type=serv proto=tcp dstport=www-http
:expr add name=HTTPs_sv_0 type=serv proto=tcp dstport=443
:expr add name=FTP_sv_0 type=serv proto=tcp dstport=ftp
:expr add name=TELNET_sv_0 type=serv proto=tcp dstport=telnet
:expr add name=RIP_sv_0 type=serv proto=udp srcport=rip dstport=rip
:expr add name=RIP-Query_sv_0 type=serv proto=udp dstport=rip
:expr add name=DNS-S_sv_0 type=serv proto=udp dstport=dns
:expr add name=SNMP_AGENT_sv_0 type=serv proto=udp dstport=snmp
:expr add name=RAS_sv_0 type=serv proto=tcp dstport=www-http
:expr add name=SRAS_sv_0 type=serv proto=tcp dstport=443
:expr add name=ICMP_LISTEN_sv_0 type=serv proto=icmp
:expr add name=SENDTO_LISTEN_sv_0 type=serv proto=udp
:expr add name=PING_RESPONDER_sv_0 type=serv proto=icmp icmptype=echo-request
:expr add name=HTTP1_sv_0 type=serv proto=tcp dstport=http proxy
:expr add name=icmp type=serv proto=icmp
:expr add name=igmp type=serv proto=igmp
:expr add name=ftp type=serv proto=tcp dstport=ftp
:expr add name=telnet type=serv proto=tcp dstport=telnet
:expr add name=http type=serv proto=tcp dstport=www-http
:expr add name=https type=serv proto=tcp dstport=443
:expr add name=RPC type=serv proto=tcp dstport=135
:expr add name=NBT type=serv proto=udp dstport=netbios-ns
:expr add name=NBT type=serv proto=udp dstport=netbios-dgm
:expr add name=NBT type=serv proto=tcp dstport=netbios-ssn
:expr add name=SMB type=serv proto=tcp dstport=445
:expr add name=imap type=serv proto=tcp dstport=imap2
:expr add name=imap3 type=serv proto=tcp dstport=imap3
:expr add name=imap4-ssl type=serv proto=tcp dstport=585
:expr add name=imaps type=serv proto=tcp dstport=993
:expr add name=pop2 type=serv proto=tcp dstport=pop2
:expr add name=pop3 type=serv proto=tcp dstport=pop3
:expr add name=pop3s type=serv proto=tcp dstport=995
:expr add name=smtip type=serv proto=tcp dstport=smtip
:expr add name=ssh type=serv proto=tcp dstport=22
:expr add name=dns type=serv proto=tcp dstport=dns
:expr add name=dns type=serv proto=udp dstport=dns
:expr add name=nntp type=serv proto=tcp dstport=nntp
:expr add name=ipsec type=serv proto=ah
:expr add name=ipsec type=serv proto=esp
:expr add name=ipsec type=serv proto=udp srcport=ike dstport=ike
:expr add name=esp type=serv proto=esp
:expr add name=ah type=serv proto=ah
:expr add name=ike type=serv proto=udp dstport=ike
:expr add name=DiffServ type=serv dscp!=cs0
:expr add name=sip type=serv proto=udp dstport=sip
:expr add name=sip type=serv proto=tcp dstport=sip
:expr add name=h323 type=serv proto=tcp dstport=h323
:expr add name=h323 type=serv proto=udp dstport=1718
:expr add name=h323 type=serv proto=udp dstport=1719
:expr add name=dhcp type=serv proto=udp dstport=bootpc
:expr add name=dhcp type=serv proto=udp dstport=bootps
:expr add name=rtsp type=serv proto=udp dstport=rtsp
:expr add name=rtsp type=serv proto=tcp dstport=rtsp
:expr add name=ssdp_serv type=serv proto=tcp dstport=1900
:expr add name=mdap_serv type=serv proto=udp dstport=3235=>
```

RELATED COMMANDS:

expr add Add a subexpression to an expression.
expr delete Delete an expression.

Expr Commands

expr modify

Modify an expression.

SYN TAX:

```
expr modify name = <string>
              type = <{intf | ip | serv}>
              index = <number>
              addr [!] = <ip-range>
              [intf [!] = <string>]
              [intfgroup [!] = <{wan|local|lan|tunnel|dmz|guest} or number>]
              [bridgeport [!] = <number>]
              [tos [!] = <number{0-255}>]
              [precedence [!] = <IP precedence type> or <number>]
              [dscp [!] = <DSCP name> or <number>]
              [proto = <supported IP protocol name> or <number>]
              [srcport [!] = <supported TCP/UDP port name> or <number>]
              [srcportend = <supported TCP/UDP port name> or <number>]
              [dstport [!] = <supported TCP/UDP port name> or <number>]
              [dstportend = <supported TCP/UDP port name> or <number>]
              [icmptype [!] = <supported ICMP type name> or <number>]
              [icmpcode [!] = <number{0-15}>]
              [icmpcodeend = <number{0-15}>]
```



If a value is preceded by a "!", it means NOT.

For example "intfgroup!=wan" means "if the interface group is different from WAN".

where:

name	The name of the expression to be modified.	REQUIRED
type	The expression type. Choose between: <ul style="list-style-type: none">▶ intf▶ ip▶ serv.	REQUIRED
number	The index of the subexpression to be modified.	REQUIRED
	Tip Use the command :expr list to obtain the indexes of the subexpressions.	
addr	The IP address (or range). Supports ip/mask notation.	REQUIRED
	Note Only for expression type <i>ip</i> .	
intf	The IP interface name.	OPTIONAL
	Note Only for expression type <i>intf</i> .	

intfgroup	The IP interface group. Choose between: ▶ wan ▶ local ▶ lan ▶ tunnel ▶ dmz ▶ guest.	OPTIONAL
	Note Only for expression type <i>intf</i> .	
bridgeport	The bridge port number.	OPTIONAL
	Note Only for expression type <i>intf</i> .	
tos	A number between 0 and 255. Represents the ToS specification in the IP packet.	OPTIONAL
	Note The parameters <i>tos</i> , <i>precedence</i> and <i>dscp</i> are mutually exclusive.	
precedence	The precedence in the IP packet (part of tos). Select an IP precedence (see "IP Precedence" on page 754) or, alternatively, specify the number.	OPTIONAL
	Note The parameters <i>tos</i> , <i>precedence</i> and <i>dscp</i> are mutually exclusive.	
dscp	The DSCP in the IP packet (part of tos). Select a DSCP (see "Differentiated Services Code Point (DSCP)" on page 755).	OPTIONAL
	Note The parameters <i>tos</i> , <i>precedence</i> and <i>dscp</i> are mutually exclusive.	
proto	The protocol (name or number) expected in the IP packet. Select one of the following protocols: <i>icmp</i> , <i>igmp</i> , <i>ipinip</i> , <i>tcp</i> , <i>udp</i> , <i>ah</i> , <i>esp</i> , <i>ipcomp</i> or, 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 749) or, alternatively, specify the protocol number.	OPTIONAL
srcportend	The source TCP/UDP port range end (inclusive). Select one of the supported TCP/UDP port names (see "Supported TCP/UDP Port Names" on page 749) or, 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 749) or, alternatively, specify the protocol number.	OPTIONAL
dstportend	The destination TCP/UDP port range end (inclusive). Select one of the supported TCP/UDP port names (see "Supported TCP/UDP Port Names" on page 749) or, alternatively, specify the protocol number.	OPTIONAL
icmptype	The ICMP type (name or number) of the packet. Select one of the supported ICMP types (see "Supported ICMP Type Names" on page 752) or, alternatively, specify the type number.	OPTIONAL
icmpcode	A number between 0 and 15. Represents the ICMP code (or beginning of range) of the packet.	OPTIONAL
icmpcodeend	A number between 0 and 15. Represents the ICMP code range end (inclusive).	OPTIONAL

Firewall Commands

Introduction

This chapter describes the commands of the **firewall** command group.

Contents

This chapter covers the following commands:

firewall config	Configure the firewall options.	306
firewall clear	Flush the firewall configuration.	308
firewall list	List the firewall configuration.	309
firewall chain add	Add a chain.	310
firewall chain delete	Delete a chain.	311
firewall chain flush	Flush all chains.	312
firewall chain list	List all chains.	313
firewall debug clear	Clear the firewall statistics.	314
firewall debug stats	Display the firewall statistics.	315
firewall debug traceconfig	Configure the firewall trace options.	316
firewall level add	Add a security level.	317
firewall level delete	Delete a security level.	318
firewall level flush	Flush the security level configuration.	319
firewall level list	List all the security levels.	320
firewall level modify	Configure a security level.	321
firewall level set	Set the security level or display the current security level.	322
firewall rule add	Add a firewall rule.	323
firewall rule delete	Delete a firewall rule.	324
firewall rule flush	Flush all firewall rules.	325
firewall rule list	Show a list of the firewall rules in a chain.	326
firewall rule modify	Modify a firewall rule.	328
firewall rule debug clear	Clear the firewall rule statistics.	329
firewall rule debug stats	Show the firewall rule statistics.	331
firewall rule debug traceconfig	Display or modify the rule trace configuration.	332

firewall config

Configure the firewall options.

SYNTAX:

```
firewall config [state = <{disabled | enabled}>]
                [keep = <{disabled | enabled}>]
                [tcpchecks = <{none | fast | exact}>]
                [udpchecks = <{disabled|enabled}>]
                [icmpchecks = <{disabled | enabled}>]
                [logdefault = <{disabled | enabled}>]
                [logthreshold = <{disabled | enabled}>]
                [tcpwindow = <number{0-1073725440}>]
```

where:

state	Enable or disable the firewall. The default is enabled .	OPTIONAL
keep	The firewall keeps active connections (enabled) or not (disabled) when the firewall rules change. The default is disabled .	OPTIONAL
tcpchecks	Select the level of TCP sequence number checks. Choose between: <ul style="list-style-type: none"> ▶ none: no TCP checks are done. ▶ fast: check all the combinations of flag and disallow all the possible illegal combinations shown below: <ul style="list-style-type: none"> ▶ SYN PSH (SYN PSH URG,...) ▶ SYN FIN (SYN FIN PSH, SYN FIN RST PSH,...) ▶ FIN flag set without ACK ▶ All flags set ▶ No flags set. ▶ exact: check and permit only combinations of flag with the TCP state of a connection: <ul style="list-style-type: none"> ▶ SYN: request to open connection ▶ SYN ACK: agree to open connection ▶ A, PA, AU, PAU: acknowledgement of receipt ▶ FA, FAP, FAU, FAP, FAPU, FAU, FPAU: request to close connection ▶ R, RA, RP, RU, RPA, RPU, RAU, RPAU: tear down connection. 	OPTIONAL
	The default is none .	
udpchecks	Disable or enable keeping UDP checks. The default is enabled .	OPTIONAL
icmpchecks	Disable or enable keeping ICMP checks. The default is enabled .	OPTIONAL
logdefault	Disable or enable logging of default firewall rule. The default is disabled .	OPTIONAL
logthreshold	Disable or enable log thresholding. The default is enabled .	OPTIONAL

tcpwindow	A number between 0 and 1073725440. This parameter permits to modify the TCP window for fast TCP checks. The default is 65536 .	OPTIONAL
-----------	---	----------

EXAMPLE:

```
=>firewall config  
:firewall config state=enabled keep=disabled tcpchecks=none udpchecks=enabled icmpchecks=enabled  
logdefault=disabled logthreshold=enabled tcpwindow=65536  
=>
```

firewall clear

Flush the firewall configuration.

SYNTAX:

```
firewall clear
```

firewall list

List the firewall configuration.

SYNTAX:

```
firewall list [format = <{pretty | cli}>]
```

where:

format	Select the output format in which the configuration must be shown. Choose between: <ul style="list-style-type: none">▶ pretty: the configuration is shown as intuitive output in clear text.▶ cli: the configuration is shown via the CLI commands configuration. The default is pretty .	OPTIONAL
--------	--	----------

EXAMPLE of output in text mode:

```
=>firewall list

Config
=====
State      : enabled
Keep       : disabled
TcpChecks  : exact
TcpWindow  : 65536
IcmpChecks: enabled
LogDefault : disabled
LogThreshold: enabled

Modules
=====
Module     State   Text                                Hooks
-----
fire       enabled Firewall Administration Module    sink, forward, source
host_service enabled Firewall Host Service Module   forward
level      enabled Firewall Level Module             forward
system_service enabled Firewall System Service Module sink
=>
```

EXAMPLE of output in CLI mode:

```
=>firewall list format=cli
:firewall config state=enabled keep=disabled tcpchecks=exact icmpchecks=enabled logdefault=disabled
| logthreshold=enabled tcpwindow=65536
:firewall debug traceconfig tcpchecks=disabled icmpchecks=disabled sink=none forward=none
| source=none
=>
```

Firewall Commands

firewall chain add

Add a chain.

SYNTAX:

```
firewall chain add chain = <string>
```

where:

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

EXAMPLE:

```
=>firewall chain list

Chains
=====
Name                               Description
-----
sink                                system
forward                             system
source                             system
sink_fire                           system
forward_fire                        system
source_fire                         system
forward_host_service                system
forward_level                       system
sink_system_service                 system
forward_level_BlockAll              system
forward_level_Standard              system
forward_level_Disabled              system
=>firewall chain add chain=myCHAIN
=>firewall chain list

Chains
=====
Name                               Description
-----
sink                                system
forward                             system
source                             system
sink_fire                           system
forward_fire                        system
source_fire                         system
forward_host_service                system
forward_level                       system
sink_system_service                 system
forward_level_BlockAll              system
forward_level_Standard              system
forward_level_Disabled              system
myCHAIN                            user
=>
```

RELATED COMMANDS:

firewall chain delete	Delete a chain.
firewall chain list	List all chains.

firewall chain delete

Delete a chain.

SYNTAX:

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

where:

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

EXAMPLE:

```
=>firewall chain list

Chains
=====
Name          Description
-----
sink          system
forward       system
source        system
sink_fire     system
forward_fire  system
source_fire   system
forward_host_service  system
forward_level  system
sink_system_service  system
forward_level_BlockAll  system
forward_level_Standard  system
forward_level_Disabled  system
myCHAIN       user

=>firewall chain delete chain=myCHAIN
=>firewall chain list

Chains
=====
Name          Description
-----
sink          system
forward       system
source        system
sink_fire     system
forward_fire  system
source_fire   system
forward_host_service  system
forward_level  system
sink_system_service  system
forward_level_BlockAll  system
forward_level_Standard  system
forward_level_Disabled  system
=>
```

RELATED COMMANDS:

firewall chain add	Add a chain.
firewall chain list	List all chains.

firewall chain flush

Flush all chains.

SYNTAX:

```
firewall chain flush
```

firewall chain list

List all chains.

SYNTAX:

```
firewall chain list      [format = <{pretty | cli}>]
```

where:

format	Select the output format in which the chains must be shown. Choose between: <ul style="list-style-type: none">▶ pretty: the chains are shown as intuitive output in clear text.▶ cli: the chains are shown via the CLI commands configuration. The default is pretty .	OPTIONAL
--------	---	----------

EXAMPLE:

```
=>firewall chain list

Chains
=====
Name                      Description
-----
sink                      system
forward                   system
source                    system
sink_fire                 system
forward_fire              system
source_fire               system
forward_host_service     system
forward_level             system
sink_system_service       system
forward_level_BlockAll   system
forward_level_Standard   system
forward_level_Disabled   system
=>
```

RELATED COMMANDS:

firewall chain add	Add a chain.
firewall chain delete	Delete a chain.

Firewall Commands

firewall debug clear

Clear the firewall statistics.

SYNTAX:

```
firewall debug clear
```

EXAMPLE:

```
=>firewall debug stats
Statistics
=====
Used rule contexts      : 0
Total rule contexts     : 256
Total packets parsed    : 2554
Packets parsed in hook sink   : 1461
Packets parsed in hook forward : 12
Packets parsed in hook source  : 1041
Packets dropped in hook sink  : 0
Packets dropped in hook forward : 0
Packets dropped in hook source  : 0
TCP flag errors detected   : 14
TCP seq/ack/win errors detected : 5
ICMP errors with partial info : 0
ICMP errors without cause    : 0
ICMP replies without request : 0
Packet replay errors       : 0
=>firewall debug clear
=>firewall debug stats

Statistics
=====
Used rule contexts      : 0
Total rule contexts     : 256
Total packets parsed    : 26
Packets parsed in hook sink   : 16
Packets parsed in hook forward : 0
Packets parsed in hook source  : 10
Packets dropped in hook sink  : 0
Packets dropped in hook forward : 0
Packets dropped in hook source  : 0
TCP flag errors detected   : 0
TCP seq/ack/win errors detected : 0
ICMP errors with partial info : 0
ICMP errors without cause    : 0
ICMP replies without request : 0
Packet replay errors       : 0
=>
```

RELATED COMMANDS:

firewall debug stats Display the firewall statistics.

firewall debug stats

Display the firewall statistics.

SYNTAX:

```
firewall debug stats
```

EXAMPLE:

```
=>firewall debug stats

Statistics
=====
Used rule contexts      : 0
Total rule contexts     : 256
Total packets parsed    : 2554
Packets parsed in hook sink : 1461
Packets parsed in hook forward : 12
Packets parsed in hook source  : 1041
Packets dropped in hook sink   : 0
Packets dropped in hook forward : 0
Packets dropped in hook source   : 0
TCP flag errors detected    : 14
TCP seq/ack/win errors detected : 5
ICMP errors with partial info  : 0
ICMP errors without cause     : 0
ICMP replies without request  : 0
Packet replay errors        : 0

=>
```

RELATED COMMANDS:

firewall debug clear Clear the firewall statistics.

firewall debug traceconfig

Configure the firewall trace options.

SYNTAX:

```
firewall debug traceconfig [tcpchecks = <{disabled | enabled}>]
                           [udpchecks = <{disabled|enabled}>]
                           [icmpchecks = <{disabled | enabled}>]
                           [sink = <{none | all | accept | deny | drop
                           | reset} or number>]
                           [forward = <{none | all | accept | deny | drop
                           reset} or number>]
                           [source = <{none | all | accept | deny | drop
                           | reset} or number>]
```

where:

tcpchecks	Disable or enable tcpchecks traces. The default is disabled .	OPTIONAL
udpchecks	Disable or enable udpchecks traces. The default is disabled .	OPTIONAL
icmpchecks	Disable or enable icmpchecks traces. The default is disabled .	OPTIONAL
sink	Specify the action traced by the firewall for sink traffic. The default is none .	OPTIONAL
forward	Specify the action traced by the firewall for forward traffic. The default is none .	OPTIONAL
source	Specify the action traced by the firewall for source traffic. The default is none .	OPTIONAL

EXAMPLE:

```
=>firewall debug traceconfig

Trace Config
=====
tcpchecks      : disabled
icmpchecks    : disabled
sink          : none
forward        : none
source         : none
=>
```

firewall level add

Add a security level.

SYNTAX:

```
firewall level add name = <string>
    [index = <number>]
    [readonly = <{disabled | enabled}>]
    [udptrackmode = <{strict | loose}>]
    [service = <{disabled | enabled}>]
    [proxy = <{disabled|enabled}>]
    [text = <quoted string>]
```

where:

name	The name of the security level to be added.	REQUIRED
index	The index of the security level.	OPTIONAL
Tip	Use the command :firewall level list to obtain a list of indexes.	
Note	If not specified, the new security level will be added at the bottom of the list.	
readonly	Select whether the security level is readonly (enabled) or not (disabled). The default is enabled .	OPTIONAL
udptrackmode	Select the UDP connection tracking mode. Choose between: <ul style="list-style-type: none">▶ strict: replies to a request from a client must be in a specific window to the client.▶ loose: inbound packets are allowed on the port that was first used to start the communication with the server (for example to allow a client of an online game to obtain peer-to-peer information from other clients of that same online game). The default is strict .	OPTIONAL
proxy	Enable or disable proxy system services for this security level.	OPTIONAL
service	Disable or enable host service definitions. The default is disabled .	OPTIONAL
text	Description of the security level. Note The maximum length is 39 characters.	OPTIONAL

RELATED COMMANDS:

firewall level delete	Delete a security level.
firewall level list	List all the security levels.

firewall level delete

Delete a security level.

SYNTAX:

```
firewall level delete name = <string>
```

where:

name	The name of the security level to be deleted.	REQUIRED
------	---	----------

RELATED COMMANDS:

firewall level add	Add a security level.
firewall level list	List all the security levels.

firewall level flush

Flush the security level configuration.

SYNTAX:

```
firewall level flush
```

firewall level list

List all the security levels.

SYNTAX:

```
firewall level list [format = <{pretty | cli}>]
```

where:

format	Select the output format in which the security levels must be shown. Choose between: <ul style="list-style-type: none">▶ pretty: the security levels are shown as intuitive output in clear text.▶ cli: the security levels are shown via the CLI commands configuration. The default is pretty .	OPTIONAL
--------	--	----------

RELATED COMMANDS:

firewall level add	Add a security level.
firewall level delete	Delete a security level.

firewall level modify

Configure a security level.

SYNTAX:

```
firewall level modify name = <string>
    [index = <number>]
    [readonly = <{disabled | enabled}>]
    [udptrackmode = <{strict | loose}>]
    [service = <{disabled | enabled}>]
    [proxy = <{disabled|enabled}>]
    [text = <quoted string>]
```

where:

name	The name of the security level to be added.	REQUIRED
index	The index of the security level.	OPTIONAL
Tip	Use the command :firewall level list to obtain a list of indexes.	
readonly	Select whether the security level is readonly (enabled) or not (disabled).	OPTIONAL
udptrackmode	Select the UDP connection tracking mode. Choose between: <ul style="list-style-type: none">▶ strict: replies to a request from a client must be in a specific window to the client.▶ loose: inbound packets are allowed on the port that was first used to start the communication with the server (for example to allow a client of an online game to obtain peer-to-peer information from other clients of that same online game). The default is strict .	OPTIONAL
proxy	Enable or disable proxy system services for this security level.	OPTIONAL
service	Disable or enable service definitions for this security level.	OPTIONAL
text	Description of the security level.	OPTIONAL
Note	The maximum length is 39 characters.	

firewall level set

Set the security level or display the current security level.

SYNTAX:

```
firewall level set [name = <string>]
```

where:

name	The name of the security level to be set.	OPTIONAL
------	---	----------

Note If no security level is specified, the current security level will be shown.

firewall rule add

Add a firewall rule.

SYNTAX:

```
firewall rule add chain = <chain name>
    [index = <number>]
    [name = <string>]
    [clink = <chain name>]
    [srcintf [!] = <string>]
    [srcip [!] = <string>]
    [dstip [!] = <string>]
    [serv [!] = <string>]
    [log = <{disabled | enabled}>]
    [state = <{disabled | enabled}>]
    action = <{accept|deny|drop|reset|count|link}>
```



If a value is preceded by a "!", it means NOT.

For example "srcintf!=wan" means "if the source interface is different from WAN".

where:

chain	The name of the chain which contains the rule.	REQUIRED
index	The number of the rule in the chain.	OPTIONAL
name	The name of the new rule.	OPTIONAL
clink	The name of the chain to be parsed when this rule applies.	OPTIONAL
srcintf	The name of the source interface expression.	OPTIONAL
srcip	The name of the source IP expression.	OPTIONAL
dstintf	The name of the destination interface expression.	OPTIONAL
dstip	The name of the destination IP expression.	OPTIONAL
serv	The name of the service expression.	OPTIONAL
log	Disable or enable logging is done when this rule applies.	OPTIONAL
state	Enable or disable this rule.	OPTIONAL
action	The action to be taken when this rule applies ('link' when clink is used).	REQUIRED

RELATED COMMANDS:

firewall rule delete	Delete a firewall rule.
expr list	List the expressions.

firewall rule delete

Delete a firewall rule.

SYNTAX:

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

where:

chain	The name of the chain in which a rule must be deleted.	REQUIRED
index	The number of the rule in the chain.	REQUIRED

RELATED COMMANDS:

firewall rule add Add a firewall rule.

firewall rule flush

Flush all firewall rules.

SYNTAX:

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

where:

chain	The name of the chain for which the rules must be flushed.	OPTIONAL
Note	If not specified, the rules of all the chains will be flushed.	

firewall rule list

Show a list of the firewall rules in a chain.

SYNTAX:

```
firewall rule list [chain = <string>]  
[format = <{pretty | cli}>]
```

where:

chain	The name of the chain for which the rules must be listed.	OPTIONAL
	Note If not specified, the rules of all the chains are shown.	
format	Select the output format in which the list must be shown. Choose between: <ul style="list-style-type: none">▶ pretty: the list is shown as intuitive output in clear text.▶ cli: the list is shown via the CLI commands configuration. The default is pretty .	OPTIONAL

EXAMPLE:

```
=>firewall rule list

Rules (flags: C=Constant, D=Dynamic, E=Enable, L=Log)
=====
Chain          Nr.  Flags   Rule
-----
sink           1    CDE      : link   sink_fire
                2    CDE      : link   sink_system_service
sink_fire      1    C E     SSDP      : accept  ssdp_serv lan.* > *.ssdp_ip
sink_system_service 1    CDE     HTTPPI    : accept  HTTPPI_sv_0 HTTPPI_if_0.* > *.*
                                         2    DE      PING_RESPONDER : accept  PING_RESPONDER_sv_0 PING_RESPONDER_
                                         if_0.* > *.*
                                         3    CD      SENDTO_LISTEN : accept  SENDTO_LISTEN_sv_0 *.* > *.*
                                         4    D       ICMP_LISTEN  : accept  ICMP_LISTEN_sv_0 *.* > *.*
                                         5    CD      SRAS        : accept  SRAS_sv_0 *.* > *.*
                                         6    CD      RAS         : accept  RAS_sv_0 *.* > *.*
                                         7    CDE     MDAP        : accept  MDAP_sv_0 *.* > *.*
                                         8    CDE     SNMP_AGENT  : accept  SNMP_AGENT_sv_0 SNMP_AGENT_
                                         if_0.* > *.*
                                         9    CD      DHCP-S     : accept  DHCP-S_sv_0 *.* > *.*
                                         10   CDE    DHCP-R     : accept  DHCP-R_sv_0 DHCP-R_if_0.* > *.*
                                         11   CDE    DNS-S      : accept  DNS-S_sv_0 DNS-S_if_0.* > *.*
                                         12   CD      RIP-Query   : accept  RIP-Query_sv_0 *.* > *.*
                                         13   CDE     RIP         : accept  RIP_sv_0 *.* > *.*
                                         14   CDE     TELNET      : accept  TELNET_sv_0 TELNET_if_0.* > *.*
                                         15   CDE     FTP         : accept  FTP_sv_0 FTP_if_0.* > *.*
                                         16   CDE     HTTPS       : accept  HTTPS_sv_0 HTTPS_if_0.* > *.*
                                         17   CDE     HTTP        : accept  HTTP_sv_0 HTTP_if_0.* > *.*

forward        1    CDE      : link   forward_fire
                2    CDE      : link   forward_host_service
                3    CDE      : link   forward_level
forward_level  1    CDE      : link   forward_level_Disabled
forward_level_Disabled 1    C E     AnyTraffic : accept  *.* > *.*
source         1    CDE      : link   source_fire
source_fire    1    C E     AnyTraffic : accept  *.* > *.*
```

=>

```
=>firewall rule list format=cli
:firewall rule add chain=sink_fire index=1 name=SSDP srcintf=lan dstip=ssdp_ip serv=ssdp_serv
| log=disabled state=enabled action=accept
:firewall rule add chain=source_fire index=1 name=AnyTraffic log=disabled state=enabled
| action=accept
:firewall rule add chain=forward_level_BlockAll index=1 name=AnyTraffic log=disabled state=enabled
| action=drop
:firewall rule add chain=forward_level_Standard index=1 name=FromLAN srcintf=lan log=disabled
| state=enabled action=accept
:firewall rule add chain=forward_level_Disabled index=1 name=AnyTraffic log=disabled state=enabled
| action=accept
=>
```

firewall rule modify

Modify a firewall rule.

SYNTAX:

```
firewall rule modify chain = <string>
                  index = <number>
                  [newindex = <number>]
                  [name = <string>]
                  [clink = <chain name>]
                  [srcintf [!] = <string>]
                  [srcip [!] = <string>]
                  [dstintf [!] = <string>]
                  [dstip [!] = <string>]
                  [serv [!] = <string>]
                  [log = <{disabled | enabled}>]
                  [state = <{disabled | enabled}>]
                  [action = <action>]
```



If a value is preceded by a "!", it means NOT.

For example "srcintf!=wan" means "if the source interface is different from WAN".

where:

chain	The name of the chain which contains the rule.	REQUIRED
index	The number of the rule in the chain.	REQUIRED
newindex	The number of the rule in the chain.	OPTIONAL
name	The name of the new rule.	OPTIONAL
clink	The name of the chain to be parsed when this rule applies.	OPTIONAL
srcintf	The name of the source interface expression.	OPTIONAL
srcip	The name of the source IP expression.	OPTIONAL
dstintf	The name of the destination interface expression.	OPTIONAL
dstip	The name of the destination IP expression.	OPTIONAL
serv	The name of the service expression.	OPTIONAL
log	Disable or enable logging when this rule applies.	OPTIONAL
state	Disable or enable this rule.	OPTIONAL
action	The action to be taken when this rule applies ('link' when clink is used).	OPTIONAL

RELATED COMMANDS:

firewall rule delete	Delete a firewall rule.
expr list	List the expressions.

firewall rule debug clear

Clear the firewall rule statistics.

SYNTAX:

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

where:

chain	The name of the chain which contains the rule.	OPTIONAL
index	The number of the rule in the chain.	OPTIONAL
Note If not specified, the statistics for all the rules in the chain will be cleared.		

EXAMPLE:

```
=>firewall rule debug stats  
chain                                index   packets   bytes  
-----  
sink                               1       0       0  
                                 2       4      192  
forward                            1       0       0  
                                 2       0       0  
                                 3      10      480  
source                             1       0       0  
sink_fire                          1       0       0  
source_fire                        1       0       0  
forward_level                      1      10      480  
sink_system_service                1       1       48  
                                 2       0       0  
...  
                                 16      0       0  
                                 17      2      96  
forward_level_BlockAll            1       0       0  
forward_level_Standard           1       0       0  
forward_level_Disabled           1      10      480  
=>firewall rule debug clear  
=>firewall rule debug stats  
chain                                index   packets   bytes  
-----  
sink                               1       0       0  
                                 2       0       0  
forward                            1       0       0  
                                 2       0       0  
                                 3       0       0  
source                             1       0       0  
sink_fire                          1       0       0  
source_fire                        1       0       0  
forward_level                      1       0       0  
sink_system_service                1       0       0  
                                 2       0       0  
...  
                                 16      0       0  
                                 17      0       0  
forward_level_BlockAll            1       0       0  
forward_level_Standard           1       0       0  
forward_level_Disabled           1       0       0  
=>
```

Firewall Commands

RELATED COMMANDS:

firewall rule debug stats Show the firewall rule statistics.

firewall rule debug stats

Show the firewall rule statistics.

SYNTAX:

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

where:

chain The name of the chain for which the statistics must be shown. OPTIONAL

Note If not specified, the statistics for all the chains will be shown.

index The number of the rule in the chain. OPTIONAL

Note If not specified, the statistics for all the rules in the chain are shown.

EXAMPLE:

```
=>firewall rule debug stats  
chain  
-----  
sink  
-----  
1 0 0  
2 4 192  
forward  
-----  
1 0 0  
2 0 0  
3 10 480  
source  
1 0 0  
sink_fire  
1 0 0  
source_fire  
1 0 0  
forward_level  
1 10 480  
sink_system_service  
1 1 48  
2 0 0  
3 0 0  
4 0 0  
5 0 0  
6 0 0  
7 0 0  
8 0 0  
9 0 0  
10 0 0  
11 0 0  
12 0 0  
13 0 0  
14 1 48  
15 0 0  
16 0 0  
17 2 96  
forward_level_BlockAll  
1 0 0  
forward_level_Standard  
1 0 0  
forward_level_Disabled  
1 10 480  
=>
```

RELATED COMMANDS:

firewall rule debug clear Clear the firewall rule statistics.

firewall rule debug traceconfig

Display or modify the rule trace configuration.

SYNTAX:

```
firewall rule debug traceconfig [trace = <{disabled | enabled}>]
```

where:

trace	Enable or disable rule traces. The default is disabled .	OPTIONAL
-------	--	----------

EXAMPLE:

```
=>firewall rule debug traceconfig  
:firewall rule debug traceconfig state=disabled  
=>
```

GRP Commands

Introduction

This chapter describes the commands of the **grp** command group.

Contents

This chapter covers the following commands:

grp config	Set the Generic Routing Protocol (GRP) configuration settings.	334
grp flush	Flush the GRP interface settings and parameters.	335
grp rtlist	Show the current routes in the GRP interfaces routing table.	336
grp rip config	Configure the RIP settings.	337
grp rip flush	Flush the RIP interface settings and global parameters.	339
grp rip ifconfig	Configure a RIP interface.	340
grp rip show	Show the RIP settings and the routes in the RIP database.	341

grp config

Set the Generic Routing Protocol (GRP) configuration settings.

SYNTAX:

```
grp config [cdistance = <number{0-255}>]
            [kdistance = <number{0-255}>]
            [rdistance = <number{0-255}>]
            [trace = <{disabled | enabled}>]
```

where:

cdistance	A number between 0 and 255. Sets the distance of the connected route type. The default is 0 .	OPTIONAL
kdistance	A number between 0 and 255. Sets the distance of the kernel route type. The default is 1 .	OPTIONAL
rdistance	A number between 0 and 255. Sets the distance of the RIP route type. The default is 120 .	OPTIONAL
trace	Enable or disable verbose console messaging. The default is disabled .	OPTIONAL

EXAMPLE:

```
=>grp config
Distance of the connected route type : 0
Distance of the kernel route type : 1
Distance of the RIP route type : 120
Tracing : disabled
=>
```

grp flush

Flush the GRP interface settings and parameters.

SYNTAX:

```
grp flush
```

grp rtlist

Show the current routes in the GRP interfaces routing table.

SYNTAX:

```
grp rtlist [dst = <ip-address>]
           [dstmask = <ip-mask (dotted or cidr)>]
```

where:

dst	The destination IP address of the route. Supports IP/mask notation.	OPTIONAL
Note	If no destination IP address is specified, all the current routes will be shown.	
dstmask	The destination IP address mask, either in dotted or in numerical cidr notation.	OPTIONAL

EXAMPLE:

```
=>grp rtlist
Codes : K - kernel, C - connected, S - static, R - rip, * - FIB route

destination : 10.0.0.0/24 *
Route type : "C"
distance : 0
Nexthop : LocalNetwork

destination : 10.0.0.0/24
Route type : "K"
distance : 1
Nexthop : 10.0.0.138

destination : 127.0.0.0/8 *
Route type : "C"
distance : 0
Nexthop : loop

destination : 192.168.1.0/24 *
Route type : "C"
distance : 0
Nexthop : LocalNetwork

destination : 192.168.1.0/24
Route type : "K"
distance : 1
Nexthop : 192.168.1.254

destination : 255.255.255.255/32 *
Route type : "K"
distance : 1
Nexthop : 127.0.0.1

=>
```

grp rip config

Configure the RIP settings.

SYNTAX:

```
grp rip config state = <{disabled | enabled}>
    [version = <{rip_unspec | rip_v1 | rip_v2}>]
    [defmetric = <number{1-16}>]
    [updatetime = <number{1-3600}>]
    [timeouttime = <number{1-3600}>]
    [garbagetime = <number{1-3600}>]
    [impqrst = <{disabled | enabled}>]
    [impkrt = <{disabled | enabled}>]
    [impsrt = <{disabled | enabled}>]
    [impdefkrt = <{disabled | enabled}>]
    [impdefsrt = <{disabled | enabled}>]
    [exprt = <{disabled | enabled}>]
    [expdefrt = <{disabled | enabled}>]
    [txdefrt = <{disabled | enabled}>]
    [trace = <{disabled | enabled}>]
```

where:

state	Enable or disable the RIP daemon. The default is enabled .	REQUIRED
version	Configure the RIP version to be applied. Choose between: <ul style="list-style-type: none"> ▶ rip_unspec: No RIP version is specified as such. The actual RIP version to be used is negotiated with the remote side. ▶ rip_v1: RIP version 1 is used. ▶ rip_v2: RIP version 2 is used. The default is rip_v2 .	OPTIONAL
defmetric	A number between 1 and 16 (hops). Represents the default RIP metric for imported routes. The default is 1 .	OPTIONAL
updatetime	A number between 1 and 3600 (seconds). Represents the update timer value of the RIP routing table. The default is 30 .	OPTIONAL
timeouttime	A number between 1 and 3600 (seconds). Represents the timeout timer value of the RIP routing info. The default is 180 .	OPTIONAL
garbagetime	A number between 1 and 3600 (seconds). Represents the garbage collection timer value. The default is 120 .	OPTIONAL
impqrst	Enable or disable the import of connected routes. The default is enabled .	OPTIONAL
impkrt	Enable or disable the import of kernel routes. The default is enabled .	OPTIONAL

impsrt	Enable or disable the import of static routes. The default is enabled .	OPTIONAL
impdefkrt	Enable or disable the import of the default kernel route. The default is enabled .	OPTIONAL
impdefsrt	Enable or disable the import of the default static route. The default is enabled .	OPTIONAL
exprt	Enable or disable the export of received RIP routes. The default is enabled .	OPTIONAL
expdefrt	Enable or disable the export of the received RIP default route. The default is enabled .	OPTIONAL
txrt	Enable or disable the transmission of the RIP default route. The default is enabled .	OPTIONAL
trace	Enable or disable verbose console messaging. The default is disabled .	OPTIONAL

EXAMPLE:

```
=>grp rip show
RIP routing protocol config dump
-----
    RIP daemon is enabled
    Global RIP queries received : 0
    Global RIP route changes : 0
    Default version : send rip_v2, receive rip_v2
    Default redistribution metric is 1
    Sending routing table updates every 30 seconds with +/-5%
    Route timeout after 180 seconds
    Route garbage collect after 120 seconds
    Import of connected routes is enabled
    Import of kernel routes is enabled
    Import of static routes is enabled
    Import of default kernel route is enabled
    Import of default static route is enabled
    Export of RIP routes is enabled
    Export of default RIP route is enabled
    Transmission of default RIP route is enabled

    Interface      Send          Recv          AuthMode      Passive      SplitHorizon  BadPackets
BadRoutes   SentUpdates
-----
-----  

0           lan1        rip_unspec    rip_unspec    none        enabled     enabled       0          0
0           Internet    rip_unspec    rip_unspec    none        enabled     enabled       0          0
0

    RIP routing table dump
-----
    Codes : K - Kernel, C - connected, S - Static, R - RIP, * - FIB route
    Network      Next Hop      Metric      From          Flags
-----
C 10.0.0.0/24          1          <>      *
C 192.168.1.0/24       1          <>      *
=>
```

RELATED COMMANDS:

grp rip show Show the RIP settings and the routes in the RIP database.

grp rip flush

Flush the RIP interface settings and global parameters.

SYNTAX:

```
grp rip flush
```

grp rip ifconfig

Configure a RIP interface.

SYNTAX:

```
grp rip ifconfig intf = <string>
    [rip = <{disabled | enabled}>]
    [rxversion = <{rip_unspec | rip_v1 | rip_v2 | rip_v1-2}>]
    [authmode = <{none | cleartext}>]
    [authstring = <quoted string>]
    [splithorizon = <{disable | enable}>]
```

where:

intf	The name of the RIP interface to be configured.	REQUIRED
Tip	Use the command :grp iflist to obtain a list of available interfaces.	
rip	Enabled or disabled RIP on this interface. The default is disabled .	OPTIONAL
rxversion	Configure the RIP receive version to be applied. Choose between: <ul style="list-style-type: none">▶ rip_unspec: No RIP version is specified as such. The actual RIP version to be used is negotiated with the remote side.▶ rip_v1: RIP version 1 is used.▶ rip_v2: RIP version 2 is used.▶ rip_v1-2: RIP version 1 and RIP version 2 are used. The default is rip_unspec .	OPTIONAL
authmode	Configure the RIP authentication mode. Enter the mode in cleartext or specify none in case no authentication mode is required. The default is none .	OPTIONAL
authstring	Configure the authentication string for the RIP authentication password.	OPTIONAL
Note	Leave authstring unspecified in case authmode is none .	
splithorizon	Enable or disable the split horizon status for this interface. The default is enabled .	OPTIONAL

EXAMPLE:

```
=>grp rip ifconfig
intf = myPPP_ppp
[rip] = disabled
[rxversion] = rip_unspec
[authmode] = none
[authstr] =
[passive] = disabled
[splithorizon] = enabled
:grp rip ifconfig intf=myPPP_ppp
=>
```

grp rip show

Show the RIP settings and the routes in the RIP database.

SYNTAX:

```
grp rip show
```

EXAMPLE:

```
=>grp rip show
RIP routing protocol config dump
-----
RIP daemon is enabled
Global RIP queries received : 0
Global RIP route changes : 3
Default version : send rip_v2, receive rip_v2
Default redistribution metric is 1
Sending routing table updates every 30 seconds with +/-5%
Route timeout after 180 seconds
Route garbage collect after 120 seconds
Import of connected routes is enabled
Import of kernel routes is enabled
Import of static routes is enabled
Import of default kernel route is enabled
Import of default static route is enabled
Export of RIP routes is enabled
Export of default RIP route is enabled
Transmission of default RIP route is enabled

      Intf     Send       Recv      AuthMode Passive   SplitHorizon BadPackets BadRoutes SentUpdates
-----
      eth0    rip_unspec rip_unspec none        off       on          0         58          62

RIP route table dump
-----
      Codes : K - Kernel, C - connected, S - Static, R - RIP, * - FIB route
Network      Next Hop      Metric   From           Flags
-----
      R 0.0.0.0/0      192.6.11.150    3        192.6.11.150  <> *
      C 10.0.0.0/8            1
      K 129.132.2.21/32    138.203.7.146  1
      C 138.203.4.0/22            1
      C 172.16.1.0/24            1
=>
```


Hostmgr Commands

Introduction

This chapter describes the commands of the **hostmgr** command group.

Contents

This chapter covers the following commands:

hostmgr add	Add host device info to the host manager.	344
hostmgr clear	Remove all the hosts from the list.	346
hostmgr config	Configure the host manager parameters.	347
hostmgr delete	Delete the host device info from the host manager.	348
hostmgr flush	Flush the host manager configuration.	349
hostmgr list	List all the host devices.	350

hostmgr add

Add host device info to the host manager.

SYNTAX:

```
hostmgr add mac_addr = <hardware-address>
          [ip_addr = <ip-address>]
          [name = <string>]
          [type = <{generic_device|desktop_computer|laptop_computer|
                     set_top_box|pda|gaming_console|phone|mobile_phone|
                     printer|mass_storage_device}>]
          [ipintf = <string>]
          [ethintf = <string>]
          [physintf = <string>]
```

where:

mac_addr	The MAC address of the host to be added.	REQUIRED
ip_addr	The host IP address.	OPTIONAL
name	The host name.	OPTIONAL
type	The host type. Choose between: <ul style="list-style-type: none"> ▶ generic_device ▶ desktop_computer ▶ laptop_computer ▶ set_top_box ▶ pda ▶ gaming_console ▶ phone ▶ mobile_phone ▶ printer ▶ mass_storage_device The default is Generic Device .	OPTIONAL
ipintf	The IP interface name.	OPTIONAL
ethintf	The Ethernet interface name.	OPTIONAL
physintf	The physical interface name.	OPTIONAL

EXAMPLE:

```
=>hostmgr list
MAC-address      IP-address      Flags   Type           Intf      Hw Intf      Hostname
-----          -----          ----   ----          ----      -----      -----
00:10:a4:ad:32:cf 192.168.1.64    C     Generic Device LocalNetwork  ethif4      MyComputer

=>hostmgr add mac_addr=00:10:a4:33:56:53 name=Play type=Playstation
=>hostmgr list
MAC-address      IP-address      Flags   Type           Intf      Hw Intf      Hostname
-----          -----          ----   ----          ----      -----      -----
00:10:a4:33:56:53 0.0.0.0        C     Playstation    ethif1      ethif1      Play
00:10:a4:ad:32:cf 192.168.1.64    C     Generic Device LocalNetwork  ethif4      MyComputer

=>
```

RELATED COMMANDS:

- | | |
|-----------------------|--|
| hostmgr delete | Delete the host device info from the host manager. |
| hostmgr list | List all the host devices. |

Hostmgr Commands

hostmgr clear

Remove all the hosts from the list.

SYNTAX:

```
hostmgr clear
```

EXAMPLE:

```
=>hostmgr list
MAC-address      IP-address      Flags    Type          Intf      Hw Intf      Hostname
-----  -----
00:10:a4:33:56:53 0.0.0.0          -        Playstation   ethif1    ethif1      Play
00:10:a4:ad:32:cf 192.168.1.64    C        Generic Device LocalNetwork  ethif4    MyComputer

=>hostmgr clear
=>hostmgr list
No hosts found.

=>
```

RELATED COMMANDS:

hostmgr delete Delete the host device info from the host manager.

hostmgr config

Configure the host manager parameters.

SYNTAX:

```
hostmgr config [state = <{disabled | enabled}>]
                [scantime = <number{10-600}>]
                [autosave = <{disabled | enabled}>]
                [trace = <{disabled | enabled}>]
```

where:

state	Enable or disable the host manager daemon. The default is enabled .	OPTIONAL
scantime	A number between 10 an 600 (seconds). Represents the time between two scans. The default is 30 (seconds).	OPTIONAL
autosave	Enable or disable automatic saves to flash memory. The default is enabled .	OPTIONAL
trace	Enable or disable the host manager traces. The default is disabled .	OPTIONAL

EXAMPLE:

```
=>hostmgr config
    state      : enabled
    scantime   : 30 sec.
    autosave   : enabled
    trace      : disabled
=>
```

Hostmgr Commands

hostmgr delete

Delete the host device info from the host manager.

SYNTAX:

```
hostmgr delete mac_addr = <hardware-address>
```

where:

mac_addr	The MAC address of the host to be removed.	REQUIRED
----------	--	----------

EXAMPLE:

```
=>hostmgr list
MAC-address      IP-address      Flags   Type           Intf      Hw Intf      Hostname
-----          -----          ----   ----          -----      -----      -----
00:10:a4:33:56:53 0.0.0.0          C       Playstation    ethif1    ethif1      Play
00:10:a4:ad:32:cf 192.168.1.64      C       Generic Device LocalNetwork  ethif4    MyComputer

=>hostmgr delete mac_addr=00:10:a4:fa:33:56
=>hostmgr list
MAC-address      IP-address      Flags   Type           Intf      Hw Intf      Hostname
-----          -----          ----   ----          -----      -----      -----
00:10:a4:ad:32:cf 192.168.1.64      C       Generic Device LocalNetwork  ethif4    MyComputer

=>
```

RELATED COMMANDS:

hostmgr add Add host device info to the host manager.

hostmgr clear Remove all the hosts from the list.

hostmgr list List all the host devices.

hostmgr flush

Flush the host manager configuration.

The host manager configuration is cleaned and returned to default values.

SYNTAX:

```
hostmgr flush
```

Hostmgr Commands

hostmgr list

List all the host devices.

SYNTAX:

```
hostmgr list
```

EXAMPLE:

```
=>hostmgr list
MAC-address      IP-address      Flags    Type          Intf      Hw Intf      Hostname
-----  -----
00:10:a4:33:56:53 0.0.0.0          -        Playstation   ethif1    ethif1      Play
00:10:a4:ad:32:cf 192.168.1.64    C        Generic Device LocalNetwork  ethif4    MyComputer
=>
```

RELATED COMMANDS:

- | | |
|----------------|--|
| hostmgr add | Add host device info to the host manager. |
| hostmgr delete | Delete the host device info from the host manager. |

IDS Commands

Introduction

This chapter describes the commands of the **ids** command group.

Contents

This chapter covers the following commands:

ids clear	Reset the IDS statistics.	352
ids config	Display/modify the Intrusion Detection System (IDS) configuration.	353
ids parser list	Display the IDS parser configuration.	354
ids parser modify	Modify the IDS parser configuration.	355
ids pattern clear	Reset the pattern tracker.	356
ids pattern list	Display the patterns in the pattern tracker.	357
ids pattern stats	Display the pattern tracker statistics.	358
ids signature list	Displays the ids signature configuration	359
ids signature modify	Modify the states of the signatures.	360
ids threshold clear	Reset the IDS thresholds to their default values.	361
ids threshold list	Display the IDS thresholds.	362
ids threshold modify	Modify the IDS thresholds.	363

ids clear

Reset the IDS statistics.

SYNTAX:

```
ids clear
```

RELATED COMMANDS:

ids parser list Display the IDS parser configuration.

ids config

Display/modify the Intrusion Detection System (IDS) configuration.

SYNTAX:

```
ids config [state = <{disabled | enabled}>]  
[trace = <{disabled|enabled}>]
```

where:

state	Enable or disable IDS checks. The default is enabled .	OPTIONAL
trace	Enable or disable tracing. The default is disabled .	OPTIONAL

EXAMPLE:

```
=>ids config  
IDS configuration:  
=====  
state : enabled  
traces ::disabled  
=>
```

ids parser list

Display the IDS parser configuration.

SYNTAX:

```
ids parser list      [parser = <parser>]
```

where:

parser	The name of the parser	REQUIRED
--------	------------------------	----------

EXAMPLE:

```
=>ids parser list
parser          state
-----
fragment        enabled
scan           enabled
dos            enabled
proto          enabled
rate           enabled
=>
```

ids parser modify

Modify the IDS parser configuration.

SYNTAX:

```
ids parser modify parser = <string>
                      state = <{disabled | enabled}>
```

where:

parser	The name of the IDS parser of which the configuration must be modified.	REQUIRED
state	Disable or enable the parser.	REQUIRED

EXAMPLE:

```
=>ids list
Resources:
=====
resource      total   maximum
-----
parsers          5       10
signatures      38      50

Signatures:
=====
parser      signature           hits action      state
-----
fragment    fragment sweep          0 log, drop  enabled
fragment    zero-length fragment size 0 log, drop  enabled
fragment    small fragment size        0 log, drop  enabled
fragment    fragment size overrun    0 log, drop  enabled
fragment    fragment overlap         0 log, drop  enabled
fragment    fragment out-of-order     0 log      enabled
...

=>ids parser modify parser=fragment state=disabled
=>ids list
Resources:
=====
resource      total   maximum
-----
parsers          5       10
signatures      38      50

Signatures:
=====
parser      signature           hits action      state
-----
fragment    fragment sweep          0 log, drop  disabled
fragment    zero-length fragment size 0 log, drop  disabled
fragment    small fragment size        0 log, drop  disabled
fragment    fragment size overrun    0 log, drop  disabled
fragment    fragment overlap         0 log, drop  disabled
fragment    fragment out-of-order     0 log      disabled
...

=>
```

ids pattern clear

Reset the pattern tracker.

SYNTAX:

```
ids pattern clear
```

EXAMPLE:

```
=>ids pattern stats
Pattern tracker statistics:
-----
memory : 32768 bytes
maximum number of patterns : 512
number of active patterns : 8
number of recycled patterns : 0
number of pattern searches : 9
number of new patterns : 8
maximum number of hash collisions : 1
% of hash entries with collisions : 0.09
% of hash entries unused : 99.31
=>ids pattern clear
=>ids pattern stats
Pattern tracker statistics:
-----
memory : 32768 bytes
maximum number of patterns : 512
number of active patterns : 0
number of recycled patterns : 0
number of pattern searches : 0
number of new patterns : 0
maximum number of hash collisions : 0
% of hash entries with collisions : 0.00
% of hash entries unused : 100.00
=>
```

RELATED COMMANDS:

- | | |
|--------------------------|--|
| ids pattern list | Display the patterns in the pattern tracker. |
| ids pattern stats | Display the pattern tracker statistics. |

ids pattern list

Display the patterns in the pattern tracker.

SYNTAX:

```
ids pattern list [size = <{10 | 100 | full} or number>]
```

where:

size	The size of the IDS pattern list that will be shown. Choose between: ▶ 10 ▶ 100 ▶ full or, alternatively, specify a number. The default is full .	OPTIONAL
------	--	----------

EXAMPLE:

```
=>ids pattern list
 1. # 40002# UDP      *.*.*.*:*
 period: [00001564..00001565]    ->      *.*.*.*:*
 2. # 30002# UDP      *.*.*.*:*
 period: [00001564..00001566]    -> 255.255.255.255:67    count:1
 3. # 5000a# UDP      *.*.*.*:*
 period: [00001411..00001425]    -> 255.255.255.255:*
=>
```

RELATED COMMANDS:

ids pattern clear	Reset the pattern tracker.
ids pattern stats	Display the pattern tracker statistics.

ids pattern stats

Display the pattern tracker statistics.

SYNTAX:

```
ids pattern stats
```

EXAMPLE:

```
=>ids pattern stats
Pattern tracker statistics:
-----
memory : 32768 bytes
maximum number of patterns : 512
number of active patterns : 8
number of recycled patterns : 0
number of pattern searches : 9
number of new patterns : 8
maximum number of hash collisions : 1
% of hash entries with collisions : 0.09
% of hash entries unused : 99.31
=>
```

RELATED COMMANDS:

ids pattern clear

Reset the pattern tracker.

ids pattern list

Display the patterns in the pattern tracker.

ids signature list

Displays the ids signature configuration

SYNTAX

```
ids signature list [signature = <fragment_sweep|zero-length_fragment_size|
                    small_fragment_size|fragment_size_overrun|
                    fragment_overlap|fragment_out-of-order|
                    ip_protocol_scan|tcp_port_scan|tcp_syn_scan|
                    stealth_tcp_null_scan|stealth_tcp_fin_scan|
                    stealth_tcp_xmas_scan|
                    stealth_tcp_full_xmas_scan|
                    stealth_tcp_vecna_scan|
                    stealth_tcp_syn-fin_scan|
                    udp_port_scan|ping_sweep_scan|tcp_syn_flood|
                    udp_flood|ping_flood|icmp_unreachable_storm|
                    smurf_broadcast_attack|smurf_storm_attack|
                    fraggle_broadcast_attack|
                    fraggle_storm_attack|land_attack}]>]
```

where

signature	The name of the signature	OPTIONAL
-----------	---------------------------	----------

EXAMPLE:

```
=>ids signature list signature fragment_sweep
signature          parser      hits action      state
-----
fragment_sweep     fragment    0  log        enabled
=>
```

ids signature modify

Modify the states of the signatures.

SYNTAX

```
ids signature modify [signature = <fragment_sweep|
                      zero_length_fragment_size|
                      small_fragment_size|
                      fragment_size_overrun|
                      fragment_overlap|
                      fragment_out-of-order|
                      ip_protocol_scan|
                      tcp_port_scan|tcp_syn_scan|
                      stealth_tcp_null_scan|
                      stealth_tcp_fin_scan|
                      stealth_tcp_xmas_scan|
                      stealth_tcp_full_xmas_scan|
                      stealth_tcp_vecna_scan|
                      stealth_tcp_syn-fin_scan|
                      udp_port_scan|ping_sweep_scan|
                      tcp_syn_flood|udp_flood|ping_flood|
                      icmp_unreachable_storm|
                      smurf_broadcast_attack|
                      smurf_storm_attack|
                      fraggle_broadcast_attack|
                      fraggle_storm_attack|land_attack}>]
                     state = <{disabled|enabled}>
```

where

signature	The name of the signature	REQUIRED
state	The desired state of the signature.	REQUIRED

EXAMPLE:

```
=>ids signature modify
[signature] = ip_protocol_scan
state = disabled
:ids signature modify signature=ip_protocol_scan state=disabled
=>
```

ids threshold clear

Reset the IDS thresholds to their default values.

SYNTAX:

```
ids threshold clear
```

EXAMPLE:

```
=>ids threshold list
index name window limit scaling
-----
1. ids scan 20 20 enabled
2. ids flood 2 100 disabled
3. ids tcp rate 10 300 enabled
4. ids udp rate 1 200 disabled
5. ids icmp rate 1 200 disabled
6. ids ip rate 1 200 disable
=>ids threshold clear
=>ids threshold list
index name window limit scaling
-----
1. ids scan 20 20 enabled
2. ids flood 2 100 disabled
3. ids tcp rate 1 200 disabled
4. ids udp rate 1 200 disabled
5. ids icmp rate 1 200 disabled
6. ids ip rate 1 200 disabled
=>
```

RELATED COMMANDS:

ids threshold list Display the IDS thresholds.

ids threshold modify Modify the IDS thresholds.

ids threshold list

Display the IDS thresholds.

SYNTAX:

```
ids threshold list
```

EXAMPLE:

```
=>ids threshold list
index   name           window      limit    scaling
-----
1. ids scan          20          20    enabled
2. ids flood         2           100   disabled
3. ids tcp rate     1           200   disabled
4. ids udp rate     1           200   disabled
5. ids icmp rate   1           200   disabled
6. ids ip rate      1           200   disabled
=>
```

RELATED COMMANDS:

- | | |
|--------------------------------------|---|
| ids threshold clear | Reset the IDS thresholds to their default values. |
| ids threshold modify | Modify the IDS thresholds. |

ids threshold modify

Modify the IDS thresholds.

SYNTAX:

```
ids threshold modify index = <number>
    [window = <number>]
    [limit = <number>]
    [scaling = <{disabled | enabled}>]
```

where:

index	The index of the IDS threshold to be modified.	REQUIRED
Tip	Use the command :ids threshold list to obtain the indexes.	
window	A number (of seconds). Represents the time window of the threshold.	OPTIONAL
limit	A number. Represents the limit of the threshold.	OPTIONAL
scaling	Enable or disable scaling of the threshold window.	OPTIONAL

EXAMPLE:

```
=>ids threshold list
index      name          window        limit      scaling
-----
1. ids scan           20            20  enabled
2. ids flood          2             100 disabled
3. ids tcp rate       1             200 disabled
4. ids udp rate       1             200 disabled
5. ids icmp rate     1             200 disabled
6. ids ip rate        1             200 disabled
=>ids threshold modify index=3 window=10 limit=300 scaling=enabled
=>ids threshold list
index      name          window        limit      scaling
-----
1. ids scan           20            20  enabled
2. ids flood          2             100 disabled
3. ids tcp rate       10            300  enabled
4. ids udp rate       1             200 disabled
5. ids icmp rate     1             200 disabled
6. ids ip rate        1             200 disabled
=>
```

RELATED COMMANDS:

- | | |
|---------------------|---|
| ids threshold clear | Reset the IDS thresholds to their default values. |
| ids threshold list | Display the IDS thresholds. |

IGMP Commands

Introduction

This chapter describes the commands of the **IGMP** command group.

Contents

This chapter covers the following commands:

igmp host config	Display or modify global IGMP configuration.	366
igmp host flush	Flush the IGMP settings.	367
igmp host list	Show the IGMP groups.	
igmp host debug clear	Show the IGMP groups.	368
igmp host ifconfig	Configure an IGMP interface.	369
igmp host iflist	Show the configuration of the IGMP interfaces.	370
igmp host debug clear	Clear IGMP statistics.	371
igmp host debug stats	Print IGMP statistics.	372
igmp proxy config	Configure the IGMP proxy.	373
igmp proxy flush	Flush all IGMP proxy settings and learned groups.	375
igmp proxy grouplist	Show the learned groups on an IGMP proxy interface.	376
igmp proxy ifconfig	Configure an IGMP proxy interface.	377
igmp proxy iflist	Show the configuration of an IGMP proxy interface.	378
igmp proxy mbslist	Show the IGMP proxy membership database (merge of all learned groups).	379
igmp proxy config	Clear IGMP proxy statistics.	380
igmp proxy debug stats	Print IGMP proxy statistics.	381
igmp proxy debug traceconfig	Modify IGMP proxy trace configuration.	382

igmp host config

Display or modify global IGMP configuration.

SYNTAX:

```
igmp host config [requirera = <{disabled | enabled}>]
```

where:

requirera	Enable or disable the router alert IP option check. The default is disabled .	OPTIONAL
-----------	---	----------

EXAMPLE:

```
=>igmp host confighostmgr list
Router alert IP option check : disabled
=>
```

RELATED COMMANDS:

igmp host debug clear Show the IGMP groups.

igmp host flush

Flush the IGMP settings.

SYNTAX:

```
igmp host flush
```

igmp host list

Show the IGMP groups.

SYNTAX:

```
igmp host list [intf = <string>]  
[expand = <{disabled | enabled}>]
```

where:

intf	The IP interface name.	OPTIONAL
expand	Enable or disable expanded listing of IGMP groups. The default is disabled .	OPTIONAL

EXAMPLE:

```
=>igmp host list  
Interface          Group          Filter-Mode  Source  
4    lan1           224.0.0.103  EXCLUDE      none  
4    lan1           239.255.255.250 EXCLUDE      none  
=>
```

igmp host ifconfig

Configure an IGMP interface.

SYNTAX:

```
igmp host ifconfig    intf = <string>
                      version = <{none | IGMPv1 | IGMPv2 | IGMPv3}>
```

where:

intf	The IP interface name.	REQUIRED
version	The IGMP version of the IP interface. Choose between: <ul style="list-style-type: none">▶ IGMPv1▶ IGMPv2▶ IGMPv3▶ none	REQUIRED

EXAMPLE:

```
=>igmp host ifconfig
intf = lan1
version = IGMPv1
:igmp host ifconfig intf=lan1 version=IGMPv1
=>
```

RELATED COMMANDS:

igmp proxy iflist Show the configuration of the IGMP interfaces.

igmp host iflist

Show the configuration of the IGMP interfaces.

SYNTAX:

```
igmp host iflist [expand = <{disabled | enabled}>]
```

where:

expand	Enable or disable expanded listing of IGMP interfaces configuration. The default is disabled .	OPTIONAL
--------	--	----------

EXAMPLE:

```
=>igmp host iflist
Interface          Version
0    loop            IGMPv3
1    Internet        IGMPv3
2    ISDN_backup     IGMPv3
3    ISDN_backup_trigger IGMPv3
4    lan1            IGMPv3
5    wan1            IGMPv3
6    dmz1            IGMPv3
7    guest1          IGMPv3
=>
```

RELATED COMMANDS:

igmp proxy ifconfig	Configure an IGMP interface.
---------------------	------------------------------

igmp host debug clear

Clear IGMP statistics.

SYNTAX:

```
igmp host debug clear
```

RELATED COMMANDS:

igmp host debug stats Print IGMP statistics.

igmp host debug stats

Print IGMP statistics.

SYNTAX:

```
igmp host debug stats
```

EXAMPLE:

```
=>igmp host debug stats
Total IGMP messages received : 0
Too small IGMP messages received : 0
Too long IGMP messages received : 0
IGMP messages with bad checksum received : 0
IGMP messages with bad TTL received : 0
IGMP messages with no router alert IP option received : 0
IGMPv1 membership queries received : 0
IGMPv2 membership queries received : 0
IGMPv3 membership queries received : 0
IGMP bad queries received : 0
IGMP failing membership queries : 0
IGMPv1/v2 membership reports received : 0
IGMPv1/v2 invalid membership reports received : 0
IGMPv1/v2 membership reports received for our groups : 0
IGMPv1/v2 membership reports transmitted : 0
IGMPv3 membership reports transmitted : 0
=>
```

RELATED COMMANDS:

igmp host debug clear Clear IGMP statistics.

igmp proxy config

Configure the IGMP proxy.

SYNTAX:

```
igmp proxy config [state = <{disabled | enabled}>]
[qi = <number{1-86400}>]
[qri = <number{1-86400}>]
[lmqi = <number{1-86400}>]
[rv = <number{2-10}>]
[requirera = <{disabled | enabled}>]
[localgroup = <{disabled | enabled}>]
```

where:

state	Enable or disable the IGMP proxy. The default is disabled .	OPTIONAL
qi	A number between 1 and 86400. Represents the interval in seconds between general queries sent by the querier.	OPTIONAL
qri	A number between 1 and 86400. Represents the maximum response time in seconds for an IGMP client in reply to general queries.	OPTIONAL
lmqi	A number between 1 and 86400. Represents the maximum response time in seconds for an IGMP client in reply to group specific queries.	OPTIONAL
rv	A number between 2 and 10. Represents the robustness variable, which allows tuning for expected IGMP packet loss.	OPTIONAL
requirera	Enable or disable the router alert IP option check. The default is disabled .	OPTIONAL
localgroup	Enable or disable the processing of a local multicast group in an IGMP packet. The default is disabled .	OPTIONAL

EXAMPLE:

```
=>igmp proxy config
IGMP proxy state          : disabled
Query Interval             : 0 days, 0:02:05
Query Response Interval    : 0 days, 0:00:10
Last Member Query Interval : 0 days, 0:00:01
Robustness variable        : 2
Router alert IP option check : disabled
Process local multicast group : disabled
=>
```

IGMP Commands

RELATED COMMANDS:

igmp proxy grouplist Show the learned groups on an IGMP proxy interface.

igmp proxy flush

Flush all IGMP proxy settings and learned groups.

SYNTAX:

```
igmp proxy flush
```

igmp proxy grouplist

Show the learned groups on an IGMP proxy interface.

SYNTAX:

```
igmp proxy grouplist      [intf = <string>]  
                          [expand = <{disabled | enabled}>]
```

where:

intf	The name of the IGMP proxy interface to be listed.	OPTIONAL
expand	Enable or disable expanded listing of the learned groups on an IGMP proxy interface. The default is disabled .	OPTIONAL

RELATED COMMANDS:

[igmp proxy config](#) Configure the IGMP proxy.

igmp proxy ifconfig

Configure an IGMP proxy interface.

SYNTAX:

```
igmp proxy ifconfig    intf = <string>
                      [state = <{inactive | downstream | upstream}>]
                      [version = <{IGMPv1 | IGMPv2 | IGMPv3}>]
                      [fastleave = <{disabled | enabled}>]
```

where:

intf	The name of the IGMP proxy interface to be configured.	REQUIRED
state	The state of the IGMP proxy interface. Choose between: <ul style="list-style-type: none"> ▶ inactive ▶ downstream ▶ upstream 	OPTIONAL
version	The IGMP version of the IGMP proxy interface. Choose between: <ul style="list-style-type: none"> ▶ IGMPv1 ▶ IGMPv2 ▶ IGMPv3 ▶ none 	OPTIONAL
fastleave	Enable or disable the immediate deletion of a group when a leave is received. The default is disabled .	OPTIONAL

EXAMPLE:

```
=>igmp proxy ifconfig
intf = lan1
[state] = downstream
[version] = IGMPv3
[fastleave] = disabled
=>
```

RELATED COMMANDS:

igmp proxy iflist	Show the configuration of an IGMP proxy interface.
-------------------	--

igmp proxy iflist

Show the configuration of an IGMP proxy interface.

SYNTAX:

```
igmp proxy iflist      [expand = <{disabled | enabled}>]
```

where:

expand	Enable or disable expanded listing of IGMP proxy interface configuration. The default is disabled .	OPTIONAL
---------------	---	----------

EXAMPLE:

```
=>igmp proxy iflist
Interface          State       Version   Querier
7 guest1           inactive   -         -
6 dmz1             inactive   -         -
5 wan1             inactive   -         -
4 lan1             downstream -         -
3 ISDN_backup_trigger inactive -         -
2 ISDN_backup      inactive   -         -
1 Internet         inactive   -         -
=>
```

RELATED COMMANDS:

igmp proxy ifconfig Configure an IGMP proxy interface.

igmp proxy mbslist

Show the IGMP proxy membership database (merge of all learned groups).

SYNTAX:

```
igmp proxy mbslist
```

igmp proxy debug clear

Clear IGMP proxy statistics.

SYNTAX:

```
igmp proxy debug clear
```

RELATED COMMANDS:

igmp proxy debug stats	Print IGMP proxy statistics.
igmp proxy debug traceconfig	Modify IGMP proxy trace configuration.

igmp proxy debug stats

Print IGMP proxy statistics.

SYNTAX:

```
igmp proxy debug stats
```

EXAMPLE:

```
=>igmp proxy debug stats
IGMP proxy statistics:
    Total IGMP packets recv : 0
    Too short IGMP packets recv : 0
    IGMP packets with bad checksum recv : 0
    IGMP packets with bad ttl recv : 0
    IGMP packets with no route alert option recv : 0
    IGMPv1 queries recv : 0
    IGMPv2 queries recv : 0
    IGMPv3 queries recv : 0
    IGMP bad queries recv : 0
    IGMP queries fail : 0
    IGMPv1 reports recv : 0
    IGMPv2 reports recv : 0
    IGMPv3 reports recv : 0
    IGMP bad reports recv : 0
    IGMP leave reports recv : 0
    IGMP bad leave reports recv : 0
    IGMPv1 queries sent : 0
    IGMPv2 queries sent : 0
    IGMPv3 queries sent : 0
    IGMP query election switch : 0
=>
```

RELATED COMMANDS:

igmp proxy debug clear	Clear IGMP proxy statistics.
igmp proxy debug traceconfig	Modify IGMP proxy trace configuration.

igmp proxy debug traceconfig

Modify IGMP proxy trace configuration.

SYNTAX:

```
igmp proxy debug traceconfig [state = <{disabled | enabled}>]
```

where:

state	Enable or disable tracing. The default is disabled .	OPTIONAL
-------	--	----------

RELATED COMMANDS:

- | | |
|------------------------|------------------------------|
| igmp proxy debug clear | Clear IGMP proxy statistics. |
| igmp proxy debug stats | Print IGMP proxy statistics. |

Interface Commands

Introduction

This chapter describes the commands of the **interface** command group.

Contents

This chapter covers the following commands:

interface list	Display interfaces.	384
----------------	---------------------	-----

interface list

Display interfaces.

SYNTAX:

```
interface list [expand = <{disabled | enabled}>]  
[reverse = <{disabled | enabled}>]
```

where:

expand	Enable or disable expanded listing of interfaces. The default is disabled .	OPTIONAL
reverse	Enable or disable reverse listing (lower layer first instead of upper layer first). The default is disabled .	OPTIONAL

EXAMPLE:

```
=>interface list
Name      Type      State      Use  UL Interfaces
ethif1    physical  connected   1    bridge
ethif4    physical  connected   1    bridge
usbif1   physical  connected   1    bridge
RELAY     eth       connected   1    Internet_ppp
bridge    eth       connected   1    eth0
atm_0_35 atm      connected   1    ethoa_0_35
atm_8_35 atm      connected   1    ethoa_8_35
ethoa_0_35 eth      connected   0
ethoa_8_35 eth      connected   0
Internet_ppp ppp     not-connected 1    Internet
Internet   ip       not-connected 0
eth0      ip       connected   0

=>
=>interface list expand=enabled
Name      Type      State      Use  UL Interfaces
ethif1    physical  connected   1    bridge
Flags.....: INTERNAL
Phys.....: intf: 0 type eth speed: 100 Mbps
ethif4    physical  connected   1    bridge
Flags.....: INTERNAL
Phys.....: intf: 3 type eth speed: 100 Mbps
usbif1   physical  connected   1    bridge
Flags.....: INTERNAL
Phys.....: intf: 4 type usb speed: 12 Mbps
RELAY     eth       connected   1    Internet_ppp
Flags.....: INTERNAL DYNAMIC RELAY
Eth.....: port: 65535 ip_cid: 200 arp_cid: 0
bridge    eth       connected   1    eth0
Flags.....: DYNAMIC
Eth.....: port: 0 ip_cid: 16385 arp_cid: 16386
atm_0_35 atm      connected   1    ethoa_0_35
Flags.....:
Atm.....: cid: 8196 atm_cid: 8196 llc_cid: 0 ppp_cid: 0
atm_8_35 atm      connected   1    ethoa_8_35
Flags.....:
Atm.....: cid: 8198 atm_cid: 8198 llc_cid: 0 ppp_cid: 0
ethoa_0_35 eth      connected   0
Flags.....:
Eth.....: port: 1 ip_cid: 16387 arp_cid: 16388
ethoa_8_35 eth      connected   0
Flags.....:
Eth.....: port: 2 ip_cid: 16389 arp_cid: 16390
Internet_ppp ppp     not-connected 1    Internet
Flags.....:
Ppp.....: cid: 0 ip_cid: 0
Internet   ip       not-connected 0
Flags.....: DYNAMIC
Ip.....: dest: Internet_ppp
eth0      ip       connected   0
Flags.....:
Ip.....: dest: bridge
=>
```


IP Commands

Introduction

This chapter describes the commands of the **ip** command group.

Contents

This chapter covers the following commands:

ip arpadd	Add an entry to the ARP cache of a broadcast Internet Protocol (IP) interface.	389
ip arpremove	Delete an entry from the ARP cache.	390
ip arplist	Display the ARP cache.	391
ip config	Show/set global IP stack configuration options.	392
ip flush	Flush all the static IP parameters.	394
ip ifadd	Create an IP interface.	395
ip ifattach	Attach an IP interface.	396
ip ifconfig	Configure the parameters of an IP interface.	397
ip ifdelete	Delete an IP interface.	399
ip ifdetach	Detach an IP interface.	400
ip iflist	Display all the IP interfaces.	401
ip ifwait	Wait for a status change of an IP interface.	402
ip ipadd	Assign an IP address to an IP interface.	403
ip ipconfig	Modify an IP address configuration.	404
ip ipdelete	Remove an IP address from an IP interface.	405
ip iplist	Display all the configured IP addresses.	406
ip mcast rtadd	Add a multicast route to the multicast routing table.	407
ip mcast rtdelete	Delete a multicast route from the multicast routing table.	408
ip mcast rtlist	Display the multicast routing table.	409
ip mcast flush	Flush the multicast routing table.	410
ip rtadd	Add a route to the routing table.	411
ip rtdelete	Delete a route from the routing table.	412
ip rtlist	Display the routing table.	413
ip auto flush	Flush the autoIP interfaces.	414
ip auto ifadd	Create an autoIP interface.	415
ip auto ifattach	Select and assign a link-local address to an autoIP interface.	416
ip auto ifconfig	Configure an autoIP interface.	417
ip auto ifdelete	Delete an existing autoIP interface.	419
ip auto ifdetach	Release the link-local address for the given autoIP interface.	420
ip auto iflist	Display the autoIP interfaces.	421
ip debug httpprobe	Send a HTTP probe.	422

IP Commands

ip debug sendto	Send UDP packets.	423
ip debug stats	Display IP statistics.	425
ip debug traceconfig	Display/modify the IP stack trace configuration.	426

ip arpadd

Add an entry to the ARP cache of a broadcast Internet Protocol (IP) interface.

SYNTAX:

```
ip arpadd intf = <string>
    ip = <ip-range>
    [hwaddr = <hardware-address>]
```

where:

intf	The IP interface name.	REQUIRED
ip	The IP address (or range) of the entry to be added to the Address Resolution Protocol (ARP) cache.	REQUIRED
hwaddr	The hardware address (for example the Ethernet MAC address) of the entry to be added.	OPTIONAL

EXAMPLE:

```
=>ip arpelist
Interface      IP-address          HW-address        Type
2   eth0        10.0.0.1           00:10:a4:ad:32:cf STATIC
=>ip arpadd intf=eth0 ip=10.0.0.2
=>ip arpelist
Interface      IP-address          HW-address        Type
2   eth0        10.0.0.1           00:10:a4:ad:32:cf STATIC
2   eth0        10.0.0.2           00:00:00:00:00:00 DYNAMIC
=>ip arpadd intf=eth0 ip=10.0.0.3 hwaddr=00:a0:24:ae:66:e1
=>ip arpelist
Interface      IP-address          HW-address        Type
2   eth0        10.0.0.1           00:10:a4:ad:32:cf STATIC
2   eth0        10.0.0.2           00:00:00:00:00:00 DYNAMIC
2   eth0        10.0.0.3           00:a0:24:ae:66:e1 STATIC
=>
```

RELATED COMMANDS:

- ip arpdelete Delete an entry from the ARP cache.
ip arpelist Display the ARP cache.

ip arpdelete

Delete an entry from the ARP cache.

SYNTAX:

```
ip arpdelete intf = <string>
              ip = <ip-range>
              [hwaddr = <hardware-address>]
```

where:

intf	The IP interface name.	REQUIRED
ip	The IP address (or range) of the entry to be deleted.	REQUIRED
hwaddr	The hardware address (for example the Ethernet MAC address) of the entry to be deleted.	OPTIONAL

EXAMPLE:

```
=>ip arplist
Interface      IP-address          HW-address        Type
2   eth0        10.0.0.1           00:10:a4:ad:32:cf STATIC
2   eth0        10.0.0.2           00:00:00:00:00:00 DYNAMIC
2   eth0        10.0.0.3           00:a0:24:ae:66:e1 STATIC
=>ip arpdelete intf=eth0 ip=10.0.0.3 hwaddr=00:a0:24:ae:66:e1
=>ip arplist
Interface      IP-address          HW-address        Type
2   eth0        10.0.0.1           00:10:a4:ad:32:cf STATIC
2   eth0        10.0.0.2           00:00:00:00:00:00 DYNAMIC
=>
```

RELATED COMMANDS:

- | | |
|-------------------|--|
| ip arpadd | Add an entry to the ARP cache of a broadcast Internet Protocol (IP) interface. |
| ip arplist | Display the ARP cache. |

ip arplist

Display the ARP cache.

SYNTAX:

```
ip arplist
```

EXAMPLE:

```
=>ip arplist
Interface      IP-address          HW-address      Type
2   eth0        10.0.0.1           00:10:a4:ad:32:cf STATIC
2   eth0        10.0.0.2           00:00:00:00:00:00 DYNAMIC
2   eth0        10.0.0.3           00:a0:24:ae:66:e1 STATIC
=>
```

RELATED COMMANDS:

- | | |
|------------------------------|--|
| ip arpadd | Add an entry to the ARP cache of a broadcast Internet Protocol (IP) interface. |
| ip arpdelete | Delete an entry from the ARP cache. |

ip config

Show/set global IP stack configuration options.

SYNTAX:

```
ip config [forwarding = <{disabled | enabled}>]
[redirects = <{disabled | enabled}>]
[checkoptions = <{disabled|enabled|transparent}>]
[netbroadcasts = <{disabled | enabled}>]
[ttl = <number{0-255}>]
[defragmode = <{disabled | enabled}>]
[addrcheck = <{off | own | static | dynamic}>]
[mssclamping = <{disabled | enabled}>]
[natloopback = <{disabled|enabled}>]
[bitrate-window(sec) = <number{10-3600}>]
```

where:

forwarding	Disable or enable the IP routing functionality. The default is enabled .	OPTIONAL
redirects	Disable or enable the sending of ICMP redirect messages. A router can send a redirect message in case a shorter path than the path followed is discovered. The default is enabled (for security reasons).	OPTIONAL
checkoptions	Disallow/Allow packets with IP options. The default is enabled .	OPTIONAL
netbroadcasts	Disable or enable net directed broadcasts. The default is 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. By limiting the TTL, continuous circulation of IP packets on the network without ever reaching a destination is avoided. The default is 64 .	OPTIONAL
defragmode	Disallow (disabled) or allow (enabled) defragmenting IP fragments. The default is enabled .	OPTIONAL

addrcheck	Set the level of IP address checks. Choose between: <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 (for example 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. The default is dynamic .	OPTIONAL
mssclamping	Disable or enable mss clamping for low MTU interfaces. Enabling mss clamping assures that the size of a TCP packet never exceeds the available Maximum Transmission Unit (MTU) of the outgoing interface. The default is on . Note It is recommended not to disable this parameter.	OPTIONAL
natloopback	Disable/Enable NAT loopback. The default is enabled .	OPTIONAL
bitrate-window(sec)	A number between 10 and 3600. Set the window to calculate the bitrate (multiple of 10). The default is 30 .	OPTIONAL

EXAMPLE:

```
=>ip config
Forwarding enabled
Sendredirects enabled
IP options enabled
NetBroadcasts disabled
Default TTL 64
Fraglimit 64 fragments
Fragcount currently 0 fragments
Defragment mode : enabled
Address checks : dynamic
Mss Clamping : enabled
NAT Loopback : enabled
Bitrate window (sec), multiple of 10 : 30
=>
```

ip flush

Flush all the static IP parameters.

Dynamic configurations (for example from PPP or CIP links) remain.



The flush command does not impact previously saved configurations.



The command **:ip flush** deletes all local IP connectivity.

Do not use this command during an IP based local connection, for example a Telnet CLI session, or web based CLI access.

SYNTAX:

```
ip flush
```

ip ifadd

Create an IP interface.

SYNTAX:

```
ip ifadd intf = <string>
      dest = <string>
```

where:

intf	The name of the IP interface to be created.	REQUIRED
dest	An network interface name.	REQUIRED

EXAMPLE:

```
=>ip iflist
Interface      Group  MTU    RX      TX      TX-Drop  Status  HW-address
0  loop        local   65535 122062  72987    0        [UP]    00:0e:50:0f:fc:2c
1  Internet    wan     1500   0       0       0        DOWN
2  RtPPPoE_ppp wan     1500   0       0       0        DOWN
3  LocalNetwork lan     1500   84105  123358   0        [UP]    00:0e:50:0f:fc:2c
=>ip ifadd intf=myIPintf dest=RtPPPoE_eth
=>ip iflist
Interface      Group  MTU    RX      TX      TX-Drop  Status  HW-address
0  loop        local   65535 123966  75177    0        [UP]    00:0e:50:0f:fc:2c
1  Internet    wan     1500   0       0       0        DOWN
2  RtPPPoE_ppp wan     1500   0       0       0        DOWN
3  LocalNetwork lan     1500   86589  125262   0        [UP]    00:0e:50:0f:fc:2c
4  myIPintf    wan     1500   0       0       0        DOWN  00:0e:50:0f:fc:2c
=>
```

RELATED COMMANDS:

ip ifdelete	Delete an IP interface.
ip iflist	Display all the IP interfaces.
ip ifwait	Wait for a status change of an IP interface.

ip ifattach

Attach an IP interface.

SYNTAX:

```
ip ifattach intf = <string>
```

where:

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

EXAMPLE:

```
=>ip iflist
Interface      Group  MTU   RX      TX      TX-Drop  Status  HW-address
0   loop        local  65535 123966  75177    0        [UP]    00:0e:50:0f:fc:2c
1   Internet    wan    1500   0       0        0        DOWN
2   RtPPPoE_ppp wan    1500   0       0        0        DOWN
3   LocalNetwork lan    1500   86589   125262   0        [UP]    00:0e:50:0f:fc:2c
4   myIPintf    wan    1500   0       0        0        DOWN  00:0e:50:0f:fc:2c
=>ip ifattach intf=myIPintf
=>ip iflist
Interface      Group  MTU   RX      TX      TX-Drop  Status  HW-address
0   loop        local  65535 123966  75177    0        [UP]    00:0e:50:0f:fc:2c
1   Internet    wan    1500   0       0        0        DOWN
2   RtPPPoE_ppp wan    1500   0       0        0        DOWN
3   LocalNetwork lan    1500   86589   125262   0        [UP]    00:0e:50:0f:fc:2c
4   myIPintf    wan    1500   0       0        0        [UP]   00:0e:50:0f:fc:2c
=>
```

RELATED COMMANDS:

ip ifdetach	Detach an IP interface.
-------------	-------------------------

ip ifconfig

Configure the parameters of an IP interface.

SYNTAX:

```
ip ifconfig intf = <string>
    [mtu = <number{68-65535}>]
    [status = <{down | up}>]
    [hwaddr = <hardware-address>]
    [group = <string> or number]
    [linksensing = <{disabled | enabled}>]
    [primary = <{disabled | enabled}>]
    [mcastpromisc = <{disabled|enabled}>]
```

where:

intf	The name of the IP interface to be configured.	REQUIRED
mtu	A number between 68 and 65535. Represents the MTU (the maximum packet size (including IP header)) to be used on this interface.	OPTIONAL
	Note The default value depends on the connection and packet service for which the interface was created.	
status	The administrative state of the interface. Choose between: ▶ down ▶ up .	OPTIONAL
hwaddr	The hardware address (for example the Ethernet MAC address) of this IP interface.	OPTIONAL
group	The group to which this interface belongs. Can be used for firewalling, for example.	OPTIONAL
linksensing	The IP interface's awareness of link state transitions. The default is enabled .	OPTIONAL
primary	Make the IP interface the primary interface (enabled) or not (disabled). The default is disabled .	OPTIONAL
mcastpromisc	Make the IP interface multicast promiscuous. Choose between: ▶ disabled ▶ enabled The default is disabled .	OPTIONAL

EXAMPLE:

```
=>ip iflist
Interface      Group  MTU    RX      TX      TX-Drop  Status  HW-address
0  loop        local   65535 123966  75177    0       [UP]    00:0e:50:0f:fc:2c
1  Internet    wan     1500   0       0       0       DOWN
2  RtPPPoE_ppp wan     1500   0       0       0       DOWN
3  LocalNetwork lan     1500   86589  125262   0       [UP]    00:0e:50:0f:fc:2c
4  myIPintf    wan     1500   0       0       0       DOWN  00:0e:50:0f:fc:2c
=>ip ifconfig
intf = myIPintf
[mtu] = 1500
[status] = up
[hwaddr] = 00:0e:50:0f:fc:2c
[group] = wan
[linksensing] = enabled
[primary] = enabled
:ip ifconfig intf=myIPintf
=>ip iflist
Interface      Group  MTU    RX      TX      TX-Drop  Status  HW-address
0  loop        local   65535 123966  75177    0       [UP]    00:0e:50:0f:fc:2c
1  Internet    wan     1500   0       0       0       DOWN
2  RtPPPoE_ppp wan     1500   0       0       0       DOWN
3  LocalNetwork lan     1500   86589  125262   0       [UP]    00:0e:50:0f:fc:2c
4  myIPintf    wan     1500   0       0       0       [UP]   00:0e:50:0f:fc:2c
=>
```



If the STATUS is shown between square brackets, then **linksensing** is disabled for that particular interface.

ip ifdelete

Delete an IP interface.

SYNTAX:

```
ip ifdelete intf = <string>
```

where:

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

EXAMPLE:

```
=>ip iflist
Interface      Group   MTU    RX      TX      TX-Drop  Status  HW-address
0   loop        local   65535 123966  75177   0        [UP]    00:0e:50:0f:fc:2c
1   Internet    wan     1500   0       0       0        DOWN
2   RtPPPoE_ppp wan     1500   0       0       0        DOWN
3   LocalNetwork lan     1500   86589  125262  0        [UP]    00:0e:50:0f:fc:2c
4   myIPintf    wan     1500   0       0       0        [UP]    00:0e:50:0f:fc:2c
=>ip ifdelete intf=myIPintf
=>ip iflist
Interface      Group   MTU    RX      TX      TX-Drop  Status  HW-address
0   loop        local   65535 123966  75177   0        [UP]    00:0e:50:0f:fc:2c
1   Internet    wan     1500   0       0       0        DOWN
2   RtPPPoE_ppp wan     1500   0       0       0        DOWN
3   LocalNetwork lan     1500   86589  125262  0        [UP]    00:0e:50:0f:fc:2c
=>
```

RELATED COMMANDS:

ip ifadd	Create an IP interface.
ip iflist	Display all the IP interfaces.
ip ifwait	Wait for a status change of an IP interface.

ip ifdetach

Detach an IP interface.

SYNTAX:

```
ip ifdetach intf = <string>
```

where:

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

EXAMPLE:

```
=>ip iflist expand=enabled
Interface      Group  MTU   RX      TX      TX-Drop  Status  HW-address
0  loop        local  65535 130597  82240    0        [UP]    00:0e:50:0f:fc:2c
  BRHW-address : ff:ff:ff:ff:ff:ff
  RX unicастpkts: 1304    брэстpkts : 0
  TX unicастpkts: 1994    брэстpkts : 0        дропpkts:0
  Oper state   : UP      Admin State: UP
  Flags        : ARP BROADCAST ARPTABLE LOOP MULTICAST INTERNAL
...
4  myIPintf    wan    1500  0       0       0        DOWN   00:0e:50:0f:fc:2c
  BRHW-address : ff:ff:ff:ff:ff:ff
  RX unicастpkts: 0       брэстpkts : 0
  TX unicастpkts: 0       брэстpkts : 0        дропpkts:0
  Oper state   : DOWN    Admin State: UP
  Flags        : PRIMARY ARP BROADCAST BOUND ARPTABLE MULTICAST LINKSENSING STATIC
=>ip ifdetach intf=myIPintf
=>ip iflist expand=enabled
Interface      Group  MTU   RX      TX      TX-Drop  Status  HW-address
0  loop        local  65535 133683  83949    0        [UP]    00:0e:50:0f:fc:2c
  BRHW-address : ff:ff:ff:ff:ff:ff
  RX unicастpkts: 1332    брэстpkts : 0
  TX unicастpkts: 2036    брэстpkts : 0        дропpkts:0
  Oper state   : UP      Admin State: UP
  Flags        : ARP BROADCAST ARPTABLE LOOP MULTICAST INTERNAL
...
4  myIPintf    wan    1500  0       0       0        DOWN   00:0e:50:0f:fc:2c
  BRHW-address : ff:ff:ff:ff:ff:ff
  RX unicастpkts: 0       брэстpkts : 0
  TX unicастpkts: 0       брэстpkts : 0        дропpkts:0
  Oper state   : DOWN    Admin State: DOWN
  Flags        : PRIMARY ARP BROADCAST ARPTABLE MULTICAST LINKSENSING STATIC
=>
```

RELATED COMMANDS:

ip ifattach	Attach an IP interface.
-----------------------------	-------------------------

ip iflist

Display all the IP interfaces.

SYNTAX:

```
ip iflist [expand = <{disabled | enabled}>]
```

where:

expand	Enable or disable expanded listing. The default is disabled .	OPTIONAL
--------	---	----------

EXAMPLE:

```
=>ip iflist
Interface      Group  MTU    RX      TX      TX-Drop  Status  HW-address
0   loop        local  65535 123966  75177   0        [UP]    00:0e:50:0f:fc:2c
1   Internet    wan    1500   0       0       0        DOWN
2   RtPPPoE_ppp wan    1500   0       0       0        DOWN
3   LocalNetwork lan    1500   86589  125262  0        [UP]    00:0e:50:0f:fc:2c
4   myIPintf    wan    1500   0       0       0        [UP]    00:0e:50:0f:fc:2c
=>
```

RELATED COMMANDS:

ip ifadd	Create an IP interface.
ip ifdelete	Delete an IP interface.
ip ifwait	Wait for a status change of an IP interface.

ip ifwait

Wait for a status change of an IP interface.

SYNTAX:

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

where:

intf	The IP interface name.	REQUIRED
timeout	A number between 1 and 600000 (seconds). Represents the timeout.	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

RELATED COMMANDS:

ip ifadd	Create an IP interface.
ip ifdelete	Delete an IP interface.
ip iflist	Display all the IP interfaces.

ip ipadd

Assign an IP address to an IP interface.

SYNTAX:

```
ip ipadd intf = <string>
      addr = <ip-address>
      [netmask = <ip-mask(dotted or cidr)>]
      [pointopoint = <ip-address>]
      [addroute = <{disabled | enabled}>]
```

where:

intf	The IP interface name.	REQUIRED
addr	The new IP address to be added.	REQUIRED
netmask	The subnetmask associated with this address.	OPTIONAL
pointopoint	The remote IP address in case of a dedicated point-to-point link.	OPTIONAL
addroute	Add typical net/subnet routes automatically according to the default (or specified) subnet mask (enabled) or not (disabled). The default is disabled .	OPTIONAL

EXAMPLE:

```
=>ip iplist
Interface      Type          IP-address      Point-to-point/Mask
1   eth0        Ethernet      10.0.0.138    255.255.255.0
1   eth0        Ethernet      169.254.141.11 255.255.0.0
0   loop         Ethernet      127.0.0.1     255.0.0.0

=>ip ipadd intf=eth0 addr=10.0.0.2/24 addroute=enabled
=>ip iplist
Interface      Type          IP-address      Point-to-point/Mask
1   eth0        Ethernet      10.0.0.2       255.255.255.0
1   eth0        Ethernet      10.0.0.138    255.255.255.0
1   eth0        Ethernet      169.254.141.11 255.255.0.0
0   loop         Ethernet      127.0.0.1     255.0.0.0
=>
```

RELATED COMMANDS:

- | | |
|-------------|--|
| ip ipdelete | Remove an IP address from an IP interface. |
| ip iplist | Display all the configured IP addresses. |

ip ipconfig

Modify an IP address configuration.

SYNTAX:

```
ip ipconfig addr = <ip-address>
    [preferred = <{disabled | enabled}>]
    [primary = <{disabled | enabled}>]
```

where:

addr	The IP address to be configured.	REQUIRED
preferred	Make this IP address the preferred address for that subnet (enabled) or not (disabled). The default is disabled .	OPTIONAL
primary	Make this IP address the primary address for the interface (enabled) or not (disabled). The default is disabled .	OPTIONAL

EXAMPLE:

```
=>ip iplist
Interface      Type          IP-address      Point-to-point/Mask
1   eth0        Ethernet      10.0.0.138     255.255.255.0
1   eth0        Ethernet      169.254.141.11  255.255.0.0
0   loop         Ethernet      127.0.0.1      255.0.0.0
=>ip ipconfig addr=169.254.141.11 preferred=enabled primary=enabled
=>ip iplist
Interface      Type          IP-address      Point-to-point/Mask
1   eth0        Ethernet      10.0.0.138     255.255.255.0
1   eth0        Ethernet      *169.254.141.11  255.255.0.0
0   loop         Ethernet      127.0.0.1      255.0.0.0
=>
```



The preferred IP address is marked with "*" in the list.

ip ipdelete

Remove an IP address from an IP interface.

SYNTAX:

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

where:

addr	The IP address to be deleted.	REQUIRED
------	-------------------------------	----------

EXAMPLE:

```
=>ip iplist
Interface      Type          IP-address      Point-to-point/Mask
1   eth0        Ethernet      10.0.0.2        255.255.255.0
1   eth0        Ethernet      10.0.0.138     255.255.255.0
1   eth0        Ethernet      169.254.141.11 255.255.0.0
0   loop         Ethernet      127.0.0.1       255.0.0.0

=>ip ipdelete addr=10.0.0.2
=>ip iplist
Interface      Type          IP-address      Point-to-point/Mask
1   eth0        Ethernet      10.0.0.138     255.255.255.0
1   eth0        Ethernet      169.254.141.11 255.255.0.0
0   loop         Ethernet      127.0.0.1       255.0.0.0

=>
```

RELATED COMMANDS:

ip ipadd	Assign an IP address to an IP interface.
ip iplist	Display all the configured IP addresses.

ip iplist

Display all the configured IP addresses.

SYNTAX:

```
ip iplist
```

EXAMPLE:

```
=>ip iplist
Interface          Type        IP-address      Point-to-point/Mask
2     LocalNetwork  Ethernet    10.0.0.138    255.255.255.0
2     LocalNetwork  Ethernet    *192.168.1.254 255.255.255.0
0     loop          Ethernet    127.0.0.1     255.255.255.255
=>
```

RELATED COMMANDS:

[ip ipadd](#) Assign an IP address to an IP interface.

[ip ipdelete](#) Remove an IP address from an IP interface.

ip mcast rtadd

Add a multicast route to the multicast routing table.

SYNTAX:

```
ip mcast rtadd    srcintf = <string>
                  [src = <ip-address>]
                  grp = <ip-address>
                  dstintf = <string>
                  [ttl = <number{1-255}>]
                  [ttlincr = {disabled | enabled}]
```

where:

srcintf	The source IP interface.	REQUIRED
src	The source IP address.	OPTIONAL
grp	The multicast group IP address.	REQUIRED
dstintf	The destination IP interface.	REQUIRED
ttl	The time-to-live for that destination IP interface.	OPTIONAL
ttlincr	Increment TTL before packet is send.	OPTIONAL

RELATED COMMANDS:

- | | |
|-------------------|--|
| ip mcast rtdelete | Delete a multicast route from the multicast routing table. |
| ip mcast rtlist | Display the multicast routing table. |
| ip mcast flush | Flush the multicast routing table. |

ip mcast rtdelete

Delete a multicast route from the multicast routing table.

SYNTAX:

```
ip mcast rtdelete    srcintf = <string>
                     [src = <ip-address>]
                     grp = <ip-address>
                     dstintf = <string>
```

where:

srcintf	The source IP interface.	REQUIRED
src	The source IP address.	OPTIONAL
grp	The multicast group IP address.	REQUIRED
dstintf	The destination IP interface.	REQUIRED

RELATED COMMANDS:

- ip mcast rtadd** Add a multicast route to the multicast routing table.
- ip mcast rtlist** Display the multicast routing table.
- ip mcast flush** Flush the multicast routing table.

ip mcast rtlist

Display the multicast routing table.

SYNTAX:

```
ip mcast rtlist [expand = <{disabled | enabled}>]
```

where:

expand	Enable or disable expanded listing. The default is disabled .	OPTIONAL
--------	---	----------

RELATED COMMANDS:

ip mcast rtadd	Add a multicast route to the multicast routing table.
ip mcast rtdelete	Delete a multicast route from the multicast routing table.
ip mcast flush	Flush the multicast routing table.

ip mcast flush

Flush the multicast routing table.

SYNTAX:

```
ip mcast flush
```

RELATED COMMANDS:

ip mcast rtadd	Add a multicast route to the multicast routing table.
ip mcast rtdelete	Delete a multicast route from the multicast routing table.
ip mcast rtlist	Display the multicast routing table.

ip rtadd

Add a route to the routing table.

SYNTAX:

```
ip rtadd dst = <ip-address>
    [dstmsk = <ip-mask(dotted or cidr)>]
    [label = <string>]
    [gateway = <ip-address>]
    [intf = <string>]
    [srcintf = <string>]
    [metric = <number{0-255}>]
```

where:

dst	The destination IP address(es) for this route.	REQUIRED
	Note Supports ip/mask notation.	
dstmsk	The destination IP address mask.	OPTIONAL
label	The name of the label.	OPTIONAL
gateway	The IP address of the next hop (direct connected gateway or extended route).	OPTIONAL
	Note The parameters gateway and intf are mutually exclusive.	
intf	Only for special interface routes: the outgoing IP interface name.	OPTIONAL
	Note The parameters gateway and intf are mutually exclusive.	
srcintf	Use this interface for source address selection.	OPTIONAL
metric	The metric for this route (weight factor). The lower the metric, the higher the weight. The default is 0 .	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 label=Interactive 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 Interactive 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 | Delete a route from the routing table. |
| ip rtlist | Display the routing table. |

ip rtdelete

Delete a route from the routing table.

SYNTAX:

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

where:

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

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 Interactive   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 label=Interactive 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 | Display the routing table. |

ip rtlist

Display the routing table.

SYNTAX:

```
ip rtlist [expand = <{disabled | enabled}>]
```

where:

expand	Enable or disable expanded listing. The default is disabled .	OPTIONAL
--------	---	----------

EXAMPLE:

```
=>ip rtlist
      Destination Label          Gateway  Interface  Metric  Status
      10.0.0.138/32            127.0.0.1  loop      0        [UP]
      127.0.0.1/32            127.0.0.1  loop      0        [UP]
      10.0.0.0/24              10.0.0.138 eth0      0        [UP]
=>ip rtlist expand=enabled
      Destination Label          Gateway  Interface  Metric  Status  Source-selection
      10.0.0.138/32            127.0.0.1  loop      0        [UP]    default    127.0.0.1
      127.0.0.1/32            127.0.0.1  loop      0        [UP]    default    127.0.0.1
      10.0.0.0/24              10.0.0.138 eth0      0        [UP]    default    10.0.0.138
=>
```

RELATED COMMANDS:

ip rtadd	Add a route to the routing table.
ip rtdelte	Delete a route from the routing table.

ip auto flush

Flush the autoIP interfaces.

SYNTAX:

```
ip auto flush
```

ip auto ifadd

Create an autoIP interface.

SYNTAX:

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

where:

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

EXAMPLE:

```
=>ip auto ifadd intf=eth0
=> ip auto iflist
eth0      : [INIT] 0.0.0.0
            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 = 0
            collisions = 0
=>
```

RELATED COMMANDS:

- | | |
|------------------|--------------------------------------|
| ip auto ifdelete | Delete an existing autoIP interface. |
| ip auto iflist | Display the autoIP interfaces. |

ip auto ifattach

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

SYNTAX:

```
ip auto ifattach intf = <string>
```

where:

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

EXAMPLE:

```
=> ip auto iflist
ipsec0    : [INIT] 0.0.0.0
          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 = 0
          collisions = 0

=>ip auto ifattach intf=ipsec0
=> ip auto iflist
ipsec0    : [SELECTING] 169.254.80.236
          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 ifdetach	Release the link-local address for the given autoIP interface.
------------------	--

ip auto ifconfig

Configure an autoIP interface.

SYNTAX:

```
ip auto ifconfig intf = <string>
    [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}>]
```

where:

intf	The name of the autoIP interface to be configured.	REQUIRED
addr	The preferred link-local IP address.	OPTIONAL
poolstart	The start IP address of the link-local address pool. The default is 169.254.1.1 .	OPTIONAL
poolend	The end IP address of the link-local address pool. The default is 169.254.254.254 .	OPTIONAL
netmask	The netmask of the link-local IP address pool. The default is 16 .	OPTIONAL
claim	A number between 0 and 65535. Represents the number of link-local address selection retries before giving up. The 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. The 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. The default is 4 .	OPTIONAL
interval	A number between 1 and 65535 (seconds). Represents the time interval between two ARP probe transmissions. The default is 2 .	OPTIONAL

EXAMPLE:

```
=> ip auto iflist
eth0      : [INIT] 0.0.0.0
    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 = 0
    collisions = 0

=>ip auto ifconfig intf=ipsec0 claim=5 probe=5
=> ip auto iflist
ipsec0    : [INIT] 0.0.0.0
    poolstart = 169.254.1.1  poolend = 169.254.254.254  netmask = 255.255.0.0
    claim : 5  defence : 5  probe : 5  interval : 2 (sec)
    probes sent = 0
    collisions = 0

=>
```

ip auto ifdelete

Delete an existing autoIP interface.

SYNTAX:

```
ip auto ifdelete intf = <string>
```

where:

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

EXAMPLE:

```
=>ip auto iflist
ipsec0    : [SELECTING] 169.254.80.236
          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

=>ip auto ifdelete intf=ipsec0
=>ip auto iflist
=>
```

RELATED COMMANDS:

ip auto ifadd	Create an autoIP interface.
ip auto iflist	Display the autoIP interfaces.

ip auto ifdetach

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

SYNTAX:

```
ip auto ifdetach intf = <string>
```

where:

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

EXAMPLE:

```
=>ip auto iflist
ipsec0 : [SELECTING] 169.254.80.236
    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

=>ip auto ifdetach intf=ipsec0
=>ip auto iflist
ipsec0 : [INIT] 169.254.80.236
    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 = 0
    collisions = 0

=>
```

RELATED COMMANDS:

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

ip auto iflist

Display the autoIP interfaces.

SYNTAX:

```
ip auto iflist [intf = <string>]
```

where:

intf The name of the autoIP interface to be listed. OPTIONAL

Note If not specified, all the autoIP interfaces are shown.

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 an autoIP interface.

ip auto ifdelete Delete an existing autoIP interface.

ip debug httpprobe

Send a HTTP probe.

This HTTP probe will measure the Round Trip Time (RTT) taken to connect and access data from a HTTP server.

SYNTAX:

```
ip debug httpprobe    url = <string>
                      [version = <{1.0 | 1.1}>]
```

where:

url	The Uniform Resource Locator (URL) identifying the HTTP server.	REQUIRED
version	The version of the HTTP server. The default is 1.0 .	OPTIONAL

EXAMPLE:

The first example shows the measured time for a file that was downloaded:

```
=>ip httpprobe url=http://download.winzip.com/wzipse22.exe
DNS Lookup-RTT      = 19 ms
TCP Connect-RTT     = 20 ms
HTTP transaction-RTT = 18772 ms
Total RTT            = 18811 ms
PageSize             = 385712 Bytes
Download speed       = 20.54 KByte/s
=>
```

If the URL of a normal HTML page is used the figures are not so relevant, as shown in the example below:

```
=>ip httpprobe url=http://www.google.be
DNS Lookup-RTT      = 19 ms
TCP Connect-RTT     = 75 ms
HTTP transaction-RTT = 401 ms
Total RTT            = 495 ms
PageSize             = 3448 Bytes
=>
```

DESCRIPTION:

- ▶ **DNS Lookup-RTT:** RTT taken to perform domain name lookup.
- ▶ **TCP Connect-RTT:** RTT taken to perform a TCP connect to the HTTP Server.
- ▶ **HTTP Transaction time-RTT:** RTT taken to send a request and get a response back from the HTTP Server (the probe retrieves the base HTML page only as body, and does not request hyperlinks within this page).

The SpeedTouch will send the HTTP request, receive the reply, and report the RTT statistics (including the size of the page returned).

ip debug sendto

Send UDP packets.

SYNTAX:

```
ip debug sendto  addr = <ip-address>
                [count = <number{1-1000000}>]
                [size = <number{0-20000}>]
                [interval = <number{1-1000000}>]
                [listen = <{disabled | enabled}>]
                [dffield = <{disabled | enabled}>]
                [srcaddr = <ip-address>]
                [srcport = <number{1-65535}>]
                dstport = <number{1-65535}>
                [dstintf = <string>]
```

where:

addr	The destination IP address.	REQUIRED
count	A number between 1 and 1000000. Represents the number of UDP packets to send. The default is 1 .	OPTIONAL
size	A number between 0 and 20000 (bytes). Represents the size of the ping packet(s). The default is 1 .	OPTIONAL
interval	A number between 1 and 10000000 (milliseconds). Represents the intermediate interval between two sent UDP packets. The default is 100 .	OPTIONAL
listen	Listen for incoming ICMP packets (enabled) or only send ICMP packets (disabled). The default is disabled .	OPTIONAL
dffield	Enable or disable setting of the don't fragment flag in the IP headers of the ping.	OPTIONAL
srcaddr	The IP source address to use.	OPTIONAL
srcport	The UDP source port number to use.	OPTIONAL
dstport	The UDP destination port number to send to.	REQUIRED
dstintf	The IP interface name. By specifying the destination interface with the dstintf parameter, a direct send is performed instead of a routed send. This means that the statefull firewall will be bypassed for the outbound packet. As a result, the returning icmp packet can not be associated with an existing udp connection (because there isn't any) and is legally dropped by the firewall. To prevent this packet from being dropped, disable the ICMPchecks and UDPchecks in the firewall configuration.	OPTIONAL

EXAMPLE:

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

ip debug stats

Display IP statistics.

SYNTAX:

```
ip debug stats
```

EXAMPLE:

```
=>ip debug stats
Total datagrams received : 8599
IP header errors : 0
Datagrams forwarded : 23
Datagram forwarding errors : 3
Datagram forwarding resource errors : 0
Total Fragments received : 0
Fragments dropped due to resources or timeouts : 0
Datagrams reassembled : 0
Datagrams fragmented successfully : 0
Datagram fragmentation errors : 0
Total Datagram fragments created successfully : 0
=>
```

ip debug traceconfig

Display/modify the IP stack trace configuration.

SYNTAX:

```
ip debug traceconfig [input = <{none | label | -telnet | -host | -broadcast
| all}>]
[forward = <{none | label | -telnet | -host | -broadcast
| all}>]
[output = <{none | label | -telnet | -host | -broadcast
| all}>]
[drop = <{none | label | -telnet | -host | -broadcast
| all}>]
[path = <{none | label | -telnet | -host | -broadcast
| all}>]
[mode = <{line | dump}>]
[arp = <{none | all}>]
```

where:

input	Define the input packets that will be traced. The default is none .	OPTIONAL
forward	Define the forward packets that will be traced. The default is none .	OPTIONAL
output	Define the output packets that will be traced. The default is none .	OPTIONAL
drop	Define the packet drops that will be traced. The default is all .	OPTIONAL
path	Define the packet that will be path-traced. The default is none .	OPTIONAL
mode	Select the packet dump method. Choose between: <ul style="list-style-type: none"> ▶ line: ▶ dump: The default is line .	OPTIONAL
arp	Define the ARP packets that will be traced. The default is none .	OPTIONAL

EXAMPLE:

```
=>ip debug traceconfig
Input traces : none
Forward traces : none
Output traces : none
Drop traces : all
Path traces : none
Trace mode : line
ARP traces : none
=>
```

EXAMPLE:

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


IPQoS Commands

Introduction

This chapter describes the commands of the **ipqos** command group.

Contents

This chapter covers the following commands:

ipqos config	Configure IPQoS for a given destination interface for the IPQoS queues instantiation.	430
ipqos list	Display the IPQoS configuration.	432
ipqos ef config	Configure the IPQoS Expedited Forwarding (EF) timer for an interface.	433
ipqos ef list	Display the IPQoS EF timers.	434
ipqos ef stats	Display the IPQoS EF timer statistics.	435
ipqos queue clear	Clear the IPQoS statistics.	436
ipqos queue config	Modify the IPQoS subqueue configuration.	437
ipqos queue list	Display the IPQoS subqueue configuration.	439
ipqos queue stats	Show the IPQoS subqueue statistics.	440

ipqos config

Configure IPQoS for a given destination interface for the IPQoS queues instantiation.



When enabling or disabling IPQoS, take the following into account:

- ▶ if the WAN interface (for example PPPoA, IP oA,...) is detached at the time of enabling/disabling IPQoS, then the WAN interface has to be attached in order for the enabling/disabling of IPQoS to take effect.
- ▶ if the WAN interface is attached at the time of enabling/disabling IPQoS, then the WAN interface has to be detached and then re-attached in order for the enabling/disabling of IPQoS to take effect.

SYNTAX:

```
ipqos config dest = <string>
    [state = <{disabled | enabled}>]
    [discard = <{tail | early}>]
    [priority = <{w fq | strict | w rr}>]
    [realtimerate = <number{1-100}>]
    [burstsize = <number{1-128}>]
    [weight1 = <number{1-97}>]
    [weight2 = <number{1-97}>]
    [weight3 = <number{1-97}>]
    [weight4 = <number{1-97}>]
    [maxpackets = <number{0-100}>]
    [maxbytes = <number{0-128}>]
```

where:

dest	The destination interface for the IPQoS queues instantiation. This is an ATM phonebook entry.	REQUIRED
state	Disable or enable IPQoS for the interface. The default is disabled .	OPTIONAL
discard	Determines the packet discard strategy in case of congestion. Choose between: <ul style="list-style-type: none"> ▶ tail: Tail Drop: arriving packets will be dropped as soon as the destination queue is in an overflow state. ▶ early: Early Packet discard: arriving packets will be dropped early according to the BLUE active queue management algorithm. The default is early .	OPTIONAL
priority	Select the subqueue priority algorithm. Choose between: <ul style="list-style-type: none"> ▶ wfq: Weighted Fair Queuing (WFQ) is used for the four AF queues. The realtime queue has priority over the WFQ queues, which have priority over the best-effort queue. ▶ strict: Priority queuing is used. Strict Priority scheduling is used between all queues. The higher the queue number, the higher the priority. ▶ wrr: Weighted Round Robin (WRR) is used for the four AF queues. Each queue is scheduled in turn, with a circular "round" wrapping. The default is wfq .	OPTIONAL

realtimerate	A number between 1 and 100. Represents a percentage of the interface bandwidth for rate-limiting of the Real Time queue. In case of congestion, the Real Time queue will only use this percentage of the interface bandwidth when there is also traffic on the other queues. The default is 80 .	OPTIONAL
burstsize	A number between 1 and 64. Represents the Real Time queue burstsize (in kilobytes) for rate limiting. The default is 2 .	OPTIONAL
weight1	A number between 1 and 97. Represents the weight of queue 1 used for WFQ or WRR. The default is 25 (%).	OPTIONAL
weight2	A number between 1 and 97. Represents the weight of queue 2 used for WFQ or WRR. The default is 25 (%).	OPTIONAL
weight3	A number between 1 and 97. Represents the weight of queue 3 used for WFQ or WRR. The default is 25 (%).	OPTIONAL
weight4	A number between 1 and 97. Represents the weight of queue 4 used for WFQ or WRR. The default is 25 (%).	OPTIONAL
maxpackets	A number between 0 and 250. Represents the maximum number of packets in all IPQoS queues instantiated for one interface. The default is 250 .	OPTIONAL
maxbytes	A number between 0 and 128. Represents the maximum size in kilobytes in all IPQoS queues instantiated for one interface. The default is 56 .	OPTIONAL

EXAMPLE:

```
=>ipqos config dest=atm_pvc_8_35 state=enabled
=>ipqos list
Name      State     Discard   Priority  Size       Size        Rate    Burst    Weights
          (Packets) (KBytes) (%)      (KBytes)   Weights
atm_pvc_0_35 disabled  early    wfq       250       56        80%     2        25% 25% 25% 25%
atm_pvc_8_35 enabled   early    wfq       250       56        80%     2        25% 25% 25% 25%
=>
```

RELATED COMMANDS:

ipqos list Display the IPQoS configuration.

ipqos list

Display the IPQoS configuration.

SYNTAX:

```
ipqos list
```

EXAMPLE:

```
=>ipqos list
Name          State     Discard   Priority  Size      Size        Rate    Burst      Weights
              (Packets) (KBytes)   (%)      (KBytes)  Weights
atm_pvc_0_35 disabled  early     wfq       250       56        80%      2        25% 25% 25% 25%
atm_pvc_8_35 enabled   early     wfq       250       56        80%      2        25% 25% 25% 25%
=>
```

RELATED COMMANDS:

ipqos config Configure IPQoS for a given destination interface for the IPQoS queues instantiation.

ipqos ef config

Configure the IPQoS Expedited Forwarding (EF) timer for an interface.

SYNTAX:

```
ipqos ef config intf = <string>
    [state = <{disabled | enabled}>]
    [timeout = <number{100-10000}>]
    [mtu = <number{68-65535}>]
```

where:

intf	The name of the IP interface.	REQUIRED
state	Enable or disable the IPQoS EF timer for the interface. The default is disabled .	OPTIONAL
timeout	A number between 100 and 10000 milliseconds. Represents the timeout. The default is 1000 .	OPTIONAL
mtu	A number between 68 and 65535. Represents the MTU of the IP interface in case of EF data. The default is 1500 .	OPTIONAL

EXAMPLE:

The example below shows the default configuration:

```
=>ipqos ef list
Interface State      Timeout     MTU
                  (ms)        (bytes)
loop      disabled   1000        65535
Rt_PPPOA2 disabled   1000        1500
eth0      disabled   1000        1500
=>ipqos ef config intf=Rt_PPPOA2 state=enabled
=>ipqos ef list
Interface State      Timeout     MTU
                  (ms)        (bytes)
loop      disabled   1000        65535
Rt_PPPOA2 enabled    1000        1500
eth0      disabled   1000        1500
=>
```

RELATED COMMANDS:

- | | |
|----------------|--|
| ipqos ef list | Display the IPQoS EF timers. |
| ipqos ef stats | Display the IPQoS EF timer statistics. |

ipqos ef list

Display the IPQoS EF timers.

SYNTAX:

```
ipqos ef list
```

EXAMPLE:

```
=>ipqos ef list
Interface  State      Timeout      MTU
           (ms)        (bytes)
loop       disabled    1000        65535
Rt_PPoA2  enabled     1000        1500
eth0      disabled    1000        1500
=>
```

RELATED COMMANDS:

ipqos ef config	Configure the IPQoS Expedited Forwarding (EF) timer for an interface.
ipqos ef stats	Display the IPQoS EF timer statistics.

ipqos ef stats

Display the IPQoS EF timer statistics.

SYNTAX:

```
ipqos ef stats
```

EXAMPLE:

```
=>ipqos ef stats
Interface  State      Remain
                  (ms)
loop        active     900
Rt_PPPOA2  active     900
eth0        disabled   0
=>
```

RELATED COMMANDS:

- | | |
|-----------------|---|
| ipqos ef config | Configure the IPQoS Expedited Forwarding (EF) timer for an interface. |
| ipqos ef list | Display the IPQoS EF timers. |

ipqos queue clear

Clear the IPQoS statistics.

SYNTAX:

```
ipqos queue clear
```

EXAMPLE:

```
=>ipqos queue stats
Name Queue      # packets    # packets    # packets    # packets    # packets    Marking
          added       marked      removed     dropped     replaced
PVC_1  0        1240         0           1240         0           0           0
      1        0           0           0           0           0           0
      2        0           0           0           0           0           0
      3        234         0           234         0           0           0
      4        0           0           0           0           0           0
      5        1345         0           1345        0           0           0
=>ipqos queue clear
=>ipqos queue stats
Name Queue      # packets    # packets    # packets    # packets    # packets    Marking
          added       marked      removed     dropped     replaced
PVC_1  0        0           0           0           0           0           0
      1        0           0           0           0           0           0
      2        0           0           0           0           0           0
      3        0           0           0           0           0           0
      4        0           0           0           0           0           0
      5        0           0           0           0           0           0
=>
```

RELATED COMMANDS:

ipqos queue stats Show the IPQoS subqueue statistics.

ipqos queue config

Modify the IPQoS subqueue configuration.

SYNTAX:

```
ipqos queue config dest = <string>
                    queue = <number{0-5}>
                    [propagate = <{disabled | enabled}>]
                    [ecnmarking = <{disabled | enabled}>]
                    [ackfiltering = <{disabled | enabled}>]
                    [maxpackets = <number{0-250}>]
                    [maxbytes = <number{0-128}>]
                    [respackets = <number{0-250}>]
                    [resbytes = <number{0-128}>]
                    [hold = <number>]
                    [markprob = <number{1-1000}>]
```

where:

dest	The destination interface for the IPQoS queues instantiation. Typically, an ATM phonebook entry.	REQUIRED
queue	A number between 0 and 5. Represents the number of the queue, where: <ul style="list-style-type: none"> ▶ 5 is the Real time queue ▶ 4 is the Assured Forwarding (AF) queue 4 ▶ 3 is the AF queue 3 ▶ 2 is the AF queue 2 ▶ 1 is the AF queue 1 ▶ 0 is the Best Effort queue. 	OPTIONAL
propagate	Higher priority packets will be queued in a lower priority queue, instead of being dropped, as soon as the destination queue is in overflow state. The packet will be put in a lower priority queue only once. Choose between disabled or enabled. The default is disabled .	OPTIONAL
	Note The propagate flag for the lowest priority subqueue (the Best Effort queue) has no meaning.	
ecnmarking	Enable Explicit Congestion Notification (ECN) for IP packets in this subqueue (enabled) or not (disabled). The default is disabled .	OPTIONAL
ackfiltering	Enable filtering of TCP ACK packets (enabled) or not (disabled). The default is disabled .	OPTIONAL
maxpackets	A number between 0 and 250. Represents the maximum number of packets in this queue. The default is 0 for the Real time queue and 100 for the other queues. Note 0 means that a maximum size is not enforced.	OPTIONAL

maxbytes	A number between 0 and 128. Represents the maximum size in kilobytes of this queue. The default is 0 for the Real time queue and 20 for the other queues.	OPTIONAL
Note	0 means that a maximum size is not enforced.	
respackets	A number between 0 and 250. Represents the reserved number of packets in this queue. The default is 30 for the Real time queue and 13 for the other queues.	OPTIONAL
resbytes	A number between 0 and 128 Represents the reserved size in kilobytes of this queue. The default is 12 for the Real time queue and 4 for the other queues.	OPTIONAL
hold	A number (of microseconds). Represents the hold time in microseconds for early discard strategy. The default is 50000 .	OPTIONAL
markprob	A number between 1 and 1000. Represents the maximum packet marking probability in parts per mille for early discard strategy. The default is 1000 .	OPTIONAL

EXAMPLE:

Name	Queue	Propagate	ECN	AckFilter	Size (Packets)	Size (KBytes)	Reserved (Packets)	Reserved (KBytes)	Holddate (usecs)
Markprob									
atm_pvc_0_35	0	disabled	disabled	disabled	100	20	13	4	50000
1000	1	disabled	disabled	disabled	100	20	13	4	50000
1000	2	disabled	disabled	disabled	100	20	13	4	50000
1000	3	disabled	disabled	disabled	100	20	13	4	50000
1000	4	disabled	disabled	disabled	100	20	13	4	50000
1000	5	disabled	disabled	disabled	0	0	30	12	50000
atm_pvc_8_35	0	disabled	disabled	disabled	100	20	13	4	50000
1000	1	disabled	disabled	disabled	100	20	13	4	50000
1000	2	disabled	disabled	disabled	100	20	13	4	50000
1000	3	disabled	disabled	disabled	100	20	13	4	50000
1000	4	disabled	disabled	disabled	100	20	13	4	50000
1000	5	disabled	disabled	disabled	0	0	30	12	50000

RELATED COMMANDS:

ipqos queue list Display the IPQoS subqueue configuration.

ipqos queue list

Display the IPQoS subqueue configuration.

SYNTAX:

```
ipqos queue list [dest = <string>]
```

where:

dest	The destination interface for the IPQoS queues instantiation. This is an ATM phonebook entry.	OPTIONAL
------	--	----------

Note If not specified, the IPQoS subqueue configuration for all the interfaces will be shown.

EXAMPLE:

Name	Queue	Propagate	ECN	AckFilter	Size (Packets)	Size (KBytes)	Reserved (Packets)	Reserved (KBytes)	Holddate (usecs)
Markprob									
atm_pvc_0_35	0	disabled	disabled	disabled	100	20	13	4	50000
1000	1	disabled	disabled	disabled	100	20	13	4	50000
1000	2	disabled	disabled	disabled	100	20	13	4	50000
1000	3	disabled	disabled	disabled	100	20	13	4	50000
1000	4	disabled	disabled	disabled	100	20	13	4	50000
1000	5	disabled	disabled	disabled	0	0	30	12	50000
atm_pvc_8_35	0	disabled	disabled	disabled	100	20	13	4	50000
1000	1	disabled	disabled	disabled	100	20	13	4	50000
1000	2	disabled	disabled	disabled	100	20	13	4	50000
1000	3	disabled	disabled	disabled	100	20	13	4	50000
1000	4	disabled	disabled	disabled	100	20	13	4	50000
1000	5	disabled	disabled	disabled	0	0	30	12	50000
=>									

RELATED COMMANDS:

ipqos queue config	Modify the IPQoS subqueue configuration.
--------------------	--

ipqos queue stats

Show the IPQoS subqueue statistics.

SYNTAX:

```
ipqos queue stats [dest = <string>]
```

where:

dest	The destination interface for the IPQoS queues instantiation. This is an ATM phonebook entry.	OPTIONAL
Note If not specified, the IPQoS subqueue statistics for all the interfaces will be shown.		

EXAMPLE:

```
=>ipqos queue stats
Name Queue    # packets   # packets   # packets   # packets   # packets   Marking
      added      marked     removed     dropped     replaced
PVC_1 0        1240       0          1240       0          0          0%
      1        0          0          0          0          0          0%
      2        0          0          0          0          0          0%
      3        234       0          234       0          0          0%
      4        0          0          0          0          0          0%
      5        145       0          145       0          0          0%
=>
```

RELATED COMMANDS:

ipqos queue clear Clear the IPQoS statistics.

Label Commands

Introduction

This chapter describes the commands of the **label** command group.

Contents

This chapter covers the following commands:

label add	Create a new label.	442
label delete	Delete a label.	443
label flush	Flush all labels.	444
label list	Display the labels.	445
label modify	Modify a label configuration.	446
label chain add	Add a new label chain.	448
label chain delete	Delete a label chain.	449
label chain flush	Flush all label chains.	450
label chain list	Display a list of chains.	451
label rule add	Add a label rule.	452
label rule delete	Delete a label rule.	455
label rule flush	Flush all label rules.	457
label rule list	Display a list of label rules.	458
label rule modify	Modify a label rule.	460
label rule debug clear	Clear the label rule statistics.	461
label rule debug stats	Display the label rule statistics.	462
label rule debug traceconfig	Display or modify the rule trace configuration.	463

Label Commands

label add

Create a new label.

SYNTAX:

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

where:

name	The name of the label to be added.	REQUIRED
------	------------------------------------	----------

EXAMPLE:

```
=>label list
Name      Class     Def      Ack      Bidirect Inherit Tosmark Type Value Ttlover Ttl
Use Trace
DSCP      overwrite dscp    defclass disabled disabled disabled tos  0      disabled 0
0      disabled
Interactive increase 8       8       disabled disabled disabled tos  0      disabled 0
0      disabled
Management increase 12      12      disabled disabled disabled tos  0      disabled 0
0      disabled
Video      increase 10       10      disabled disabled disabled tos  0      disabled 0
0      disabled
VoIP       overwrite 14       14      enabled  enabled  disabled tos  0      disabled 0
0      disabled
default    increase default  prioritize disabled disabled disabled tos  0      disabled 0
0      disabled
=>label add name=myLABEL
=>label list
Name      Class     Def      Ack      Bidirect Inherit Tosmark Type Value Ttlover Ttl
Use Trace
DSCP      overwrite dscp    defclass disabled disabled disabled tos  0      disabled 0
0      disabled
Interactive increase 8       8       disabled disabled disabled tos  0      disabled 0
0      disabled
Management increase 12      12      disabled disabled disabled tos  0      disabled 0
0      disabled
Video      increase 10       10      disabled disabled disabled tos  0      disabled 0
0      disabled
VoIP       overwrite 14       14      enabled  enabled  disabled tos  0      disabled 0
0      disabled
default    increase default  prioritize disabled disabled disabled tos  0      disabled 0
0      disabled
myLABEL   ignore    0        0       disabled disabled disabled tos  0      disabled 0
0      disabled
=>
```

RELATED COMMANDS:

label delete	Delete a label.
label list	Display the labels.

label delete

Delete a label.

SYNTAX:

```
label delete name = <string>
    [force = <{disabled | enabled}>]
```

where:

name	The name of the label to be deleted.	REQUIRED
force	Force delete and cleanup references even when the label is still in use (enabled) or not (disabled). The default is disabled .	OPTIONAL

EXAMPLE:

```
=>label list
Name      Class     Def      Ack      Bidirect Inherit Tosmark Type Value   Ttlover Ttl
Use      Trace
DSCP      overwrite dscp     defclass disabled disabled disabled tos  0       disabled 0
0        disabled
Interactive increase 8        8        disabled disabled disabled tos  0       disabled 0
0        disabled
Management increase 12       12       disabled disabled disabled tos  0       disabled 0
0        disabled
Video      increase 10        10       disabled disabled disabled tos  0       disabled 0
0        disabled
VoIP       overwrite 14        14       enabled  enabled  disabled tos  0       disabled 0
0        disabled
default    increase default  prioritize disabled disabled disabled tos  0       disabled 0
0        disabled
myLABEL   ignore   0         0        disabled disabled disabled tos  0       disabled 0
0        disabled
=>label delete name=myLABEL force=yes
=>label list
Name      Class     Def      Ack      Bidirect Inherit Tosmark Type Value   Ttlover Ttl
Use      Trace
DSCP      overwrite dscp     defclass disabled disabled disabled tos  0       disabled 0
0        disabled
Interactive increase 8        8        disabled disabled disabled tos  0       disabled 0
0        disabled
Management increase 12       12       disabled disabled disabled tos  0       disabled 0
0        disabled
Video      increase 10        10       disabled disabled disabled tos  0       disabled 0
0        disabled
VoIP       overwrite 14        14       enabled  enabled  disabled tos  0       disabled 0
0        disabled
default    increase default  prioritize disabled disabled disabled tos  0       disabled 0
0        disabled
=>
```

RELATED COMMANDS:

- | | |
|-------------------|---------------------|
| label add | Create a new label. |
| label list | Display the labels. |

label flush

Flush all labels.



The flush command does not impact previously saved configurations.

SYNTAX:

```
label flush
```

label list

Display the labels.

SYNTAX:

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

where:

name	The name of the label to be displayed.	OPTIONAL
Note	If not specified, all the labels will be displayed.	

EXAMPLE:

```
=>label list
Name      Class     Def      Ack      Bidirect Inherit  Tosmark  Type   Value   Ttlover  Ttl
Use      Trace
DSCP      overwrite dscp      defclass  disabled disabled disabled tos  0       disabled 0
0        disabled
Interactive increase 8        8        disabled disabled disabled tos  0       disabled 0
0        disabled
Management increase 12      12      disabled disabled disabled tos  0       disabled 0
0        disabled
Video      increase 10        10      disabled disabled disabled tos  0       disabled 0
0        disabled
VoIP      overwrite 14        14      enabled  enabled  disabled tos  0       disabled 0
0        disabled
default    increase default  prioritize disabled disabled disabled tos  0       disabled 0
0        disabled
=>
=>label list name=Interactive
Name      Class     Defclass Ackclass Ttlover  Tt      Tosmark  Tos      Use      Trace
Interactive increase 8        8        disabled disabled disabled tos  0       disabled 0
0        disabled
=>
```

RELATED COMMANDS:

label add	Create a new label.
label delete	Delete a label.

label modify

Modify a label configuration.

SYNTAX:

```
label modify name = <string>
    [classification = <{ignore | overwrite | increase}>]
    [defclass = <number{0-15} | dscp | default>]
    [ackclass = <number{0-15} | defclass | prioritize>]
    [bidirectional = <{disabled | enabled}>]
    [inheritance = <{disabled | enabled}>]
    [tosmarking = <{disabled | enabled}>]
    [tos = <number{0-255}>]
    [dscp = <{DSCP name} | <number>>]
    [precedence = <{IP precedence type} | <number>>]
    [trace = <{disabled | enabled}>]
```

where:

name	The name of the label to be configured.	REQUIRED
classification	Select the method of classification, in other words, determine what the Layer 3 class assignment must do with the priority of the data packet (as set by Layer 2). Choose between: <ul style="list-style-type: none"> ▶ ignore: Ignore the class parameters defclass and ackclass, but use the class as set by Layer 2 (VLAN user priority, ATM QoS). ▶ overwrite: Change the class to defclass and ackclass, overwriting the value set by Layer 2 (VLAN user priority, ATM QoS). ▶ increase: Change the class according to defclass and ackclass, but only if the defclass value is higher than the class value already set by Layer 2. The default is ignore .	OPTIONAL
Note	The class as set by Layer 2 is derived from: <ul style="list-style-type: none"> ▶ the VLAN user priority (in case of VLAN or priority tagged frames) ▶ the ATM PVC QoS class (in case the packet is received from an ATM PVC). For non-VLAN frames or non-PVC received data, Layer 2 sets the internal class (priority) to 4 by default.	
defclass	The default priority class of the assigned connection. Choose between: <ul style="list-style-type: none"> ▶ A number between 0 and 15 ▶ dscp ▶ default. The default is 0 .	OPTIONAL
ackclass	The priority class of the ACK segments of the TCP connection. Choose between: <ul style="list-style-type: none"> ▶ A number between 0 and 15 ▶ prioritize ▶ defclass. The default is 0 .	OPTIONAL

bidirectional	The label applies to the initiator stream, the returning stream (the current connection) as well as to the child connections (enabled) or not (disabled). The default is disabled .	OPTIONAL
inheritance	The label will be copied to all child connection streams in the same direction (enabled) or not (disabled). The default is disabled .	OPTIONAL
tosmarking	Enable or disable tos marking. The default is disabled .	OPTIONAL
tos	A number between 0 and 255. Represents the ToS specification in the IP packet (used for tosmarking). The default is 0 .	OPTIONAL
	Note The parameters tos , precedence and dscp are mutually exclusive.	
dscp	The DSCP in the IP packet (part of tos). Select a DSCP (see " Differentiated Services Code Point (DSCP)" on page 755) or, alternatively, specify the number.	OPTIONAL
	Note The parameters tos , precedence and dscp are mutually exclusive.	
precedence	The precedence in the IP packet (part of tos). Select an IP precedence (see " IP Precedence" on page 754) or, alternatively, specify the number.	OPTIONAL
	Note The parameters tos , precedence and dscp are mutually exclusive.	
trace	Enable or disable IP tracing for this label. The default is disabled .	OPTIONAL

EXAMPLE:

```
=>label list
Name      Class     Def   Ack     Bidirect Inherit Tosmark Type  Value  Use   Trace
DSCP      overwrite dscp  defclass disabled disabled disabled tos   0      0     0     dis
abled
Interactive increase 8      8      disabled disabled disabled tos   0      0     0     dis
abled
Management increase 12     12     disabled disabled disabled tos   0      0     0     dis
abled
Video     increase 10      10     disabled disabled disabled tos   0      0     0     dis
abled
VoIP      overwrite 14     14     enabled  enabled  disabled tos   0      0     0     dis
abled
default   increase  default prioritize disabled disabled disabled tos   0      0     0     dis
abled
myLABEL   ignore    0      0      disabled disabled disabled tos   0      0     0     dis
abled
=>label modify name=myLABEL classification=increase defclass=7 ackclass=7 bidirectional=enabled
inheritance=enabled tosmarking=enabled tos=234 trace=enabled
=>label list
Name      Class     Def   Ack     Bidirect Inherit Tosmark Type  Value  Use   Trace
DSCP      overwrite dscp  defclass disabled disabled disabled tos   0      0     0     dis
abled
Interactive increase 8      8      disabled disabled disabled tos   0      0     0     dis
abled
Management increase 12     12     disabled disabled disabled tos   0      0     0     dis
abled
Video     increase 10      10     disabled disabled disabled tos   0      0     0     dis
abled
VoIP      overwrite 14     14     enabled  enabled  disabled tos   0      0     0     dis
abled
default   increase  default prioritize disabled disabled disabled tos   0      0     0     dis
abled
myLABEL   increase 7       7      enabled  enabled  enabled  tos   234   0     0     ena
=>
```

label chain add

Add a new label chain.

SYNTAX:

```
label chain add  chain = <string>
```

where:

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

EXAMPLE:

```
=>label chain list

Chains
=====
Name          Description
-----
routing_labels      system
rt_user_labels     user
rt_default_labels   user
qos_labels         system
qos_user_labels    user
qos_default_labels user
=>label chain add chain=myChain
=>label chain list

Chains
=====
Name          Description
-----
routing_labels      system
rt_user_labels     user
rt_default_labels   user
qos_labels         system
qos_user_labels    user
qos_default_labels user
myChain           user
=>
```

RELATED COMMANDS:

label chain delete	Delete a label chain.
label chain list	Display a list of chains.

label chain delete

Delete a label chain.

SYNTAX:

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

where:

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

EXAMPLE:

```
=>label chain list

Chains
=====
Name          Description
-----
routing_labels      system
rt_user_labels     user
rt_default_labels   user
qos_labels         system
qos_user_labels    user
qos_default_labels user
myChain           user
=>label chain delete chain=myChain
=>label chain list

Chains
=====
Name          Description
-----
routing_labels      system
rt_user_labels     user
rt_default_labels   user
qos_labels         system
qos_user_labels    user
qos_default_labels user
=>
```

RELATED COMMANDS:

label chain add	Add a new label chain.
label chain list	Display a list of chains.

label chain flush

Flush all label chains.

SYNTAX:

```
label chain flush
```

label chain list

Display a list of chains.

SYNTAX:

```
label chain list [format = <{pretty | cli}>]
```

where:

format	Select the output format of the list. Choose between: <ul style="list-style-type: none">▶ pretty: the configuration is shown as intuitive output in clear text.▶ cli: the configuration is shown via the CLI commands configuration. The default is pretty .	OPTIONAL
--------	---	----------

EXAMPLE:

```
=>label chain list

Chains
=====
Name          Description
-----
routing_labels      system
rt_user_labels     user
rt_default_labels   user
qos_labels         system
qos_user_labels    user
qos_default_labels user

=>label chain list format=cli

:label chain add chain=rt_user_labels
:label chain add chain=rt_default_labels
:label chain add chain=qos_user_labels
:label chain add chain=qos_default_labels
=>
```

RELATED COMMANDS:

label chain add	Add a new label chain.
label chain delete	Delete a label chain.

label rule add

Add a label rule.

SYNTAX:

```
label rule add chain = <chain name>
    [index = <number>]
    [name = <string>]
    [clink = <chain name>]
    [srcintf [!] = <string>]
    [srcip [!] = <{ip address | private}>]
    [dstip [!] = <{ip address | private}>]
    [serv [!] = <{service name}>]
    [log = <{disabled | enabled}>]
    [state = <{disabled | enabled}>]
    label = <string>
```



If a value is preceded by a "!", it means NOT.
For example "srcintf!=wan" means "if srcintf is different from WAN".

where:

chain	The name of the chain in which the rule must be inserted.	REQUIRED
index	The index number of the rule before which the new rule must be added.	OPTIONAL
Tip	Use the command :label rule list to obtain the index number of the applicable rule.	
name	The name of the new rule.	OPTIONAL
clink	The name of the chain to be parsed when this rule applies.	OPTIONAL
srcintf	The name of the source interface expression.	OPTIONAL
srcip	The name of the source IP expression.	OPTIONAL
dstip	The name of the destination IP expression.	OPTIONAL
serv	The name of the service expression.	OPTIONAL
log	Disable or enable logging when this rule applies. The default is disabled .	OPTIONAL
state	Disable or enable this rule. The default is enabled .	OPTIONAL
label	Choose between: ▶ None ▶ link (when clink is used) ▶ label name.	REQUIRED

EXAMPLE:

```
=>label rule list

Rules (flags: C=Constant, D=Dynamic, E=Enable, L=Log)
=====
Chain          Nr.  Flags   Rule
-----
routing_labels    1    CDE      : link      rt_user_labels
                  2    CDE      : link      rt_default_labels
qos_labels        1    CDE      : link      qos_user_labels
                  2    CDE      : link      qos_default_labels
qos_default_labels 1    C E     : VoIP      sip *.* > *.*
                  2    C E     : VoIP      h323 *.* > *.*
                  3    C E     : Interactive telnet *.* > *.*
                  4    C E     : Interactive smtp *.* > *.*
                  5    C E     : Interactive imap4-ssl *.* > *.*
                  6    C E     : Interactive imap3 *.* > *.*
                  7    C E     : Interactive imap *.* > *.*
                  8    C E     : Interactive imaps *.* > *.*
                  9    C E     : Interactive pop3s *.* > *.*
                  10   C E     : Interactive pop3 *.* > *.*
                  11   C E     : Interactive pop2 *.* > *.*
                  12   C E     : Interactive httppproxy *.* > *.*
                  13   C E     : Interactive http *.* > *.*
                  14   C E     : Interactive https *.* > *.*
                  15   C E     : Interactive esp *.* > *.*
                  16   C E     : Interactive ah *.* > *.*
                  17   C E     : Management dns *.* > *.*
                  18   C E     : Management ike *.* > *.*
                  19   E       : Management icmp *.* > *.*
                  20   C E     : Video      rtsp *.* > *.*
                  21   C E     : Video      igmp *.* > *.*
                  22   C E     default   : default   !wan.* > *.*

=>label rule add chain=myChain name=myRule dstip=150.150.150.150 serv=DiffServ log=enabled state=enabled
| label=myLABEL
=>label rule list

Rules (flags: C=Constant, D=Dynamic, E=Enable, L=Log)
=====
Chain          Nr.  Flags   Rule
-----
routing_labels    1    CDE      : link      rt_user_labels
                  2    CDE      : link      rt_default_labels
qos_labels        1    CDE      : link      qos_user_labels
                  2    CDE      : link      qos_default_labels
qos_default_labels 1    C E     : VoIP      sip *.* > *.*
                  2    C E     : VoIP      h323 *.* > *.*
                  3    C E     : Interactive telnet *.* > *.*
                  4    C E     : Interactive smtp *.* > *.*
                  5    C E     : Interactive imap4-ssl *.* > *.*
                  6    C E     : Interactive imap3 *.* > *.*
                  7    C E     : Interactive imap *.* > *.*
                  8    C E     : Interactive imaps *.* > *.*
                  9    C E     : Interactive pop3s *.* > *.*
                  10   C E     : Interactive pop3 *.* > *.*
                  11   C E     : Interactive pop2 *.* > *.*
                  12   C E     : Interactive httppproxy *.* > *.*
                  13   C E     : Interactive http *.* > *.*
                  14   C E     : Interactive https *.* > *.*
                  15   C E     : Interactive esp *.* > *.*
                  16   C E     : Interactive ah *.* > *.*
                  17   C E     : Management dns *.* > *.*
                  18   C E     : Management ike *.* > *.*
                  19   E       : Management icmp *.* > *.*
                  20   C E     : Video      rtsp *.* > *.*
                  21   C E     : Video      igmp *.* > *.*
                  22   C E     default   : default   !wan.* > *.*

myChain           1    C EL    myRule    : myLABEL DiffServ *.* > *.150.150.150.150
=>
```

Label Commands

RELATED COMMANDS:

<code>label rule delete</code>	Delete a label rule.
<code>label rule list</code>	Display a list of label rules.

label rule delete

Delete a label rule.

SYNTAX:

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

where:

chain	The name of the chain in which a rule must be deleted.	REQUIRED
index	The index number of the rule in the chain.	REQUIRED
Tip	Use the command :label rule list to obtain the index number of the applicable rule.	

EXAMPLE:

```
=>label rule list
Rules (flags: C=Constant, D=Dynamic, E=Enable, L=Log)
=====
Chain          Nr.  Flags   Rule
-----
routing_labels    1    CDE      : link       rt_user_labels
                  2    CDE      : link       rt_default_labels
qos_labels        1    CDE      : link       qos_user_labels
                  2    CDE      : link       qos_default_labels
qos_default_labels 1    C E     : VoIP      sip *.* > *.*
                  2    C E     : VoIP      h323 *.* > *.*
                  3    C E     : Interactive telnet *.* > *.*
...
                  19   E       : Management  icmp *.* > *.*
                  20   C E     : Video      rtsp *.* > *.*
                  21   C E     : Video      igmp *.* > *.*
                  22   C E     default    : default   !wan.* > *.*
myChain          1    C EL    myRule
: myLABEL DiffServ *.* > *.150.150.150.150
=>label rule delete chain=myChain index=1
=>label rule list
Rules (flags: C=Constant, D=Dynamic, E=Enable, L=Log)
=====
Chain          Nr.  Flags   Rule
-----
routing_labels    1    CDE      : link       rt_user_labels
                  2    CDE      : link       rt_default_labels
qos_labels        1    CDE      : link       qos_user_labels
                  2    CDE      : link       qos_default_labels
qos_default_labels 1    C E     : VoIP      sip *.* > *.*
                  2    C E     : VoIP      h323 *.* > *.*
                  3    C E     : Interactive telnet *.* > *.*
...
                  19   E       : Management  icmp *.* > *.*
                  20   C E     : Video      rtsp *.* > *.*
                  21   C E     : Video      igmp *.* > *.*
                  22   C E     default    : default   !wan.* > *.*
=>
```

Label Commands

RELATED COMMANDS:

label rule add	Add a label rule.
label rule list	Display a list of label rules.

label rule flush

Flush all label rules.

The chains themselves are not removed.



The flush command does not impact previously saved configurations.

SYNTAX:

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

where:

chain	The name of the chain to be flushed.	OPTIONAL
-------	--------------------------------------	----------

Note If not specified, all the rules for all the chains are flushed.

label rule list

Display a list of label rules.

SYNTAX:

```
label rule list [chain = <string>]  
[format = <{pretty | cli}>]
```

where:

chain	The name of the chain for which the rules must be listed.	OPTIONAL
	Note If not specified, all rules for all chains are shown.	
format	Select the output format of the list. Choose between: <ul style="list-style-type: none">▶ pretty: the configuration is shown as intuitive output in clear text.▶ cli: the configuration is shown via the CLI commands configuration. The default is pretty .	OPTIONAL

EXAMPLE:

```
=>label rule list format=cli
:label rule add chain=qos_default_labels index=1 serv=sip log=disabled state=enabled label=VoIP
:label rule add chain=qos_default_labels index=2 serv=h323 log=disabled state=enabled label=VoIP
:label rule add chain=qos_default_labels index=3 serv=telnet log=disabled state=enabled
label=Interactive
:label rule add chain=qos_default_labels index=4 serv=smtp log=disabled state=enabled
label=Interactive
:label rule add chain=qos_default_labels index=5 serv=imap4-ssl log=disabled state=enabled
label=Interactive
:label rule add chain=qos_default_labels index=6 serv=imap3 log=disabled state=enabled
label=Interactive
:label rule add chain=qos_default_labels index=7 serv=imap log=disabled state=enabled
label=Interactive
:label rule add chain=qos_default_labels index=8 serv=imaps log=disabled state=enabled
label=Interactive
:label rule add chain=qos_default_labels index=9 serv=pop3s log=disabled state=enabled
label=Interactive
:label rule add chain=qos_default_labels index=10 serv=pop3 log=disabled state=enabled
label=Interactive
:label rule add chain=qos_default_labels index=11 serv=pop2 log=disabled state=enabled
label=Interactive
:label rule add chain=qos_default_labels index=12 serv=httpproxy log=disabled state=enabled
label=Interactive
:label rule add chain=qos_default_labels index=13 serv=http log=disabled state=enabled
label=Interactive
:label rule add chain=qos_default_labels index=14 serv=https log=disabled state=enabled
label=Interactive
:label rule add chain=qos_default_labels index=15 serv=esp log=disabled state=enabled
label=Interactive
:label rule add chain=qos_default_labels index=16 serv=ah log=disabled state=enabled
label=Interactive
:label rule add chain=qos_default_labels index=17 serv=dns log=disabled state=enabled
label=Management
:label rule add chain=qos_default_labels index=18 serv=ike log=disabled state=enabled
label=Management
:label rule add chain=qos_default_labels index=19 serv=icmp log=disabled state=enabled
label=Management
:label rule add chain=qos_default_labels index=20 serv=rtsp log=disabled state=enabled label=Video
:label rule add chain=qos_default_labels index=21 serv=igmp log=disabled state=enabled label=Video
:label rule add chain=qos_default_labels index=22 name=default srcintf!=wan log=disabled
state=enabled
| label=default
=>
```

RELATED COMMANDS:

- | | |
|--------------------------|----------------------|
| label rule add | Add a label rule. |
| label rule delete | Delete a label rule. |

label rule modify

Modify a label rule.

SYNTAX:

```
label rule modify chain = <chain name>
                  index = <number>
                  [newindex = <number>]
                  [name = <string>]
                  [clink = <chain name>]
                  [srcintf [!] = <string>]
                  [srcip [!] = <{ip address | private}>]
                  [dstip [!] = <{ip address | private}>]
                  [serv [!] = <{service name}>]
                  [log = <{disabled | enabled}>]
                  [state = <{disabled | enabled}>]
                  [label = <string>]
```



If a value is preceded by a "!", it means "NOT".

For example "srcintf!=wan" means "if srcintf is different from WAN".

where:

chain	The name of the chain which contains the rule.	REQUIRED
index	The number of the rule in the chain.	REQUIRED
newindex	The new number of the rule in the chain.	OPTIONAL
name	The name of the new rule.	OPTIONAL
clink	The name of the chain to be parsed when this rule applies.	OPTIONAL
srcintf	The name of the source interface expression.	OPTIONAL
srcip	The name of the source IP expression.	OPTIONAL
dstip	The name of the destination IP expression.	OPTIONAL
serv	The name of the device expression.	OPTIONAL
log	Disable or enable logging when this rule applies.	OPTIONAL
state	Disable or enable this rule.	OPTIONAL
label	Choose between: ▶ None ▶ link (when clink is used) ▶ label name.	OPTIONAL

label rule debug clear

Clear the label rule statistics.

SYNTAX:

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

where:

chain	The name of the chain in which the rule is to be found.	OPTIONAL
Note If not specified, the statistics for all the rules in all chains will be cleared.		
index	The index number (determined by the position) of the rule in the chain.	OPTIONAL
Note If not specified, the statistics for all the rules in a chain will be cleared.		

EXAMPLE:

```
=>label rule debug stats chain=qos_labels  
chain index packets bytes  
-----  
qos_labels 1 0 0  
2 203 19146  
=>label rule debug clear chain=qos_labels  
=>label rule debug stats  
chain index packets bytes  
-----  
routing_labels 1 0 0  
2 0 0  
qos_labels 1 0 0  
2 0 0  
qos_default_labels 1 0 0  
2 0 0  
3 4 168  
4 0 0  
5 0 0  
6 0 0  
7 0 0  
8 0 0  
9 0 0  
10 0 0  
11 0 0  
12 0 0  
13 10 480  
14 3 144  
15 0 0  
16 0 0  
17 5 329  
18 0 0  
19 4 452  
20 177 17573  
=>
```

RELATED COMMANDS:

label rule debug stats Display the label rule statistics.

label rule debug stats

Display the label rule statistics.

SYNTAX:

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

where:

chain	The name of the chain for which the statistics must be shown.	OPTIONAL
-------	---	----------

Note If not specified, the statistics for the rules applicable to all chains are shown.

index	The index number of the rule for which the statistics must be shown.	OPTIONAL
-------	--	----------

Tip Use the command :label rule list to obtain the index number of the applicable rule.

Note If not specified, the statistics for all rules applicable to the specified chain are shown.

EXAMPLE:

```
=>label rule debug stats chain=qos_labels
chain                         index    packets      bytes
-----
qos_labels                     1          0          0
                                2         167        15690
=>
=>label rule debug stats chain=qos_default_labels index=20
chain                         index    packets      bytes
-----
qos_default_labels             19         165        16421
=>
```

RELATED COMMANDS:

label rule debug clear	Clear the label rule statistics.
------------------------	----------------------------------

label rule debug traceconfig

Display or modify the rule trace configuration.

SYNTAX:

```
label rule debug traceconfig [trace = <{disabled | enabled}>]
```

where:

trace	Disable or enable rule traces. The default is disabled .	OPTIONAL
-------	--	----------

EXAMPLE:

```
=>label rule debug traceconfig  
:label rule debug traceconfig state=disabled  
=>
```


Language Commands

Introduction

This chapter describes the commands of the **Language** command group.

Contents

This chapter covers the following commands:

language config	Select a language.	466
language delete	Delete one or all language archives.	467
language list	List the available language archives.	468

language config

Select a language.

SYNTAX:

```
language config [language = <string>]  
[complete = <{yes | no}>]
```

where:

language	Language code: OSI language code (2 chars) for language. Example: en for english.	OPTIONAL
complete	Enable translation for expert pages. The default is yes .	OPTIONAL

EXAMPLE:

```
=>language config  
language      : en  
complete     : no  
=>
```

RELATED COMMANDS:

language delete	Delete one or all language archives.
language list	List the available language archives.

language delete

Delete one or all language archives.

SYNTAX:

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

where:

file	The filename of the language archive to be removed.	OPTIONAL
all	Remove all languages archives (yes) or not (no). The default is no .	OPTIONAL

RELATED COMMANDS:

language config	Select a language.
language list	List the available language archives.

language list

List the available language archives.

SYNTAX:

```
language list
```

EXAMPLE:

```
=>language list
CODE LANGUAGE      VERSION      FILENAME
en* English        5.3.0.10.0  <system>
=>
```



The currently selected language is indicated by a "*" next to the OSI language code.

RELATED COMMANDS:

[language config](#)

Select a language.

[language delete](#)

Delete one or all language archives.

MBUS Commands

Introduction

This chapter describes the commands of the **mbus** command group.

Contents

This chapter covers the following commands:

mbus config	Modify mbus parameters.	470
mbus client config	Modify client parameters.	471
mbus client exec	Execute mbus command.	472
mbus client register	Register cli client to mbus.	474
mbus client test	Test functionality invoked.	475
mbus debug stats	Display mbus statistics.	476
mbus debug traceconfig	Modify mbus trace settings.	477
mbus listobjects	Display the object instances.	478
mbus listtypes	Display the registered objecttypes.	479
mbus loadobjects	Load object instances of registered objecttypes.	480
mbus unloadobjects	Unload object instances.	481

mbus config

Modify mbus parameters.

SYNTAX:

```
mbus config [autoload = <{disabled | enabled}>]
```

where:

autoload	Enable or disable autoload object tree at boot time. The default is enabled .	OPTIONAL
----------	---	----------

mbus client config

Modify client parameters.

SYNTAX:

```
mbus client config [writelock = <{enabled | disabled}>]
                    [wlgetimeout = <number>]
                    [wlidletimeout = <number>]
                    [path = <quoted string>]
                    [type = <idpath | keypath>]
```

where:

writelock	Enable or disable mbus writelock. The default is enabled .	OPTIONAL
wlgetimeout	Writelock get timeout in seconds. <forever=-1, *default=nowait=0>	OPTIONAL
wlidletimeout	Writelock idle timeout in seconds. <forever=-1, *default=nowait=30>	OPTIONAL
path	Fully qualified mbus path.	OPTIONAL
type	Path type. The default is idpath .	OPTIONAL

RELATED COMMANDS:

mbus client exec	Execute mbus command.
mbus client register	Register cli client to mbus.
mbus client test	Test functionality invoked.

mbus client exec

Execute mbus command.

SYNTAX:

```
mbus client exec      cmd = <{addobject | deleteobject | getparamattributes |
                           getparamcount | getparamnames | getparamvalues | getpath
                           | setparamvalue}>
                           [param = <quoted string>]
                           [value = <quoted string>]
                           [gettextype = <{object | parameter | path}>]
                           [depth = <nnumber>]
                           [processcmd = <{disabled | enabled}>]
                           [readcommitted = <{disabled | enabled}>]
                           [onerrorrollback = <{disabled | enabled}>]
```

where:

cmd	Mbus command name. Choose between: <ul style="list-style-type: none">▶ <i>addobject</i>▶ <i>deleteobject</i>▶ <i>getparamattributes</i>▶ <i>getparamcount</i>▶ <i>getparamnames</i>▶ <i>getparamvalues</i>▶ <i>getpath</i>▶ <i>setparamvalue</i>	REQUIRED
param	Parameter name.	OPTIONAL
value	Parameter value to set.	OPTIONAL
gettextype	Get type elements. Choose between: <ul style="list-style-type: none">▶ <i>object</i>▶ <i>parameter</i>▶ <i>path</i> The default is <i>parameter</i> .	OPTIONAL
depth	Number of levels to recurse, default=-1 (all).	OPTIONAL
processcmd	Enable or disable the process SET_PARAMVALUE command flag. The default is <i>enabled</i> .	OPTIONAL
readcommitted	Enable or disable the read committed data only (MBUS_CMD_GET_PARAMVALUES) flag. The default is <i>enabled</i> .	OPTIONAL
onerrorrollback	Enable or disable the on error rollback flag. The default is <i>enabled</i> .	OPTIONAL

RELATED COMMANDS:

mbus client config	Modify client parameters.
mbus client register	Register cli client to mbus.
mbus client test	Test functionality invoked.

mbus client register

Register cli client to mbus.

SYNTAX:

```
mbus client register
```

RELATED COMMANDS:

mbus client config	Modify client parameters.
mbus client exec	Execute mbus command.
mbus client test	Test functionality invoked.

mbus client test

Test functionality invoked.

SYNTAX:

```
mbus client test      [path = <quoted string>]
                      [type = <{idpath | keypath}>]
```

where:

path	Fully qualified mbus path.	OPTIONAL
type	Path type. Choose between: ▶ <i>idpath</i> ▶ <i>keypath</i> . The default is <i>idpath</i> .	OPTIONAL

EXAMPLE:

```
=>mbus client test
pool name           bytes cur,max,avail (max.%)   count cur,max,avail      ref
cur,max,avail
-----
mbus_dynstr         2004,    2004,    4920  ( 40%)   167,     167,    410
0,      0,      -
- strings buffer    2553,    2563,    8192  ( 31%)
-----
Total usage          4557,    4567,   13112  ( 34%)
=>
```

RELATED COMMANDS:

mbus client config	Modify client parameters.
mbus client exec	Execute mbus command.
mbus client register	Register cli client to mbus.

mbus debug stats

Display mbus statistics.

SYNTAX:

```
mbus debug stats
```

RELATED COMMANDS:

[mbus debug traceconfig](#) Modify mbus trace settings.

mbus debug traceconfig

Modify mbus trace settings.

SYNTAX:

```
mbus debug traceconfig [level = <number{0-4}>]
```

where:

level	A number between 0 and 4. Represents the mbus trace level.	OPTIONAL
-------	---	----------

RELATED COMMANDS:

mbus debug stats Display mbus statistics.

mbus listobjects

Display the object instances.

SYNTAX:

```
mbus listobjects      [path = <quoted string>]
                      [type = <{idpath | keypath}>]
                      [output = <{list | tree}>]
                      [expand = <{disabled | enabled}>]
```

where:

path	Fully qualified mbus path to list.	OPTIONAL
type	Path type. Choose between: <ul style="list-style-type: none"> ▶ <i>idpath</i> ▶ <i>keypath</i>. The default is <i>idpath</i> .	OPTIONAL
output	List output type. Choose between: <ul style="list-style-type: none"> ▶ <i>list</i> ▶ <i>tree</i>. The default is <i>tree</i> .	OPTIONAL
expand	Enable or disable details. The default is <i>disabled</i> .	OPTIONAL

RELATED COMMANDS:

mbus listtypes Display the registered objecttypes.

mbus listtypes

Display the registered objecttypes.

SYNTAX:

```
mbus listtypes      [path = <quoted string>]  
                   [expand = <{disabled | enabled}>]
```

where:

path	Fully qualified mbus path to list.	OPTIONAL
expand	Enable or disable details. The default is disabled .	OPTIONAL

EXAMPLE:

```
=>mbus listtypes  
- root  
  + InternetGatewayDevice  
    * DeviceInfo  
    * Layer3Forwarding  
    * LANDevice  
      - WLANConfiguration  
        + WEPKey  
        + PreSharedKey  
      -  
    * WANDevice  
      - WANCommonInterfaceConfig  
      - WANDSLLinkConfig  
      - WANConnectionDevice  
        + WANDSLLinkConfig  
        + WANIPConnection  
          * PortMapping  
        + WANPPPConnection  
          * PortMapping  
    * Services  
    * ManagementServer  
=>
```

RELATED COMMANDS:

mbus listobjects Display the object instances.

mbus loadobjects

Load object instances of registered objecttypes.

SYNTAX:

```
mbus loadobjects      [path = <quoted string>]  
                      [type = <{idpath | keypath}>]
```

where:

path	Fully qualified mbus path.	OPTIONAL
type	Path type. Choose between: ▶ <i>idpath</i> ▶ <i>keypath</i> . The default is <i>idpath</i> .	OPTIONAL

RELATED COMMANDS:

mbus unloadobjects Unload object instances.

mbus unloadobjects

Unload object instances.

SYNTAX:

```
mbus unloadobjects      [path = <quoted string>]
                        [type = <{idpath | keypath}>]
```

where:

path	Fully qualified mbus path.	OPTIONAL
type	Path type. Choose between: ▶ <i>idpath</i> ▶ <i>keypath</i> . The default is <i>idpath</i> .	OPTIONAL

RELATED COMMANDS:

mbus loadobjects Load object instances of registered objecttypes.

MEMM Commands

Introduction

This chapter describes the commands of the **memm** command group.

Contents

This chapter covers the following commands:

memm debug lock traceconfig	Modify lock trace settings.	484
memm debug traceconfig	Modify memm trace settings.	485
memm listobjects	Display objects.	486
memm stats	Display memm statistics.	487

memm debug lock traceconfig

Modify lock trace settings.

SYNTAX:

```
memm debug lock traceconfig [level = <number{0-4}>]  
[name = <string>]
```

where:

level	A number between 0 and 4. Represents the memm trace level.	OPTIONAL
name	Application name filter for lock traces (empty displays all lock traces).	OPTIONAL

memm debug traceconfig

Modify memm trace settings.

SYNTAX:

```
memm debug traceconfig [level = <number{0-4}>]
```

where:

level	A number between 0 and 4. Represents the memm trace level.	OPTIONAL
-------	---	----------

memm listobjects

Display objects.

SYNTAX:

```
memm listobjects [name = <quoted string>]
```

where:

name	Select typename(s) to list (supports partial typename).	OPTIONAL
------	---	----------

memm stats

Display memm statistics.

SYNTAX:

memm stats	[name = <quoted string>]
------------	--------------------------

where:

name	Select typename(s) to list (supports partial typename).	OPTIONAL
------	---	----------

EXAMPLE:

=>memm stats	pool name	bytes	cur,max,avail	(max.%)	count	cur,max,avail	ref	cur,max,avail
	memm_pooledesc	1196,	1196,	1300 (92%)	23,	23,	25	obj: 0, 255
	dynstr_pool	72,	72,	120 (60%)	3,	3,	5	0, 0, -
	mbus_llist	2904,	2904,	4920 (59%)	242,	242,	410	0, 0, -
	mbus_dynstr	2928,	2964,	4920 (60%)	244,	247,	410	0, 0, -
	- strings buffer	3545,	3563,	8192 (43%)				
	mbus_client	12,	12,	120 (10%)	1,	1,	10	0, 0, -
	mbus_cmddata	0,	88,	3520 (2%)	0,	2,	80	0, 0, -
	mbus_objecttype	744,	744,	960 (77%)	31,	31,	40	0, 0, -
	mbus_paramtype	5908,	5908,	7000 (84%)	211,	211,	250	0, 0, -
	mbus_object	28,	28,	1960 (1%)	1,	1,	70	0, 0, -
	mbus_objectindex	0,	0,	840 (0%)	0,	0,	70	0, 0, -
	upnp_dynstr	36,	60,	600 (10%)	3,	5,	50	0, 0, -
	-strings buffer	138,	151,	2048 (7%)				
	upnp_handles	332,	332,	996 (33%)	1,	1,	3	0, 0, -
	upnp_devices	816,	816,	8160 (10%)	3,	3,	30	0, 0, -
	upnp_services	2560,	2560,	19200 (13%)	4,	4,	30	0, 0, -
	upnp_subscriptions	0,	0,	30400 (0%)	0,	0,	50	0, 0, -
	upnp_firstdeviceevents	0,	0,	4000 (0%)	0,	0,	50	0, 0, -
	mbus_cliclient	0,	0,	240 (0%)	0,	0,	5	0, 0, -
	cwmp_elem_pool	0,	0,	560 (0%)	0,	0,	20	0, 0, -
	cwmp_attr_pool	0,	0,	400 (0%)	0,	0,	20	0, 0, -
	cwmp_llist_pool	0,	24,	480 (5%)	0,	2,	40	0, 0, -
	cwmp_dynstr_pool	0,	0,	240 (0%)	0,	0,	20	0, 0, -
	-strings buffer	0,	0,	8192 (0%)				
	cwmp_event_pool	0,	0,	64 (0%)	0,	0,	4	0, 0, -
	cwmp_cmddata_pool	0,	0,	1320 (0%)	0,	0,	30	0, 0, -
	Total usage			21219, 21422,110752 (19%)				
	=>							

EXAMPLE:

pool name cur,max,avail	bytes cur,max,avail (max.%)	count cur,max,avail	ref
memm_pooldesc j: 0, 255	1196, 1196, 1300 (92%)	23, 23, 25	---
dynstr_pool 0, 0, -	72, 72, 120 (60%)	3, 3, 5	
mbus_llist 0, 0, -	2460, 2532, 4920 (51%)	205, 211, 410	
mbus_dynstr 0, 0, -	2316, 2424, 4920 (49%)	193, 202, 410	
- strings buffer	2761, 2839, 8192 (34%)		
mbus_client 0, 0, -	12, 12, 120 (10%)	1, 1, 10	
mbus_cmddata 0, 0, -	0, 88, 3520 (2%)	0, 2, 80	
mbus_objecttype 0, 0, -	480, 480, 960 (50%)	20, 20, 40	
mbus_paramtype 0, 0, -	3976, 3976, 7000 (56%)	142, 142, 250	
mbus_object 0, 0, -	840, 840, 1960 (42%)	30, 30, 70	
mbus_objectindex 0, 0, -	168, 168, 840 (20%)	14, 14, 70	
upnp_dynstr 0, 0, -	36, 72, 600 (12%)	3, 6, 50	
- strings buffer	132, 152, 2048 (7%)	14, 14, 70	
upnp_handles 0, 0, -	332, 332, 996 (33%)	1, 1, 3	
upnp_devices 0, 0, -	816, 816, 8160 (10%)	3, 3, 30	
upnp_services 0, 0, -	2560, 2560, 19200 (13%)	4, 4, 30	
upnp_subscriptions 0, 0, -	1216, 2432, 30400 (8%)	2, 4, 50	
upnp_firstdeviceevents 0, 0, -	0, 320, 4000 (8%)	0, 4, 50	
mbus_cliclient 0, 0, -	0, 0, 240 (0%)	0, 0, 5	
cwmp_elem_pool 0, 0, -	0, 0, 560 (0%)	0, 0, 20	
cwmp_attr_pool 0, 0, -	0, 0, 400 (0%)	0, 0, 20	
cwmp_llist_pool 0, 0, -	0, 24, 480 (5%)	0, 2, 40	
cwmp_dynstr_pool 0, 0, -	0, 0, 240 (0%)	0, 0, 20	
- strings buffer	0, 0, 8192 (0%)		
cwmp_event_pool 0, 0, -	0, 0, 64 (0%)	0, 0, 4	
cwmp_cmddata_pool 0, 0, -	0, 0, 13202 (0%)	0, 0, 30	
Total usage	19373, 21335, 110752 (19%)		
=>			

MLP Commands

Introduction

This chapter describes the commands of the **mlp** command group.

Contents

This chapter covers the following commands:

mlp flush	Flush all the Multi-Level access Policies (MLP) structures.	490
mlp import	Import all the scores.	491
mlp debug export	Export all the scores.	492
mlp debug stats	Display MLP statistics.	493
mlp debug traceconfig	Modify the MLP trace settings.	494
mlp privilege add	Add a privilege.	495
mlp privilege addzone	Add a zone to a privilege.	497
mlp privilege config	Modify a privilege.	498
mlp privilege delete	Delete a privilege.	499
mlp privilege list	Display the privileges.	500
mlp privilege removezone	Remove a zone from a privilege.	502
mlp role add	Add a role.	503
mlp role addpriv	Add a privilege to a role.	504
mlp role config	Modify the role.	505
mlp role delete	Delete a role.	506
mlp role list	Display the roles.	507
mlp role removepriv	Remove a privilege from a role.	508

mlp flush

Flush all the Multi-Level access Policies (MLP) structures.

SYNTAX:

```
mlp flush
```

mlp import

Import all the scores.

SYNTAX:

```
mlp import [trace = <{disabled | enabled | full}>]
```

where:

trace	Select the import trace level. Choose between: ▶ disabled ▶ enabled ▶ full. The default is disabled .	OPTIONAL
-------	---	----------

RELATED COMMANDS:

mlp debug export Export all the scores.

mlp debug export

Export all the scores.

SYNTAX:

```
mlp debug export
```

RELATED COMMANDS:

mlp import Import all the scores.

mlp debug stats

Display MLP statistics.

SYNTAX:

```
mlp debug stats
```

EXAMPLE:

```
=>mlp debug stats
Roles (cur/free/max)      : 8/7/15
Privileges (cur/free/max)  : 35/25/60
ListItems (cur/free/max)   : 31/119/150
=>
```

mlp debug traceconfig

Modify the MLP trace settings.

SYNTAX:

```
mlp debug traceconfig [trace = <{disabled | enabled | full}>]
```

where:

trace	Select the trace level. Choose between: ▶ disabled ▶ enabled ▶ full. The default is disabled .	OPTIONAL
-------	--	----------

EXAMPLE:

```
=>mlp debug traceconfig
mlp trace: disabled
=>
```

mlp privilege add

Add a privilege.

SYNTAX:

```
mlp privilege add name = <quoted string>
                  type = <{access | service}>
                  [descr = <quoted string>]
```

where:

name	The name of the new privilege.	REQUIRED
Note The maximum number of privileges is 60.		
type	Select the privilege type. Choose between: <ul style="list-style-type: none">▶ access: the privilege is an access privilege.▶ service: the privilege is a service privilege.	REQUIRED
descr	A description of the privilege. Note The maximum length is 63 characters.	OPTIONAL

EXAMPLE:

```
=>mlp privilege list type=access
Privilege (type) Description
-----
anyaccess (access) All access privileges granted
AP1 (access) LAN_Local_all
AP2 (access) LAN_Local with all secured channels
AP3 (access) LAN_Local with all channels & access to sensitive files
...
AP15 (access) WAN from all channels
AP16 (access) LAN from HTTP/HTTPs

=>mlp privilege add name=myPrivilege type=access descr="My access privilege"
=>mlp privilege list type=access
Privilege (type) Description
-----
anyaccess (access) All access privileges granted
AP1 (access) LAN_Local_all
AP2 (access) LAN_Local with all secured channels
AP3 (access) LAN_Local with all channels & access to sensitive files
...
AP15 (access) WAN from all channels
AP16 (access) LAN from HTTP/HTTPs
myPrivilege (access) My access privilege

=>
```

MLP Commands

RELATED COMMANDS:

<code>mlp privilege delete</code>	Delete a privilege.
<code>mlp privilege list</code>	Display the privileges.

mlp privilege addzone

Add a zone to a privilege.



Only one zone can be added at a time to a privilege. If multiple zones need to be added, the command **:mlp privilege addzone** must be executed for each zone to be added.

SYNTAX:

```
mlp privilege addzone    name = <string>
                        zone = <string>
```

where:

name	The name of the privilege in which a zone must be added.	REQUIRED
zone	The name of the new zone.	REQUIRED

EXAMPLE:

```
=>mlp privilege list type=access name=myPrivilege verbose=all
myPrivilege (access) My access privilege

=>mlp privilege addzone name=myPrivilege zone=channel_http
=>mlp privilege list type=access name=myPrivilege verbose=all
myPrivilege (access) My access privilege
    channel_http

=>
```

RELATED COMMANDS:

mlp privilege removezone Remove a zone from a privilege.

mlp privilege config

Modify a privilege.

SYNTAX:

```
mlp privilege config      name = <string>
                           [descr = <quoted string>]
                           [score = <{hex-word}[:{hex-word}] ex: 'a12:c30f'>]
```

where:

name	The name of the privilege to be modified.	REQUIRED
descr	Set the privilege description.	OPTIONAL
Note	The maximum length is 63 characters.	
score	Set the score of the privilege.	OPTIONAL
Note	This is a hexadecimal value, for example "a12:c30f".	

EXAMPLE:

```
=>mlp privilege config name=myPrivilege descr="My test privilege" score=a12:def4
=>mlp privilege list verbose=all
anyaccess (access) All access privileges granted
  unsecure_connection, channel_ftp, channel_telnet, channel_http,
  channel_mdap, channel_serial, origin_lan, origin_wan, origin_local

anyservice (service) All service privileges granted
...
AP15 (access) WAN from all channels
  unsecure_connection, channel_ftp, channel_telnet, channel_http,
  channel_mdap, channel_serial, origin_wan

AP16 (access) LAN from HTTP/HTTPPs
  unsecure_connection, channel_http, origin_lan

myPrivilege (service) My test privilege
  r_lan, r_fs_view, r_fs_retrieve, r_rtg, r_fwdg, r_frwl, r_ipsec_norm,
  r_ipsec_adv, r_certificates, r_local, r_qos, and_lan, and_local, cli,
  ftp
=>
```

mlp privilege delete

Delete a privilege.

SYNTAX:

```
mlp privilege delete      name = <string>
```

where:

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

EXAMPLE:

```
=>mlp privilege list type=access
Privilege (type) Description
-----
anyaccess (access) All access privileges granted
AP1 (access) LAN_Local_all
AP2 (access) LAN_Local with all secured channels
AP3 (access) LAN_Local with all channels & access to sensitive files
AP4 (access) LAN_Local with all secured channels & access to sensitive files
AP5 (access) WAN from a secure HTTP
AP6 (access) WAN from a secure HTTP & access to sensitive files
AP7 (access) LAN from all channels except telnet
AP8 (access) LAN from all secured channels except telnet
AP9 (access) LAN from all channels except telnet & access to sensitive file
AP10 (access) LAN access to sensitive file & secured channels except telnet
AP11 (access) All origins_noHTTP
AP12 (access) All origins_from all secured channels no http
AP13 (access) All origins_noHTTP & access to sensitive files
AP14 (access) All origins_secured channels &access to sensitive files no http
AP15 (access) WAN from all channels
AP16 (access) LAN from HTTP/HTTPs
myPrivilege (access) My access privilege

=>mlp privilege delete name=myPrivilege
=>mlp privilege list type=access
Privilege (type) Description
-----
anyaccess (access) All access privileges granted
AP1 (access) LAN_Local_all
AP2 (access) LAN_Local with all secured channels
AP3 (access) LAN_Local with all channels & access to sensitive files
...
AP13 (access) All origins_noHTTP & access to sensitive files
AP14 (access) All origins_secured channels &access to sensitive files no http
AP15 (access) WAN from all channels
AP16 (access) LAN from HTTP/HTTPs

=>
```

RELATED COMMANDS:

mlp privilege delete	Add a privilege.
mlp privilege list	Display the privileges.

mlp privilege list

Display the privileges.

SYNTAX:

```
mlp privilege list      [name = <string>]
                      [type = <{access | service}>]
                      [verbose = <{minimal | medium | all}>]
```

where:

name	The name of the privilege to be listed.	OPTIONAL
Note If not specified, all the privileges will be shown.		
type	Select the privilege type to be shown. Choose between:	OPTIONAL
<ul style="list-style-type: none"> ▶ access: only the access privileges will be shown ▶ service: only the service privileges will be shown. 		
Note If not specified, all the privilege types will be shown.		
verbose	Limit the output list. Choose between:	OPTIONAL
<ul style="list-style-type: none"> ▶ minimal ▶ medium ▶ all. 		
The default is minimal .		

EXAMPLE:

```
=>mlp privilege list type=service
Privilege (type) Description
-----
anyService (service) All service privileges granted
SP1 (service) Limited_Read
SP2 (service) detailed_Read
SP3 (service) Write_to_LAN
SP4 (service) Write_FW_RT_normal_FW_cfgs
SP5 (service) cooperative_cfgs
SP6 (service) Remote_Configurations
SP7 (service) Advanced_sink & source_FW
SP8 (service) Write_to_U_Connection
SP9 (service) Write_to_Local
SP10 (service) Simple_network_debugging
SP11 (service) Advance_Network_debugging
SP12 (service) User_admin
SP13 (service) MLP_Admin
SP14 (service) Backup & Restore
SP15 (service) CLI_mngt_Interface_only
SP16 (service) CGI_mngt_Interface_only
SP17 (service) FTP_mngt_Interface_only
=>mlp privilege list name=SP5 verbose=all
SP5 (service) cooperative_cfgs
  r_ipsec_norm, r_ipsec_adv, r_certificates, w_frwl_norm, w_frwl_adv,
  w_ipsec, w_certificates
=>
```

RELATED COMMANDS:

mlp privilege delete Add a privilege.
mlp privilege delete Delete a privilege.

mlp privilege removezone

Remove a zone from a privilege.



Only one zone can be removed at a time from a privilege. If multiple zones need to be removed, the command :**mlp privilege removezone** must be executed for each zone to be removed.

SYNTAX:

```
mlp privilege removezone    name = <string>
                           zone = <string>
```

where:

name	The name of the privilege in which a zone must be removed.	REQUIRED
zone	The name of the zone to be removed.	REQUIRED

EXAMPLE:

```
=>mlp privilege list name=myPrivilege verbose=all
myPrivilege (access) My access privilege
  channel_http, channel_serial

=>mlp privilege removezone name=myPrivilege zone=channel_serial
=>mlp privilege list name=myPrivilege verbose=all
myPrivilege (access) My access privilege
  channel_http

=>
```

RELATED COMMANDS:

mlp privilege addzone Add a zone to a privilege.

mlp role add

Add a role.

SYNTAX:

```
mlp role add name = <quoted string>
    parent = <string>
    [descr = <quoted string>]
```

where:

name	The name of the new role.	REQUIRED
	Note The maximum number of roles is 15.	
parent	The name of the parent role.	REQUIRED
descr	The role description.	OPTIONAL
	Note The maximum length is 63 characters.	

EXAMPLE:

```
=>mlp role list
Role (parent) Description
-----
root (-) The superuser
SuperUser (root) Any service and any access from LAN/WAN/LOCAL
TechnicalSupport (SuperUser) Any service and any channels from WAN
Administrator (TechnicalSupport) Any service and any access from LAN/Local origin only
PowerUser (Administrator) GUI(Service/overview page)via http/https from LAN origin
User (PowerUser) GUI(Overview pages, remote assistance) via HTTP/HTTPS from LAN
LAN_Admin (Administrator) Only LAN related configurations from any Channel/Origin
WAN_Admin (Administrator) Only WAN related configurations from any Channels/Origin

=>mlp role create name=myRole parent=User descr="My user access"
=>mlp role list
Role (parent) Description
-----
root (-) The superuser
SuperUser (root) Any service and any access from LAN/WAN/LOCAL
TechnicalSupport (SuperUser) Any service and any channels from WAN
Administrator (TechnicalSupport) Any service and any access from LAN/Local origin only
PowerUser (Administrator) GUI(Service/overview page)via http/https from LAN origin
User (PowerUser) GUI(Overview pages, remote assistance) via HTTP/HTTPS from LAN
LAN_Admin (Administrator) Only LAN related configurations from any Channel/Origin
WAN_Admin (Administrator) Only WAN related configurations from any Channels/Origin
myRole (User) My user access

=>
```

RELATED COMMANDS:

mlp role delete	Delete a role.
mlp role list	Display the roles.

mlp role addpriv

Add a privilege to a role.

SYNTAX:

```
mlp role addpriv    name = <string>
                    access = <string>
                    service = <string>
```

where:

name	The name of the role.	REQUIRED
access	The name of the access privilege to be added.	REQUIRED
service	The name of the service privilege to be added.	REQUIRED

EXAMPLE:

```
=>mlp role list name=myRole verbose=all
myRole (User) My user access

=>mlp role addpriv name=myRole access=myPrivilege service=SP7
=>mlp role list name=myRole verbose=all
myRole (User) My user access
    myPrivilege & SP7

=>
```

RELATED COMMANDS:

mlp role removepriv Remove a privilege from a role.

mlp role config

Modify the role.

SYNTAX:

```
mlp role config    name = <string>
                  [parent = <string>]
                  [descr = <quoted string>]
```

where:

name	The name of the role to be configured.	REQUIRED
parent	The name of the parent role.	OPTIONAL
descr	The role description.	OPTIONAL
Note The maximum length is 63 characters.		

EXAMPLE:

```
=>mlp role list name=myRole verbose=all
myRole (User) My user access
  myPrivilege & SP7

=>mlp role config name=myRole parent=Guest descr="My user access"
=>mlp role list name=myRole verbose=all
myRole (Guest) My user access
  myPrivilege & SP7

=>
```

mlp role delete

Delete a role.

SYNTAX:

```
mlp role delete name = <string>
```

where:

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

EXAMPLE:

```
=>mlp role list
Role (parent) Description
-----
root (-) The superuser
SuperUser (root) Any service and any access from LAN/WAN/LOCAL
TechnicalSupport (SuperUser) Any service and any channels from WAN
Administrator (TechnicalSupport) Any service and any access from LAN/Local origin only
PowerUser (Administrator) GUI(Service/overview page)via http/https from LAN origin
User (PowerUser) GUI(Overview pages, remote assistance) via HTTP/HTTPS from LAN
LAN_Admin (Administrator) Only LAN related configurations from any Channel/Origin
WAN_Admin (Administrator) Only WAN related configurations from any Channels/Origin
myRole (Guest) My user access

=>mlp role delete name=myRole
=>mlp role list
Role (parent) Description
-----
root (-) The superuser
SuperUser (root) Any service and any access from LAN/WAN/LOCAL
TechnicalSupport (SuperUser) Any service and any channels from WAN
Administrator (TechnicalSupport) Any service and any access from LAN/Local origin only
PowerUser (Administrator) GUI(Service/overview page)via http/https from LAN origin
User (PowerUser) GUI(Overview pages, remote assistance) via HTTP/HTTPS from LAN
LAN_Admin (Administrator) Only LAN related configurations from any Channel/Origin
WAN_Admin (Administrator) Only WAN related configurations from any Channels/Origin

=>
```

RELATED COMMANDS:

mlp role delete	Add a role.
mlp role list	Display the roles.

mlp role list

Display the roles.

SYNTAX:

```
mlp role list      [name = <string>]
                  [verbose = <{minimal | medium | all}>]
```

where:

name	The name of the role to be listed.	OPTIONAL
	Note If not specified, all the roles will be listed.	
verbose	Limit the output list. Choose between: <ul style="list-style-type: none">▶ minimal▶ medium▶ all. The default is minimal .	OPTIONAL

EXAMPLE:

```
=>mlp role list
Role (parent) Description
-----
root (-) The superuser
SuperUser (root) Any service and any access from LAN/WAN/LOCAL
TechnicalSupport (SuperUser) Any service and any channels from WAN
Administrator (TechnicalSupport) Any service and any access from LAN/Local origin only
PowerUser (Administrator) GUI(Service/overview page)via http/https from LAN origin
User (PowerUser) GUI(Overview pages, remote assistance) via HTTP/HTTPS from LAN
LAN_Admin (Administrator) Only LAN related configurations from any Channel/Origin
WAN_Admin (Administrator) Only WAN related configurations from any Channels/Origin
myRole (Guest) My user access

=>mlp role list name=PowerUser verbose=all
PowerUser (Administrator) GUI(Service/overview page)via http/https from LAN origin
  AP16 & SP1
  AP16 & SP2
  AP16 & SP10
  AP16 & SP16
  AP16 & SP8
=>
```

RELATED COMMANDS:

mlp role delete	Add a role.
mlp role delete	Delete a role.

mlp role removepriv

Remove a privilege from a role.

SYNTAX:

```
mlp role removepriv    name = <string>
                      access = <string>
                      [service = <string>]
```

where:

name	The name of the role.	REQUIRED
access	The name of the access privilege to be deleted.	REQUIRED
service	The name of the service privilege to be deleted.	OPTIONAL

EXAMPLE:

```
=>mlp role list name=myRole verbose=all
myRole (Guest) My user access
  myPrivilege & SP7

=>mlp role removepriv name=myRole access=myPrivilege
=>mlp role list name=myRole verbose=all
myRole (Guest) My user access

=>
```

RELATED COMMANDS:

mlp role addpriv Add a privilege to a role.

NAT Commands

Introduction

This chapter describes the commands of the **nat** command group.

Contents

This chapter covers the following commands:

nat config	Display/modify global NAT configuration options.	510
nat flush	Flush the current NAT configuration.	511
nat ifconfig	Modify address translation on an IP interface.	512
nat iflist	Display the address translation configuration on all the interfaces.	513
nat mapadd	Add an address mapping to a Network Address Translation (NAT) enabled interface.	514
nat mapdelete	Delete an address mapping from a NAT enabled interface.	517
nat maplist	Display the address mapping for a NAT enabled interface.	518
nat tmpladd	Add an address mapping template.	519
nat tmpldelete	Delete an address mapping template.	522
nat tmpllist	Display the address mapping templates.	523
nat tmplinst	Instantiate address mapping templates for a given dynamic address.	524

nat config

Display/modify global NAT configuration options.

SYNTAX:

```
nat config [trace = <{disabled | enabled}>]
```

where:

trace	Enable or disable traces. The default is disabled .	REQUIRED
-------	---	----------

EXAMPLE:

```
=>nat config  
NAT traces : disabled  
=>nat config trace=enabled  
=>nat config  
NAT traces : enabled  
=>
```

nat flush

Flush the current NAT configuration.

SYNTAX:

```
nat flush
```

nat ifconfig

Modify address translation on an IP interface.

SYNTAX:

```
nat ifconfig  intf = <string>
              translation = <{disabled | enabled | transparent}>
```

where:

intf	The IP interface name.	REQUIRED
translation	Enable or disable address translation. Choose between: <ul style="list-style-type: none"> ▶ disabled: Address translation is disabled. ▶ enabled: Address translation is enabled. ▶ transparent: An inside address is translated into the same outside IP address. In fact, no translation happens and the IP packets passing through the SpeedTouch™ remain untouched. The default is disabled .	REQUIRED

EXAMPLE:

```
=>nat iflist
Interface      NAT
loop           disabled
myPPP_ppp     disabled
eth0           transparent
=>nat ifconfig intf=myPPP_ppp translation=enabled
=>nat iflist
Interface      NAT
loop           disabled
myPPP_ppp     enabled
eth0           transparent
=>
```

RELATED COMMANDS:

nat iflist	Display the address translation configuration on all the interfaces.
------------	--

nat iflist

Display the address translation configuration on all the interfaces.

SYNTAX:

```
nat iflist
```

EXAMPLE:

```
=>nat iflist
Interface      NAT
loop          disabled
Internet       enabled
RtPPPoE_ppp   enabled
LocalNetwork   transparent
=>
```

RELATED COMMANDS:

nat ifconfig Modify address translation on an IP interface.

nat mapadd

Add an address mapping to a Network Address Translation (NAT) enabled interface.

SYNTAX:

```
nat mapadd  intf = <string>
            [type = <{napt | nat}>]
            [outside_addr = <ip-range>]
            [inside_addr = <ip-range>]
            [access_list = <ip-range>]
            [foreign_addr = <ip-range>]
            [protocol = <{supported IP protocol} or number>]
            [outside_port = <port-range>]
            [inside_port = <port-range>]
            [mode = <{auto | inbound | outbound}>]
```

where:

intf	The IP interface name.	REQUIRED
Note The specified interface must be an existing NAT enabled interface.		
type	The type of NAT map to be used. Choose between: <ul style="list-style-type: none">▶ napt: The map is a Network Address Port Translation (NAPT) map or a port-shifting NAT map.▶ nat: The map is a basic NAT map or a two-way NAT map.	OPTIONAL
outside_addr	The outside (typically public) IP address. <ul style="list-style-type: none">▶ Inbound: This map will only apply if the destination IP address is part of the outside address(es).▶ Outbound: Represents the address(es) to be used as source address after translation. Allowed values are:<ul style="list-style-type: none">▶ An IP address (A.B.C.D or A.D, meaning A.0.0.D).▶ A range of IP addresses (A.B.C.[D-E]) (only for NAT).▶ An IP subnet (A.B.C.D/M) (only for NAT).	REQUIRED for NAT maps and NAPT maps. OPTIONAL for port-shifting maps.
inside_addr	The inside (typically private) IP address. <ul style="list-style-type: none">▶ Inbound: Represents the address(es) to be used as destination address after translation.▶ Outbound: This map will only apply if the source IP address is part of the inside address(es). Allowed values are:<ul style="list-style-type: none">▶ An IP address (A.B.C.D or A.D, meaning A.0.0.D).▶ A range of IP addresses (A.B.C.[D-E]) (only for NAT).▶ An IP subnet (A.B.C.D/M (only for NAT)).	OPTIONAL for NAT maps and NAPT maps. REQUIRED for port-shifting maps.
The default is "" (unmapped).		
Note If specified, the number of inside addresses must be equal to the number of outside addresses.		

access_list	The range of inside addresses to which the mapping is restricted. This parameter is used for outbound traffic only. Represents the list of inside IP addresses (LAN devices) allowed to make use of this map. Allowed values are: <ul style="list-style-type: none">▶ An IP address (A.B.C.D or A.D, meaning A.0.0.D).▶ A range of IP addresses (A.B.C.[D-E]).▶ An IP subnet (A.B.C.D/M).▶ * (all addresses). The default is the <i>inside_addr</i> .	OPTIONAL
foreign_addr	The range of destination addresses to which the mapping is restricted. This parameter is used as filter for inbound/outbound traffic. <ul style="list-style-type: none">▶ Inbound: This map only applies if the source IP address is part of the foreign address(es).▶ Outbound: This map only applies if the destination IP address is part of the foreign address(es). Allowed values are: <ul style="list-style-type: none">▶ An IP address (A.B.C.D or A.D, meaning A.0.0.D).▶ A range of IP addresses (A.B.C.[D-E]).▶ An IP subnet (A.B.C.D/M).▶ * (all addresses). The default is * (meaning all addresses).	OPTIONAL
protocol	The IP protocol to be used as filter for inbound/outbound traffic. The NAT map only applies if the protocol of the IP packet matches the map protocol. Select an IP protocol (see “ Supported IP Protocols ” on page 748) or, alternatively, type the protocol number. The default is 0 (meaning any protocol).	OPTIONAL
outside_port	The outside port number or range (only for maps of type <i>napt</i>). <ul style="list-style-type: none">▶ For a NAPT map: Represents the range of TCP/UDP ports to be used as dynamic port range during the NAPT process.▶ For a port-shifting map: Represents the source port(s) to be translated (shifted) into <i>inside_port</i>. Allowed values are: <ul style="list-style-type: none">▶ TCP/UDP port range [P-Q] (with Q > P)▶ TCP/UDP port number (only for port-shifting maps).	OPTIONAL for NAPT map. REQUIRED for port-shifting map.
inside_port	The inside port number or range (only for maps of type <i>napt</i>). If this parameter is specified for a map of type <i>napt</i> , then this map is a port-shifting map. Represents the target TCP/UDP ports to which the traffic has to be shifted. Allowed values are: <ul style="list-style-type: none">▶ TCP/UDP port range [P-Q] (with Q > P)▶ TCP/UDP port number (only for port-shifting maps).	REQUIRED for port-shifting map

Note The size of *inside_port* must be identical to the size of *outside_port*.

NAT Commands

mode	The mode to create the portmap. Choose between: ▶ auto ▶ inbound ▶ outbound.	OPTIONAL
The standard is auto .		

EXAMPLE:

```
=>nat maplist
Idx Type Interface      Outside Address          Inside Address     Use
 1 NAPT eth0           any:80                  127.0.0.1:8080    0
 2 NAPT eth0           any:1080                127.0.0.1:8080    0
 3 NAPT eth0           any:8080                127.0.0.1:8080    0
=>nat mapadd intf=myPPP_ppp type=napt outside_addr=100.100.100.1 access_list=10.0.0.0/24
=>nat maplist
Idx Type Interface      Outside Address          Inside Address     Use
 1 NAPT myPPP_ppp     100.100.100.1        unmapped          0
 1 NAPT eth0           any:80                  127.0.0.1:8080    0
 2 NAPT eth0           any:1080                127.0.0.1:8080    0
 3 NAPT eth0           any:8080                127.0.0.1:8080    0
=>maplist intf=myPPP_ppp expand=enabled
Idx Type Interface      Outside Address          Inside Address     Use
 1 NAPT myPPP_ppp     100.100.100.1        unmapped          0
Access List.....       10.0.0.0/24
Foreign Address.....   any
Protocol.....         any
Flags.....            Static
Description.....       Outbound NAPT without defserver
=>
```

RELATED COMMANDS:

- | | |
|---------------|--|
| nat mapdelete | Delete an address mapping from a NAT enabled interface. |
| nat maplist | Display the address mapping for a NAT enabled interface. |

nat mapdelete

Delete an address mapping from a NAT enabled interface.

SYNTAX:

```
nat mapdelete intf = <string>
               index = <number>
```

where:

intf	The IP interface name.	REQUIRED
index	The map index.	REQUIRED

Tip Use the command :nat maplist to obtain a list of the mapindexes.

EXAMPLE:

```
=>nat maplist
Idx Type Interface      Outside Address          Inside Address      Use
 1 NAPT myPPP_ppp       100.100.100.1           unmapped            0
 1 NAPT eth0             any:80                 127.0.0.1:8080        0
 2 NAPT eth0             any:1080              127.0.0.1:8080        0
 3 NAPT eth0             any:8080              127.0.0.1:8080        0
=>nat mapdelete intf=myPPP_ppp index=1
=>nat maplist
Idx Type Interface      Outside Address          Inside Address      Use
 1 NAPT eth0             any:80                 127.0.0.1:8080        0
 2 NAPT eth0             any:1080              127.0.0.1:8080        0
 3 NAPT eth0             any:8080              127.0.0.1:8080        0
=>
```

RELATED COMMANDS:

nat mapadd	Add an address mapping to a Network Address Translation (NAT) enabled interface.
nat maplist	Display the address mapping for a NAT enabled interface.

nat maplist

Display the address mapping for a NAT enabled interface.

SYNTAX:

```
nat maplist [intf = <string>]
            [expand = <{disabled | enabled}>]
```

where:

intf	The IP interface name.	OPTIONAL
Note If not specified, the address mapping for all the NAT enabled interfaces will be shown.		
expand	Enable or disable expanded listing. The default is disabled .	OPTIONAL

EXAMPLE:

```
=>nat maplist
Idx Type Interface      Outside Address          Inside Address      Use
 1 NAPT myPPP_ppp       100.100.100.1        unmapped           0
 1 NAPT eth0            any:80                 127.0.0.1:8080     0
 2 NAPT eth0            any:1080              127.0.0.1:8080     0
 3 NAPT eth0            any:8080              127.0.0.1:8080     0
=>maplist intf=myPPP_ppp expand=enabled
Idx Type Interface      Outside Address          Inside Address      Use
 1 NAPT myPPP_ppp       100.100.100.1        unmapped           0
                                         Access List..... 10.0.0.0/24
                                         Foreign Address.... any
                                         Protocol..... any
                                         Flags..... Static
                                         Description..... Outbound NAPT without defserver
=>
```

RELATED COMMANDS:

- | | |
|---------------|--|
| nat mapadd | Add an address mapping to a Network Address Translation (NAT) enabled interface. |
| nat mapdelete | Delete an address mapping from a NAT enabled interface. |

nat tmpladd

Add an address mapping template.

SYNTAX:

```
nat tmpladd [intf = <string>]
            [group = <{wan|local|lan|tunnel|dmz|guest} or number>]
            [timeout = <number{0-65535}>]
            [type = <{napt | nat}>]
            [outside_addr = <ip-range>]
            [inside_addr = <ip-range>]
            [access_list = <ip-range>]
            [foreign_addr = <ip-range>]
            [protocol = <{supported IP protocol} or number>]
            [outside_port = <port-range>]
            [inside_port = <port-range>]
            [mode = <{auto | inbound | outbound}>]
```

where:

intf	The IP interface name.	OPTIONAL
group	The IP interface group scope for this template. Choose between: <ul style="list-style-type: none">▶ wan▶ local▶ lan▶ tunnel▶ dmz▶ guest.	OPTIONAL
	Note If an interface has been defined with the parameter intf , then the setting for this parameter is ignored.	
timeout	A number between 0 and 6553 (seconds). Represents the lifetime for this template.	OPTIONAL
type	The type of NAT map to be used. Choose between: <ul style="list-style-type: none">▶ napt: The map is a NAPT map or a port-shifting NAT map.▶ nat: The map is a basic NAT map or a two-way NAT map.	OPTIONAL
outside_addr	The outside (typically public) IP address. <ul style="list-style-type: none">▶ Inbound: This map will only apply if the destination IP address is part of the outside address(es).▶ Outbound: Represents the address(es) to be used as source address after translation. Allowed values are: <ul style="list-style-type: none">▶ An IP address (A.B.C.D or A.D, meaning A.0.0.D).▶ A range of IP addresses (A.B.C.[D-E]) (only for NAT).▶ An IP subnet (A.B.C.D/M) (only for NAT).	REQUIRED for NAT maps and NAPT maps. OPTIONAL for port-shifting maps.

inside_addr	<p>The inside (typically private) IP address.</p> <ul style="list-style-type: none"> ▶ Inbound: Represents the address(es) to be used as destination address after translation. ▶ Outbound: This map will only apply if the source IP address is part of the inside address(es). <p>Allowed values are:</p> <ul style="list-style-type: none"> ▶ An IP address (A.B.C.D or A.D, meaning A.0.0.D). ▶ A range of IP addresses (A.B.C.[D-E]) (only for NAT). ▶ An IP subnet (A.B.C.D/M (only for NAT)). <p>Default value is "" (meaning unmapped).</p> <p>Note If specified, the number of inside addresses must be equal to the number of outside addresses.</p>	OPTIONAL for NAT maps and NAPT maps.
access_list	<p>The range of inside addresses to which the mapping is restricted. This parameter is used for outbound traffic only. Represents the list of inside IP addresses (LAN devices) allowed to make use of this map.</p> <p>Allowed values are:</p> <ul style="list-style-type: none"> ▶ An IP address (A.B.C.D or A.D, meaning A.0.0.D). ▶ A range of IP addresses (A.B.C.[D-E]). ▶ An IP subnet (A.B.C.D/M). ▶ * (all addresses). <p>The default is the inside_addr.</p>	OPTIONAL
foreign_addr	<p>The range of destination addresses to which the mapping is restricted. This parameter is used as filter for inbound/outbound traffic.</p> <ul style="list-style-type: none"> ▶ Inbound: This map only applies if the source IP address is part of the foreign address(es). ▶ Outbound: This map only applies if the destination IP address is part of the foreign address(es). <p>Allowed values are:</p> <ul style="list-style-type: none"> ▶ An IP address (A.B.C.D or A.D, meaning A.0.0.D). ▶ A range of IP addresses (A.B.C.[D-E]). ▶ An IP subnet (A.B.C.D/M). ▶ * (all addresses). <p>The default is * (meaning all addresses).</p>	OPTIONAL
protocol	<p>The IP protocol to be used as filter for inbound/outbound traffic. The NAT map only applies if the protocol of the IP packet matches the map protocol. Select an IP protocol (see "Supported IP Protocols" on page 748) or, alternatively, type the protocol number. The default is 0 (meaning any protocol).</p>	OPTIONAL

outside_port	The outside port number or range (only for maps of type napt). <ul style="list-style-type: none"> ▶ For a NAPT map: Represents the range of TCP/UDP ports to be used as dynamic port range during the NAPT process. ▶ For a port-shifting map: Represents the source port(s) to be translated (shifted) into inside_port. <p>Allowed values are:</p> <ul style="list-style-type: none"> ▶ TCP/UDP port range [P-Q] (with Q > P) ▶ TCP/UDP port number (only for port-shifting maps). 	OPTIONAL for NAPT map. REQUIRED for port-shifting map.
inside_port	The inside port number or range (only for maps of type napt). If this parameter is specified for a map of type napt , then this map is a port-shifting map. Represents the target TCP/UDP ports to which the traffic has to be shifted. Allowed values are: <ul style="list-style-type: none"> ▶ TCP/UDP port range [P-Q] (with Q > P) ▶ TCP/UDP port number (only for port-shifting maps). <p>Note The size of inside_port must be identical to the size of outside_port.</p>	REQUIRED for port-shifting map
mode	The mode to create the portmap. Choose between: <ul style="list-style-type: none"> ▶ auto ▶ inbound ▶ outbound. <p>The standard is auto.</p>	OPTIONAL

EXAMPLE:

```
=>nat tmpladd intf=myPPP_ppp type=napt outside_addr=100.100.100.1
=>nat tmpllist
Idx Type Interface Ifgroup Outside Address Inside Address Use
 1 NAPT myPPP_ppp any      100.100.100.1 unmapped          0
=>nat tmpladd group=lan type=napt outside_addr=100.200.100.1
=>nat tmpllist
Idx Type Interface Ifgroup Outside Address Inside Address Use
 1 NAPT any           lan      100.200.100.1 unmapped          0
 2 NAPT myPPP_ppp   any      100.100.100.1 unmapped          0
=>
```

RELATED COMMANDS:

- | | |
|-----------------------|--|
| nat tmpldelete | Delete an address mapping template. |
| nat tmpllist | Display the address mapping templates. |
| nat tmplinst | Instantiate address mapping templates for a given dynamic address. |

nat tmpldelete

Delete an address mapping template.

SYNTAX:

```
nat tmpldelete index = <number>
```

where:

index	The index of the address mapping template to be deleted.	REQUIRED
-------	--	----------

Tip Use the command :**nat tmpllist** to view the address mapping templates.

EXAMPLE:

```
=>nat tmpllist
Idx Type Interface      Ifgroup  Outside Address           Inside Address          Use
  1 NAPT any             lan      100.200.100.1         unmapped               0
  2 NAPT myPPP_ppp      any      100.100.100.1         unmapped               0
=>nat tmpldelete index=1
=>nat tmpllist
Idx Type Interface      Ifgroup  Outside Address           Inside Address          Use
  1 NAPT myPPP_ppp      any      100.100.100.1         unmapped               0
=>
```

RELATED COMMANDS:

nat tmpladd	Add an address mapping template.
nat tmpllist	Display the address mapping templates.
nat tmplinst	Instantiate address mapping templates for a given dynamic address.

nat tmpllist

Display the address mapping templates.

SYNTAX:

```
nat tmpllist [expand = <{disabled | enabled}>]
```

where:

expand	Enable or disable expanded listing. The default is disabled .	OPTIONAL
--------	---	----------

EXAMPLE:

```
=>nat tmpllist
Idx Type Interface      Ifgroup  Outside Address           Inside Address          Use
  1 NAPT any             lan      100.200.100.1         unmapped                0
  2 NAPT myPPP_ppp       any      100.100.100.1         unmapped                0
=>
```

RELATED COMMANDS:

nat tmpladd	Add an address mapping template.
nat tmpldelete	Delete an address mapping template.
nat tmplinst	Instantiate address mapping templates for a given dynamic address.

nat tmplinst

Instantiate address mapping templates for a given dynamic address.

SYNTAX:

```
nat tmplinst intf = <string>
            addr_index = <ip-address>
            dynamic_addr = <ip-address>
```

where:

intf	The IP interface name.	REQUIRED
addr_index	The outside IP address index/key to instantiate for.	REQUIRED
dynamic_addr	The dynamic address to substitute the index/key with.	REQUIRED

EXAMPLE:

```
=>nat tmplinst intf=myPPP_ppp addr_index=100.100.100.1 dynamic_addr=200.200.200.1
=>nat tmpllist
Idx Type Interface      Ifgroup   Outside Address           Inside Address      Use
  1 NAPT myPPP_ppp       any      100.100.100.1          unmapped           1
=>
```

RELATED COMMANDS:

- | | |
|----------------|--|
| nat tmpladd | Add an address mapping template. |
| nat tmpldelete | Delete an address mapping template. |
| nat tmpllist | Display the address mapping templates. |

PPP Commands

Introduction

This chapter describes the commands of the **ppp** command group.

Contents

This chapter covers the following commands:

ppp flush	Flush all PPP interfaces.	526
ppp ifadd	Create a new PPP interface.	527
ppp ifattach	Attach a PPP interface.	528
ppp ifconfig	Configure a PPP interface.	530
ppp ifdelete	Delete a PPP interface.	534
ppp ifdetach	Detach a PPP interface.	535
ppp iflist	Display the PPP interfaces.	537
ppp ifscan	Scan a PPPoE interface for available Access Concentrator and Service names.	538
ppp rtadd	Add a route to the routing table when the PPP link comes up.	539
ppp rtdelete	Delete the route for a PPP link.	541
ppp relay flush	Remove all Ethernet interfaces from the PPP relay agent list and terminate all sessions.	543
ppp relay ifadd	Add an Ethernet interface to the PPP relay list.	544
ppp relay ifconfig	Modify an ethernet interface from the PPP relay agent list.	545
ppp relay ifdelete	Delete an Ethernet interface from the PPP relay agent list.	546
ppp relay iflist	Display all Ethernet interfaces added to the PPP relay agent list.	547
ppp relay sesslist	Add an Ethernet interface to the PPP relay list.	548

ppp flush

Flush all PPP interfaces.



The flush command does not impact previously saved configurations.

SYNTAX:

```
ppp flush
```

ppp ifadd

Create a new PPP interface.

SYNTAX:

```
ppp ifadd intf = <string>
```

where:

intf	The name for the new PPP interface.	REQUIRED
Note If not specified, the destination parameter must be specified. In this case the name of the destination will double as interface name.		

EXAMPLE:

```
=>ppp iflist
Internet: dest : RELAY      [00:00:00]
    Retry : 10
    mode = IP routing
    flags = echo magic accomp restart mru addr savepwd demanddial
    dns metric = 0   mru = 1492   Tx inactivity = 1200s   left = 0s
    auth = auto   user =   password =
    admin state = down   oper state = down   link state = not-connected
    LCP : state = initial retrasm = 10   term. reason =
    IPCP: state = initial retrasm = 0   term. reason =
    acname : ---   service : ---


=>ppp ifadd intf=Rt_PPPOA
=>ppp iflist
Internet: dest : RELAY      [00:00:00]
    Retry : 10
    mode = IP routing
    flags = echo magic accomp restart mru addr savepwd demanddial
    dns metric = 0   mru = 1492   Tx inactivity = 1200s   left = 0s
    auth = auto   user =   password =
    admin state = down   oper state = down   link state = not-connected
    LCP : state = initial retrasm = 10   term. reason =
    IPCP: state = initial retrasm = 0   term. reason =
    acname : ---   service : ---


Rt_PPPOA: dest :      [00:00:00]
    Retry : 10
    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 retrasm = 10   term. reason =
    IPCP: state = initial retrasm = 0   term. reason =

=>
```

RELATED COMMANDS:

ppp ifdelete	Delete a PPP interface.
ppp iflist	Display the PPP interfaces.

ppp ifattach

Attach a PPP interface.

SYNTAX:

```
ppp ifattach      intf = <string>
```

where:

intf The name of the PPP interface to be attached.	REQUIRED
---	----------

EXAMPLE:

```
=>ppp iflist
Internet: dest : RELAY      [00:00:00]
    Retry : 10
    mode = IP routing
    flags = echo magic accomp restart mru addr savepwd demanddial
    dns metric = 0   mru = 1492   Tx inactivity = 1200s   left = 0s
    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 : --- 

Rt_PPPOA: dest : RtPPPoA_atm      [00:00:00]
    Retry : 10
    mode = IP routing
    flags = echo magic accomp restart mru addr route savepwd
    dns metric = 0   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 = down   oper state = down   link state = not-connected
    LCP : state = initial   retransm = 10   term. reason =
    IPCP: state = initial   retransm = 0   term. reason =

=>ppp ifattach intf=Rt_PPPOA
=>ppp iflist
Internet: dest : RELAY      [00:00:00]
    Retry : 10
    mode = IP routing
    flags = echo magic accomp restart mru addr savepwd demanddial
    dns metric = 0   mru = 1492   Tx inactivity = 1200s   left = 0s
    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 : --- 

Rt_PPPOA: dest : RtPPPoA_atm      [00:00:00]
    Retry : 10
    mode = IP routing
    flags = echo magic accomp restart mru addr route savepwd
    dns metric = 0   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 = connected
    LCP : state = request   retransm = 1   term. reason =
    IPCP: state = initial   retransm = 0   term. reason =

=>
```

RELATED COMMANDS:

ppp ifdetach Detach a PPP interface.

ppp ifconfig

Configure a PPP interface.



The interface to be configured must not be connected at the time of configuration. If this should be the case, use the command :**ppp ifdetach** before using the command :**ppp ifconfig**.

SYNTAX:

```
ppp ifconfig    intf = <string>
                [dest = <string>]
                [user = <string>]
                [password = <password>]
                [acname = <quoted string>]
                [servicename = <quoted string>]
                [pcOMP = <{disabled | enabled}>]
                [accomp = <{enabled | disabled | negotiate}>]
                [trace = <{disabled | enabled}>]
                [concentrator = <{disabled | enabled}>]
                [auth = <{pap | chap | auto}>]
                [restart = <{disabled | enabled}>]
                [retryinterval = <number{0-65535}>]
                [passive = <{disabled | enabled}>]
                [silent = <{disabled | enabled}>]
                [echo = <{disabled | enabled}>]
                [mru = <number{293-8192}>]
                [laddr = <ip-address>]
                [raddr = <ip-address>]
                [netmask = <ip-mask(dotted or cidr)>]
                [format = <{cidr | dotted | none}>]
                [pool = <none>]
                [savepwd = <{disabled | enabled}>]
                [demanddial = <{disabled | enabled}>]
                [doddelay = <number{0-3600}>]
                [primdns = <ip-address>]
                [secdns = <ip-address>]
                [dnsmetric = <number{0-100}>]
                [idletime = <number{0-1000000}>]
                [idletrigger = <{RxTx | Rx | Tx}>]
                [unnumbered = <{disabled | enabled}>]
```

where:

intf	The name of the PPP interface to be configured.	REQUIRED
dest	The destination for this PPP interface. Typically, a phonebook entry. If an Ethernet interface is given as destination, then the connection will be a PPPoE connection. If an ATM interface is given as destination, then the connection will be a PPPoA connection.	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 session.	OPTIONAL
	Tip Use the command :ppp ifscan to obtain the names of available access concentrators, if any.	
servicename	The Service Name for a PPPoE session.	OPTIONAL
	Tip Use the command :ppp ifscan to obtain the available service names, if any.	
pcomp	Try (enabled) or do not try (disabled) to negotiate PPP protocol compression (LCP PCOMP). The default is disabled .	OPTIONAL
accomp	Try (enabled), do never try (disabled) or negotiate (negotiate) to negotiate PPP address & control field compression (LCP ACCOMP). In most cases, LCP ACCOMP should not be disabled nor negotiated, in other words, the address field FF-03 should not be sent over ATM. The default is enabled . Note If the accomp parameter is set to "negotiate", the local side of the PPP connection demands to do ACCOMP and adapts itself to the result of this negotiation.	OPTIONAL
trace	Enable or disable verbose console logging. The default is disabled .	OPTIONAL
concentrator	The access concentrator is on this side of the PPPoE connection. Choose between: <ul style="list-style-type: none">▶ enabled: the PPP connection is terminated on the Access Concentrator (here the SpeedTouch™ itself)▶ disabled: the SpeedTouch™ is PPP client. The default is disabled .	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. The default is auto .	OPTIONAL
restart	Automatically restart the connection when Link Control Protocol (LCP) link goes down (enabled) or not (disabled). The default is disabled .	OPTIONAL
retryinterval	A number between 0 and 65535 (seconds). Represents the intermediate interval between two retries to establish the connection on ATM level. The default is 10 .	OPTIONAL
passive	Put the link in listening state in case LCP times out (enabled) or not (disabled). 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. The default is disabled .	OPTIONAL
silent	Do not send anything at startup and just listen for incoming LCP messages (enabled) or retry up to 10 times to establish the connection (disabled). The default is disabled .	OPTIONAL

echo	Send LCP echo requests at regular intervals (enabled) or not (disabled). The default is disabled .	OPTIONAL
mru	A number between 293 and 8192. Represents the maximum packet size the SpeedTouch™ should negotiate to be able to receive. The default is 1492 .	OPTIONAL
laddr	The local IP address of the peer-to-peer connection. Specifying a local IP address forces the remote side of the PPP link (if it allows to) to accept this IP address as the SpeedTouch™ PPP 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 PPP link (if it allows to) to accept this IP address as its PPP 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 PPP 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, in other words, 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). The default is cidr .	OPTIONAL
pool	The name of the free DHCP server pool to which the acquired IP subnet must be assigned.	OPTIONAL
savepwd	Save password (enabled), if supplied, or do not save the password (disabled). The default is disabled .	OPTIONAL
demanddial	Enable or disable the dial-on-demand feature. The default is disabled .	OPTIONAL
doddelay	A number between 0 and 3600 (seconds). During this initial interval, packets do not trigger the PPP interface.	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. Note 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. Note 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. The default is 1 .	OPTIONAL
idletime	A number between 0 and 1000000 (seconds). Represents the maximum time the link may be idle. The 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. ▶ 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. <p>The default is RxTx.</p>	OPTIONAL
unnumbered	Takes the local IP address from the <i>laddr</i> field and remote IP address from the IP address pool assigned to the incoming PPP 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:

```
=>ppp iflist
Internet: dest : RELAY      [00:00:00]
    Retry : 10
    mode = IP routing
    flags = echo magic accomp restart mru addr savepwd demanddial
    dns metric = 0   mru = 1492   Tx inactivity = 1200s   left = 0s
    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 : ---


Rt_PPPOA: dest :      [00:00:00]
    Retry : 10
    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 =


=> ppp ifconfig intf=Rt_PPPOA dest=RtPPPoA_atm user=johndoe@ISP password=johndoe
[ppp]=>iflist
Internet: dest : RELAY      [00:00:00]
    Retry : 10
    mode = IP routing
    flags = echo magic accomp restart mru addr savepwd demanddial
    dns metric = 0   mru = 1492   Tx inactivity = 1200s   left = 0s
    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 : ---


Rt_PPPOA: dest : RtPPPoA_atm      [00:00:00]
    Retry : 10
    mode = IP routing
    flags = echo magic accomp restart mru addr savepwd
    dns metric = 0   mru = 1500
    auth = auto   user = johndoe@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 =
=>
```

ppp ifdelete

Delete a PPP interface.

SYNTAX:

```
ppp ifdelete      intf = <intfname>
```

where:

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

EXAMPLE:

```
=>ppp iflist
Internet: dest : RELAY      [00:00:00]
    Retry : 10
    mode = IP routing
    flags = echo magic accomp restart mru addr savepwd demanddial
    dns metric = 0   mru = 1492   Tx inactivity = 1200s   left = 0s
    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 : --- 

Rt_PPPOA: dest : RtPPPoA_atm [local disconnect]      [00:00:00]
    Retry : 10
    mode = IP routing
    flags = echo magic accomp restart mru addr savepwd
    dns metric = 0   mru = 1500
    auth = auto   user = johndoe@ISP   password = *****
    admin state = down   oper state = down   link state = not-connected
    LCP : state = initial   retransm = 9   term. reason = User kill
    IPCP: state = initial   retransm = 0   term. reason =

=>ppp ifdelete intf=Rt_PPPOA
=>ppp iflist
Internet: dest : RELAY      [00:00:00]
    Retry : 10
    mode = IP routing
    flags = echo magic accomp restart mru addr savepwd demanddial
    dns metric = 0   mru = 1492   Tx inactivity = 1200s   left = 0s
    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:

ppp ifadd	Create a new PPP interface.
ppp iflist	Display the PPP interfaces.

ppp ifdetach

Detach a PPP interface.

SYNTAX:

```
ppp ifdetach      intf = <intfname>
```

where:

intf	The name of the PPP interface to be detached.	REQUIRED
------	---	----------

EXAMPLE:

```
=>ppp iflist
Internet: dest : RELAY      [00:00:00]
    Retry : 10
    mode = IP routing
    flags = echo magic accomp restart mru addr savepwd demanddial
    dns metric = 0   mru = 1492   Tx inactivity = 1200s   left = 0s
    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 : --- 

Rt_PPPOA: dest : RtPPPoA_atm      [00:00:00]
    Retry : 10
    mode = IP routing
    flags = echo magic accomp restart mru addr route savepwd
    dns metric = 0   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 = connected
    LCP : state = reqsent   retransm = 5   term. reason =
    IPCP: state = initial   retransm = 0   term. reason =

=>ppp ppp ifdetach intf=Rt_PPPOA
=>ppp iflist
Internet: dest : RELAY      [00:00:00]
    Retry : 10
    mode = IP routing
    flags = echo magic accomp restart mru addr savepwd demanddial
    dns metric = 0   mru = 1492   Tx inactivity = 1200s   left = 0s
    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 : --- 

Rt_PPPOA: dest : RtPPPoA_atm [local disconnect]      [00:00:00]
    Retry : 10
    mode = IP routing
    flags = echo magic accomp restart mru addr route savepwd
    dns metric = 0   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 = down   oper state = down   link state = not-connected
    LCP : state = initial   retransm = 9   term. reason = User kill
    IPCP: state = initial   retransm = 0   term. reason =
=>
```

PPP Commands

RELATED COMMANDS:

ppp ifattach Attach a PPP interface.

ppp iflist

Display the PPP interfaces.

SYNTAX:

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

where:

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

EXAMPLE INPUT/OUTPUT :

```
=>ppp iflist
Internet: dest : RELAY      [00:00:00]
    Retry : 10
    mode = IP routing
    flags = echo magic accomp restart mru addr savepwd demanddial
    dns metric = 0   mru = 1492   Tx inactivity = 1200s   left = 0s
    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 : ---

Rt_PPPOA: dest : RtPPPoA_atm      [00:00:00]
    Retry : 10
    mode = IP routing
    flags = echo magic accomp restart mru addr route savepwd
    dns metric = 0   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 = connected
    LCP : state = reqsent   retransm = 1   term. reason =
    IPCP: state = initial   retransm = 0   term. reason =

=>
```

RELATED COMMANDS:

ppp ifadd	Create a new PPP interface.
ppp ifdelete	Delete a PPP interface.

ppp ifscan

Scan a PPPoE interface for available Access Concentrator and Service names.

 Use the command :**ppp ifdetach** for this interface before performing a scan on it.

SYNTAX:

```
ppp ifscan      intf = <string>
                [time = <number{0-36000}>]
```

where:

intf	The name of the PPPoE interface to be scanned.	REQUIRED
time	A number between 0 and 36000 (seconds). Represents the time to scan for services.	OPTIONAL

EXAMPLE:

```
=>ppp iflist
myRtPPPoE: dest : RtPPPoE_eth      [00:00:00]
    Retry : 10
    mode = IP routing
    flags = echo magic accomp restart mru addr route savepwd
    dns metric = 0   mru = 1492
    route : dst=0.0.0.0/0 - src=10.0.0.0/1 (metric 1)
    auth = auto   user = johndoe@ISP  password = *****
    admin state = down   oper state = down   link state = not-connected
    LCP : state = initial  retransm = 9  term. reason =
    IPCP: state = initial  retransm = 0  term. reason =
    acname : ---  service : ---
=>ppp ifscan intf=myRtPPPoE time=45
          Service Name                  Access Concentrator
Done !
=>
```

ppp rtadd

Add a route to the routing table when the PPP 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.

 The interface must not be connected when a route configuration is added. If this should be the case, use the command :**ppp ifdetach** for this interface prior to configuring routes.

SYNTAX:

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

where:

intf	The name of the PPP interface.	REQUIRED
dst	The IP destination address specification 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, in other words, the PPP connection acts as default route (<i>dstmsk=0</i>) ▶ Only the remote (sub)net is reachable (<i>dstmsk=1</i>) ▶ The actual destination mask will be the default netmask applicable for destination IP address ▶ Only the single remote host is reachable (<i>dstmsk=32</i>) ▶ Any valid (contiguous) netmask in case of Variable Length Subnet Masking (VLSM). 	OPTIONAL
label	The name of the label.	OPTIONAL
src	The IP source address specification for the route (in other words, who can use this link).	OPTIONAL
srcmsk	The source IP mask. Depending on the source netmask: <ul style="list-style-type: none"> ▶ Everybody is allowed to use this PPP connection (<i>dstmsk=0</i>) ▶ Only members of the same subnet as the host which opened the PPP connection are allowed to use the PPP connection (<i>dstmsk=1</i>) ▶ The actual destination mask will be the netmask applicable for the IP address of the host which opened the PPP connection ▶ Only the host which opened the PPP connection is allowed to use the PPP connection (<i>dstmsk=32</i>) ▶ Any valid (contiguous) netmask in case of VLSM. 	OPTIONAL
metric	The route metric, in other words, the cost factor of the route. Practically, the cost is determined by the hop count.	OPTIONAL

EXAMPLE:

```
=>ppp iflist
Internet: dest : RELAY      [00:00:00]
  Retry : 10
  mode = IP routing
  flags = echo magic accomp restart mru addr savepwd demanddial
  dns metric = 0   mru = 1492   Tx inactivity = 1200s   left = 0s
  auth = auto   user =   password =
  admin state = down   oper state = down   link state = not-connected
  LCP : state = initial   retrasm = 10   term. reason =
  IPCP: state = initial   retrasm = 0   term. reason =
  acname : ---   service : --- 

Rt_PPPoA: dest : RtPPPoA_atm      [00:00:00]
  Retry : 10
  mode = IP routing
  flags = echo magic accomp restart mru addr savepwd
  dns metric = 0   mru = 1500
  auth = auto   user = johndoe@ISP   password = *****
  admin state = down   oper state = down   link state = not-connected
  LCP : state = initial   retrasm = 10   term. reason =
  IPCP: state = initial   retrasm = 0   term. reason =

=>ppp rtadd intf=Rt_PPPoA dst=0.0.0.0/0 src=10.0.0.0/1
=>ppp iflist
Internet: dest : RELAY      [00:00:00]
  Retry : 10
  mode = IP routing
  flags = echo magic accomp restart mru addr savepwd demanddial
  dns metric = 0   mru = 1492   Tx inactivity = 1200s   left = 0s
  auth = auto   user =   password =
  admin state = down   oper state = down   link state = not-connected
  LCP : state = initial   retrasm = 10   term. reason =
  IPCP: state = initial   retrasm = 0   term. reason =
  acname : ---   service : --- 

Rt_PPPoA: dest : RtPPPoA_atm      [00:00:00]
  Retry : 10
  mode = IP routing
  flags = echo magic accomp restart mru addr route savepwd
  dns metric = 0   mru = 1500
  route : dst=0.0.0/0 - src=10.0.0.0/1 (metric 1)
  auth = auto   user = johndoe@ISP   password = *****
  admin state = down   oper state = down   link state = not-connected
  LCP : state = initial   retrasm = 10   term. reason =
  IPCP: state = initial   retrasm = 0   term. reason =
=>
```

RELATED COMMANDS:

ppp rtdelete Delete the route for a PPP link.

ppp rtdelete

Delete the route for a PPP link.



The interface must not be connected when a route configuration must be deleted. If the interface is connected, use the command :**ppp ifdetach** for this interface.

SYNTAX:

```
ppp rtdelete      intf = <intfname>
```

where:

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

EXAMPLE:

```
=>ppp iflist
Internet: dest : RELAY      [00:00:00]
  Retry : 10
  mode = IP routing
  flags = echo magic accomp restart mru addr savepwd demanddial
  dns metric = 0   mru = 1492   Tx inactivity = 1200s   left = 0s
  auth = auto   user =   password =
  admin state = down   oper state = down   link state = not-connected
  LCP : state = initial   retrasm = 10   term. reason =
  IPCP: state = initial   retrasm = 0   term. reason =
  acname : ---   service : --- 

Rt_PPPoA: dest : RtPPPoA_atm [local disconnect]      [00:00:00]
  Retry : 10
  mode = IP routing
  flags = echo magic accomp restart mru addr route savepwd
  dns metric = 0   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 = down   oper state = down   link state = not-connected
  LCP : state = initial   retrasm = 9   term. reason = User kill
  IPCP: state = initial   retrasm = 0   term. reason =

=>ppp ppp rtdelete intf=Rt_PPPoA
=>ppp iflist
Internet: dest : RELAY      [00:00:00]
  Retry : 10
  mode = IP routing
  flags = echo magic accomp restart mru addr savepwd demanddial
  dns metric = 0   mru = 1492   Tx inactivity = 1200s   left = 0s
  auth = auto   user =   password =
  admin state = down   oper state = down   link state = not-connected
  LCP : state = initial   retrasm = 10   term. reason =
  IPCP: state = initial   retrasm = 0   term. reason =
  acname : ---   service : --- 

Rt_PPPoA: dest : RtPPPoA_atm [local disconnect]      [00:00:00]
  Retry : 10
  mode = IP routing
  flags = echo magic accomp restart mru addr savepwd
  dns metric = 0   mru = 1500
  auth = auto   user = johndoe@ISP   password = *****
  admin state = down   oper state = down   link state = not-connected
  LCP : state = initial   retrasm = 9   term. reason = User kill
  IPCP: state = initial   retrasm = 0   term. reason =

=>
```

RELATED COMMANDS:

ppp rtadd Add a route to the routing table when the PPP link comes up.

ppp relay flush

Remove all Ethernet interfaces from the PPP relay agent list and terminate all sessions.



The flush command does not impact previously saved configurations.

SYNTAX:

```
ppp relay flush
```

ppp relay ifadd

Add an Ethernet interface to the PPP relay list.

SYNTAX:

```
ppp relay ifadd intf = <string>
```

where:

intf	The Ethernet interface to be added to the PPP relay agent list.	REQUIRED
------	---	----------

EXAMPLE:

```
=>ppp relay iflist
# Interface          HWaddr          Status
1 ethoa_0_35         00-0e-50-0f-fc-2d connected
=>ppp relay ifadd intf=ethoa_8_35
=>ppp relay iflist
# Interface          HWaddr          Status
1 ethoa_0_35         00-0e-50-0f-fc-2d connected
2 ethoa_8_35         00-0e-50-0f-fc-2d connected
=>
```

RELATED COMMANDS:

[ppp relay ifdelete](#) Delete an Ethernet interface from the PPP relay agent list.

[ppp relay iflist](#) Display all Ethernet interfaces added to the PPP relay agent list.

ppp relay ifconfig

Modify an ethernet interface from the PPP relay agent list.

SYNTAX:

```
ppp relay ifconfig    intf = <string>
                      hwaddr = <hardware-address>
```

where:

intf	The ethernet intf to be added to the PPP relay agent list.	REQUIRED
hwaddr	The hardware address (e.g. Ethernet MAC address) of this interface.	REQUIRED

ppp relay ifdelete

Delete an Ethernet interface from the PPP relay agent list.

SYNTAX:

```
ppp relay ifdelete intf = <string>
```

where:

intf	The Ethernet interface to be deleted from the PPP relay agent list.	REQUIRED
------	---	----------

EXAMPLE:

```
=>ppp relay iflist
# Interface          HWaddr          Status
1 ethoa_0_35         00-0e-50-0f-fc-2d connected
2 ethoa_8_35         00-0e-50-0f-fc-2d connected
=>ppp relay ifdelete intf=ethoa_8_35
=>ppp relay iflist
# Interface          HWaddr          Status
1 ethoa_0_35         00-0e-50-0f-fc-2d connected
=>
```

RELATED COMMANDS:

ppp relay ifadd Add an Ethernet interface to the PPP relay list.

ppp relay iflist Display all Ethernet interfaces added to the PPP relay agent list.

ppp relay iflist

Display all Ethernet interfaces added to the PPP relay agent list.

SYNTAX:

```
ppp relay iflist
```

where:

intf	The Ethernet interface to be added to the PPP relay agent list.	REQUIRED
------	---	----------

EXAMPLE:

```
=>ppp relay iflist
# Interface          HWaddr        Status
1 ethoa_0_35         00-0e-50-0f-fc-2d connected
2 ethoa_8_35         00-0e-50-0f-fc-2d connected
=>
```

RELATED COMMANDS:

ppp relay ifadd Add an Ethernet interface to the PPP relay list.

ppp relay ifdelete Delete an Ethernet interface from the PPP relay agent list.

ppp relay sesslist

Add an Ethernet interface to the PPP relay list.

SYNTAX:

```
ppp relay ifadd intf = <string>
```

where:

intf	The Ethernet interface to be added to the PPP relay agent list.	REQUIRED
------	---	----------

PPTP Commands

Introduction

This chapter describes the commands of the **pptp** command group.

Contents

This chapter covers the following commands:

pptp ifadd	Add a Point-to-Point Tunneling Protocol (PPTP) profile.	550
pptp flush	Flush the complete PPTP configuration.	551
pptp list	Show the current PPTP configuration.	552
pptp profadd	Define a new PPTP profile.	553
pptp profdelete	Delete a PPTP profile.	554
pptp proflist	Display all the current PPTP profiles.	555

pptp ifadd

Add 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}>]
```

where:

dest	The WAN destination for this PPTP tunnel. Typically a phonebook entry.	REQUIRED
rate	A number between 10 and 10000. Represents the transmission speed (in bits/s) for the WAN link.	OPTIONAL
encaps	The type of WAN encapsulation to be used with this PPTP profile. Choose between: <ul style="list-style-type: none">▶ VC_MUX▶ Network Layer Protocol IDentifiers (NLPID). The default is vcmux .	OPTIONAL
ac	The High-level Data Link Control (HDLC) framing option applicable to PPTP interfaces using this PPTP profile. Choose between: <ul style="list-style-type: none">▶ always: 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.▶ never: Before relaying the encapsulated PPP frames over the PPPoA link, make sure the address and control field will never be found in front of the frames.▶ keep: Do not change the frames arriving via the PPTP tunnel. The default is never .	OPTIONAL
Note	The default setting is compliant to RFC2364, therefore it is recommended to keep this setting.	

pptp flush

Flush the complete PPTP configuration.



The flush command does not impact previously saved configurations.

SYNTAX:

```
pptp flush
```

pptp list

Show the current PPTP configuration.

SYNTAX:

```
pptp list
```

EXAMPLE:

```
=>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}>]
```

where:

name	The name of the new PPTP profile.	REQUIRED
qos	The name of the qosbook entry, containing the settings for this profile.	OPTIONAL
Note	This parameter never needs to be specified.	
encaps	The type of WAN protocol encapsulation to be used with this PPTP profile. Choose between: <ul style="list-style-type: none">▶ VC-MUX▶ NLPIID. The default is vcmux .	OPTIONAL
ac	The HDLC framing option applicable to PPTP interfaces using this PPTP profile. Choose between: <ul style="list-style-type: none">▶ always: 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.▶ never: Before relaying the encapsulated PPP frames over the PPPoA link, make sure the address and control field will never be found in front of the frames.▶ keep: Do not change the frames arriving via the PPTP tunnel. The default is never .	OPTIONAL
Note	The default setting is compliant to RFC2364, therefore it is recommended to keep this setting.	

EXAMPLE:

```
=>pptp proplist
Profile      QoS      Encaps      AC
Relay_PPP1   default  nlpid      always
=>pptp profadd name=PPTPLink encaps=vcmux ac=never
=>pptp proplist
Profile      QoS      Encaps      AC
Relay_PPP1   default  nlpid      always
PPTPLink    default  vcmux     never
=>
```

RELATED COMMANDS:

pptp profdelete	Define a new PPTP profile.
pptp proplist	Display all the current PPTP profiles.

pptp profdelete

Delete a PPTP profile.

SYNTAX:

```
pptp profdelete name <string>
```

where:

name The name of the PPTP profile to be deleted.

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 Display all the current PPTP profiles.

pptp proplist

Display all the current PPTP profiles.

SYNTAX:

```
pptp proplist
```

EXAMPLE:

```
=>pptp proplist
Profile      QoS      Encaps      AC
Relay_PPP1    default  nlpid      always
PPTPLink     default  vcmux     never
=>
```

RELATED COMMANDS:

pptp profadd	Define a new PPTP profile.
pptp profdelete	Delete a PPTP profile.

Script Commands

Introduction

This chapter describes the commands of the **script** command group.

Scripting is not a general purpose mechanism but is only used in the autoPVC/ILMI mechanism.



It is recommended not to change the default scripts.

Contents

This chapter covers the following commands:

script add	Add a line to a script.	558
script delete	Delete a complete script or a line from a script.	559
script flush	Flush all scripts.	560
script list	Display a script or all scripts.	561
script run	Run a script.	562

script add

Add a line to a script.

SYNTAX:

```
script add name = <string>
            [index = <number>]
            command = <quoted string>
```

where:

name	Name of the script in which a line must be added.	REQUIRED
index	Line number	OPTIONAL
Note	Use 0 to add a line.	
command	Command.	REQUIRED

RELATED COMMANDS:

script delete	Delete a complete script or a line from a script.
script list	Display a script or all scripts.

script delete

Delete a complete script or a line from a script.

SYNTAX:

```
script delete name = <string>
               [index = <number>]
```

where:

name	Name of the script to be deleted.	REQUIRED
index	Line number to be deleted.	OPTIONAL
Tip Use the command :script list to view the line numbers.		
Note If not specified, the complete script will be deleted.		

RELATED COMMANDS:

script add	Add a line to a script.
script list	Display a script or all scripts.

script flush

Flush all scripts.



The flush command does not impact previously saved configurations.

SYNTAX:

```
script flush
```

script list

Display a script or all scripts.

SYNTAX:

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

where:

name	Name of the script to be listed.	OPTIONAL
Note	If not specified, all the scripts are displayed.	

EXAMPLE

Some of the default scripts are shown below:

```
=>script list
Script: autopvc_add_qos
  0: qosbook add name _auto_$1_$2 class $3 tx_peakrate $4 tx_sustrate $5 tx_maxburst $6
     rx_peakrate $4 rx_sustrate $5 rx_maxburst $6 dynamic yes
...
Script: autopvc_add_bridge
  0: qosbook add name _auto_$1_$2 class $3 tx_peakrate $4 tx_sustrate $5 tx_maxburst $6
     rx_peakrate $4 rx_sustrate $5 rx_maxburst $6 dynamic yes
  1: phonebook add name _auto_$1_$2 addr $1.$2 type any dynamic yes
  2: bridge ifadd intf _auto_$1_$2 dest _auto_$1_$2
  3: bridge ifconfig intf _auto_$1_$2 qos _auto_$1_$2
  4: bridge ifattach intf _auto_$1_$2

Script: autopvc_delete_bridge
  0: bridge ifdetach intf _auto_$1_$2
  1: bridge ifdelete intf _auto_$1_$2
  2: phonebook delete name _auto_$1_$2
  3: qosbook delete name _auto_$1_$2

Script: autopvc_add_ppoerelay
  0: qosbook add name _auto_$1_$2 class $3 tx_peakrate $4 tx_sustrate $5 tx_maxburst $6
     rx_peakrate $4 rx_sustrate $5 rx_maxburst $6 dynamic yes
  1: phonebook add name _auto_$1_$2 addr $1.$2 type any dynamic yes
  2: ethoa ifadd intf _auto_$1_$2 dest _auto_$1_$2
  3: ethoa ifconfig intf _auto_$1_$2 qos _auto_$1_$2
  4: ethoa ifattach intf _auto_$1_$2
  5: ip iwait intf _auto_$1_$2 timeout 15 adminstatus up
  6: pppoe relay add port _auto_$1_$2
...
=>
```

RELATED COMMANDS:

script add	Add a line to a script.
script delete	Delete a complete script or a line from a script.

script run

Run a script.

SYNTAX:

```
script run    name = <string>
              pars = <string>
```

where:

name	Name of the script to be run.	REQUIRED
Tip Use the command :script list to obtain the names of the different scripts.		
pars	Parameters separated with comma. For example a,b,c.	REQUIRED

Service Commands

Introduction

This chapter describes the commands of the **service** command group.

Contents

This chapter covers the following commands:

service host assign	Assign a host service to a LAN device.	564
service host config	Get/set global host service configuration options.	565
service host add	Add a host service.	566
service host delete	Delete a host service.	567
service host disable	Disable a host service.	568
service host flush	Flush all host services.	569
service host list	Display a list of host services.	570
service host stats	Show the host service statistics.	571
service host triggerlist	List all triggers.	572
service host rule add	Create/define a host service portmap.	573
service host rule delete	Delete a host service portmap.	574
service system ifadd	Add an interface group to the access list.	575
service system ifdelete	Delete an interface group from the access list.	576
service system ipadd	Add an IP address (range) to the access list.	577
service system ipdelete	Delete an IP address (range) from the access list.	578
service system list	Display the system services.	579
service system mapadd	Add a port map for a system service.	581
service system mapdelete	Delete a port map for a system service.	582
service system modify	Modify a system service.	583

service host assign

Assign a host service to a LAN device.

SYNTAX:

```
service host assign name = <quoted string>
                  [host = <ip-address>]
                  [log = <{disabled|enabled}>]
```

where:

name	The name of an existing host service.	REQUIRED
host	The IP address of the LAN device.	OPTIONAL
log	Enable or disable logging.	OPTIONAL

EXAMPLE:

```
=>service host assign name="MSN Messenger" host=192.168.1.64
=>service host list
Service Name                      Host          User-Defined Mode
-----
Age of Empires                    unassigned      server
AIM Talk                          unassigned      client
Aliens vs. Predator              unassigned      server
..
MSN Messenger                     192.168.1.64    server
...
=>
```

RELATED COMMANDS:

service host disable Disable a host service.

service host config

Get/set global host service configuration options.

SYNTAX:

```
service host config [trace = <{disabled | enabled}>]
```

where:

trace	Enable or disable traces. The default is disabled .	REQUIRED
-------	---	----------

EXAMPLE:

```
=>service host config
Service traces : disabled
l=>
```

Service Commands

service host add

Add a host service.

SYNTAX:

```
service host add      name = <quoted string>
                      [mode = <{server | client | custom}>]
```

where:

name	The name of the new host service.	REQUIRED
mode	The service mode. Choose between: <ul style="list-style-type: none">▶ server: The service is defined for server purposes.▶ client: The service is defined for client purposes.▶ custom: The service is a user created service (in other words, a customized service).	OPTIONAL
The default is custom .		

EXAMPLE:

```
=>service host list
Service Name          Host           User-Defined Mode
-----
Age of Empires        unassigned     server
AIM Talk              unassigned     client
...
MSN Messenger         unassigned     server
=>service host add name=myService
=>service host list
Service Name          Host           User-Defined Mode
-----
Age of Empires        unassigned     server
AIM Talk              unassigned     client
...
MSN Messenger         unassigned     server
myService             unassigned     yes
=>
```

RELATED COMMANDS:

service host delete	Delete a host service.
service host list	Display a list of host services.

service host delete

Delete a host service.

SYNTAX:

```
service host delete name = <quoted string>
```

where:

name	The name of the host service to be deleted.	REQUIRED
------	---	----------

EXAMPLE:

```
=>service host list
Service Name          Host      User-Defined Mode
-----
Age of Empires       unassigned
AIM Talk             unassigned
...
MSN Messenger        unassigned
myService            unassigned    yes
=>service host delete name=myService
=>service host list
Service Name          Host      User-Defined Mode
-----
Age of Empires       unassigned
AIM Talk             unassigned
...
MSN Messenger        unassigned
=>
```

RELATED COMMANDS:

service host add	Add a host service.
service host list	Display a list of host services.

service host disable

Disable a host service.

SYNTAX:

```
service host disable [name = <quoted string>]
```

where:

name	The name of the host service to be disabled.	OPTIONAL
Note	If not specified, all the host services will be disabled.	

EXAMPLE:

```
=>service host list
Service Name          Host      User-Defined Mode
-----
Age of Empires        unassigned    server
AIM Talk              unassigned    client
Aliens vs. Predator   unassigned    server
..
MSN Messenger         192.168.1.64  server
...
=>service host disable name="MSN Messenger"
=>service host list
Service Name          Host      User-Defined Mode
-----
Age of Empires        unassigned    server
AIM Talk              unassigned    client
Aliens vs. Predator   unassigned    server
..
MSN Messenger         unassigned    server
...
=>
```

RELATED COMMANDS:

service host assign Assign a host service to a LAN device.

service host flush

Flush all host services.

SYNTAX:

```
service host flush
```

service host list

Display a list of host services.

SYNTAX:

```
service host list [name = <quoted string>]
```

where:

name	The name of the host service to be listed.	REQUIRED
Note	If not specified, all the host services will be listed.	

EXAMPLE:

```
=>service host list
Service Name          Host      User-Defined Mode
-----
Aliens vs. Predator   unassigned    server
Asheron's Call        unassigned    client
Battlecom              unassigned    server
Black and White       unassigned    server
Buddy Phone           unassigned    client
Bungie.net             unassigned    server
Citrix Metaframe      unassigned    client
CU-SeeMe               unassigned    client
Dark Reign 2           unassigned    server
...
Westwood Online         unassigned    client
Yahoo Messenger Chat   unassigned    server
=>
=>service host list name="MSN Messenger"
Service Name:"MSN Messenger" Host:unassigned User-Defined:no Mode:server
-----
Port 1863 for protocol tcp will be forwarded to host port 1863
Portrange 6891 - 6900 for protocol tcp will be forwarded to host portrange 6891 - 6900
Port 6901 for protocol tcp or udp will be forwarded to host port 6901
=>
```

RELATED COMMANDS:

service host add	Add a host service.
service host delete	Delete a host service.

service host stats

Show the host service statistics.

SYNTAX:

```
service host stats
```

EXAMPLE:

```
=>service host stats
Services      : 113 of 150 in use.
Service maps  : 225 of 300 in use.
Trigger ports: 0 of 25 in use.
=>
```

service host triggerlist

List all triggers.

SYNTAX:

```
service host triggerlist
```

EXAMPLE:

=>service triggerlist			
Ip	Triggerport	Portrange	Timeout
--	-----	-----	-----
=>			

service host rule add

Create/define a host service portmap.

SYNTAX:

```
service host rule add name = <quoted string>
    [protocol = <{any|tcp|udp} or number>]
    [baseport = <supported UDP port | number>]
    portrange = <port-range>
    [triggerport = <supported UDP port | number>]
    [triggerprotocol = <{any|tcp|udp} or number>]
```

where:

name	The name of the host service.	REQUIRED
protocol	The IP protocol type. Choose between: <ul style="list-style-type: none">▶ any▶ tcp▶ udp▶ a number.	OPTIONAL
baseport	The inbound base port. Select one of the supported TCP/UDP port names (see “ Supported TCP/UDP Port Names ” on page 749) or, alternatively, specify the port number. Note If not specified, baseport is equal to the first port of portrange .	OPTIONAL
portrange	The outbound port range.	REQUIRED
triggerport	The outbound trigger port. Select one of the supported TCP/UDP port names (see “ Supported TCP/UDP Port Names ” on page 749) or, alternatively, specify the port number.	OPTIONAL
triggerprotocol	Protocol of the trigger port. Choose between: <ul style="list-style-type: none">▶ any▶ tcp▶ udp▶ a number. Note If not specified, triggerprotocol is equal to protocol .	OPTIONAL

RELATED COMMANDS:

service host rule delete Delete a host service portmap.

service host rule delete

Delete a host service portmap.

SYNTAX:

```
service host rule delete name = <quoted string>
                           [protocol = <{any|tcp|udp} or number>]
                           [baseport = <supported UDP port | number>]
                           portrange = <port-range>
                           [triggerport = <supported UDP port | number>]
                           [triggerprotocol = <{any|tcp|udp} or number>]
```

where:

name	The name of the host service.	REQUIRED
protocol	The IP protocol type. Choose between: <ul style="list-style-type: none">▶ any▶ tcp▶ udp▶ a number.	OPTIONAL
baseport	The inbound base port. Select one of the supported TCP/UDP port names (see “Supported TCP/UDP Port Names” on page 749) or, alternatively, specify the port number. Note If not specified, baseport is equal to the first port of portrange .	OPTIONAL
portrange	The outbound port range.	REQUIRED
triggerport	The outbound trigger port. Select one of the supported TCP/UDP port names (see “Supported TCP/UDP Port Names” on page 749) or, alternatively, specify the port number.	OPTIONAL
triggerprotocol	Protocol of the trigger port. Choose between: <ul style="list-style-type: none">▶ any▶ tcp▶ udp▶ a number. Note If not specified, triggerprotocol is equal to protocol .	OPTIONAL

RELATED COMMANDS:

service host rule add Create/define a host service portmap.

service system ifadd

Add an interface group to the access list.

SYNTAX:

```
service system ifadd    name = <string>
                      group = <{wan|local|lan|tunnel|dmz|guest} or number>
```

where:

name	The name of the system service for this access list.	REQUIRED
group	The interface group for this access list.	REQUIRED

EXAMPLE:

```
=>service system list name=SIPPBX expand=enabled
Idx Name          Protocol      SrcPort DstPort Group        State
-----
1 SIPPBX         udp           5060      disabled
Description..... SIP PBX, registrar and proxy
Properties..... server
Managed parameters... state port acl map log
Source Ip Selection.... auto
Interface Access List.... any
Ip Access List..... any
NAT Port List..... 5060
=>service system ifadd name=SIPPBX group=lan
=>service system list name=SIPPBX expand=enabled
Idx Name          Protocol      SrcPort DstPort Group        State
-----
1 SIPPBX         udp           5060      disabled
Description..... SIP PBX, registrar and proxy
Properties..... server
Managed parameters... state port acl map log
Source Ip Selection.... auto
Interface Access List.... lan
Ip Access List..... any
NAT Port List..... 5060
=>
```

RELATED COMMANDS:

service system ifdelete Delete an interface group from the access list.

Service Commands

service system ifdelete

Delete an interface group from the access list.

SYNTAX:

```
service system ifdelete name = <string>
                  group = <{wan|local|lan|tunnel|dmz|guest} or number>
```

where:

name	The name of the system service for this access list.	REQUIRED
group	The interface group for this access list.	REQUIRED

EXAMPLE:

```
=>service system list name=SIPPBX expand=enabled
Idx Name          Protocol      SrcPort DstPort Group        State
-----
1 SIPPBX         udp           5060      disabled
Description..... SIP PBX, registrar and proxy
Properties..... server
Managed parameters... state port acl map log
Source Ip Selection.... auto
Interface Access List.... lan
Ip Access List..... any
NAT Port List..... 5060
=>service system ifdelete name=SIPPBX group=lan
=>service system list name=SIPPBX expand=enabled
Idx Name          Protocol      SrcPort DstPort Group        State
-----
1 SIPPBX         udp           5060      disabled
Description..... SIP PBX, registrar and proxy
Properties..... server
Managed parameters... state port acl map log
Source Ip Selection.... auto
Interface Access List.... any
Ip Access List..... any
NAT Port List..... 5060
=>
```

RELATED COMMANDS:

service system ifadd Add an interface group to the access list.

service system ipadd

Add an IP address (range) to the access list.

SYNTAX:

```
service system ipadd    name = <string>
                      ip = <ip-range>
```

where:

name	The name of the system service for this access list.	REQUIRED
ip	The IP address (range) for this access list.	REQUIRED

EXAMPLE:

```
=>service system list name=SIPPBX expand=enabled
Idx Name          Protocol      SrcPort   DstPort  Group        State
-----
1 SIPPBX         udp           5060      disabled
Description..... SIP PBX, registrar and proxy
Properties..... server
Managed parameters... state port acl map log
Source Ip Selection.... auto
Interface Access List.... any
Ip Access List..... any
NAT Port List..... 5060
=>service system ipadd name=SIPPBX ip=192.168.1.64
=>service system list name=SIPPBX expand=enabled
Idx Name          Protocol      SrcPort   DstPort  Group        State
-----
1 SIPPBX         udp           5060      disabled
Description..... SIP PBX, registrar and proxy
Properties..... server
Managed parameters... state port acl map log
Source Ip Selection.... auto
Interface Access List.... any
Ip Access List..... 192.168.1.64
NAT Port List..... 5060
=>
```

RELATED COMMANDS:

service system ipdelete Delete an IP address (range) from the access list.

Service Commands

service system ipdelete

Delete an IP address (range) from the access list.

SYNTAX:

```
service system ipdelete name = <string>
                  ip = <ip-range>
```

where:

name	The name of the system service for this access list.	REQUIRED
ip	The IP address (range) for this access list.	REQUIRED

EXAMPLE:

```
=>service system list name=SIPPBX expand=enabled
Idx Name          Protocol      SrcPort DstPort Group      State
-----
1 SIPPBX         udp           5060      disabled
Description..... SIP PBX, registrar and proxy
Properties..... server
Managed parameters.... state port acl map log
Source Ip Selection.... auto
Interface Access List.... any
Ip Access List..... 192.168.1.64
NAT Port List..... 5060
=>service system ipdelete name=SIPPBX ip=192.168.1.64
=>service system list name=SIPPBX expand=enabled
Idx Name          Protocol      SrcPort DstPort Group      State
-----
1 SIPPBX         udp           5060      disabled
Description..... SIP PBX, registrar and proxy
Properties..... server
Managed parameters.... state port acl map log
Source Ip Selection.... auto
Interface Access List.... any
Ip Access List..... any
NAT Port List..... 5060
=>
```

RELATED COMMANDS:

service system ipadd Add an IP address (range) to the access list.

service system list

Display the system services.

SYNTAX:

```
service system list [name = <string>]
                    [expand = <{disabled | enabled}>]
                    [dynamics = <{disabled | enabled}>]
                    [members = <{disabled | enabled}>]
```

where:

name	The name of the system service to be displayed.	OPTIONAL
	Note If not specified, all the system services will be displayed.	
expand	Enable or disable expanded listing. The default is disabled .	OPTIONAL
dynamics	Display dynamic sevices. The default is disabled .	OPTIONAL
members	Display service group members. The default is disabled .	OPTIONAL

Service Commands

EXAMPLE:

```
=>service system list
Idx Name          Protocol      SrcPort  DstPort  Group      State
-----
1  DNS-C         udp           53
2  SNTP          udp           123     123
3  SLA_ICMP_PING  icmp
4  SLA_UDP_PING  udp           8
5  SLA_ICMP_TRRT  icmp
6  SLA_UDP_TRRT  udp           7
7  SYSLOG         udp           33434
8  HTTP           tcp           514
9  HTTPs          tcp           80
10 HTTPPI         tcp           443
11 FTP            tcp           8080
12 TELNET         tcp           21
13 RIP            udp           23
14 RIP-Query      udp           520
15 DNS-S          udp           520
16 Dynamic DNS
17 DHCP-S         udp           53
18 SNMP_AGENT     udp           49152
19 SNMP_TRAPS    udp           161
20 MDAP           udp           3235
21 SIPPBX         udp           5060
22 IKE             udp           500
23 IP_COMMANDS
24 PING_RESPONDER  icmp          8
=>
=>service system list name=SIPPBX expand=enabled
Idx Name          Protocol      SrcPort  DstPort  Group      State
-----
1  SIPPBX         udp           5060
Description..... SIP PBX, registrar and proxy
Properties..... server
Managed parameters..... state port acl map log
Source Ip Selection..... auto
Interface Access List..... any
Ip Access List..... any
NAT Port List..... 5060
=>
```

RELATED COMMANDS:

service system modify Modify a system service.

service system mapadd

Add a port map for a system service.

SYNTAX:

```
service system mapadd name = <{string}>
                    intf = <{auto|loop|Internet|LocalNetwork}>
                    port = <supported port or number>
```

where:

name	The name of the system service for this map.	REQUIRED
intf	The interface for this map.	REQUIRED
port	The port for this map. Select one of the supported TCP/UDP port names (see “ Supported TCP/UDP Port Names ” on page 749) or, alternatively, specify the port number.	REQUIRED

RELATED COMMANDS:

[service system mapdelete](#) Delete a port map for a system service.

service system mapdelete

Delete a port map for a system service.

SYNTAX:

```
service system mapdelete name = <string>
                           intf = <{auto|loop|Internet|LocalNetwork}>
                           port = <supported port or number>
```

where:

name	The name of the system service for this map.	REQUIRED
intf	The interface for this map.	REQUIRED
port	The port for this map. Select one of the supported TCP/UDP port names (see “ Supported TCP/UDP Port Names ” on page 749) or, alternatively, specify the port number.	REQUIRED

RELATED COMMANDS:

service system mapadd Add a port map for a system service.

service system modify

Modify a system service.

SYNTAX:

```
service system modify name = <string>
    [state = <{disabled | enabled}>]
    [port = <supported port or number>]
    [srcintf = <string>]
    [log = <{disabled|enabled}>]
    [forward = <{disabled|enabled}>]
```

where:

name	The name of the system service for this map.	REQUIRED
state	Disable or enable this system service.	OPTIONAL
port	The port for this map. Select one of the supported TCP/UDP port names (see “ Supported TCP/UDP Port Names ” on page 749) or, alternatively, specify the port number.	OPTIONAL
srcintf	The primary IP interface for this system service.	OPTIONAL
log	Disable or enable service logging	OPTIONAL
forward	Disable or enable service forwarding	OPTIONAL

RELATED COMMANDS:

[service system list](#) Display the system services.

SNMP Commands

Introduction

This chapter describes the commands of the **snmp** command group.

Contents

This chapter covers the following commands:

snmp config	Show/set global Simple Network Management Protocol (SNMP) parameters.	586
snmp get	Get from the supplied SNMP Object IDentifier (OID).	587
snmp getnext	GetNext from the supplied SNMP OID.	588
snmp walk	Walk from the supplied SNMP OID.	589
snmp community add	Configure an SNMP community string to allow SNMP access over IP.	590
snmp community delete	Delete an SNMP community string to prevent SNMP access over IP.	591
snmp community list	List all SNMP community strings in use for SNMP access over IP.	592
snmp community modify	Modify an SNMP community string to allow SNMP access over IP.	593
snmp ifadd	Create a new SNMP interface.	594
snmp ifdelete	Delete an SNMP interface.	595
snmp ifattach	Attach an SNMP interface.	596
snmp ifdetach	Detach an SNMP interface.	597
snmp ifconfig	Modify an SNMP interface.	598
snmp iflist	Display the SNMP interfaces.	599

snmp config

Show/set global Simple Network Management Protocol (SNMP) parameters.

SYNTAX:

```
snmp config [sysContact = <quoted string>]
              [sysName = <quoted string>]
              [sysLocation = <quoted string>]
```

where:

sysContact	The SNMP system contact. The default is Service Provider .	OPTIONAL
sysName	The SNMP system name.	OPTIONAL
sysLocation	The SNMP system location. The default is Customer Premises .	OPTIONAL

EXAMPLE:

```
=>snmp config
SNMP System Contact      : Service Provider
SNMP System Name         : SpeedTouch 620
SNMP System Location     : Customer Premises
All SNMP traps           : ENABLED
Delay, in secs before first trap is sent      : 90
=>
```

snmp get

Get from the supplied SNMP Object IDentifier (OID).

For example: get ObjectId=.1.3.6.1.2.1.1.1.0.

SYNTAX:

```
snmp get [ObjectId = <string>]
```

where:

ObjectID The Object Identifier. Object ID to get from ... must include the instance which is 0 for scalar objects, for example .1.3.6.1.2.1.1.1.0 sysDescription.

Note If not specified, the sysDescription OID .1.3.6.1.2.1.1.1.0 is assumed. Its value is SpeedTouch™.

EXAMPLE:

```
=>snmp get  
VB_octetStr .1.3.6.1.2.1.1.1.0 SpeedTouch™ 620
```

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 getnext [ObjectId = <string>]
```

where:

ObjectId	The Object Identifier.	OPTIONAL
	Object ID to getnext from for example .1.3.6.1.2.1.1 system returns sysDescription.	

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 Object Identifier (OID).
snmp walk	Walk from the supplied SNMP OID.

snmp walk

Walk from the supplied SNMP OID.

SYNTAX:

```
snmp walk [ObjectId = <string>]
```

where:

ObjectId	The Object Identifier.	OPTIONAL
	Object ID to walk from for example .1.3.6.1.2.1.1 system walks the system group.	

EXAMPLE:

```
=>snmp walk ObjectId=.1.3.6.1.2.1.1
VB_octetStr   .1.3.6.1.2.1.1.1.0          SpeedTouch™ 620
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 Object IDentifier (OID).
snmp getnext	GetNext from the supplied SNMP OID.

snmp community add

Configure an SNMP community string to allow SNMP access over IP.

SYNTAX:

```
snmp community add securityname = <{RWCommunity | ROCommunity}>
communityname = <password>
```

where:

securityname	Configure access rights/restrictions control for the community name. Choose between: ▶ RWCommunity : read/write access rights ▶ ROCommunity : read only access rights.	REQUIRED
communityname	Configure the SNMP community name.	REQUIRED

EXAMPLE:

```
=>snmp community add
securityname = RWCommunity
communityname = *****
Please retype communityname for verification.
communityname = *****
:>snmp community add securityname=RWCommunity communityname=_DEV_2C6A78E1C41E7B01
=>snmp community add
securityname = ROCommunity
communityname = *****
Please retype communityname for verification.
communityname = *****
:>snmp community add securityname=ROCommunity communityname=_DEV_184B05F89719A74E
=>
```

RELATED COMMANDS:

snmp community delete	Delete an SNMP community string to prevent SNMP access over IP.
snmp community list	List all SNMP community strings in use for SNMP access over IP.

snmp community delete

Delete an SNMP community string to prevent SNMP access over IP.

SYNTAX:

```
snmp community delete securityname = <{RWCommunity | ROCommunity}>
```

where:

securityname	Select the access rights/restrictions control for the community name to be deleted. Choose between: <ul style="list-style-type: none">▶ RWCommunity: read/write access rights▶ ROCommunity: read only access rights.	REQUIRED
--------------	--	----------

EXAMPLE:

```
=>snmp community list
Read-write SNMP community name : *****
Read-only SNMP community name : *****
=>snmp community delete securityname=ROCommunity
=>snmp community list
Read-write SNMP community name : *****
Read-only SNMP community name : not specified
=>
```

RELATED COMMANDS:

snmp community add	Configure an SNMP community string to allow SNMP access over IP.
snmp community list	List all SNMP community strings in use for SNMP access over IP.

snmp community list

List all SNMP community strings in use for SNMP access over IP.

SYNTAX:

```
snmp community list
```

EXAMPLE:

```
=>snmp community list
Read-write SNMP community name : *****
Read-only  SNMP community name : *****
=>
```

RELATED COMMANDS:

snmp community add	Configure an SNMP community string to allow SNMP access over IP.
snmp community delete	Delete an SNMP community string to prevent SNMP access over IP.

snmp community modify

Modify an SNMP community string to allow SNMP access over IP.

SYNTAX:

```
snmp community modify securityname = <{RWCommunity | ROCommunity}>
communityname = <password>
```

where:

securityname	Configure access rights/restrictions control for the community name. Choose between: <ul style="list-style-type: none">▶ RWCommunity: read/write access rights▶ ROCommunity: read only access rights.	REQUIRED
communityname	Configure the SNMP community name.	REQUIRED

snmp ifadd

Create a new SNMP interface.

SYNTAX:

```
snmp ifadd          intf = <string>
                    dest = <string>
```

where:

intf	The name for the new SNMP interface.	REQUIRED
dest	The destination interface for this SNMP interface.	REQUIRED

EXAMPLE:

```
=>snmp ifadd
intf = new
dest = Internet
:snmp ifadd intf=new dest=Internet
=>
```

RELATED COMMANDS:

snmp ifdelete	Delete an SNMP interface.
snmp ifattach	Attach an SNMP interface.
snmp ifdetach	Detach an SNMP interface.
snmp ifconfig	Modify an SNMP interface.
snmp iflist	Display the SNMP interfaces.

snmp ifdelete

Delete an SNMP interface.

SYNTAX:

```
snmp ifdelete      intf = <string>
```

where:

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

EXAMPLE:

```
=>snmp ifdelete
intf = new
:snmp ifdelete intf=new
=>
```

RELATED COMMANDS:

snmp ifadd	Create a new SNMP interface.
snmp ifattach	Attach an SNMP interface.
snmp ifdetach	Detach an SNMP interface.
snmp ifconfig	Modify an SNMP interface.
snmp iflist	Display the SNMP interfaces.

snmp ifattach

Attach an SNMP interface.

SYNTAX:

```
snmp ifattach      intf = <string>
```

where:

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

EXAMPLE:

```
=>snmp ifattach
intf = Internet
:snmp ifattach intf=Internet
=>
```

RELATED COMMANDS:

snmp ifadd	Create a new SNMP interface.
snmp ifdelete	Delete an SNMP interface.
snmp ifdetach	Detach an SNMP interface.
snmp ifconfig	Modify an SNMP interface.
snmp iflist	Display the SNMP interfaces.

snmp ifdetach

Detach an SNMP interface.

SYNTAX:

```
snmp ifdetach      intf = <string>
```

where:

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

EXAMPLE:

```
=>snmp ifdetach
intf = Internet
:snmp ifdetach intf=Internet
=>
```

RELATED COMMANDS:

snmp ifadd	Create a new SNMP interface.
snmp ifdelete	Delete an SNMP interface.
snmp ifattach	Attach an SNMP interface.
snmp ifconfig	Modify an SNMP interface.
snmp iflist	Display the SNMP interfaces.

snmp ifconfig

Modify an SNMP interface.

SYNTAX:

```
snmp ifconfig      intf = <string>
                  [securityname = <{RWCommunity | ROCommunity}>]
                  [communityname = <password>]
```

where:

intf	The name of the SNMP interface to configure.	REQUIRED
securityname	Configure access rights/restrictions control for the community name. Choose between: ▶ RWCommunity ▶ ROCommunity. The default is <i>RWCommunity</i>.	OPTIONAL
communityname	Configure SNMP community name.	OPTIONAL

EXAMPLE:

```
=>snmp ifconfig
intf = Internet
[securityname] =
[communityname] =
:snmp ifconfig intf=Internet
=>
```

RELATED COMMANDS:

snmp ifadd	Create a new SNMP interface.
snmp ifdelete	Delete an SNMP interface.
snmp ifattach	Attach an SNMP interface.
snmp ifdetach	Detach an SNMP interface.
snmp iflist	Display the SNMP interfaces.

snmp iflist

Display the SNMP interfaces.

SYNTAX:

```
snmp iflist [intf = <string>]
```

where:

intf	The name of an SNMP interface to configure.	OPTIONAL
------	---	----------

600

speedtouch™

E-DOC-CTC-20060414-0005 v1.0

SNTP Commands

Introduction

This chapter describes the commands of the Simple Network Time Protocol (SNTP) command group.

Contents

This chapter covers the following commands:

sntp add	Add a Network Time Protocol (NTP) server to the NTP server list.	602
sntp config	Modify/display the SNTP client configuration.	603
sntp delete	Delete an NTP server from the NTP server list.	604
sntp flush	Flush the NTP server list and the SNTP client configuration.	605
sntp list	List the NTP servers.	606

sntp add

Add a Network Time Protocol (NTP) server to the NTP server list.

The internal SpeedTouch™ real time clock (SNTP client) will be synchronized with the NTP server.

SYNTAX:

```
sntp add [addr = <ip-address>]
          [name = <string>]
          [version = <number{1-4}>]
```

where:

addr	The IP address of the NTP server to add to the list.	OPTIONAL
Note This parameter is optional in this respect that either an IP address or the name of an NTP server must be specified.		
name	The DNS name of NTP server to be added to list. If both the IP address and the DNS name are provided, the IP address is ignored.	OPTIONAL
Note This parameter is optional in this respect that either an IP address or the name of an NTP server must be specified		
version	The SNTP version of the NTP server. Select either 1, 2, 3, or 4 following NTP server supported versions. The default is 4 .	OPTIONAL

EXAMPLE:

```
=>sntp list
IP Address      Version    Status
100.101.110.113 4          Synchronized
=>sntp add addr=100.101.110.111
=>sntp list
IP Address      Version    Status
100.101.110.111 4          contacting ...
100.101.110.113 4          Synchronized
=>
```

RELATED COMMANDS:

sntp delete	Delete an NTP server from the NTP server list.
sntp list	List the NTP servers.

sntp config

Modify/display the SNTP client configuration.

SYNTAX:

```
sntp config [enable = <{enabled | disabled}>]  
[poll = <number{1-60}>]
```

where:

enable	Enable or disable the SpeedTouch™ SNTP client. The default is enabled .	OPTIONAL
poll	A number between 1 and 60 (minutes). Represents the time interval for the SNTP client to poll the configured NTP server and, if needed, (re-)synchronize its internal clock. The default is 60 .	OPTIONAL

EXAMPLE:

```
=>sntp config  
SNTP configuration:  
    state = enabled  
    poll interval = 60 minute(s)  
    poll interval (before first sync) = 60 minute(s)  
=>
```

sntp delete

Delete an NTP server from the NTP server list.

SYNTAX:

```
sntp delete [addr = <ip-address>]
            [name = <string>]
```

where:

addr	The IP address of the NTP server to be removed from the list.	OPTIONAL
------	---	----------

Note This parameter is optional in this respect that either an IP address or the name of an NTP server must be specified.

name	The DNS name of the NTP server to be removed to the list.	OPTIONAL
------	---	----------

Note This parameter is optional in this respect that either an IP address or the name of an NTP server must be specified.

EXAMPLE:

```
=>sntp list
IP Address      Version   Status
100.101.110.111    4       contacting ...
100.101.110.113    4       Synchronized
=>sntp del addr=100.101.110.111
=>sntp list
IP Address      Version   Status
100.101.110.113    4       Synchronized
=>
```

RELATED COMMANDS:

sntp add	Add a Network Time Protocol (NTP) server to the NTP server list.
sntp list	List the NTP servers.

sntp flush

Flush the NTP server list and the SNTP client configuration.

SYNTAX:

```
sntp flush
```

sntp list

List the NTP servers.

SYNTAX:

```
sntp list
```

EXAMPLE:

```
=>sntp list
IP Address      Version    Status          Name
100.101.110.111   4          contacting ...
100.101.110.112   4          Unable to contact
100.101.110.113   4          Synchronized
=>
```

DESCRIPTION:

The status of an NTP server can be:

- ▶ **Not used:** The SpeedTouch™ SNTP client is disabled. As a consequence, none of the NTP servers are used.
- ▶ **Contacting...:** The SpeedTouch™ SNTP client is trying to contact this NTP server.
- ▶ **Unable to contact:** The SpeedTouch™ SNTP client is unable to contact this NTP server. It may be down, or no end-to-end connectivity exists (no connection, no DSL,...).
- ▶ **Synchronized:** The SpeedTouch™ SNTP client was able to contact this NTP server. If required the internal clock has been synchronized with this NTP server.

RELATED COMMANDS:

sntp add	Add a Network Time Protocol (NTP) server to the NTP server list.
sntp delete	Delete an NTP server from the NTP server list.

Software Commands

Introduction

This chapter describes the commands of the **software** command group.

Contents

This chapter covers the following commands:

software download	Download parameters.	608
software upgrade	Reboot the modem to initiate the SW upgrade.	609
software version	Display the software version.	610

software download

Download parameters.

SYNTAX:

```
software download    filetype = <{firmware | configuration}>
                      url = <string>
                      [username = <string>]
                      [password = <string>]
                      filesize = <string>
                      [targetfilename = <string>]
```

where:

filetype	The type of the file to be downloaded.	REQUIRED
url	Set the HTTP URL where the file is to be found including remote filename.	REQUIRED
username	Set the name to be used to authenticate to the download server.	OPTIONAL
password	Set the password to be used to authenticate to the download server.	OPTIONAL
filesize	Set the size of the file to be downloaded.	REQUIRED
targetfilename	Set the filename used to save the file on the modem.	OPTIONAL

software upgrade

Reboot the modem to initiate the SW upgrade.

New software available on a remote LAN host will be uploaded to the modem.

SYNTAX:

```
software upgrade
```

software version

Display the software version.

SYNTAX:

```
software version
```

System Commands

Introduction

This chapter describes the commands of the **system** command group.

Contents

This chapter covers the following commands:

system config	Set/change the system configuration parameters.	612
system flush	Flush the current system configuration.	614
system locale	Set/get the regional settings.	615
system reboot	Reboot the SpeedTouch™.	617
system reset	Reset the SpeedTouch™ to its factory or ISP defaults and reboot the device.	618
system timedreboot	Set or change a timed reboot for the SpeedTouch™.	619
system debug autosave	Autosave debugging commands	620
system debug stats	Show the SpeedTouch™ CPU and memory statistics.	621
system ra config	Configure Remote management access parameters.	622
system rtc synchronize	Do a SNTP update and synchronize the realtime clock.	623
system rtc settime	Set/get date, time, timezone, daylight savings time, uptime.	624

system config

Set/change the system configuration parameters.

 For a good operation of UPnP and the discovery mechanism, it is highly recommended not to change the system configuration settings.

SYNTAX:

```
system config [upnp = <{disabled | enabled}>]
[tr64 = <{disabled | enabled}>]
[mdap = <{disabled | enabled}>]
[drst = <{disabled | enabled}>]
[led = <{green | red | orange | flash | off}>]
[resetbutton = <{disabled | enabled}>]
[digestauth = <{disabled | enabled}>]
[defaultconnection = <string>]
[rtc = <{disabled|enabled}>]
[autosave = <{disabled | enabled}>]
[autosavedelay = <number{0-600}>]
```

where:

upnp	Enable or disable UPnP discovery. The default is enabled .	OPTIONAL
tr64	Enable or disable TR-64 discovery. The default is disabled .	OPTIONAL
mdap	Enable or disable proprietary discovery protocol. The default is enabled .	OPTIONAL
drst	Enable or disable DrSpeedTouch access. The default is disabled .	OPTIONAL
led	Set the system LED colour. Choose between: <ul style="list-style-type: none"> ▶ green: solid green ▶ red: solid red ▶ orange: solid orange ▶ flash: toggle between green and orange ▶ off: LED is off. The default is green .	OPTIONAL
resetbutton	Enable or disable reset-to-factory-defaults pushbutton. The default is enabled .	OPTIONAL
digestauth	Enable or disable HTTP digest authentication. The default is enabled .	OPTIONAL
defaultconnection	The name of the default internet connection.	OPTIONAL
rtc	Enable or disable RTC. The default is disabled	OPTIONAL
autosave	Enable or disable autosaves. The default is enabled .	OPTIONAL

autosavedelay	A number between 0 and 600. Represents the autosave delay in seconds (0 for immediate save).	OPTIONAL
---------------	---	----------

EXAMPLE:

```
=>system config
upnp discovery      : enabled
TR-64 discovery    : disabled
mdap discovery     : enabled
drst support        : disabled
reset button        : enabled
digest authentication: enabled
rtc                 : enabled
defaultconnection   : Internet
autosave            : enabled
autosave delay      : 10s
=>
```

System Commands

system flush

Flush the current system configuration.

The system password and the system config settings (dcache excluded) are flushed.



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 locale

Set/get the regional settings.

SYNTAX:

```
system locale [dec_symbol = <{, | .}>]
[group_symbol = <{. | ,}>]
[date_separator = <{/ | - | .}>]
[date_format = <{iso | ddmm/yyyy | ddmmyy | mmddyyyy | mmddyy}>]
[time_format = <{iso | hmmss}>]
[datetime_format = <{iso | date+time | time+date}>]
[duration_format = <{dhmmss | hmmss}>]
```

where:

dec_symbol	Set the decimal symbol. Choose between: ▶ . ▶ , The default is ",".	OPTIONAL
group_symbol	Set the group symbol. Choose between: ▶ . ▶ , The default is ".".	OPTIONAL
date_separator	Set the date separator. Choose between: ▶ / ▶ - ▶ . The default is "-".	OPTIONAL
date_format	Set the date format. Choose between: ▶ iso ▶ ddmm/yyyy ▶ ddmmyy ▶ mmddyyyy ▶ mmddyy The default is ddmm/yyyy .	OPTIONAL
Note dd = day; mm = month; yyyy or yy = year.		

System Commands

time_format	Set the time format. Choose between: ▶ iso ▶ hmmss. The default is <i>iso</i> .	OPTIONAL
datetime_format	Set the date-time format. Choose between: ▶ iso ▶ date+time ▶ time+date. The default is <i>date+time</i> .	OPTIONAL
duration_format	Set the duration format. Choose between: ▶ dhmmss ▶ hmmss. The default is <i>dhmmss</i> .	OPTIONAL

Note h = hours; mm = minutes; ss = seconds.

EXAMPLE:

```
=>system locale
Decimal symbol      =
Digit grouping symbol =
Date separator      =
Date format          = ddmmmyyyy
Time format          = iso
Date-time format    = date+time
Duration format     = dhmmss
=>
```

system reboot

Reboot the SpeedTouch™.



Non-saved configuration settings will be lost after reboot.

SYNTAX:

```
system reboot
```

EXAMPLE:

```
=>system reboot
.....
(lost session connectivity due to reboot)
....
```

RELATED COMMANDS:

system timedreboot

Set or change a timed reboot for the SpeedTouch™..

System Commands

system reset

Reset the SpeedTouch™ to its factory or ISP defaults and reboot the device.



All user specific settings and all saved configuration changes are lost after reboot.

SYNTAX:

```
system reset  factory yes/no = <{yes | no}>
              proceed no/yes = <{no | yes}>
```

where:

factory yes/no	Choose between: ▶ yes: delete user and ISP specific settings. ▶ no: delete user specific settings only.	REQUIRED
proceed no/yes	Confirmation for resetting the modem. If no confirmation is given, the SpeedTouch™ will not be reset.	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)
.....
```

RELATED COMMANDS:

sntp config

Modify/display the SNTP client configuration.

system timedreboot

Set or change a timed reboot for the SpeedTouch™.

SYNTAX

```
timedreboot      [state = <{disabled|enabled}>]  
                  [time = <number{0-16384}>]  
                  [date = <string>]
```

where:

state	Enable/Disable timed reboot. The default is disabled .	OPTIONAL
time	Change default countdown time (Min). A number between 0 and 16384. The default is 60 .	OPTIONAL
date	Set date/time to reboot (DD/MM/YYYY/HH:MM). The default is the current day and time	OPTIONAL

EXAMPLE:

```
=>:system timedreboot state=enabled time=60  
input timeout (min): 60  
=>
```

RELATED COMMANDS:

system reboot Reboot the SpeedTouch™.

system debug autosave

Autosave debugging commands

SYNTAX:

system debug autosave	[trace = <{disabled enabled}>]
-----------------------	--------------------------------

where:

trace	Enable or disable autosave traces	OPTIONAL
-------	-----------------------------------	----------

system debug stats

Show the SpeedTouch™ CPU and memory statistics.

SYNTAX:

```
system debug stats [reset = <{disabled | enabled}>]
```

where:

reset	Reset the CPU statistics. The default is disabled .	OPTIONAL
-------	---	----------

DESCRIPTION:

- ▶ **CHIP memory:** Memory used by the CPU (first MB from the RAM) – not cached since it has to be realtime.
- ▶ **Application memory:** Memory used by the applications.
- ▶ **min:** The least amount of free memory detected during the uptime of the SpeedTouch™.

system ra config

Configure Remote management access parameters.

SYNTAX:

```
system ra config      [secure = <{disabled | enabled}>]
                      [port = <number>]
                      [timeout = <number>]
```

where:

secure	Enable or disable https. The default is enabled .	OPTIONAL
port	Set the destination port for remote access. The default is 51003 .	OPTIONAL
timeout	Set the connection timeout in minutes.	OPTIONAL

EXAMPLE:

```
=>system ra config
Remote management access configuration

Remote access port    : 51003
Secure remote access  : enabled
=>
```

system rtc synchronize

Do a SNTP update and synchronize the realtime clock.

SYNTAX:

```
system rtc synchronize
```

system rtc settim

Set/get date, time, timezone, daylight savings time, uptime.

When synchronization with an external NTP server via the SpeedTouch™ SNTP client fails, this command allows to manually configure its internal clock.



In cases without regular synchronization, accurate realtime clock settings can not be guaranteed.

SYNTAX:

```
system rtc settim [date = <dd/mm/yyyy>]
                  [time = <hh:mm:ss>]
                  [timezone = <(+ or -)hh:mm>]
                  [daylightsaving = <{disabled | enabled}>]
```

where:

date	The system date formatted as dd/mm/yyyy.	OPTIONAL
	Note If not specified, the current date is preserved.	
time	The system time formatted as hh:mm:ss.	OPTIONAL
	Note If not specified, the current time is preserved.	
timezone	The system timezone formatted as (+ or -)hh:mm. Valid timezones are possible from -12:00 to +14:00 with a resolution of 15 minutes.	OPTIONAL
	Note If not specified, the current timezone is preserved.	
daylightsaving	Enable or disable daylight saving. The default is disabled .	OPTIONAL
	Note If not specified, the current daylight saving setting is preserved.	

EXAMPLE:

```
=>system settim
date = 31/01/2005
time = 18:05:16
timezone = +00:00
daylightsaving = off
=>
```

Systemlog Commands

Introduction

This chapter describes the commands of the **systemlog** command group.

Contents

This chapter covers the following commands:

systemlog flush	Flush all messages in the internal Syslog message buffer.	626
systemlog show	Show messages in the internal Syslog message buffer.	627
systemlog send	Send messages from the internal Syslog message buffer to a specified local or remote syslog server host.	628

Systemlog Commands

systemlog flush

Flush all messages in the internal Syslog message buffer.

SYNTAX:

```
systemlog flush
```

systemlog show

Show messages in the internal Syslog message buffer.

SYNTAX:

```
systemlog show [fac = <supported facility name>]  
[sev = <supported severity name>]  
[hist = <{disabled | enabled}>]
```

where:

fac	Specify the facility name of the syslog messages to be shown. Use one of the supported facility names (see “ Supported Facilities ” on page 753).	OPTIONAL
Note	If not specified, the messages of all the facilities will be shown.	
sev	Specify the lowest priority severity of the syslog messages to be shown. All the syslog messages with severity as specified or higher will be shown. Use one of the supported severity names (see “ Supported Severities ” on page 754).	OPTIONAL
Note	If not specified, the messages of all the severities will be shown.	
hist	Show messages over several SpeedTouch™ reboots (enabled) or show only messages since latest startup (disabled).	OPTIONAL
Note	If not specified, only the recent messages will be shown.	

EXAMPLE:

```
=>syslog msgbuf show fac=kern sev=emerg hist=enabled  
<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=enabled  
<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 messages from the internal Syslog message buffer to a specified local or remote syslog server host.



There will be no notification on whether the host has received the messages or not.

SYNTAX:

```
systemlog send [fac = <supported facility name>]
                [sev = <supported severity name>]
                [hist = <{disabled | enabled}>]
                dest = <ip-address>
```

where:

fac	Specify the facility name of the syslog messages to show. Use one of the supported facility names (see “ Supported Facilities ” on page 753).	OPTIONAL
Note	If not specified, the messages of all the facilities will be shown.	
sev	Specify the lowest priority severity of the syslog messages to be shown. All the syslog messages with severity as specified or higher will be shown. Use one of the supported severity names (see “ Supported Severities ” on page 754).	OPTIONAL
Note	If not specified, the messages of all the severities will be shown.	
hist	Show messages over several SpeedTouch™ reboots (disabled) or show only messages since latest startup (enabled).	OPTIONAL
Note	If not specified, only the recent messages will be shown.	
dest	The IP address of the remote host on the local or remote network, in other words, the collector's IP address, to send the syslog messages to.	REQUIRED

Upgrade Commands

Introduction

This chapter describes the commands of the **upgrade** command group.

Contents

This chapter covers the following commands:

upgrade config	Change the upgrade daemon configuration.	630
upgrade start	Start a new upgrade.	631
upgrade debug traceconfig	Enable/disable upgrade daemon tracing.	632
upgrade debug sesslist	Display the list of sessions.	633
upgrade profile add	Add a new file profile description.	634
upgrade profile modify	Modify a file profile description.	635
upgrade profile delete	Delete a file profile description.	636
upgrade profile list	Display a list of the file profile descriptions.	636

Upgrade Commands

upgrade config

Change the upgrade daemon configuration.

SYNTAX:

```
upgrade config [state = <{disabled|enabled}>]  
[protocol = <{tftp|ftp|http|https}>]
```

where:

state	Set the upgrade daemon state to enabled or disabled. The default is disabled .	OPTIONAL
protocol	The transfer protocol to be used for file download. Choose between: <ul style="list-style-type: none">▶ tftp▶ ftp▶ http▶ https The default is tftp .	OPTIONAL

EXAMPLE:

```
=>:upgrade config  
Upgrade daemon state: disabled  
Upgrade daemon transfer protocol: tftp  
Upgrade daemon option 66:  
Upgrade daemon option 67:  
=>
```

upgrade start

Start a new upgrade.

SYNTAX:

```
upgrade start      protocol = <{tftp|ftp|http|https}>
                   [server = <string>]
                   [filename = <string>]
```

where:

protocol	The protocol to be used for the download. Choose between: <ul style="list-style-type: none">▶ tftp▶ ftp▶ http▶ https The default is disabled .	REQUIRED
server	The server name or IP address	OPTIONAL
filename	The file to be downloaded	OPTIONAL

Upgrade Commands

upgrade debug traceconfig

Enable/disable upgrade daemon tracing.

SYNTAX:

```
upgrade debug traceconfig state = <{disabled|enabled}>
```

where:

state	Set the upgraded daemon state to enabled or disabled.	REQUIRED
-------	---	----------

upgrade debug sesslist

Display the list of sessions.

SYNTAX:

```
upgrade debug sesslist
```

EXAMPLE:

```
=>upgrade debug sesslist
Session 0:
-----
SessionId : 0x80c25850
State      : IDLE
Protocol   : TFTP
ServerName:
FileName   :
DestFile   :
ServerIP   : 0.0.0.0
Error      :

Session 1:
-----
SessionId : 0x80c25c20
State      : IDLE
Protocol   : TFTP
ServerName:
FileName   :
DestFile   :
ServerIP   : 0.0.0.0
Error      :
=>
```

upgrade profile add

Add a new file profile description.

SYNTAX:

```
upgrade profile add  extension = <{sts|ini}>
                    [maxsize(KBytes) = <number>]
                    action = <{none|config_load|script_load|reboot}>
```

where:

extension	The extension of the new file profile. Choose between: ▶ sts ▶ ini	REQUIRED
maxsize	The maximum filesize (in KBytes) allowed for files with the extension of this profile. The default is 100 .	OPTIONAL
action	The action to be done when the file with the extension of this profile is downloaded. Choose between: ▶ none ▶ config_load ▶ script_load ▶ reboot	REQUIRED

RELATED COMMANDS:

- | | |
|------------------------|--|
| upgrade profile delete | Delete a file profile description. |
| upgrade profile list | Display a list of the file profile descriptions. |

upgrade profile modify

Modify a file profile description.

SYNTAX:

```
upgrade profile modify extension = <{sts|ini}>
                                [maxsize(KBytes) = <number>]
                                action = <{none|config_load|script_load|reboot}>
```

where:

extension	The extension of the new file profile. Choose between: <ul style="list-style-type: none">▶ sts▶ ini	REQUIRED
maxsize	The maximum filesize (in KBytes) allowed for files with the extension of this profile. The default is 100 .	OPTIONAL
action	The action to be done when the file with the extension of this profile is downloaded. Choose between: <ul style="list-style-type: none">▶ none▶ config_load▶ script_load▶ reboot	REQUIRED

RELATED COMMANDS:

upgrade profile delete	Delete a file profile description.
upgrade profile list	Display a list of the file profile descriptions.

Upgrade Commands

upgrade profile delete

Delete a file profile description.

SYNTAX:

```
upgrade profile delete extension = <{sts|ini}>
```

where:

extension	The extension of the profile to be deleted. Choose between: ▶ sts ▶ ini	REQUIRED
-----------	--	----------

RELATED COMMANDS:

- | | |
|----------------------|--|
| upgrade profile add | Add a new file profile description. |
| upgrade profile list | Display a list of the file profile descriptions. |

upgrade profile list

Display a list of the file profile descriptions.

SYNTAX:

```
upgrade profile list
```

RELATED COMMANDS:

- | | |
|------------------------|------------------------------------|
| upgrade profile delete | Delete a file profile description. |
| upgrade profile modify | Modify a file profile description. |

638

speedtouch™

E-DOC-CTC-20060414-0005 v1.0

UPnP Commands

Introduction

This chapter describes the commands of the **upnp** command group.

Contents

This chapter covers the following commands:

upnp config	Configure the UPnP™ parameter(s).	640
upnp flush	Flush the UPnP™ configuration.	641
upnp list	List all registered devices.	642

upnp config

Configure the UPnP™ parameter(s).

SYNTAX:

```
upnp config [maxage = <number{60-999999}>]
            [writemode = <{full | natonly | readonly}>]
            [safenat = <{disabled | enabled}>]
            [preferredaddress = <ip-address>]
            [httpport = <number{1-65535}>]
```

where:

maxage	A number between 60 and 999999. 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. The default is 1800 .	OPTIONAL
writemode	Note 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.	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. The default is natonly .
safenat	Enable or 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. The default is enabled .	OPTIONAL
preferredaddress	CSV list of preferred IP address for UPnP advertisements (1 per LAN IP interface).	OPTIONAL
httpport	Note Enter "0.0.0.0" for none.	A number between 1 and 65535. Represents the web server port. The default is 80 .

upnp flush

Flush the UPnP™ configuration.

The UPnP configuration will be reset to the default configuration.

SYNTAX:

```
upnp flush
```

upnp list

List all registered devices.



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}>]
```

where:

verbose	Verbose level. The default is 1 .	OPTIONAL
---------	---	----------

EXAMPLE:

```
=>upnp list
----- device: IGD.xml -----
Advertised on: LocalNetwork (10.0.0.138)
----- devices/services -----
++ 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:RELAY
-- Service 5: urn:upnp-org:serviceId:wanpppc:Internet
----- end -----
=>
```

User Commands

Introduction

This chapter describes the commands of the **user** command group.

Contents

This chapter covers the following commands:

user add	Add a user.	644
user config	Modify a user.	646
user delete	Delete a user.	648
user flush	Flush the users.	649
user list	Display the users.	650
user rights	Display the session rights.	652

user add

Add a user.



You can only add a user whose privileges are the same or lower than your own privileges.

SYNTAX:

```
user add name = <quoted string>
        password = <password>
        role = <string>
        [hash2 = <string>]
        [descr = <quoted string>]
        [defuser = <{disabled | enabled}>]
        [defremadmin = <{disabled | enabled}>]
        [deflocadmin = <{disabled | enabled}>]
```

where:

name	The new user name.	REQUIRED
	Note 1. The maximum number of users is 10. 2. The maximum length is 32 characters.	
password	The password.	REQUIRED
	Note The maximum length is 32 characters.	
role	The role name.	REQUIRED
	Tip Use the command :mlp role list to obtain the role name (see " mlp role list" on page 507 for more information).	
hash2	The MD5 hash.	OPTIONAL
	Note The maximum length is 32 characters.	
descr	A user description.	OPTIONAL
	Note The maximum length is 63 characters.	
defuser	Set this user as the default user (enabled) or not (disabled).	OPTIONAL
	Note When the Web interface is accessed, the account of this user will be used by default. The user will not need to authenticate himself with user name or password. The default is disabled .	
defremadmin	Set this user as the default remote administrator (enabled) or not (disabled). The default is disabled .	OPTIONAL
deflocadmin	Set this user as the default local administrator (enabled) or not (disabled). The default is disabled .	OPTIONAL

EXAMPLE:

In the example below, the user Administrator creates a new user JohnDoe. This user has lower access rights than the user "Poweruser".

```
{Administrator}=>user list
User                         Flags Role
----                         ----- -----
Administrator                  U     Administrator
tech                           R     TechnicalSupport

{Administrator} [user]=>add
name = JohnDoe
password = *****
Please retype password for verification.
password = *****
role = Administrator
[hash2] =
[descr] =
[defuser] =
[defremadmin] =
:user add name=JohnDoe password=_CYP_x90/lhxuRyMME role=Administrator
{Administrator}=>user list
User                         Flags Role
----                         ----- -----
Administrator                  U     Administrator
tech                           R     TechnicalSupport
JohnDoe                       Administrator

{Administrator}=>
```

DESCRIPTION:

- ▶ **U:** indicates the default user.
- ▶ **R:** indicates the default remote administrator.

RELATED COMMANDS:

user delete	Delete a user.
user list	Display the users.

user config

Modify a user.



You can only modify a user whose privileges are the same or lower than your own privileges.

SYNTAX:

```
user config name = <string>
            [password = <password>]
            [role = <string>]
            [descr = <quoted string>]
            [defuser = <{disabled | enabled}>]
            [defremadmin = <{disabled | enabled}>]
            [deflocadmin = <{disabled | enabled}>]
```

where:

name	The user name.	REQUIRED
password	The password.	OPTIONAL
	Note The maximum length is 32 characters.	
role	The role name.	OPTIONAL
	Tip Use the command :mlp role list to obtain the role names (see " mlp role list" on page 507 for more information).	
descr	A user description.	OPTIONAL
	Note The maximum length is 63 characters.	
defuser	Set this user as the default user (enabled) or not (disabled).	OPTIONAL
	Note When the Web interface is accessed, the account of this user will be used by default. The user will not need to authenticate himself with user name or password. The default is disabled .	
defremadmin	Set this user as the default remote administrator (enabled) or not (disabled). The default is disabled .	OPTIONAL
deflocadmin	Set this user as the default local administrator (enabled) or not (disabled). The default is disabled .	OPTIONAL

EXAMPLE:

```
{Superuser}>user list
      User          Role
      ====
      Superuser      root
      aPoweruser    Poweruser

{Superuser}>user config name=aPoweruser role=LAN_admin
{Superuser}>user list
      User          Role
      ====
      Superuser      root
      aPoweruser    LAN_admin

{Superuser}>
```

user delete

Delete a user.



You can only delete a user whose privileges are the same or lower than your own privileges.

SYNTAX:

```
user delete name = <string>
```

where:

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

EXAMPLE:

```
{Administrator}=>user list
User           Flags Role
-----
Administrator   U     Administrator
tech           R     TechnicalSupport
JohnDoe        Administrator

{Administrator}=>user delete name = JohnDoe
{Administrator}=>user list
User           Flags Role
-----
Administrator   U     Administrator
tech           R     TechnicalSupport

{Administrator}=>
```

RELATED COMMANDS:

user add Add a user.

user list Display the users.

user flush

Flush the users.

After execution of this command, the access to the SpeedTouch™ will no longer be protected by a password. The first user that logs on, after initiating a new Telnet session, will have the same administrative rights as a user with the role **root**.

SYNTAX

user flush

EXAMPLE:

```
{Administrator}=>user list
User                                Flags  Role
----                                ----  ---
Administrator                         U     Administrator
tech                                 R     TechnicalSupport
JohnDoe                             Administrator

{Administrator}=>user flush
{Administrator}=>user list

{Administrator}=>
```

When this Telnet session is terminated and a new Telnet session is initiated, the following welcome screen will appear:

The user will not have to authenticate himself to gain access to the SpeedTouch™.

user list

Display the users.

SYNTAX:

```
user list [name = <string>]
          [channel = <{ftp | telnet | http | mdap | serial}>]
          [origin = <{lan | wan | local}>]
          [secure = <{disabled | enabled}>]
```

where:

name	The user name.	OPTIONAL
Note If not specified, all the users will be listed.		
channel	The selected channel. Choose between: <ul style="list-style-type: none">▶ ftp▶ telnet▶ http▶ mdap▶ serial.	OPTIONAL
Note If not specified, all the channels will be listed.		
origin	The selected origin. Choose between: <ul style="list-style-type: none">▶ lan▶ wan▶ local.	OPTIONAL
Note If not specified, all the origins will be listed.		
secure	The selected security level. Choose between: <ul style="list-style-type: none">▶ disabled▶ enabled.	OPTIONAL
Note If not specified, all the security levels will be listed.		

EXAMPLE:

```
{Administrator}=>user list
User          Flags Role
-----
Administrator      U      Administrator
tech            R      TechnicalSupport

{Administrator}=>user list name=tech channel=http origin=wan
service(s): r_lan, r_wan, r_fs_view, r_fs_retrieve, r_rtg, r_fwdg, r_nat, r_frwl,
r_ipsec_norm, r_ipsec_adv, r_certificates, r_remote_mngnt, r_local, r_qos,
w_lan, w_wan, w_fs_passive, w_rtg, w_fwdg, w_nat, w_frwl_norm, w_frwl_adv,
w_frwe_mngnt, w_ipsec, w_certificates, w_remote_mngnt, w_local, w_qos,
snd_lan, snd_wan, snd_local, and_lan, and_wan, and_frwl, and_local,
user_admin, mlp_admin, secure_ipsec_term, secure_br, cli, cgi, ftp, mdap,
zone_45, zone_46, zone_47, zone_48, zone_49, zone_50, zone_51, zone_52,
zone_53, zone_54, zone_55, zone_56, zone_57, zone_58, zone_59, zone_60,
zone_61, zone_62, swk_activation, sensitve file, zone_71, zone_72, zone_73,
zone_74, zone_75, zone_76, zone_77, zone_78, zone_79, zone_83, zone_84,
zone_85, zone_86, zone_87, zone_88, zone_89, zone_90, zone_91, zone_92,
traces

{Administrator}=>
```

RELATED COMMANDS:

- | | |
|--------------------|----------------|
| user add | Add a user. |
| user delete | Delete a user. |

user rights

Display the session rights.



The user rights of the currently logged-in user are shown.

SYNTAX:

```
user rights
```

EXAMPLE:

```
{Administrator}=>user rights
Current session info:
user:           name='Administrator', role='Administrator'
access:         lan (origin_lan), telnet (channel_telnet), unsecure (unsecure_connection)
service(s):     r_lan, r_wan, r_fs_view, r_fs_retrieve, r_rtg, r_fwdg, r_nat, r_frwl,
                r_ipsec_norm, r_ipsec_adv, r_certificates, r_remote_mgnt, r_local, r_qos,
                w_lan, w_wan, w_fs_passive, w_rtg, w_fwdg, w_nat, w_frwl_norm, w_frwl_adv,
                w_frwe_mgnt, w_ipsec, w_certificates, w_remote_mgnt, w_local, w_qos,
                snd_lan, snd_wan, snd_local, and_lan, and_wan, and_frwl, and_local,
                user_admin, mlp_admin, secure_ipsec_term, secure_br, cli, cgi, ftp, mdap,
                zone_45, zone_46, zone_47, zone_48, zone_49, zone_50, zone_51, zone_52,
                zone_53, zone_54, zone_55, zone_56, zone_57, zone_58, zone_59, zone_60,
                zone_61, zone_62, swk_activation, sensitve_file, zone_71, zone_72, zone_73,
                zone_74, zone_75, zone_76, zone_77, zone_78, zone_79, zone_83, zone_84,
                zone_85, zone_86, zone_87, zone_88, zone_89, zone_90, zone_91, zone_92,
                traces

{Administrator}=>
```

Voice Commands

Introduction

This chapter describes the commands of the **voice** command group.

Contents

This chapter covers the following commands:

voice config	Configure general voice settings.	655
voice state	Show VoIP service states.	657
voice codec list	Show the CODEC capability configuration.	658
voice codec config	Configure a CODEC capability.	659
voice country config	Configure country.	660
voice country list	List available countries.	661
voice dialplan add	Add an entry in the dialplan.	662
voice dialplan delete	Delete an entry from the dialplan.	664
voice dialplan modify	Modify an entry in the dialplan.	665
voice dialplan list	Display an entry or the complete dialplan.	667
voice dialplan flush	Delete all user-entered dialplan entries.	668
voice fax config	Configure the fax parameters.	669
voice fxoport config	Configure the FXO port parameters.	670
voice fxsport config	Configure the FXS port parameters.	671
voice mgcp config	Configure the MGCP general parameters.	672
voice pb add	Add phonebook entry in the phonebook.	674
voice phonebook add	Add a phone number to the phonebook.	675
voice phonebook delete	Delete a phone number from the phonebook.	676
voice phonebook list	Show all entries in the phonebook.	677
voice profile add	Add profile.	678
voice profile delete	Delete profile.	680
voice profile modify	Modify profile.	681
voice profile list	Show all profiles.	683
voice profile flush	Flush all profiles.	684
voice qos list	Show the QoS configuration.	685
voice qos config	Configure the QoS parameters.	686
voice services config	General supplementary service configuration.	687
voice services list	Show the supplementary service configuration.	689
voice services provision	Provision a supplementary service.	690
voice services withdraw	Withdraw a supplementary service.	691
voice services activate	Activate a supplementary service.	692

Voice Commands

voice services deactivate	Deactivate a supplementary service.	693
voice services assign	Assign a servicecode to a supplementary service in standard mode.	694
voice services assign_pxsc	Assign a servicecode to a supplementary service in non_standard mode.	695
voice services flush	Show voice statistics.	696
voice sip config	Configure the SIP general parameters.	697
voice stats list	Show voice statistics.	699
voice stats reset	Reset the voice statistics.	700

voice config

Configure general voice settings.

SYNTAX:

```
voice config [autofxo = <{disabled | enabled}>]
              [digitrelay = <{auto | inband | rfc2833 | signalling}>]
              [click2dial_ports = <{FXS1 | FXS2 | all}>]
              [rtp_portrange = <port-range>] [sign_internal = <{external | internal}>]
              [static_intf = <disabled | enabled>]
              [intf = <string>]
              [endofnumber = <#{|*}>
```

where:

autofxo	Enable or disable automatic FXO calls when not registered. The default is enabled .	OPTIONAL
digitrelay	Set the digit relay mode. Choose between: <ul style="list-style-type: none">▶ auto▶ inband▶ rfc2833▶ signalling. The default is auto .	OPTIONAL
click2dial_ports	Set the click to dial port. Choose between: <ul style="list-style-type: none">▶ FXS1▶ FXS2▶ all. The default is FXS1 .	OPTIONAL
rtp_portrange	The RTP port range.	OPTIONAL
sign_internal	Signalling for local calls is kept local or external. The default is external .	OPTIONAL
static_intf	Use a static (configured) interface to look for the IP address or not. The default is disabled .	OPTIONAL
intf	The name of an IP interface to be used for VoIP traffic.	OPTIONAL
endofnumber	The end of number character for dialled number starting with a cipher.	OPTIONAL

EXAMPLE:EXAMPLE:

```
=>voice config
autofxo      : enabled
digitrelay   : auto
click2dial_port : FXS1
rtp_portrange : 1024-49151
sign_internal : external
static_interface: disabled
interface     :
=>
```

```
=>voice config
autofxo      : enabled
rtp_portrange : 1024-49151
=>
```

voice state

Show VoIP service states.

SYNTAX:

```
voice state
```

EXAMPLE:

```
=>voice state
VOIP_SIP-admin. state : disabled
VOIP_SIP-oper. state : disabled
VOIP_SIP-IP address   : 0.0.0.0
=>
```

voice codec list

Show the CODEC capability configuration.

SYNTAX:

```
voice codec  
list
```

EXAMPLE:

```
=>voice codec list  
Codec Type      Packet Delay    Voice Act. Detection  Priority   Status  
-----  
    g711u          30            enabled           1  enabled  
    g711a          30            enabled           1  enabled  
    f726_40         30            enabled           4  disabled  
    g726_32         30            enabled           5  disabled  
    g726_24         30            enabled           6  disabled  
    g726_16         30            enabled           7  disabled  
    g729           30            enabled           2  enabled  
=>
```

RELATED COMMANDS:

voice codec config Configure a CODEC capability.

voice codec config

Configure a CODEC capability.

SYNTAX:

```
voice codec config type = <{g711u | g711a | g726_40 | g726_32 | g726_24 | g726_16 |  
g729}>  
[ptime = <{10 | 30 | 20}>]  
[ptime_g723 = <{30}>]  
[vad = <{disabled | enabled}>]  
[priority = <number{1-8}>]  
[status = <{disabled | enabled}>]
```

where:

type	The codec type.	REQUIRED
ptime	The packet time.	OPTIONAL
ptime_g723	The g723 packet time.	OPTIONAL
vad	Enable or disable Voice Activity Detection.	OPTIONAL
priority	A number between 1 and 8. Represents the codec capability priority.	OPTIONAL
status	Enable or disable the capability.	OPTIONAL

EXAMPLE:

```
=>voice codec config  
type = g711u  
[ptime] = 10  
[vad] = enabled  
[priority] = 2  
:voice codec config type=g711u ptime=10 vad=enabled priority=2  
=>
```

RELATED COMMANDS:

voice codec list

Show the CODEC capability configuration.

voice country config

Configure country.

SYNTAX:

```
voice country config [country = <country>]
```

where:

country	Load this country's specific settings into the DSP.	OPTIONAL
---------	---	----------

EXAMPLE:

```
=>voice country config  
Country : spain  
=>
```

RELATED COMMANDS:

voice country list	List available countries.
--------------------	---------------------------

voice country list

List available countries.

SYNTAX:

```
voice country list
```

EXAMPLE:

```
=>voice country list
Available countries:
belgium      etsi
france1      france2
netherlands  northamerica
norway       spain
sweden       uk
=>
```

RELATED COMMANDS:

voice country config Configure country.

voice dialplan add

Add an entry in the dialplan.

SYNTAX:

```
voice dialplan    prefix = <0...9 - * #>
add              defaultport = <{FXS1 | FXS2 | FXO | VoIP | NA}>
                  fallbackport = <{FXS1 | FXS2 | FXO | VoIP | NA}>
                  priority = <{NA | Low | High}>
                  fallback = <{disabled | enabled}>
                  minimumdigits = <number{1-31}>
                  maximumdigits = <number{1-31}>
                  posofmodify = <number{0-31}>
                  remnumdigits = <number{0-31}>
                  [insert = <0...9]
                  rescan = <{no | yes}>
                  [data = <{no | yes}>]
                  action = <{none | ROUTE_excl_# | ROUTE_incl_#}>
```

where:

prefix	The prefix which identifies this entry.	REQUIRED
defaultport	The default outgoing port. Choose between: <ul style="list-style-type: none"> ▶ FXS1 ▶ FXS2 ▶ FXO ▶ VoIP ▶ NA The default is FXS1 .	REQUIRED
fallbackport	The fallback outgoing port. Choose between: <ul style="list-style-type: none"> ▶ FXS1 ▶ FXS2 ▶ FXO ▶ VoIP ▶ NA The default is FXS1 .	REQUIRED
priority	The priority of the entry. Choose between: <ul style="list-style-type: none"> ▶ NA ▶ Low ▶ High The default is NA .	REQUIRED

fallback	Enable or disable the fallback mechanism status of the entry. The default is disabled .	REQUIRED
minimumdigits	A number between 1 and 31. Represents the minimum number of digits.	REQUIRED
maximumdigits	A number between 1 and 31. Represents the maximum number of digits.	REQUIRED
posofmodify	A number between 0 and 31. Represents the startposition at which a number of digits must be removed.	REQUIRED
remnumdigits	A number between 0 and 31. Represents the number of digits that need to be removed from the complete number.	REQUIRED
insert	A number between 0 and 9. String which must be inserted at PosOfModify after removing RemNumDigits.	OPTIONAL
rescan	Rescanning of the number needed or not.	REQUIRED
data	Use the entry for data calls or not.	OPTIONAL
action	Action parameter.	REQUIRED

RELATED COMMANDS:

- | | |
|------------------------------|--|
| voice dialplan delete | Delete an entry from the dialplan. |
| voice dialplan modify | Modify an entry in the dialplan. |
| voice dialplan list | Display an entry or the complete dialplan. |
| voice dialplan flush | Delete all user-entered dialplan entries. |

voice dialplan delete

Delete an entry from the dialplan.

SYNTAX:

```
voice dialplan    prefix = <{ }>
delete
```

where:

prefix	The prefix which identifies this entry.	REQUIRED
--------	---	----------

RELATED COMMANDS:

voice dialplan add	Add an entry in the dialplan.
voice dialplan modify	Modify an entry in the dialplan.
voice dialplan list	Display an entry or the complete dialplan.
voice dialplan flush	Delete all user-entered dialplan entries.

voice dialplan modify

Modify an entry in the dialplan.

SYNTAX:

```
voice dialplan    prefix = <{}>
modify          newprefix = <0...9 - * #>
                  defaultport = <{FXS1 | FXS2 | FXO | VoIP | NA}>
                  fallbackport = <{FXS1 | FXS2 | FXO | VoIP | NA}>
                  priority = <{NA | Low | High}>
                  fallback = <{disabled | enabled}>
                  minimumdigits = <number{1-31}>
                  maximumdigits = <number{1-31}>
                  posofmodify = <number{0-31}>
                  remnumdigits = <number{0-31}>
                  [insert = <0...9]
                  rescan = <{no | yes}>
                  [data = <{no | yes}>]
                  action = <{none | ROUTE_excl_# | ROUTE_incl_#}>
```

where:

prefix	The prefix which identifies this entry.	REQUIRED
newprefix	The prefix which identifies this entry.	REQUIRED
defaultport	<p>The default outgoing port. Choose between:</p> <ul style="list-style-type: none"> ▶ FXS1 ▶ FXS2 ▶ FXO ▶ VoIP ▶ NA <p>The default is FXS1.</p>	REQUIRED
fallbackport	<p>The fallback outgoing port. Choose between:</p> <ul style="list-style-type: none"> ▶ FXS1 ▶ FXS2 ▶ FXO ▶ VoIP ▶ NA <p>The default is FXS1.</p>	REQUIRED

priority	The priority of the entry. Choose between: ▶ NA ▶ Low ▶ High The default is NA .	REQUIRED
fallback	Enable or disable the fallback mechanism status of the entry. The default is disabled .	REQUIRED
minimumdigits	A number between 1 and 31. Represents the minimum number of digits.	REQUIRED
maximumdigits	A number between 1 and 31. Represents the maximum number of digits.	REQUIRED
posofmodify	A number between 0 and 31. Represents the startposition at which a number of digits must be removed.	REQUIRED
remnumdigits	A number between 0 and 31. Represents the number of digits that need to be removed from the complete number.	REQUIRED
insert	A number between 0 and 9. String which must be inserted at PosOfModify after removing RemNumDigits.	OPTIONAL
rescan	Rescanning of the number needed or not.	REQUIRED
data	Use the entry for data calls or not.	OPTIONAL
action	Action parameter.	REQUIRED

RELATED COMMANDS:

voice dialplan add	Add an entry in the dialplan.
voice dialplan delete	Delete an entry from the dialplan.
voice dialplan list	Display an entry or the complete dialplan.
voice dialplan flush	Delete all user-entered dialplan entries.

voice dialplan list

Display an entry or the complete dialplan.

SYNTAX:

```
voice dialplan    prefix = <{all | *96 | *95 | *94 | *43* | *43 | *30 | #96 |
list          #95 | #94 | #43* | #43 | #30}>
```

where:

prefix	The prefix which identifies this entry.	REQUIRED
--------	---	----------

EXAMPLE:

```
=>voice dialplan list
prefix = *43
:voice dialplan list prefix=*43
Prefix  DefPort  FallbPort  Prior  Fallback  Min.Dig  Max.Dig  PostMod  Rem.Dig
-----
           Insert  Rescan  Data
-----
      *43      NA        NA      NA      No       3        30      0        0
                  No      No
=>
-----
```

RELATED COMMANDS:

voice dialplan add	Add an entry in the dialplan.
voice dialplan delete	Delete an entry from the dialplan.
voice dialplan modify	Modify an entry in the dialplan.
voice dialplan flush	Delete all user-entered dialplan entries.

voice dialplan flush

Delete all user-entered dialplan entries.

SYNTAX:

```
voice dialplan flush
```

RELATED COMMANDS:

voice dialplan add	Add an entry in the dialplan.
voice dialplan delete	Delete an entry from the dialplan.
voice dialplan modify	Modify an entry in the dialplan.
voice dialplan list	Display an entry or the complete dialplan.

voice fax config

Configure the fax parameters.

SYNTAX:

```
voice fax config [detect_timeout = <number{0-120}>]  
[transport = <{inband_auto|inband_reneg|t38}>]  
[udptl_redun = <number{0-3}>]
```

where:

detect_timeout	A number between 0 and 120. Represents the time-out in seconds used to detect FAX (0 is no time-out)	OPTIONAL
transport	The type of fax transport. Choose between: <ul style="list-style-type: none">▶ inband_auto▶ inband_reneg▶ t38	OPTIONAL
udptl_redun	A number between 0 and 3. Represents the number of secondary IPP packets.	OPTIONAL

voice fxoport config

Configure the FXO port parameters.

SYNTAX:

```
voice fxoport config      [incfxo = <{enabled|disabled}>]  
                           [fxodisconnect = <number{500-5000}>]
```

where:

incfxo	Enable or disable incoming FXO calls. The default is enabled .	OPTIONAL
fxodisconnect	The FXO disconnect timer (in ms). Enter a number between 500 and 5000.	OPTIONAL

voice fxsport config

Configure the FXS port parameters.

SYNTAX:

```
voice fxsport config [interdigit = number{10-10000}]
```

where:

interdigit	A number between 10 and 10000. The interdigit timer (in ms)	OPTIONAL
------------	--	----------

EXAMPLE:

```
=>voice fxsport config
Inter digit timer      : 5000
=>
```

voice mgcp config

Configure the MGCP general parameters.

SYNTAX:

```
voice mgcp config [callagentaddr = <string>]
[callagentport = <string>]
[seccallagentaddr = <string>]
[seccallagentport = <string>]
[failoverdelay = <number>]
[maxrsipdelay = <number>]
[maxretries = <number>]
[retransmittimer = <number>]
[cpedomain = <{IP-address|Reverse-DNS|MAC-
Address|Serial-Number|Unique-Random-Id|Secure-Unique-
Random-Id|Custom}>]
[cpedomaincustom = <string>]
[cpedomainbrackets = {automatic|enabled|disabled}]
[starstarashookflash = <disabled|enabled>]
```

where:

callagentaddr	Primary call agent IP address or FQDN.	OPTIONAL
callagentport	Primary call agent port. See "Supported TCP/UDP Port Names" on page 749.	OPTIONAL
seccallagentaddr	Secondary call agent IP address or FQDN.	OPTIONAL
seccallagentport	Secondary call agent port. See "Supported TCP/UDP Port Names" on page 749.	OPTIONAL
failoverdelay	Time-out in seconds after which the fail-over procedure is started. The default is -1 s .	OPTIONAL
maxrsipdelay	Maximum waiting time in milliseconds between when an endpoint comes up and the corresponding restart message is sent. The default is 0 ms .	OPTIONAL
maxretries	Maximum number of times a command is retransmitted.	OPTIONAL
retransmittimer	Maximum waiting time in milliseconds before retransmitting a command. The default is 4000 ms .	OPTIONAL
cpedomain	Type of MGCP domain. The default is IP-address .	OPTIONAL
cpedomaincustom	Custom local MGCP domain name.	OPTIONAL
cpedomainbrackets	Use brackets around MGCP domain name. The default is automatic .	OPTIONAL
starstarashookflash	Enable or disable translation of ** into hookflash. The default is disabled .	OPTIONAL

EXAMPLE:

```
=>voice mgcp config
Primary call agent address      :
Primary call agent port        : 2727
Secondary call agent address   :
Secondary call agent port      : 2727
Fail-over delay                 : -1s
CPE domain                      : IP-Address
CPE domain brackets             : automatic
Maximum restart delay           : 0ms
Request retransmit retries      : 3
Maximum retransmit time-out     : 4000ms
Translate ** into hookflash     : disabled
=>
```

voice pb add

Add phonebook entry in the phonebook.

SYNTAX:

```
voice pb add
    Lastname = <quoted string>
    Firstname = <quoted string>
    [Business = <quoted string>]
    [Home = <quoted string>]
    [Mobile = <quoted string>]
    [Other = <quoted string>]
    [E-mail = <quoted string>]
```

where:

Lastname	The last name.	REQUIRED
Firstname	The first name.	REQUIRED
Business	The business number.	OPTIONAL
Home	The home number.	OPTIONAL
Mobile	The mobile number.	OPTIONAL
Other	Any other number.	OPTIONAL
E-mail	The e-mail address.	OPTIONAL

EXAMPLE:

```
=>voice pb add
Lastname = Doe
Firstname = John
[Business] = 111222333
[Home] =
[Mobile] =
[Other] =
[E-mail] = johndoe@mail
:voice pb add Lastname =Doe Firstname=John Business=111222333 E-mail=johndoe@mail
=>
```

voice phonebook add

Add a phone number to the phonebook.

SYNTAX:

```
voice phonebook add e164number = <string>
```

where:

e164number The telephone number to be added.

REQUIRED

EXAMPLE:

```
=>voice phonebook list
Phonebook ( 1 entries)
-----
003292321513
=>voice phonebook add e164number=00315674890
=>voice phonebook list
Phonebook ( 2 entries)
-----
003292321513
00315674890
=>
```

RELATED COMMANDS:

voice phonebook delete Delete a phone number from the phonebook.

voice phonebook list Show all entries in the phonebook.

voice phonebook delete

Delete a phone number from the phonebook.

SYNTAX:

```
voice phonebook delete e164number = <string>
```

where:

e164number	Enter the telephone number to be deleted or all to delete all the phonenumbers.	REQUIRED
------------	---	----------

EXAMPLE:

```
=>voice phonebook list
Phonebook ( 2 entries)
-----
003292321513
00315674890
=>voice phonebook delete e164number=003292321513
=>voice phonebook list
Phonebook ( 1 entries)
-----
00315674890
=>
=>voice phonebook delete e164number=all
=>voice phonebook list
Phonebook ( 0 entries)
-----
=>
```

RELATED COMMANDS:

voice phonebook add	Add a phone number to the phonebook.
voice phonebook list	Show all entries in the phonebook.

voice phonebook list

Show all entries in the phonebook.

SYNTAX:

```
voice phonebook list
```

EXAMPLE:

```
=>voice phonebook list
Phonebook ( 2 entries)
-----
003292321513
00315674890
=>
```

RELATED COMMANDS:

- voice phonebook add Add a phone number to the phonebook.
- voice phonebook delete Delete a phone number from the phonebook.

voice profile add

Add profile.

SYNTAX:

```
voice profile add      endpointname = <string>
                      SIP_URI = <string>
                      [username = <string>]
                      [password = <password>]
                      [displayname = <string>]
                      voiceport = <{FXS1 | FXS2 | COMMON}>
                      [abbr = <string>]
                      [status = <{disabled|enabled}>]
```

where:

endpointname	The analogue line endpoint name.	REQUIRED
SIP_URI	The SIP URI related to this voice port.	REQUIRED
username	The authentication username related to this voice port.	OPTIONAL
password	The authentication password related to this voice port.	OPTIONAL
displayname	An alias name for the SIP_URI.	OPTIONAL
voiceport	The analog line number. Choose between: <ul style="list-style-type: none"> ▶ FXS1 ▶ FXS2 ▶ COMMON. 	REQUIRED
voiceport	The analog line number. Choose between: <ul style="list-style-type: none"> ▶ FXS1 ▶ FXS2 	REQUIRED
abbr	An abbreviated number that will be mapped to the SIP_URI.	OPTIONAL
status	Enable or disable this port.	OPTIONAL

EXAMPLE:

```
=>voice profile add
SIP_URI = new
[username] =
[password] =
[displayname] =
voiceport = FXS1
[abbr] =
:voice profile add SIP_URI=new voiceport=FXS1
=>
```

EXAMPLE:

```
=>voice profile add  
endpointname = new  
voiceport = FXS1  
[status] = disabled  
:voice profile add endpointname=new voiceport=FXS1 status=disabled  
=>
```

RELATED COMMANDS:

voice profile delete	Delete profile.
voice profile modify	Modify profile.
voice profile list	Show all profiles.
voice profile flush	Flush all profiles.

voice profile delete

Delete profile.

SYNTAX:

```
voice profile delete SIP_URI = <string>
                    endpointname = <all|aalm/2|aalm/1}>
```

where:

SIP_URI	The SIP URI related to this voice port.	REQUIRED
endpointname	The analogue line endpoint name.	REQUIRED

EXAMPLE:

```
=>voice profile delete
SIP_URI = new
:voice profile delete SIP_URI=new
=>
```

EXAMPLE:

```
=>voice profile delete
endpointname = all
:voice profile delete endpointname=all
=>
```

RELATED COMMANDS:

voice profile add	Add profile.
voice profile modify	Modify profile.
voice profile list	Show all profiles.
voice profile flush	Flush all profiles.

voice profile modify

Modify profile.

SYNTAX:

```
voice profile modify endpointname = <{aaln/2|aaln/1}>
    SIP_URI = <string>
    NEW_URI = <string>
    [username = <string>]
    [password = <password>]
    [displayname = <string>]
    voiceport = <{FXS1 | FXS2 | COMMON}>
    [abbr = <string>]
    [status = <{disabled|enabled}>]
```

where:

endpointname	The analogue line endpoint name.	REQUIRED
SIP_URI	The SIP URI related to this voice port.	REQUIRED
NEW_URI	The new SIP URI related to this voice port.	REQUIRED
username	The authentication username related to this voice port.	OPTIONAL
password	The authentication password related to this voice port.	OPTIONAL
displayname	An alias name for the SIP_URI.	OPTIONAL
voiceport	<p>The analog line number. Choose between:</p> <ul style="list-style-type: none"> ▶ FXS1 ▶ FXS2 ▶ COMMON. 	REQUIRED
voiceport	<p>The analog line number. Choose between:</p> <ul style="list-style-type: none"> ▶ FXS1 ▶ FXS2 	REQUIRED
abbr	An abbreviated number that will be mapped to the SIP_URI.	OPTIONAL
status	Enable or disable this port. The default is disabled .	OPTIONAL

Voice Commands

EXAMPLE:

```
=>voice profile modify  
SIP_URI = new  
NEW_URI = newer  
[username] =  
[password] =  
[displayname] =  
voiceport = FXS2  
[abbr] =  
:voice profile modify SIP_URI=new NEW_URI=newer voiceport=FXS2  
=>
```

RELATED COMMANDS:

voice profile add	Add profile.
voice profile delete	Delete profile.
voice profile list	Show all profiles.
voice profile flush	Flush all profiles.

voice profile list

Show all profiles.

SYNTAX:

```
voice profile list      SIP_URI = <{all}>
```

where:

SIP_URI	The SIP URI related to this voice port.	REQUIRED
---------	---	----------

RELATED COMMANDS:

voice profile add	Add profile.
voice profile delete	Delete profile.
voice profile modify	Modify profile.
voice profile flush	Flush all profiles.

voice profile flush

Flush all profiles.

SYNTAX:

```
voice profile flush
```

RELATED COMMANDS:

voice profile add	Add profile.
voice profile delete	Delete profile.
voice profile modify	Modify profile.
voice profile list	Show all profiles.

voice qos list

Show the QoS configuration.

SYNTAX:

```
voice qos list
```

EXAMPLE:

```
=>voice qos list
Traffic Type    QOS field      Value
-----
sign&control    dscp          af42
realtime        dscp          ef
=>
```

RELATED COMMANDS:

voice qos config Configure the QoS parameters.

voice qos config

Configure the QoS parameters.

SYNTAX:

```
voice qos config      type = <{sign&control | realtime}>
                      qosfield = <{dscp | precedence}>
                      [dscp = <{ef | af11 | af12 | af13 | af21 | af22 | af23 |
                        af31 | af32 | af33 | af41 | af42 | af43 | cs0 | cs1 | cs2 |
                        | cs3 | cs4 | cs5 | cs6 | cs7} or number>]
                      [precedence = <{routine | priority | immediate | flash |
                        flash-override | CRITIC-ECP | internetwork-control |
                        network-control} or number>]
```

where:

type	The type of traffic that needs QoS. Choose between: ▶ sign&control ▶ realtime The default is <i>sign&control</i> .	REQUIRED
qosfield	The QoS field to be used. Choose between: ▶ dscp ▶ precedence The default is <i>dscp</i> .	REQUIRED
dscp	The DSCP value.	OPTIONAL
precedence	The precedence value.	OPTIONAL

EXAMPLE:

```
=>voice qos config
type = sign&control
qosfield = precedence
[precedence] = priority
=>
```

RELATED COMMANDS:

voice qos list Show the QoS configuration.

voice services config

General supplementary service configuration.

SYNTAX:

```
voice services config
  [mwi_phone = <{immediate | deferred}>]
  [mwi_network = <{sollicited | unsollicited}>]
  [prefix_servcode = <{standard | nonstandard}>]
  [servcode_cmd = <{standard | nonstandard}>]
  [cfu_dest = <string>]
  [cfnr_timer = <number{0-600}>]
  [cfnr_dest = <string>]
  [cfbs_dest = <string>]
  [hfhf_feature = <{disabled|enabled}>]
```

where:

mwi_phone	The way the phone reacts on a message waiting indication. The default is immediate .	OPTIONAL
mwi_network	The way the network protocol implements the message waiting indication. The default is sollicited .	OPTIONAL
prefix_servcode	The way the service codes are assigned. The default is standard .	OPTIONAL
servcode_cmd	The way the service code command is used. The default is standard .	OPTIONAL
cfu_dest	The destination for the CFU service.	
cfnr_timer	A number between 0 and 600. Represents the CFNR timer value. The default is 10 .	OPTIONAL
cfnr_dest	The destination for the CFNR service.	OPTIONAL
cfbs_dest	The destination for the CFBS service.	OPTIONAL
hfhf_feature	Enable/disable hfhf_feature.	OPTIONAL

EXAMPLE:

```
=>voice services config
Phone MWI type      : immediate
Network MWI type    : sollicited
Prefix service code : standard
Service code command: standard
CFNR timer          : 10
CFNR destination    :
=>
```

Voice Commands

RELATED COMMANDS:

voice services list	Show the supplementary service configuration.
voice services provision	Provision a supplementary service.
voice services withdraw	Withdraw a supplementary service.
voice services activate	Activate a supplementary service.
voice services deactivate	Deactivate a supplementary service.
voice services assign	Assign a servicecode to a supplementary service in standard mode.
voice services assign_pxsc	Assign a servicecode to a supplementary service in non_standard mode.
voice services flush	Flush all supplementary services.

voice services list

Show the supplementary service configuration.

SYNTAX:

```
voice services list
```

EXAMPLE:

Service	Status	ActCode	DeactCode	RegCode	Provisioned
transfer	activated	*96	#96		yes
hold	activated	*94	#94		yes
waiting	activated	*43	#43		yes
mwii	deactivated	*98	#98		no
clip	activated	*30	#30		yes
clir	deactivated	*31	#31		no
colp	deactivated	*76	#76		no
3pty	activated	*95	#95		yes
forcedFXO	deactivated	*01*	#94		no
cfnr	deactivated	*61	#61	**61*	no
callreturn	deactivated	*69			no
ccbs	deactivated	*37	#37		no
clironcall	deactivated	*31*	#31*		no
waitingoncall	activated	*43*	#43*		yes

RELATED COMMANDS:

- | | |
|----------------------------|---|
| voice services config | General supplementary service configuration. |
| voice services provision | Provision a supplementary service. |
| voice services withdraw | Withdraw a supplementary service. |
| voice services activate | Activate a supplementary service. |
| voice services deactivate | Deactivate a supplementary service. |
| voice services assign | Assign a servicecode to a supplementary service in standard mode. |
| voice services assign_pxsc | Assign a servicecode to a supplementary service in non_standard mode. |
| voice services flush | Flush all supplementary services. |

voice services provision

Provision a supplementary service.

SYNTAX:

```
voice services provision type = <string>
```

where:

type	The supplementary service type.	REQUIRED
------	---------------------------------	----------

EXAMPLE:

```
=>voice services provision
type = transfer
:voice services provision type=transfer
=>
```

RELATED COMMANDS:

voice services config	General supplementary service configuration.
voice services list	Show the supplementary service configuration.
voice services withdraw	Withdraw a supplementary service.
voice services activate	Activate a supplementary service.
voice services deactivate	Deactivate a supplementary service.
voice services assign	Assign a servicecode to a supplementary service in standard mode.
voice services assign_pxsc	Assign a servicecode to a supplementary service in non_standard mode.
voice services flush	Flush all supplementary services.

voice services withdraw

Withdraw a supplementary service.

SYNTAX:

```
voice services      type = <string>
withdraw
```

where:

type	The supplementary service type.	REQUIRED
------	---------------------------------	----------

EXAMPLE:

```
=>voice services withdraw
type = transfer
:voice services withdraw type=transfer
=>
```

RELATED COMMANDS:

voice services config	General supplementary service configuration.
voice services list	Show the supplementary service configuration.
voice services provision	Provision a supplementary service.
voice services activate	Activate a supplementary service.
voice services deactivate	Deactivate a supplementary service.
voice services assign	Assign a servicecode to a supplementary service in standard mode.
voice services assign_pxsc	Assign a servicecode to a supplementary service in non_standard mode.
voice services flush	Flush all supplementary services.

voice services activate

Activate a supplementary service.

SYNTAX:

```
voice services      type = <string>
activate
```

where:

type	The supplementary service type.	REQUIRED
------	---------------------------------	----------

EXAMPLE:

```
=>voice services activate
type = hold
:voice services activate type=hold
=>
```

RELATED COMMANDS:

voice services config	General supplementary service configuration.
voice services list	Show the supplementary service configuration.
voice services provision	Provision a supplementary service.
voice services withdraw	Withdraw a supplementary service.
voice services deactivate	Deactivate a supplementary service.
voice services assign	Assign a servicecode to a supplementary service in standard mode.
voice services assign_pxsc	Assign a servicecode to a supplementary service in non_standard mode.
voice services flush	Flush all supplementary services.

voice services deactivate

Deactivate a supplementary service.

SYNTAX:

```
voice services      type = <string>
deactivate
```

where:

type	The supplementary service type.	REQUIRED
------	---------------------------------	----------

EXAMPLE:

```
=>voice services deactivate
type = hold
:voice services deactivate type=hold
=>
```

RELATED COMMANDS:

voice services config	General supplementary service configuration.
voice services list	Show the supplementary service configuration.
voice services provision	Provision a supplementary service.
voice services withdraw	Withdraw a supplementary service.
voice services activate	Activate a supplementary service.
voice services assign	Assign a servicecode to a supplementary service in standard mode.
voice services assign_pxsc	Assign a servicecode to a supplementary service in non_standard mode.
voice services flush	Flush all supplementary services.

voice services assign

Assign a servicecode to a supplementary service in standard mode.

SYNTAX:

```
voice services      type = <string>
assign            servicecode = <0...9>
```

where:

type	The supplementary service type.	REQUIRED
servicecode	The service activation code.	REQUIRED

EXAMPLE:

```
=>voice services assign
type = transfer
servicecode = 2
:voice services assign type=transfer servicecode=2
=>
```

RELATED COMMANDS:

voice services config	General supplementary service configuration.
voice services list	Show the supplementary service configuration.
voice services provision	Provision a supplementary service.
voice services withdraw	Withdraw a supplementary service.
voice services activate	Activate a supplementary service.
voice services deactivate	Deactivate a supplementary service.
voice services assign_pxsc	Assign a servicecode to a supplementary service in non_standard mode.
voice services flush	Flush all supplementary services.

voice services assign_pxsc

Assign a servicecode to a supplementary service in non_standard mode.

SYNTAX:

```
voice services      type = <string>
assign_pxsc        action = <{activate | deactivate}>
                    action_cfnr = <{activate | deactivate | register}>
                    action_act = <{activate}>
                    servicecode = <0...9 * #>
```

where:

type	The supplementary service type.	REQUIRED
action	The supplementary service action.	REQUIRED
action_cfnr	The supplementary service action.	REQUIRED
action_act	The supplementary service action.	REQUIRED
servicecode	The service activation code.	REQUIRED

RELATED COMMANDS:

voice services config	General supplementary service configuration.
voice services list	Show the supplementary service configuration.
voice services provision	Provision a supplementary service.
voice services withdraw	Withdraw a supplementary service.
voice services activate	Activate a supplementary service.
voice services deactivate	Deactivate a supplementary service.
voice services assign	Assign a servicecode to a supplementary service in standard mode.
voice services flush	Flush all supplementary services.

voice services flush

Flush all supplementary services.

SYNTAX:

```
voice services flush
```

RELATED COMMANDS:

voice services config	General supplementary service configuration.
voice services list	Show the supplementary service configuration.
voice services provision	Provision a supplementary service.
voice services withdraw	Withdraw a supplementary service.
voice services activate	Activate a supplementary service.
voice services deactivate	Deactivate a supplementary service.
voice services assign	Assign a servicecode to a supplementary service in standard mode.
voice services assign_pxsc	Assign a servicecode to a supplementary service in non_standard mode.

voice sip config

Configure the SIP general parameters.

SYNTAX:

```
voice sip config [useragentdomain = <string>]
[primproxyaddr = <string>]
[secproxyaddr = <string>]
[proxyport = <{string} or number>]
[primregaddr = <string>]
[secregaddr = <string>]
[regport = {string} or number]
[regexpire = <number{60-65535}>]
[notifier_addr = <string>]
[notifier_port = <{string} or number>]
[subscribe_expire = <number{60-65535}>]
[CWreply = <{182 | 180}>]
[transport = <{UDP | TCP}>]
[rtpmapstaticPT = <{disabled | enabled}>]
[reinvite_stop_audio = <{disabled | enabled}>]
[PRACK = <{disabled | enabled}>]
[clirformat = <{standard | nonstandard}>]
[DTMF*#inINFO = <{*#|1011}>]
[clip_consider_displayname = <{no|yes}>]
[sdp_ptime = <{10|20|30|notsent}>]
```

where:

useragentdomain	User Agent domain.	OPTIONAL
primproxyaddr	Primary outbound proxy address or FQDN.	OPTIONAL
secproxyaddr	Secondary outbound proxy address or FQDN.	OPTIONAL
proxyport	Outbound proxy port. See “ Supported TCP/UDP Port Names ” on page 749.	OPTIONAL
primregaddr	Primary registrar address or domain.	OPTIONAL
secregaddr	Secondary registrar address or domain.	OPTIONAL
regport	Registrar port. See “ Supported TCP/UDP Port Names ” on page 749.	OPTIONAL
regexpire	A number between 60 and 65535. Represents the registration expire time.	OPTIONAL
notifier_addr	Notifier address or FQDN used for MWI.	OPTIONAL
notifier_port	Notifier port used for MWI. See “ Supported TCP/UDP Port Names ” on page 749.	OPTIONAL
subscribe_expire	A number between 60 and 65535. The subscription expire time.	OPTIONAL

CWreply	The reply to be sent for a waiting call. Choose between: ▶ 180 ▶ 182.	OPTIONAL
transport	The transport type. Choose between: ▶ TCP ▶ UDP.	OPTIONAL
rtpmapstaticPT	Enable or disable adding RTP map i.c.o. static PT. The default is disabled .	OPTIONAL
reinvite_stop_audio	Enable or disable the explicit stop of audio stream in the SDP when sending a re-invite. The default is disabled .	OPTIONAL
PRACK	Enable or disable Provisional ACK on provisional responses. The default is disabled .	OPTIONAL
clirformat	The clir format. Choose between: ▶ standard ▶ nonstandard. The default is standard .	OPTIONAL
DTMF*#inINFO	Format used for */# in INFO method.	OPTIONAL
clip_consider_displayname	Consider the displayname for CLIP or not.	OPTIONAL
sdp_ptime	Consider the displayname for CLIP or not.	OPTIONAL

voice stats list

Show voice statistics.

SYNTAX:

```
voice stats list  voiceport = <{FXS1 | FXS2 | BT | all | VoIP}>
                  type = <{detailed | generic | all}>
```

where:

voiceport	Select the voice port to be used to get the statistics. Choose between: <ul style="list-style-type: none">▶ FXS1▶ FXS2▶ all	REQUIRED
type	Select the type of statistics to be listed. Choose between: <ul style="list-style-type: none">▶ detailed▶ generic▶ all: both the detailed and the generic statistics are shown.	REQUIRED

EXAMPLE:

```
=>voice stats list voiceport=FXS1 type=generic
Generic statistics for FXS1:
Total nbr of incoming calls      : 0
Successful nbr of incoming calls : 0
Missed nbr of incoming calls    : 0
Total nbr of outgoing calls     : 0
Successful nbr of outgoing calls : 0
Failed nbr of outgoing calls    : 0
=>
```

RELATED COMMANDS:

voice stats reset Reset the voice statistics.

voice stats reset

Reset the voice statistics.

SYNTAX:

```
voice stats reset  voiceport = <{FXS1 | FXS2 | BT | all | VoIP}>
                    type = <{detailed | generic | all}>
```

where:

voiceport	The voice port used to reset the statistics. Choose between: <ul style="list-style-type: none">▶ FXS1▶ FXS2▶ all	REQUIRED
type	Type of statistics to be reset. Choose between: <ul style="list-style-type: none">▶ detailed▶ generic▶ all: both the detailed and the generic statistics are reset.	REQUIRED

EXAMPLE:

```
=>voice stats reset voiceport=FXS1 type=generic
generic statistics for FXS1 successfully reset
=>voice stats list voiceport=FXS1 type=generic
Generic statistics for FXS1:

Total nbr of incoming calls      : 0
Successful nbr of incoming calls : 0
Missed nbr of incoming calls    : 0
Total nbr of outgoing calls     : 0
Successful nbr of outgoing calls : 0
Failed nbr of outgoing calls    : 0
=>
```

RELATED COMMANDS:

voice stats list Show voice statistics.

Wireless Commands

Introduction

This chapter describes the commands of the **wireless** command group.

Contents

This chapter covers the following commands:

wireless ifconfig	Configure the wireless settings.	702
wireless reset	Reset the wireless parameters.	704
wireless debug stats	Show the transfer statistics.	706
wireless macacl add	Add an ACL MAC entry.	707
wireless macacl config	Configure the ACL mode.	708
wireless macacl delete	Delete an ACL MAC entry.	709
wireless macacl flush	Flush ACL entries.	710
wireless macacl list	Show a list of all configured ACL entries.	711
wireless macacl modify	Modify an existing ACL entry.	712
wireless macacl register	Start a registration process (only for register mode).	713
wireless multiuni add	Add a multicast to unicast mapping.	714
wireless multiuni delete	Delete all multicast mappings.	715
wireless multiuni flush	Flush all multicast to unicast mappings.	716
wireless multiuni list	Show configured multicast to unicast mappings.	717
wireless multiuni scanresults	Display detected multicast mac addresses.	718
wireless qos config	Configure QoS settings.	719
wireless qos apacconfig	Configure AP EDCA parameters.	720
wireless qos staacconfig	Configure STA EDCA parameters.	721
wireless secmode config	Configure/show the security mode.	722
wireless secmode wep	Configure the WEP settings.	723
wireless secmode wpa-psk	Configure the WPA-PSK settings.	724
wireless secmode wpa	Configure the WPA settings.	725
wireless stations list	List the currently associated stations.	726
wireless wds add	Add a Wireless Distribution System (WDS) station.	727
wireless wds config	WDS configuration parameters.	728
wireless wds delete	Delete a WDS station.	729
wireless wds flush	Flush all WDS stations.	730
wireless wds list	List the WDS stations.	731
wireless wds scanresults	Scan all networks.	732

wireless ifconfig

Configure the wireless settings.

SYNTAX:

```
wireless ifconfig [state = <{enabled | disabled}>]
                  [channel = <{auto | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10
                                | 11 | 12 | 13}>]
                  [ssid = <quoted string>]
                  [any = <{disabled | enabled}>]
                  [interop = <{802.11b | 802.11b/g | 802.11g}>]
                  [frameburst = <{disabled | enabled}>]
```

where:

state	Enable or disable the interface operational status of wireless access point. The default is enabled .	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™ (recommended setting). ▶ a number between 1 and 14. <p>Note Depending on local regulations, the number of channels actually allowed to be used, may be restricted.</p> <p>The default is auto.</p>	OPTIONAL
ssid	The network name, also known as Service Set ID (SSID).	OPTIONAL
any	Make the SSID public or not. Choose between: <ul style="list-style-type: none"> ▶ enabled: the SpeedTouch™ broadcasts its SSID and accepts every client. ▶ disabled: the SpeedTouch™ does not broadcast its SSID and accepts only those clients who have the correct SSID. <p>The default is enabled.</p>	OPTIONAL
interop	Select the interoperability mode. Choose between: <ul style="list-style-type: none"> ▶ 802.11b: only IEEE 802.11b clients can connect to the SpeedTouch™. ▶ 802.11b/g: IEEE 802.11g and b clients can connect to the SpeedTouch™. <p>Note When a IEEE 802.11b client connects to the SpeedTouch™, the modulation rate drops to 11Mbps for this client.</p> <ul style="list-style-type: none"> ▶ 802.11g: only IEEE 802.11g clients can connect to the SpeedTouch™. <p>The default is 802.11b/g.</p>	OPTIONAL
frameburst	Enable or disable framebursting. The default is disabled .	OPTIONAL

EXAMPLE:

```
=>wireless] ifconfig
State : enabled
Network name (SSID) : SpeedTouchBF4511
Public network (any) : enabled
Channel : 1 [auto]
RTS Threshold : 2347
Short Retry Limit : 7
Long Retry Limit : 4
Beacon Period : 100
Rate : 54 Mbps
Interoperability : 802.11b/g
Protection : auto
Protection Mode : rtscts
Protection Trigger : local&overlap
Shortslot : auto
Framebursting : disabled
Locale : Europe
Rate Set : 1(b) 2(b) 5.5(b) 6 9 11(b) 12 18 24 36 48 54
Dtim interval : 3 (every 300 msec)
=>
```

wireless reset

Reset the wireless parameters.

The following wireless parameters are reset to their default settings:

- ▶ Access Point (AP) + communication channel
- ▶ Security mode
- ▶ MAC Access Control List (ACL) (MACACL mode)
- ▶ Pre-shared keys
- ▶ MACACL list is cleared.

SYNTAX:

```
wireless reset proceed = <{disabled | enabled}>
```

where:

proceed	Confirm to reset the settings to factory defaults. Choose between: ▶ disabled ▶ enabled.	REQUIRED
---------	---	----------

EXAMPLE:

```
=>wireless ifconfig
State : enabled
Network name (SSID) : SpeedTouchBF4511
Public network (any) : enabled
Channel : 8 [manual]
RTS Threshold : 2347
Short Retry Limit : 7
Long Retry Limit : 4
Beacon Period : 100
Rate : 54 Mbps
Interoperability : 802.11b/g
Protection : auto
Protection Mode : rtscts
Protection Trigger : local&overlap
Shortslot : auto
Framebursting : disabled
Locale : Europe
Rate Set : 1(b) 2(b) 5.5(b) 6 9 11(b) 12 18 24 36 48 54
Dtim interval : 3 (every 300 msec)
=>wireless reset proceed=enabled
=>wireless] ifconfig
State : enabled
Network name (SSID) : SpeedTouchBF4511
Public network (any) : enabled
Channel : 1 [auto]
RTS Threshold : 2347
Short Retry Limit : 7
Long Retry Limit : 4
Beacon Period : 100
Rate : 54 Mbps
Interoperability : 802.11b/g
Protection : auto
Protection Mode : rtscts
Protection Trigger : local&overlap
Shortslot : auto
Framebursting : disabled
Locale : Europe
Rate Set : 1(b) 2(b) 5.5(b) 6 9 11(b) 12 18 24 36 48 54
Dtim interval : 3 (every 300 msec)
=>
```

wireless debug stats

Show the transfer statistics.

SYNTAX:

```
wireless debug stats
```

EXAMPLE:

```
=>wireless debug stats
Good Rx Packets      : 1040
Bad Rx Packets       : 0
Good Tx Packets      : 1234
Bad Tx Packets       : 3
=>
```

wireless macacl add

Add an ACL MAC entry.

SYNTAX:

```
wireless macacl add hwaddr = <hardware-address>
    permission = <{allow|deny}>
        [name = <quoted string>]
```

where:

hwaddr	The Ethernet MAC address of the ACL entry.	REQUIRED
permission	The action to be performed on ACL entry. Choose between: ▶ allow : the wireless client is allowed entry. ▶ deny : the wireless client is not allowed entry.	REQUIRED
name	The name of the wireless client.	OPTIONAL

EXAMPLE:

```
=>wireless macacl list
Station Name      hardware address      Permission
ST120g           00:02:2d:5f:b6:9e    accept
=>wireless macacl add hwaddr=01:67:d0:23:56:cd permission=allow name=Test2
=>wireless macacl list
Station Name      hardware address      Permission
ST120g           00:02:2d:5f:b6:9e    accept
Test2            01:67:d0:23:56:cd    accept
=>
```

wireless macacl config

Configure the ACL mode.

SYNTAX:

```
wireless macacl config [control = <{lock | unlock | register}>]
```

where:

control	Select the ACL mode. Choose between: ▶ lock : The ACL is locked. No new entries are allowed in the ACL. ▶ unlock : Every wireless client can associate to the SpeedTouch™. ▶ register : A wireless client is only added to the ACL after using the Register Button. The default is unlock .	OPTIONAL
---------	--	----------

EXAMPLE:

```
=>wireless macacl config
Access Control : unlock
=>wireless macacl config control=register
=>wireless macacl config
Access Control : register
=>
```

wireless macacl delete

Delete an ACL MAC entry.

SYNTAX:

```
wireless macacl delete    hwaddr = <hardware-address>
```

where:

hwaddr The Ethernet MAC address of the ACL entry.	REQUIRED
--	----------

EXAMPLE:

```
=>wireless macacl list
Station Name      hardware address      Permission
ST120g           00:02:2d:5f:b6:9e    accept
Test2            01:67:d0:23:56:cd    accept
=>wireless macacl delete hwaddr=01:67:d0:23:56:cd
=>wireless macacl list
Station Name      hardware address      Permission
ST120g           00:02:2d:5f:b6:9e    accept
=>
```

wireless macacl flush

Flush ACL entries.

SYNTAX:

```
wireless macacl flush      proceed = <{disabled | enabled}>
```

where:

proceed	Confirm to reset the ACL settings to factory defaults. Choose between: ▶ disabled ▶ enabled.	REQUIRED
---------	---	----------

EXAMPLE:

```
=>wireless macacl list
Station Name          hardware address      Permission
ST120g               00:02:2d:5f:b6:9e    accept
=>wireless macacl flush proceed=disabled
=>wireless macacl list
Station Name          hardware address      Permission
ST120g               00:02:2d:5f:b6:9e    accept
=>wireless macacl flush proceed = enabled
=>wireless macacl list
No stations defined.
=>
```

wireless macacl list

Show a list of all configured ACL entries.

SYNTAX:

```
wireless macacl list
```

EXAMPLE:

```
=>wireless macacl list
Station Name           hardware address      Permission
ST120g                00:02:2d:5f:b6:9e    accept
Test2                 01:67:d0:23:56:cd    accept
=>
```

wireless macacl modify

Modify an existing ACL entry.

SYNTAX:

```
wireless macacl modify    hwaddr = <hardware-address>
                           [permission = <{allow | deny}>]
                           [name = <quoted string>]
```

where:

hwaddr	The Ethernet MAC address of the ACL entry.	REQUIRED
	Note If the MAC address is not present in the ACL list, the command will not be excuted.	
permission	The action to be performed on ACL entry. Choose between: <ul style="list-style-type: none"> ▶ allow: the wireless client is allowed entry. ▶ deny: the wireless client is not allowed entry. 	OPTIONAL
name	The name of the wireless client.	OPTIONAL

EXAMPLE:

```
=>wireless macacl list
Station Name      hardware address      Permission
ST120g           00:02:2d:5f:b6:9e    accept
Test2            01:67:d0:23:56:cd    deny
=>wireless macacl modify hwaddr=01:67:d0:23:56:cd permission=allow name=Test3
=>wireless macacl list
Station Name      hardware address      Permission
ST120g           00:02:2d:5f:b6:9e    accept
Test3            01:67:d0:23:56:cd    accept
=>
```

wireless macacl register

Start a registration process (only for register mode).

Prerequisite

The register mode must be enabled (with the command :**wireless macacl config control=register**).



If the register mode is not enabled when the registration process is started, a warning will be displayed.

Registration Process

After executing this command, the ACL will be unlocked for a time period of 60 seconds.

Any wireless client who tries to associate with the SpeedTouch™ during this period and who has the correct wireless settings, will be added to the ACL.

SYNTAX:

```
wireless macacl register proceed = <{disabled | enabled}>
```

where:

proceed	Confirm to start the registration process. Choose between: ▶ disabled ▶ enabled.	REQUIRED
---------	---	----------

EXAMPLE:

```
=>wireless macacl register proceed=enabled
Registration process will be started, new stations can associate during 60 seconds.
=>wireless macacl register proceed=disabled
=>
```

wireless multiuni add

Add a multicast to unicast mapping.

SYNTAX:

```
wireless multiuni add    multicast = <hardware-address>
                           unicast = <hardware-address>
```

where:

multicast	Multicast MAC.	REQUIRED
unicast	Unicast MAC.	REQUIRED

wireless multiuni delete

Delete all multicast mappings.

SYNTAX:

```
wireless multiuni delete multicast = <hardware-address>
```

where:

multicast	Multicast MAC.	REQUIRED
-----------	----------------	----------

wireless multiuni flush

Flush all multicast to unicast mappings.

SYNTAX:

```
wireless multiuni flush
```

wireless multiuni list

Show configured multicast to unicast mappings.

SYNTAX:

```
wireless multiuni list
```

wireless multiuni scanresults

Display detected multicast mac addresses.

SYNTAX:

```
wireless multiuni scanresults [rescan = <{no | yes}>]
```

where:

rescan	Rescan for new multicast addresses. The default is <i>no</i> .	OPTIONAL
--------	---	----------

wireless qos config

Configure QoS settings.

SYNTAX:

```
wireless qos config [mode = <{disabled | wmm}>]
```

where:

mode	Active WMM. The default is disabled .	OPTIONAL
------	---	----------

EXAMPLE:

```
=>wireless qos config
mode      : wmm
ackpolicy : normal
=>
```

wireless qos apacconfig

Configure AP EDCA parameters.

SYNTAX:

```
wireless qos apacconfig [class = <{AC_BE | AC_BK | AC_VI | AC_VO}>]
[cwmax = <{1 | 3 | 7 | 15 | 31 | 63 | 127 | 255 | 511
| 1023 | 2047 | 4095 | 8191 | 16383 | 32767}>]
[cwmin = <{0 | 1 | 3 | 7 | 15 | 31 | 63 | 127 | 255}>]
[aifsn = <number{1-15}>]
[txop = <number>]
```

where:

class	AC class.	OPTIONAL
cwmax	CWmax configuration.	OPTIONAL
cwmin	CWmin configuration.	OPTIONAL
aifsn	A number between 1 and 15. Represents the aifsn configuration.	OPTIONAL
txop	Txoplimit [us] configuration.	OPTIONAL

EXAMPLE:

```
=>wireless qos apacconfig
AP EDCA table
  Class   CWmin    CWmax     AIFSN      ACM      TXOP      Units
  AC_BK    15       1023      7          0          0          0
  AC_BE    15       63         3          0          0          0
  AC_VI    7         15         1          0        3008        94
  AC_VO    3         7          1          0        1504        47
=>
```

wireless qos staacconfig

Configure STA EDCA parameters.

SYNTAX:

```
wireless qos staacconfig [class = <{AC_BE | AC_BK | AC_VI | AC_VO}>]
[cwmax = <{1 | 3 | 7 | 15 | 31 | 63 | 127 | 255 | 511
| 1023 | 2047 | 4095 | 8191 | 16383 | 32767}>]
[cwmin = <{0 | 1 | 3 | 7 | 15 | 31 | 63 | 127 | 255}>]
[aifsn = <number{1-15}>]
[txop = <number>]
```

where:

class	AC class.	OPTIONAL
cwmax	CWmax configuration.	OPTIONAL
cwmin	CWmin configuration.	OPTIONAL
aifsn	A number between 1 and 15. Represents the aifsn configuration.	OPTIONAL
txop	Txoplimit [us] configuration.	OPTIONAL

EXAMPLE:

```
=>wireless qos apacconfig
STA EDCA table
 Class   CWmin   CWmax     AIFSN     ACM      TXOP    Units
 AC_BK    15      1023       7        0        0        0
 AC_BE    15      1023       3        0        0        0
 AC_VI    7       15         2        0      3008       94
 AC_VO    3       7          2        0      1504       47
=>
```

wireless secmode config

Configure/show the security mode.

SYNTAX:

```
wireless secmode config [mode = <{disable | wep | wpa-psk | wpa}>]
```

where:

mode	The security mode. Choose between: ▶ disable : no security ▶ wep : Wired Equivalent Privacy (WEP) ▶ wpa-psk : Wi-Fi Protected Access (WPA) - Pre-Shared Key (PSK) ▶ wpa : Wi-Fi Protected Access (WPA).	OPTIONAL
------	--	----------

The default is **disable** (in other words, by default, there is no security. This is to enable an easy first use).

EXAMPLE:

```
=>wireless secmode config
No security enabled, open authentication, no data encryption
=>wireless secmode config mode=wep
=>wireless secmode config
Security level WEP:
  encryption key : 26FF871306
=>wireless secmode config mode=wpa-psk
=>wireless secmode config
Security level WPA-PSK:
  preshared key   : 26FF871306
  encryption      : TKIP
  rekeying        : every 0 seconds
=>
```

wireless secmode wep

Configure the WEP settings.



If the current security mode is WEP, then the security mode must first be changed, before the WEP settings can be (re)configured.

SYNTAX:

```
wireless secmode wep      [encryptionkey = <quoted string>]
```

where:

encryptionkey	The secret key to be used for WEP encryption. Two key lengths (64 or 128 bits) and two formats (alphanumeric or hexadecimal) are permitted: <ul style="list-style-type: none">▶ for 64-bit encryption: the key must consist of 10 hexadecimal digits or 5 alphanumeric characters▶ for 128-bit encryption: the key must consist of 26 hexadecimal digits or 13 alphanumeric characters.	OPTIONAL
---------------	--	----------

Note If no encryptionkey is filled in, then the default encryptionkey is used.

EXAMPLE:

```
=>wireless secmode wep
Security level WEP:
  encryption key  : FFA22E9CCA
  enabled         : yes
=>
```

wireless secmode wpa-psk

Configure the WPA-PSK settings.



If the current security mode is WPA-PSK, then the security mode must first be changed, before the WPA-PSK settings can be (re)configured.

SYNTAX:

```
wireless secmode wpa-psk [presharedkey = <quoted string>]
[version = <{WPA | WPA2 | WPA+WPA2}>]
```

where:

presharedkey	The preshared key is an alphanumeric string with a length from 8 to 64 characters.	OPTIONAL
Note If no presharedkey is filled in, then the default presharedkey is used.		
version	The WPA version. Choose between: ▶ WPA ▶ WPA2 ▶ WPA+WPA2	OPTIONAL

EXAMPLE:

```
=>wireless secmode wpa-psk
Security level WPA-PSK:
preshared key      : E61BFC42AF
encryption          : TKIP
rekeying            : every 0 seconds
version             : WPA
enabled             : no
=>
```

wireless secmode wpa

Configure the WPA settings.

-  If the current security mode is WPA, then the security mode must first be changed, before the WPA settings can be (re)configured.

SYNTAX:

```
wireless secmode wpa      [radiuskey = <quoted string>]
                           [rekeysec = <number{0-86400}>]
                           [radiusip = <ip-address>]
                           [radiusport = <number{0-65535}>]
                           [radiussto = <number{1800-172800}>]
                           [version = <{WPA | WPA2 | WPA+WPA2}>]
```

where:

radiuskey	The radius key.	OPTIONAL
rekeysec	A number between 0 and 86400. Represents the rekey interval.	OPTIONAL
radiusip	The radius server IP address.	OPTIONAL
radiusport	The radius server port.	OPTIONAL
radiussto	The Radius Session Timeout.	OPTIONAL
version	The WPA version. Choose between: <ul style="list-style-type: none"> ▶ WPA ▶ WPA2 ▶ WPA+WPA2 	OPTIONAL

EXAMPLE:

```
=>wireless secmode wpa
Security level WPA:
  radius key      :
  radius ip:port  :0.0.0.0:1812
  encryption      : TKIP
  rekeying        : every 0 seconds
  version         : WPA
  enabled          : no
=>
```

wireless stations list

List the currently associated stations.

SYNTAX:

```
wireless stations list      [hwaddr = <hardware-address>]
```

where:

hwaddr	The MAC address of the station.	OPTIONAL
--------	---------------------------------	----------

EXAMPLE:

```
=>wireless stations list
Station Name          Hardware Address   Authentication
ST120g                00:02:2D:5F:B6:9E  WEP-64 used

Total number of associated stations : 1
=>
```

wireless wds add

Add a Wireless Distribution System (WDS) station.



Up to four WDS stations can be added.

SYNTAX:

```
wireless wds add bssid = <hardware-address>
    [name = <quoted string>]
```

where:

bssid	The WDS Base Service Set Identifier (BSSID).	REQUIRED
name	A user specified reference name.	OPTIONAL

EXAMPLE:

```
=>wireless wds list
SpeedTouch180          00:90:96:4F:38:E4
=>wireless wds add bssid=00:90:96:62:5d:69 name="SpeedTouch Sascha"
=>wireless wds list
SpeedTouch180          00:90:96:4F:38:E4
SpeedTouch Sascha       00:90:96:62:5D:69
=>
```

Wireless Commands

wireless wds config

WDS configuration parameters.

SYNTAX:

```
wireless wds config [state = <{disabled | enabled}>]
```

where:

state	Enable or disable the WDS functionality. The default is enabled .	OPTIONAL
-------	---	----------

EXAMPLE:

```
=>wireless wds config
State                  : enabled
Own BSSID              : 00:90:96:52:2D:89
=>
```

wireless wds delete

Delete a WDS station.

SYNTAX:

```
wireless wds delete bssid = <hardware-address>
```

where:

bssid The BSSID of the WDS station to be deleted.

REQUIRED

EXAMPLE:

```
=>wireless wds list
SpeedTouch180          00:90:96:4F:38:E4
SpeedTouch Sascha       00:90:96:62:5D:69
=>wireless wds delete bssid=00:90:96:62:5d:69
=>wireless wds list
SpeedTouch180          00:90:96:4F:38:E4
=>
```

Wireless Commands

wireless wds flush

Flush all WDS stations.

SYNTAX:

```
wireless wds flush
```

wireless wds list

List the WDS stations.

SYNTAX:

```
wireless wds list
```

EXAMPLE:

```
=>wireless wds list
SpeedTouch180          00:90:96:4F:38:E4
SpeedTouch Sascha       00:90:96:62:5D:69
=>
```

Wireless Commands

wireless wds scanresults

Scan all networks.

SYNTAX:

```
wireless wds scanresults [rescan = <{disabled | enabled}>]
```

where:

rescan	Perform a new scan (enabled) or not (disabled). The default is disabled .	OPTIONAL
--------	---	----------

EXAMPLE:

```
=>wireless wds scanresults rescan=yes
All associations with the AP will be lost, due to the wireless network scan!
SSID                           channel bssid          noise      rssi
SpeedTouchB39DEB                9  00:90:96:af:48:5a -89 dBm    -76 dBm
<Protected Network>            11 00:02:8a:37:01:c3 -76 dBm    -52 dBm
<Protected Network>            11 00:02:2d:2d:f9:e2 -76 dBm    -48 dBm
SpeedTouchDE0566                 1  00:30:f1:9f:06:0c -80 dBm    -56 dBm
=>
```

Abbreviations

The table below lists all the abbreviations used in the CLI Reference Guide.

Abbreviation	Description
AAL5	ATM Adaption Layer 5
ACL	Access Control List
ADSL	Asymmetric Digital Subscriber Line
AES	Advanced Encryption System
AF	Assured Forwarding
AP	Access Point
ARP	Address Resolution Protocol
ATM	Asynchronous Transfer Mode
BSSID	Base Service Set IDentifier
CA	Certificate Authority
CAPI	Common ISDN Application Interface
CC	Continuity Check
CEP	Certificate Enrollment Protocol
CHAP	Challenge Handshake Authentication Protocol
CIDR	Classless Inter Domain Routing
CLI	Command Line Interface
CLP	Cell Loss Priority
CO	Central Office
CPE	Customer Premises Equipment
CRL	Certificate Revocation List
CTD	Conformance Traffic Descriptor
CWMP	CPE WAN Management Protocol
DHCP	Dynamic Host Configuration Protocol
DN	Distinguished Name
DNS	Domain Name System
DPD	Dead Peer Detection
DSCP	Differentiated Services Code Point
DSD	Differentiated Service Delivery
DSL	Digital Subscriber Line
ECN	Explicit Congestion Notification
EF	Expedited Forwarding
FCS	Frame Check Sequence
FTP	File Transfer Protocol
GRE	General Routing Encapsulation
GRP	Gateway Routing Protocol

Abbreviations

Abbreviation	Description
HDLC	High-level Data Link Control
HTTP	HyperText Transfer Protocol
ICMP	Internet Control Message Protocol
IDS	Intrusion Detection System
IGD	Internet Gateway Device
IGMP	Internet Group Management Protocol
IKE	Internet Key Exchange
IP	Internet Protocol
IP oA	IP over ATM
IPCP	Internet Protocol Control Protocol
IPSec	IP Security
ISDN	Integrated Services Digital Network
ISI	Independent Set ID
KB	Kilobytes
Kbps	Kilobits per second
LAN	Local Area Network
LCP	Link Control Protocol
LDAP	Light-weight Directory Access Protocol
LLC	Logical Link Control
MAC	Medium Access Control
MC	MultiCast
MD5	Message Digest 5
MGCP	Media Gateway Control Protocol
MLP	Multi-Level access Policy
MPLS	Multiprotocol Label Switching
MTU	Maximum Transmission Unit
NAPT	Network Address and Port Translation
NAT	Network Address Translation
NLPID	Network Layer Protocol IDentifiers
NTP	Network Time Protocol
OAM	Operation and Maintenance
OBG	On Board Controller
OID	Object IDentifier
PAP	Password Authentication Protocol
PBX	Private Branch Exchange
PFS	Perfect Forward Secrecy
PKCS	Public Key Cryptography Standard
PKI	Public Key Infrastructure
POTS	Plain Old Telephone Service

Abbreviation	Description
PPP	Point-to-Point Protocol
PPPoA	PPP over ATM
PPPoE	PPP over Ethernet
PPTP	Point-to-Point Tunneling Protocol
PSD	Power Spectral Density
PSK	Pre-Shared Key
PVC	Permanent Virtual Channel
QoS	Quality of Service
RIP	Routing Information Protocol
RTC	Real Time Clock
RTT	Round Trip Time
SAs	Security Associations
SFTP	Secure File Transfer Protocol
SHDSL	Single Pair High-speed Digital Subscriber Line
SIP	Session Initiation Protocol
SLA	Service Level Agreement
SMTP	Simple Mail Transfer Protocol
SNAP	Sub Network Access Protocol
SNMP	Simple Network Management Protocol
SNR	Signal-to-Noise Ratio
SNTP	Simple Network Time Protocol
SSH	Secure SHell
SSID	Service Set IDentifier
TCP	Transmission Control Protocol
TFTP	Trivial File Transfer Protocol
TKIP	Temporal Key Integrity Protocol
ToS	Type of Service
TTL	Time To Live
UA	User Agent
UDP	User Datagram Protocol
ULP	Upper Layer Protocol
UPnP	Universal Plug and Play
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
USB	Universal Serial Bus
VC	Virtual Channel
VCI	Virtual Channel Identifier
VCMUX	Virtual Channel MUltipleXing
VDSL	Very high speed Digital Subscriber Line

Abbreviations

Abbreviation	Description
VLAN	Virtual Local Area Network
VLSM	Variable Length Subnet Masking
VP	Virtual Path
VPI	Virtual Path Identifier
VPN	Virtual Private Networking
WAN	Wide Area Network
WDS	Wireless Distribution System
WEP	Wired Equivalent Privacy
WFQ	Weighted Fair Queueing
WINS	Windows Internet Naming Service
WLAN	Wireless LAN
WPA	Wi-Fi Protected Access
WRR	Weighted Round Robin

System Logging Messages

Introduction

This chapter lists the different System Logging messages.

Contents

This chapter lists the different System Logging messages of the following modules:

Auto-PVC Module	738
Configuration Module	738
DHCP Client Module	738
DHCP Relay Module	739
DHCP Server Module	739
Dyndns Module	740
Firewall Module	740
LOGIN Module	741
Kernel Module	741
Linestate Module	741
NAPT Module	741
PPP Module	742
PPTP Module	742
RIP Module	743
Routing Module	744
Session Module	744
SNTP Module	744
Software Module	745
UPnP Module	745

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	DHCC lease ip-address <ip-address> bound to intf <intf_id>
LOCAL2	WARNING	DHCC intf <intf_id> renews lease ip-address <ip-address>
LOCAL2	WARNING	DHCC intf <intf_id> rebinds lease ip-address <ip-address> from server<\$paratext><Default ¶ Font>(<ip-address>)
LOCAL2	WARNING	DHCC offer received from <ip-address> (can be relay agent) for intf <intf_id>
LOCAL2	WARNING	DHCC server (<ip-address>) offers <ip-address> to intf <intf_id>
LOCAL2	WARNING	DHCC unable to configure ip address: <ip-address> (bootp-reply)
LOCAL2	WARNING	DHCC bootp lease ip-address <ip-address> bound to intf <intf_id> from server (<ip-address>)
LOCAL2	WARNING	DHCC <ip-address> already configured on intf <intf_id>: failure
LOCAL2	WARNING	DHCC <ip-address> (<ip-address>) set on intf <intf_id>: {failure} {lock}
LOCAL2	WARNING	DHCC <ip-address> deleted: {failure} {lock}

DHCP Relay Module

Facility	Severity	Contents
LOCAL2	WARNING	DHCR relay: Dropping boot rqs on interface <intf_id> due to invalid giaddr for server (<ip-address>)
LOCAL2	WARNING	DHCR relay: Dropping boot reply with invalid relay agent option from <intf_id>
LOCAL2	WARNING	DHCR relay: Dropping boot request containing the relay agent option from <intf_id>
LOCAL2	WARNING	DHCR relay: Dropping boot reply to unknown interface from <intf_id>
LOCAL2	WARNING	DHCR relay: Dropping boot reply to inactive interface <intf_id>
LOCAL2	WARNING	DHCR relay: Dropping boot reply to inactive interface <intf_id>
LOCAL2	WARNING	DHCR relay: Dropping boot request packet with spoofed giaddr field from <intf_id>
LOCAL2	WARNING	DHCR relay: Dropping boot request received on unknown interface from <intf_id>
LOCAL2	WARNING	DHCR relay: Dropping boot request on inactive interface <intf_id>
LOCAL2	WARNING	DHCR relay: Dropping boot request with invalid hops field on interface <intf_id>
LOCAL2	WARNING	DHCR relay: Dropping boot request with invalid giaddr on interface <intf_id>

DHCP Server Module

Facility	Severity	Contents
LOCAL2	WARNING	DHCS server: <DHCP Offer DHCP ACK> cannot be send due to invalid server identifier
LOCAL2	WARNING	DHCS server: DHCPACK cannot be send due to invalid server identifier
LOCAL2	WARNING	DHCS server: DHCPNAK cannot be send due to invalid server identifier
LOCAL2	WARNING	DHCS server up
LOCAL2	WARNING	DHCS server went down

Dyndns Module

Facility	Severity	Contents
DYNDNS	WARNING	<DYNDNS_STR_ID> Connection failed to <dyndns_service> for client <dyndns_client>
DYNDNS	WARNING	<DYNDNS_STR_ID> Failed to resolve host name <dyndns_service> for client <dyndns_client>
DYNDNS	WARNING	<DYNDNS_STR_ID> Server Timeout(<dyndns_service>) for client <dyndns_client>
DYNDNS	WARNING	<DYNDNS_STR_ID> Update failed to server <dyndns_service> for client <dyndns_client>
DYNDNS	NOTICE	<DYNDNS_STR_ID> client <dyndns_client> <"started" "stopped">
DYNDNS	WARNING	<DYNDNS_STR_ID> Update failed for client <dyndns_client>, incomplete configuration
DYNDNS	WARNING	<DYNDNS_STR_ID> Update failed for client <dyndns_client> (<message>), host <hostname> has not been updated
DYNDNS	WARNING	<DYNDNS_STR_ID> Update failed for client <dyndns_client>
DYNDNS	NOTICE	<DYNDNS_STR_ID> Host <hostname> has been updated successfully by <dyndns_service>
DYNDNS	WARNING	<DYNDNS_STR_ID> Authentication failed to server <dyndns_service>

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>

LOGIN 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 up (downstream: <bitrate_in> kbit/s, upstream: <bitrate_out> kbit/s; output Power Down: <outputPowerDn> dBm, Up: <outputPowerUp> dBm; line Attenuation Down: <lineAttenuationDn> dB, Up: <lineAttenuationUp> dB; snr Margin Down: <snrMarginDn> dB, Up: <snrMarginUp> dB)"

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 authenticate-request
AUTH	NOTICE	PPP PAP remote user <remote_user_name> successful authenticated
AUTH	ERROR	PPP PAP authentication for remote user <remote_user> failed
AUTH	INFO	PPP PAP Authenticate Ack received
AUTH	INFO	PPP PAP Authenticate Nack received
AUTH	INFO	PPP PAP Authenticate Request sent
AUTH	ERROR	PPP PAP authentication failed (<intf name>)
AUTH	ERROR	PPP CHAP authentication failed (<intf name>)
AUTH	INFO	PPP CHAP authentication failed [protocol reject(server)]
AUTH	INFO	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}
LOCAL0	ERROR	PPP IPCP cannot determine remote IP address (<intf name>)
LOCAL0	ERROR	PPP IPCP cannot determine locale IP address (<intf name>)

PPTP Module

Facility	Severity	Contents
LOCAL0	WARNING	PPTP tunnel (<Pbname>) up:(<ip addr>)
LOCAL0	WARNING	PPTP tunnel (<Pbname>) down:(<ip addr>)

RIP Module

Facility	Severity	Contents
LOCAL1	INFO	RIP Packet received from unknown interface
LOCAL1	INFO	RIP Packet size is smaller than minimum size
LOCAL1	INFO	RIP Packet size is greater than maximum size
LOCAL1	INFO	RIP Wrong RIP packet alignment
LOCAL1	INFO	RIP RIP version 0 with command <command-name> received
LOCAL1	INFO	RIP Reserved field not zero in RIP header
LOCAL1	INFO	RIP RIP is not enabled for network address <ip-address>
LOCAL1	INFO	RIP Packet's v<version_nr> does not match to RIP v<version_nr>
LOCAL1	INFO	RIP Packet's v<version_nr> does not match to RIP v<version_nr> on interface <intf-name>
LOCAL1	INFO	RIP Packet v<version_nr> is dropped because authentication is disabled on interface <intf-name>
LOCAL1	INFO	RIP Simple password authentication failed on interface<intf-name>
LOCAL1	INFO	RIP No authentication in RIP packet
LOCAL1	INFO	RIP Obsolete command <command-name> received
LOCAL1	INFO	RIP Unknown RIP command received
LOCAL1	INFO	RIP Response does not come from default RIP port
LOCAL1	INFO	RIP Datagram doesn't come from a valid neighbor: <ip-address>
LOCAL1	INFO	RIP Unsupported family from <ip-address>
LOCAL1	INFO	RIP Network is net 127, class D or class E network
LOCAL1	INFO	RIP Address <ip-address> is my own address, net 0 or not unicast
LOCAL1	INFO	RIP RIPv1 packet with incorrect must be zero fields
LOCAL1	INFO	RIP Route metric is not in the 1-16 range
LOCAL1	INFO	RIP Nexthop address is not directly reachable <ip-address>
LOCAL1	INFO	RIP RIPv2 address <ip-address> is not correct mask /<mask> applied
LOCAL1	INFO	RIP Not configured for sending RIPv1 packet on interface <intf-name>
LOCAL1	INFO	RIP RIP routing table is full
LOCAL1	INFO	RIP Neighbor <ip-address> is not connected to direct network
LOCAL1	INFO	RIP Interface <intf-name> has not any valid local IP address for sending RIPv2 packets
LOCAL1	INFO	RIP Interface <intf-name> has not any valid %s address for sending RIPv<version_nr> packets
LOCAL1	INFO	RIP IP Address <ip-address> not found in RIP table

Routing Module

Facility	Severity	Contents
LOCAL1	INFO	GRP Default destination is routed via gateway <ip_address>
LOCAL1	INFO	GRP Default destination is not routed anymore via gateway <ip_address>
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>

Session Module

Facility	Severity	Contents
AUTH	NOTICE	SESSION TIMEOUT Timeout! (after <seconds> sec)

SNTP Module

Facility	Severity	Contents
NTP	WARNING	SNTP Unable to contact server: <SNTP server ip>
NTP	WARNING	SNTP Server not synchronized: <SNTP server ip>
NTP	WARNING	SNTP Invalid response from server: <SNTP server ip>
NTP	INFO	SNTP Synchronized to server: <SNTP server ip>
NTP	INFO	SNTP Synchronized again to server: <SNTP server ip>
NTP	WARNING	SNTP Roundtrip exceeds limits
NTP	ERROR	SNTP No server(s) configured, check configuration
NTP	ERROR	SNTP Systemtime update: time setting <systemtime> > new time setting: <new time>

Software Module

Facility	Severity	Contents
KERN	INFO	SOFTWARE File <Filename> <receive transmit> initiated
KERN	INFO	SOFTWARE File <Filename> <receive transmit><"" not> successful terminated"

UPnP Module

Facility	Severity	Contents
WARNING	SECURITY	UPnP<ActionName> refused for ip=<ip_address>
NOTICE	SECURITY	UPnP <ActionName> (<Error_string>) for ip=<ip_address>
NOTICE	SECURITY	UPnP action <ActionName> from ip=<ip_address> (<Error_string>)

VOIP Module

Facility	Severity	Contents
INFO	USER	VOIP: Send RAS message <msgName> (rejectReason = <reason>)
INFO	USER	VOIP: Recv RAS message <msgName>
INFO	USER	VOIP: Send RAS message <msgName>
INFO	USER	VOIP: Recv message <msgName> <reason> <causeValue>
INFO	USER	VOIP: Recv message <msgName> (rejectReason = <reason>)
INFO	USER	VOIP: Recv message <msgName>
INFO	USER	VOIP: Send message <msgName> <reason> <causeValue>
INFO	USER	VOIP: Send message <msgName> (rejectReason = <reason>)
INFO	USER	VOIP: Send message <msgName>
INFO	USER	VOIP: Recv <buffer>
INFO	USER	VOIP: Send <buffer>

System Logging Messages

746

speedtouch™

E-DOC-CTC-20060414-0005 v1.0

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 IP Protocols	748
Supported TCP/UDP Port Names	749
Supported ICMP Type Names	752
Supported Facilities	753
Supported Severities	754
IP Precedence	754
Differentiated Services Code Point (DSCP)	755

Supported IP Protocols

For more information on the listed IP protocols, see RFC1340 or www.iana.org.

Protocol name	Number	Description
ah	51	Authentication Header (AH)
egp	8	Exterior Gateway Protocol (EGP)
esp	50	Encapsulating Security Payload (ESP)
ggp	3	Gateway Gateway Protocol (GGP)
gre	47	General Routing Encapsulation (GRE)
hmp	20	Host Monitoring Protocol (HMP)
icmp	1	Internet Control Message Protocol (ICMP)
igmp	2	Internet Group Management Protocol (IGMP)
pup	12	PUP Protocol
rdp	27	Reliable Data Protocol (RDP)
rsvp	46	Resource Reservation Protocol (RSVP)
tcp	6	Transmission Control Protocol (TCP)
udp	17	User Datagram Protocol (UDP)
vines	83	Vines
xns-idp	22	Xerox NS IDP
6to4		

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
nntp	119	Y	Y	Network News Transfer Protocol (NNTP)
ntp	123	Y	Y	Network Time Protocol (NTP)

Port name	Number	TCP	UDP	Description
sntp	123	Y	Y	Simple Network Time Protocol (SNTP)
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	Internet Message 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	Novell
imap3	220	Y	Y	Internet Message Access Protocol (IMAP) v3
clearcase	371	Y	Y	ClearCase
ulistserv	372	Y	Y	UNIX Listserv
ldap	389	Y	Y	Lightweight 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
biff	512	-	Y	Used by mail system to notify users of new mail received
exec	512	Y	-	Remote process execution
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
syslog	514	-	Y	Syslog
printer	515	Y	Y	Spooler
talk	517	Y	Y	Like Tenex link, but across machine
ntalk	518	Y	Y	NTalk

Port name	Number	TCP	UDP	Description
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
new-rwho	540	Y	Y	uucpd remote who is
uucp	540	Y	Y	uucpd
uucp-rlogin	540	Y	Y	uucpd remote login
rtsp	554	Y	Y	Real Time Streaming Protocol (RTSP)
whoami	565	Y	Y	whoami
ipcserver	600	Y	Y	SUN IPC Server
doom	666	Y	Y	Doom ID Software
ils	1002	Y	Y	Internet Locator Service (ILS)
h323	1720	Y	Y	H323 Host Call Secure
nfsd	2049	Y	Y	NFS deamon
sip	5060	Y	Y	Session Initiation Protocol (SIP)
xwindows	6000	Y	Y	X windows
irc-u	6667	Y	Y	Internet Relay Chat (IRC) Protocol
realaudio	7070	Y	Y	realaudio
httpproxy	8080	Y	Y	HTTP Proxy

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 Facilities

For more information on the listed 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 Severities

For more information on the listed 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

IP Precedence

Precedence	Number
Routine	0
Priority	1
Immediate	2
Flash	3
Flash-Override	4
CRITIC-ECP	5
Internetwork-Control	6
Network-Control	7

Differentiated Services Code Point (DSCP)

For more information on DSCP, see RFC3260.

PHB	PHB Name	Binary value
ef	Expedited Forwarding	101110
af41	Assured Forwarding 41	100110
af42	Assured Forwarding 42	100100
af43	Assured Forwarding 43	100010
af31	Assured Forwarding 31	011110
af32	Assured Forwarding 32	011100
af33	Assured Forwarding 33	011010
af21	Assured Forwarding 21	010110
af22	Assured Forwarding 22	010110
af23	Assured Forwarding 23	010010
af11	Assured Forwarding 11	001110
af12	Assured Forwarding 12	001100
af13	Assured Forwarding 13	001010
cs7	Class Selector 7	111000
cs6	Class Selector 6	110000
cs5	Class Selector 5	101000
cs4	Class Selector 4	100000
cs3	Class Selector 3	011000
cs2	Class Selector 2	010000
cs1	Class Selector 1	001000
cs0	Class Selector 0	000000

Supported Key Names

756

speedtouch™

E-DOC-CTC-20060414-0005 v1.0

Index

A

aal5stats
 atm debug aal5stats 35
 actlist
 eth bridge dynvlan actlist 246
 add
 atm bundle add 21
 atm oam vclb add 46
 atm phonebook add 49
 atm qosbook add 54
 dhcp client roptions add 118
 dhcp client txoptions add 122
 dhcp relay add 126
 dhcp rule add 139
 dhcp server lease add 151
 dhcp server pool add 163
 dns server host add 191
 dns server route add 195
 dsd urlfilter rule add 210
 dyndns add 216
 dyndns host add 222
 eth bridge dynvlan add 247
 eth bridge rule add 266
 eth switch share add 282
 eth vlan add 285
 expr add 290
 firewall chain add 310
 firewall level add 317
 firewall rule add 323
 hostmgr add 344
 label add 442
 label chain add 448
 label rule add 452
 mlp privilege add 495
 mlp role add 503
 script add 558
 service host add 566
 service host rule add 573
 snmp community add 590
 ntp add 602
 upgrade profile add 634
 user add 644
 voice phonebook add 675
 wireless macacl add 707
 wireless wds add 727
 addpriv
 mlp role addpriv 504
 addzone
 mlp privilege addzone 497

 adsl
 config 8
 info 9
 appconfig
 connection appconfig 76
 appinfo
 connection appinfo 78
 applist
 connection applist 79
 arpadd
 ip arpadd 389
 arpdelete
 ip arpdelete 390
 arplist
 ip arplist 391
 assign
 service host assign 564
 atm
 bundle
 add 21
 attach 22
 clear 24
 config 23
 delete 25
 detach 26
 flush 27
 ifadd 28
 ifconfig 29
 ifdelete 30
 list 31
 cac
 config 32
 list 33
 overbooking 34
 debug
 aal5stats 35
 gstats 36
 portstats 37
 flush 13
 ifadd 14
 ifattach 15
 ifconfig 16
 ifdelete 18
 ifdetach 19
 iflist 20
 oam
 cc
 list 43

Index

modify 44
 send 45
 config 38
 list 39
 modify 40
 ping 42
 vclb
 add 46
 del 47
 list 48
 phonebook
 add 49
 autolist 50
 delete 51
 flush 52
 list 53
 qosbook
 add 54
 config 55
 ctdadd 56
 ctddelete 58
 ctdlist 59
 delete 60
 flush 61
 list 62
attach
 atm bundle attach 22
autolist
 atm phonebook autolist 50
autopvc
 config 64
 info 65
autosave
 system debug autosave 620

B

bind
 connection bind 80
bindlist
 connection bindlist 81

C

capture
 eth switch mirror capture 279
clean
 connection clean 82

clear
 atm bundle clear 24
 connection clear 83
 dhcp client debug clear 115
 dhcp server debug clear 147
 dns server debug clear 184
 dns server debug spoof clear 186
 eth bridge clear 244
 firewall debug clear 314
 firewall rule debug clear 329
 hostmgr clear 346
 ids clear 352
 ids pattern clear 356
 ids threshold clear 361
 igmp host debug clear 371
 igmp proxy debug clear 380
 ipqos queue clear 436
 label rule debug clear 461

config
 adsl config 8
 atm bundle config 23
 atm cac config 32
 atm oam config 38
 atm qosbook config 55
 autopvc config 64
 connection config 84
 cwmp config 96
 cwmp server config 99
 delete 68
 dhcp relay config 129
 dhcp server config 144
 dhcp server pool config 164
 dns client config 176
 dns server config 182
 dsd config 200
 dsd intercept config 206
 dsd syslog config 207
 dsd urlfilter config 209
 dump 69
 eth bridge config 245
 eth bridge dynvlan config 248
 firewall config 306
 flush 70
 grp config 334
 grp rip config 337
 hostmgr config 347
 ids config 353
 ids parser config 355
 igmp host config 366
 igmp proxy config 373
 ip config 392
 ipqos config 430
 ipqos ef config 433
 ipqos queue config 437
 language config 466
 list 71
 load 72
 mbus client config 471

mlp privilege config 498
 mlp role config 505
 nat config 510
 save 73
 service host config 565
 snmp config 586
 sntp config 603
 system config 612
 system ra config 622
 upgrade
 config 630
 upnp config 640
 user config 646
 voice codec config 659
 voice country config 660
 wireless macacl config 708
 wireless secmode config 722
 wireless wds config 728

connection

appconfig 76
 appinfo 78
 applist 79
 bind 80
 bindlist 81
 clean 82
 clear 83
 config 84
 debug 85
 describe 86
 flush 87
 info 88
 list 89
 refresh 90
 stats 91
 timerclear 92
 timerconfig 93
 unbind 94

ctdadd

atm qosbook ctdadd 56

ctddelete

atm qosbook ctddelete 58

ctdlist

atm qosbook ctdlist 59

cwmp

config 96
 debug
 traceconfig 98
 server
 config 99

D

debug

 connection debug 85
 exec 102

del

 atm oam vclb del 47

delete

 atm bundle delete 25
 atm phonebook delete 51
 atm qosbook delete 60
 config delete 68
 dhcp client roptions delete 119
 dhcp client txoptions delete 123
 dhcp relay delete 130
 dhcp rule delete 141
 dhcp server lease delete 153
 dhcp server pool delete 166
 dns server host delete 192
 dns server route delete 196
 dsd urlfilter rule delete 211
 dyndns delete 217
 dyndns host delete 223
 eth bridge dynvlan delete 249
 eth bridge rule delete 267
 eth switch share delete 283
 eth vlan delete 286
 expr delete 293
 firewall chain delete 311
 firewall level delete 318
 firewall rule delete 324
 hostmgr delete 348
 label chain delete 449
 label delete 443
 label rule delete 455
 language delete 467
 mlp privilege delete 499
 mlp role delete 506
 script delete 559
 service host delete 567
 service host rule delete 574
 snmp community delete 591
 sntp delete 604
 upgrade profile delete 636
 user delete 648
 voice phonebook delete 676
 wireless macacl delete 709
 wireless wds delete 729

describe

 connection describe 86

detach

 atm bundle detach 26

dhcp

 client
 debug

```

clear 115
stats 116
traceconfig 117
flush 105
ifadd 106
ifattach 107
ifconfig 108
ifdelete 109
ifdetach 110
iflist 111
ifrenew 113
rqoptions
    add 118
    delete 119
    list 120
    optionlist 121
txoptions
    add 122
    delete 123
    list 124
relay
    add 126
    config 129
    debug
        stats 127
        traceconfig 128
    delete 130
    flush 131
    ifconfig 132
    iflist 134
    list 135
    modify 136
    ruleadd 137
    ruledelete 138
rule
    add 139
    debug
        traceconfig 140
    delete 141
    flush 142
    list 143
server
    config 144
    debug
        clear 147
        stats 148
        traceconfig 150
flush 145
lease
    add 151
    delete 153
    flush 154
    list 155
option
    flush 156
    instadd 157
    instdelete 158
    instlist 159
    tmpladd 160
tmpdelete 161
tmpplist 162
policy 146
pool
    add 163
    config 164
    delete 166
    flush 167
    list 168
    optadd 169
    optdelete 170
    rtadd 171
    rtdelete 172
    ruleadd 173
    ruledelete 174
disable
    service host disable 568
dns
    client
        config 176
        dnsadd 177
        dnsdelete 178
        dnslist 179
        flush 180
        nslookup 181
    server
        config 182
        debug
            clear 184
            spoof
                clear 186
                getaddress 187
                getflags 188
                list 189
                update 190
            stats 185
        flush 183
    host
        add 191
        delete 192
        flush 193
        list 194
route
    add 195
    delete 196
    flush 197
    list 198
dnsadd
    dns client dnsadd 177
dnsdelete
    dns client dnsdelete 178
dnslist
    dns client dnslist 179
dsd
    config 200

```

```

debug                                         config 248
    connection                                delete 249
        list 202                               flush 250
    proxy 203                                list 251
    recycling 204                            flush 252
    stats 205                                ifadd 253
intercept                                 ifattach 254
    config 206                                ifconfig 255
syslog                                    ifdelete 257
    config 207                                ifdetach 258
        list 208                               iflist 259
urlfilter                                 macadd 260
    config 209                                macdelete 261
        rule                                     maclist 262
            add 210                                mcdadd 263
            delete 211                             mcddelete 264
            flush 212                             mcdlist 265
            list 213                                rule
            modify 214                           add 266
dump                                         delete 267
    config dump 69                           flush 268
dyndns                                    list 269
    add 216
    delete 217
    flush 218
    host
        add 222
        delete 223
        flush 224
        list 225
    list 219
    modify 220
    service
        list 226
        modify 227

E

egress
    eth switch mirror egress 280

env
    flush 230
    get 231
    list 232
    set 233
    unset 234

eth
    bridge
        clear 244
        config 245
    dynvlan
        actlist 246
        add 247

                                         config 248
                                         delete 249
                                         flush 250
                                         list 251
                                         flush 252
                                         ifadd 253
                                         ifattach 254
                                         ifconfig 255
                                         ifdelete 257
                                         ifdetach 258
                                         iflist 259
                                         macadd 260
                                         macdelete 261
                                         maclist 262
                                         mcdadd 263
                                         mcddelete 264
                                         mcdlist 265
                                         rule
                                             add 266
                                             delete 267
                                             flush 268
                                             list 269
                                         vlan
                                             ifadd 270
                                             ifconfig 271
                                             ifdelete 272
                                             iflist 273
                                         device
                                             ifconfig 274
                                             iflist 275
                                             flush 243
                                             ifadd 237
                                             ifattach 238
                                             ifconfig 239
                                             ifdelete 240
                                             ifdetach 241
                                             iflist 242
                                         switch
                                             group
                                                 flush 276
                                                 list 277
                                                 move 278
                                         mirror
                                             capture 279
                                             egress 280
                                             ingress 281
                                         share
                                             add 282
                                             delete 283
                                             list 284
                                         vlan
                                             add 285
                                             delete 286
                                             flush 287
                                             list 288
                                         exec
                                             debug exec 102
                                             mbus client exec 472

```

Index

export
 mlp debug export 492

expr
 add 290
 delete 293
 flush 295
 list 296
 modify 302

F

firewall

chain
 add 310
 delete 311
 flush 312
 list 313
 config 306
 debug
 clear 314
 stats 315
 traceconfig 316

flush 308

level
 add 317
 delete 318
 flush 319
 list 320
 modify 321
 set 322

list 309

rule
 add 323
 debug
 clear 329
 stats 331
 traceconfig 332
 delete 324
 flush 325
 list 326
 modify 328

flush

atm bundle flush 27
 atm flush 13
 atm phonebook flush 52
 atm qosbook flush 61
 config flush 70
 connection flush 87
 dhcp client flush 105
 dhcp relay flush 131
 dhcp rule flush 142
 dhcp server flush 145
 dhcp server lease flush 154
 dhcp server option flush 156

dhcp server pool flush 167
 dns client flush 180
 dns server flush 183
 dns server host flush 193
 dns server route flush 197
 dsd urlfilter rule flush 212
 dyndns flush 218
 dyndns host flush 224
 env flush 230
 eth bridge dynvlan flush 250
 eth bridge flush 252
 eth bridge rule flush 268
 eth flush 243
 eth switch group flush 276
 eth vlan flush 287
 expr flush 295
 firewall chain flush 312
 firewall flush 308
 firewall level flush 319
 firewall rule flush 325
 grp flush 335
 grp rip flush 339
 hostmgr flush 349
 igmp host flush 367
 igmp proxy flush 375
 ip auto flush 414
 ip flush 394
 ip mcast flush 410
 label chain flush 450
 label flush 444
 label rule flush 457
 mlp flush 490
 nat flush 511
 ppp flush 526
 ppp relay flush 543
 pptp flush 551
 script flush 560
 service host flush 569
 sntp flush 605
 system flush 614
 systemlog flush 626
 upnp flush 641
 user flush 649
 wireless macacl flush 710
 wireless wds flush 730

G

get

env get 231
 snmp get 587
 snmp getnext 588

getaddress

dns server debug spoof getaddress 187

getflags
 dns server debug spoof getflags 188
 grouplist
 igmp proxy grouplist 376
 grp
 config 334
 flush 335
 rip
 config 337
 flush 339
 ifconfig 340
 show 341
 rtlist 336
 gstats
 atm debug gstats 36

H

hostmgr
 add 344
 clear 346
 config 347
 delete 348
 flush 349
 list 350
 httpprobe
 ip debug httpprobe 422

I

ids
 clear 352
 config 353
 list 354
 parser
 config 355
 pattern
 clear 356
 list 357
 stats 358
 signature
 list 359
 modify 360
 threshold
 clear 361
 list 362
 modify 363

ifadd
 atm bundle ifadd 28

atm ifadd 14
 dhcp client ifadd 106
 eth bridge ifadd 253
 eth bridge vlan ifadd 270
 eth ifadd 237
 ip auto ifadd 415
 ip ifadd 395
 ppp ifadd 527
 ppp relay ifadd 544
 pptp ifadd 550
 service system ifadd 575
 snmp ifadd 594

ifattach
 atm ifattach 15
 dhcp client ifattach 107
 eth bridge ifattach 254
 eth ifattach 238
 ip auto ifattach 416
 ip ifattach 396
 ppp ifattach 528
 snmp ifattach 596

ifconfig
 atm bundle ifconfig 29
 atm ifconfig 16
 dhcp client ifconfig 108
 dhcp relay ifconfig 132
 eth bridge ifconfig 255
 eth bridge vlan ifconfig 271
 eth device ifconfig 274
 eth ifconfig 239
 grp rip ifconfig 340
 igmp host ifconfig 369
 igmp proxy ifconfig 377
 ip auto ifconfig 417
 ip ifconfig 397
 nat ifconfig 512
 ppp ifconfig 530
 ppp relay ifconfig 545
 snmp ifconfig 598
 voice ifconfig 655
 wireless ifconfig 702

ifdelete
 atm bundle ifdelete 30
 atm ifdelete 18
 dhcp client ifdelete 109
 eth bridge ifdelete 257
 eth bridge vlan ifdelete 272
 eth ifdelete 240
 ip auto ifdelete 419
 ip ifdelete 399
 ppp ifdelete 534
 ppp relay ifdelete 546
 service system ifdelete 576
 snmp ifdelete 595

ifdetach
 atm ifdetach 19

dhcp client ifdetach 110
 eth bridge ifdetach 258
 eth ifdetach 241
 ip auto ifdetach 420
 ip ifdetach 400
 ppp ifdetach 535
 snmp ifdetach 597

iflist
 atm iflist 20
 dhcp client iflist 111
 dhcp relay iflist 134
 eth bridge iflist 259
 eth bridge vlan iflist 273
 eth device iflist 275
 eth iflist 242
 igmp host iflist 370
 igmp proxy iflist 378
 ip auto iflist 421
 ip iflist 401
 nat iflist 513
 ppp iflist 537
 ppp relay iflist 547
 snmp iflist 599

ifrenew
 dhcp client ifrenew 113

ifscan
 ppp ifscan 538

ifwait
 ip ifwait 402

igmp
 host
 config 366
 debug
 clear 371
 stats 372
 flush 367
 ifconfig 369
 iflist 370
 list 368
 proxy
 config 373
 debug
 clear 380
 flush 375
 grouplist 376
 ifconfig 377
 iflist 378
 mbslist 379

import
 mlp import 491

info
 adsl info 9
 autopvc info 65
 connection info 88

ingress
 eth switch mirror ingress 281

instadd
 dhcp server option instadd 157

instdelete
 dhcp server option instdelete 158

instlist
 dhcp server option instlist 159

interface
 list 384

ip
 arpadd 389
 arpdelete 390
 arpplist 391
 auto
 flush 414
 ifadd 415
 ifattach 416
 ifconfig 417
 ifdelete 419
 ifdetach 420
 iflist 421
 config 392
 debug
 httpprobe 422
 sendto 423
 stats 425
 traceconfig 426
 flush 394
 ifadd 395
 ifattach 396
 ifconfig 397
 ifdelete 399
 ifdetach 400
 iflist 401
 ifwait 402
 ipadd 403
 ipconfig 404
 ipdelete 405
 iplist 406
 mcast
 flush 410
 rtadd 407
 rtdelete 408
 rtlist 409
 rtadd 411
 rtdelete 412
 rtlist 413

ipadd
 ip ipadd 403
 service system ipadd 577

ipconfig
 ip ipconfig 404

ipdelete
 ip ipdelete 405
 service system ipdelete 578

iplist
 ip iplist 406

ipqos
 config 430
 ef
 config 433
 list 434
 stats 435
 list 432
 queue
 clear 436
 config 437
 list 439
 stats 440

modify 460
 language
 config 466
 delete 467
 list 468
 list
 atm bundle list 31
 atm cac list 33
 atm oam cc config 43
 atm oam list 39
 atm oam vclb list 48
 atm phonebook list 53
 atm qosbook list 62
 config list 71
 connection list 89
 dhcp client roptions list 120
 dhcp client txoptions list 124
 dhcp relay list 135
 dhcp rule list 143
 dhcp server lease list 155
 dhcp server pool list 168
 dns server debug spoof list 189
 dns server host list 194
 dns server route list 198
 dsd syslog list 208
 dsd urlfilter rule list 213
 dyndns host list 225
 dyndns list 219
 dyndns service list 226
 env list 232
 eth bridge dynvlan list 251
 eth bridge rule list 269
 eth switch group list 277
 eth switch share list 284
 eth vlan list 288
 expr list 296
 firewall chain list 313
 firewall level list 320
 firewall list 309
 firewall rule list 326
 hostmgr list 350
 ids list 354
 ids pattern list 357
 ids signature list 359
 ids threshold list 362
 igmp host list 368
 interface list 384
 ipqos ef list 434
 ipqos list 432
 ipqos queue list 439
 label chain list 451
 label list 445
 label rule list 458
 language list 468
 mlp privilege list 500
 mlp role list 507
 pptp list 552

K

Key names

DSCP 755
 ICMP type 752
 IP precedence 754
 IP protocols 748
 Syslog facilities 753
 Syslog severities 754
 TCP/UDP port 749

L

label

add 442
 chain
 add 448
 delete 449
 flush 450
 list 451
 delete 443
 flush 444
 list 445
 modify 446
 rule
 add 452
 debug
 clear 461
 stats 462
 traceconfig 463
 delete 455
 flush 457
 list 458

script list 561
 service host list 570
 service system list 579
 snmp community list 592
 sntp list 606
 upnp list 642
 user list 650
 voice codec list 658
 voice country list 661
 voice phonebook list 677
 voice stats list 699
 wireless macacl list 711
 wireless stations list 726
 wireless wds list 731
list dsd debug connection list 202
listobjects
 mbus listobjects 478
 memm listobjects 486
listtypes
 mbus listtypes 479
load
 config load 72
loadobjects
 mbus loadobjects 480
locale
 system locale 615

M

macadd
 eth bridge macadd 260
macdelete
 eth bridge macdelete 261
maclist
 eth bridge maclist 262
mapadd
 nat mapadd 514
 service system mapadd 581
mapdelete
 nat mapdelete 517
 service system mapdelete 582
maplist
 nat maplist 518
mbslist
 igmp proxy mbslist 379
mbus
 client

config 471
 exec 472
 register 474
 test 475
debug
 stats 476
 traceconfig 477
listobjects 478
listtypes 479
loadobjects 480
unloadobjects 481
mcdadd
 eth bridge mcdadd 263
mcddelete
 eth bridge mcddelete 264
mcclist
 eth bridge mcclist 265
memm
 debug
 lock
 traceconfig 484
 traceconfig 485
listobjects 486
 stats 487
mlp
 debug
 export 492
 stats 493
 traceconfig 494
 flush 490
 import 491
privilege
 add 495
 addzone 497
 config 498
 delete 499
 list 500
 removezone 502
role
 add 503
 addpriv 504
 config 505
 delete 506
 list 507
 removepriv 508
modify
 atm oam cc modify 44
 atm oam modify 40
 dhcp relay modify 136
 dsd urlfilter rule modify 214
 dyndns modify 220
 dyndns service modify 227
 expr modify 302
 firewall level modify 321
 firewall rule modify 328

ids signature modify 360
 ids threshold modify 363
 label modify 446
 label rule modify 460
 service system modify 583
 snmp community modify 593
 upgrade profile modify 635
 wireless macacl modify 712
move
 eth switch group move 278

N

nat
 config 510
 flush 511
 ifconfig 512
 iflist 513
 mapadd 514
 mapdelete 517
 maplist 518
 tmpladd 519
 tmpldelete 522
 tmplinst 524
 tmpllist 523
nslookup
 dns client nslookup 181

O

optadd
 dhcp server pool optadd 169
optdelete
 dhcp server pool optdelete 170
optionlist
 dhcp client roptions optionlist 121
overbooking
 atm cac overbooking 34

P

ping
 atm oam ping 42

ping root command 4
policy
 dhcp server policy 146
portstats
 atm debug portstats 37
ppp
 flush 526
 ifadd 527
 ifattach 528
 ifconfig 530
 ifdelete 534
 ifdetach 535
 iflist 537
 ifscan 538
relay
 flush 543
 ifadd 544
 ifconfig 545
 ifdelete 546
 iflist 547
 sesslist 548
 rtadd 539
 rtdelte 541

pptp

flush 551
 ifadd 550
 list 552
 profadd 553
 profdelete 554
 proflist 555

profadd

pptp profadd 553

profdelete

pptp profdelete 554

proflist

pptp proflist 555

proxy

dsd debug proxy 203

R

reboot
 system reboot 617
recycling
 dsd debug recycling 204
refresh
 connection refresh 90

register
mbus client register 474
wireless macacl register 713

removepriv
mlp role removepriv 508

removezone
mlp privilege removezone 502

reset
system reset 618
voice stats reset 700
wireless reset 704

rights
user rights 652

root commands
ping 4
traceroute 5

rtadd
dhcp server pool rtadd 171
ip mcast rtadd 407
ip rtadd 411
ppp rtadd 539

rtdelete
dhcp server pool rtdelete 172
ip mcast rtdelete 408
ip rtdelete 412
ppp rtdelete 541

rtlist
grp rtlist 336
ip mcast rtlist 409
ip rtlist 413

ruleadd
dhcp relay ruleadd 137
dhcp server pool ruleadd 173

ruledelete
dhcp relay ruledelete 138
dhcp server pool ruledelete 174

run
script run 562

S

save
config save 73

scanresults
wireless wds scanresults 732

script
add 558
delete 559
flush 560
list 561
run 562

send
atm oam cc send 45
systemlog send 628

sendto
ip debug sendto 423

service
host
add 566
assign 564
config 565
delete 567
disable 568
flush 569
list 570
rule
add 573
delete 574
stats 571
triggerlist 572

system
ifadd 575
ifdelete 576
ipadd 577
ipdelete 578
list 579
mapadd 581
mapdelete 582
modify 583

sesslist
ppp relay sesslist 548
upgrade debug sesslist 633

set
env set 233
firewall level set 322

setttime
system rtc setttime 624

show
grp rip show 341
systemlog show 627

snmp
community
add 590
delete 591
list 592
modify 593
config 586
get 587

getnext 588
 ifadd 594
 ifattach 596
 ifconfig 598
 ifdelete 595
 ifdetach 597
 iflist 599
 walk 589

sntp
 add 602
 config 603
 delete 604
 flush 605
 list 606

software
 upgrade 609
 version 610

start
 upgrade start 631

state
 voice state 657

stats
 connection stats 91
 dhcp client debug stats 116
 dhcp relay debug stats 127
 dhcp server debug stats 148
 dns server debug stats 185
 dsd debug stats 205
 firewall debug stats 315
 firewall rule debug stats 331
 ids pattern stats 358
 igmp host debug stats 372
 ip debug stats 425
 ipqos ef stats 435
 ipqos queue stats 440
 label rule debug stats 462
 mbus debug stats 476
 memm stats 487
 mlp debug stats 493
 service host stats 571
 system debug stats 621
 wireless debug stats 706

synchronize
 system rtc synchronize 623

Syslog messages
 Auto-PVC 738
 Configuration 738
 DHCP client 738
 DHCP relay 739
 DHCP server 739
 Dyndns 740
 Firewall 740
 Kernel 741
 Linestate 741

LOGIN 741
 NAPT 741
 PPP 742
 PPTP 742
 RIP 743
 Routing 744
 Session 744
 SNTP 744
 Software 745
 UPnP 745
 VOIP 745

system
 config 612
 debug
 autosave 620
 stats 621
 flush 614
 locale 615
 ra
 config 622
 reboot 617
 reset 618
 rtc
 settime 624
 synchronize 623

systemlog
 flush 626
 send 628
 show 627

T

test
 mbus client test 475

timerclear
 connection timerclear 92

timerconfig
 connection timerconfig 93

tmpladd
 dhcp server option tmpladd 160
 nat tmpladd 519

tmpldelete
 dhcp server option tmpldelete 161
 nat tmpldelete 522

tmplinst
 nat tmplinst 524

tmpllist
 dhcp server option tmpllist 162
 nat tmpllist 523

traceconfig

- cwmp debug traceconfig 98
- dhcp client debug traceconfig 117
- dhcp relay debug traceconfig 128
- dhcp rule debug traceconfig 140
- dhcp server debug traceconfig 150
- firewall debug traceconfig 316
- firewall rule debug traceconfig 332
- ip debug traceconfig 426
- label rule debug traceconfig 463
- mbus debug traceconfig 477
- memm debug lock traceconfig 484
- memm debug traceconfig 485
- mlp debug traceconfig 494
- upgrade debug traceconfig 632

traceroute

- traceroute root command 5

triggerlist

- service host triggerlist 572

U
unbind

- connection unbind 94

unloadobjects

- mbus unloadobjects 481

unset

- env unset 234

update

- dns server debug spoof update 190

upgrade

- config 630
- debug
 - sesslist 633
 - traceconfig 632
- profile
 - add 634
 - delete 636
 - modify 635
- software upgrade 609
- start 631

upnp

- config 640
- flush 641
- list 642

user

- add 644
- config 646
- delete 648
- flush 649

- list 650
- rights 652

V
version

- software version 610

voice

- codec
 - config 659
 - list 658
- country
 - config 660
 - list 661
- ifconfig 655
- phonebook
 - add 675
 - delete 676
 - list 677
- state 657
- stats
 - list 699
 - reset 700

W
walk

- snmp walk 589

wep

- wireless seemode wep 723

wireless

- debug
 - stats 706
- ifconfig 702
- macacl
 - add 707
 - config 708
 - delete 709
 - flush 710
 - list 711
 - modify 712
 - register 713
- reset 704
- seemode
 - config 722
 - wep 723
 - wpa-psk 724
- stations
 - list 726
- wds
 - add 727

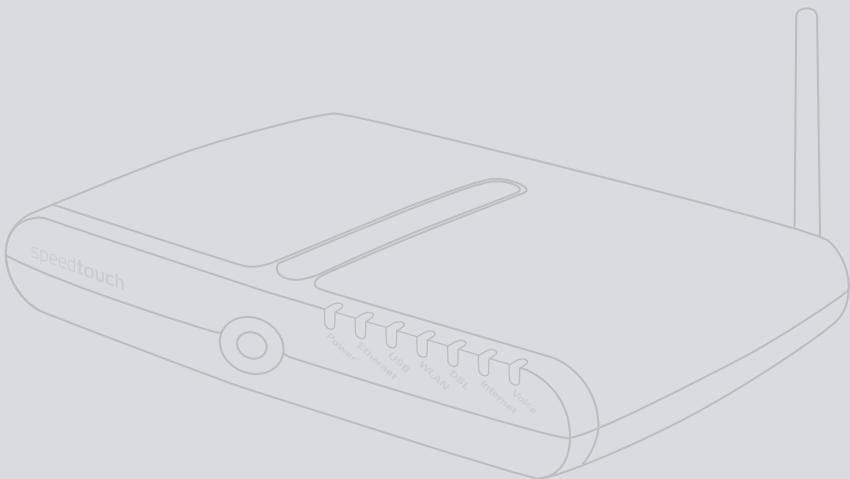
config 728
delete 729
flush 730
list 731
scanresults 732

wpa-psk
 wireless secmode wpa-psk 724

772

speedtouch™

E-DOC-CTC-20060414-0005 v1.0



Need more help?

Additional help is available online at www.speedtouch.com