

Mediant™ 4000 SBC

Enterprise Session Border Controller

High-Availability System

User's Manual



Version 6.6

March 2014

Document # LTRT-41536



Table of Contents

1	Overview	15
<hr/>		
	Getting Started with Initial Connectivity.....	17
2	Assigning the OAMP IP Address	19
2.1	Web Interface.....	19
2.2	CLI.....	21
<hr/>		
	Management Tools	23
3	Introduction	25
4	Web-Based Management.....	27
4.1	Getting Acquainted with the Web Interface.....	27
4.1.1	Computer Requirements.....	27
4.1.2	Accessing the Web Interface.....	28
4.1.3	Areas of the GUI	29
4.1.4	Toolbar Description.....	30
4.1.5	Navigation Tree	31
4.1.5.1	Displaying Navigation Tree in Basic and Full View.....	31
4.1.5.2	Showing / Hiding the Navigation Pane.....	32
4.1.6	Working with Configuration Pages	33
4.1.6.1	Accessing Pages.....	33
4.1.6.2	Viewing Parameters	34
4.1.6.3	Modifying and Saving Parameters	35
4.1.6.4	Working with Tables.....	36
4.1.7	Searching for Configuration Parameters	39
4.1.8	Working with Scenarios	40
4.1.8.1	Creating a Scenario.....	41
4.1.8.2	Accessing a Scenario.....	43
4.1.8.3	Editing a Scenario	44
4.1.8.4	Saving a Scenario to a PC	45
4.1.8.5	Loading a Scenario to the Device	45
4.1.8.6	Deleting a Scenario	46
4.1.8.7	Quitting Scenario Mode.....	46
4.1.9	Creating a Login Welcome Message.....	47
4.1.10	Getting Help.....	48
4.1.11	Logging Off the Web Interface.....	48
4.2	Viewing the Home Page.....	49
4.3	Configuring Web User Accounts	51
4.3.1	Basic User Accounts Configuration	53
4.3.2	Advanced User Accounts Configuration.....	54
4.4	Displaying Login Information upon Login	58
4.5	Configuring Web Security Settings	58
4.6	Web Login Authentication using Smart Cards	59
4.7	Configuring Web and Telnet Access List	60
4.8	Configuring RADIUS Settings	61
5	CLI-Based Management.....	63
5.1	Enabling CLI using Telnet.....	63

5.2	Enabling CLI using SSH and RSA Public Key	63
5.3	Establishing a CLI Session	65
5.4	CLI Commands	66
5.4.1	Status Commands	66
5.4.2	Ping Command	68
5.4.3	Management Commands	69
5.4.4	Configuration Commands	69
5.4.5	LDAP Commands	70
6	SNMP-Based Management	71
6.1	Configuring SNMP Community Strings	71
6.2	Configuring SNMP Trap Destinations	72
6.3	Configuring SNMP Trusted Managers	73
6.4	Configuring SNMP V3 Users.....	74
7	EMS-Based Management.....	77
8	INI File-Based Management.....	79
8.1	INI File Format	79
8.1.1	Configuring Individual ini File Parameters	79
8.1.2	Configuring Table ini File Parameters	79
8.1.3	General ini File Formatting Rules	81
8.2	Loading an ini File	81
8.3	Modifying an ini File	82
8.4	Secured Encoded ini File	82
General System Settings		83
9	Configuring Certificates	85
9.1	Replacing the Device's Certificate	85
9.2	Loading a Private Key	86
9.3	Mutual TLS Authentication	88
9.4	Configuring Certificate Revocation Checking (OCSP)	89
9.5	Self-Signed Certificates.....	89
9.6	Loading Certificate Chain for Trusted Root.....	90
10	Date and Time.....	91
10.1	Configuring Date and Time Manually.....	91
10.2	Automatic Date and Time through SNTP Server	91
General VoIP Configuration		93
11	Network.....	95
11.1	Configuring Physical Ethernet Ports	95
11.2	Configuring Tx/Rx for Ethernet Port-Pair Groups.....	97
11.3	Configuring IP Network Interfaces	98
11.3.1	Bootting using DHCP	102
11.3.2	Assigning NTP Services to Application Types	102
11.3.3	Multiple Interface Table Configuration Rules.....	102
11.3.4	Troubleshooting the Multiple Interface Table	104
11.3.5	Networking Configuration Examples	104

11.3.5.1	One VoIP Interface for All Applications	104
11.3.5.2	VoIP Interface per Application Type.....	105
11.3.5.3	VoIP Interfaces for Combined Application Types	106
11.3.5.4	VoIP Interfaces with Multiple Default Gateways	107
11.4	Configuring the IP Routing Table	108
11.4.1	Interface Column	110
11.4.2	Routing Table Configuration Summary and Guidelines	110
11.4.3	Troubleshooting the Routing Table	111
11.5	Configuring Quality of Service.....	111
11.6	Disabling ICMP Redirect Messages.....	114
11.7	DNS.....	114
11.7.1	Configuring the Internal DNS Table.....	114
11.7.2	Configuring the Internal SRV Table.....	116
11.8	Configuring NFS Settings.....	117
11.9	Network Address Translation Support	118
11.9.1	Device Located behind NAT	119
11.9.1.1	Configuring a Static NAT IP Address for All Interfaces	120
11.9.1.2	Configuring NAT Translation per IP Interface	120
11.9.2	Remote UA behind NAT	122
11.9.2.1	First Incoming Packet Mechanism	122
11.9.2.2	No-Op Packets	123
11.10	Robust Receipt of Media Streams	124
11.11	Multiple Routers Support.....	124
12	Security	125
12.1	Configuring Firewall Settings	125
12.2	Configuring General Security Settings	129
12.3	Intrusion Detection System	130
12.3.1	Enabling IDS.....	130
12.3.2	Configuring IDS Policies	131
12.3.3	Assigning IDS Policies.....	134
12.3.4	Viewing IDS Alarms	136
13	Media	139
13.1	Configuring Voice Settings.....	139
13.1.1	Configuring Voice Gain (Volume) Control	139
13.1.2	Silence Suppression (Compression)	139
13.1.3	Echo Cancellation.....	140
13.2	Fax and Modem Capabilities.....	141
13.2.1	Fax/Modem Operating Modes	142
13.2.2	Fax/Modem Transport Modes	143
13.2.2.1	T.38 Fax Relay Mode	143
13.2.2.2	G.711 Fax / Modem Transport Mode	144
13.2.2.3	Fax Fallback	145
13.2.2.4	Fax/Modem Bypass Mode	145
13.2.2.5	Fax / Modem NSE Mode	147
13.2.2.6	Fax / Modem Transparent with Events Mode	148
13.2.2.7	Fax / Modem Transparent Mode.....	148
13.2.2.8	RFC 2833 ANS Report upon Fax/Modem Detection	149
13.2.3	V.34 Fax Support.....	149
13.2.3.1	Bypass Mechanism for V.34 Fax Transmission.....	150
13.2.3.2	Relay Mode for T.30 and V.34 Faxes	150
13.2.4	V.152 Support.....	151
13.3	Configuring RTP/RTCP Settings.....	152

13.3.1	Configuring the Dynamic Jitter Buffer	152
13.3.2	Comfort Noise Generation	153
13.3.3	Dual-Tone Multi-Frequency Signaling	154
13.3.3.1	Configuring DTMF Transport Types.....	154
13.3.3.2	Configuring RFC 2833 Payload	155
13.3.4	Configuring RTP Base UDP Port.....	156
13.4	Configuring Various Codec Attributes	157
13.5	Configuring Media Realms.....	157
13.5.1	Configuring Quality of Experience per Media Realm	159
13.5.2	Configuring Bandwidth Management per Media Realm.....	162
13.6	Configuring Server for Media Quality of Experience	164
13.7	Configuring Media Security	165
14	Services	167
14.1	Routing Based on LDAP Active Directory Queries	167
14.1.1	Configuring the LDAP Server	167
14.1.2	Configuring the Device's LDAP Cache.....	168
14.1.3	Active Directory based Tel-to-IP Routing for Microsoft Lync.....	170
14.1.3.1	Querying the AD and Routing Priority	170
14.1.3.2	Configuring AD-Based Routing Rules.....	174
14.2	Least Cost Routing.....	175
14.2.1	Overview	175
14.2.2	Configuring LCR	178
14.2.2.1	Enabling the LCR Feature.....	178
14.2.2.2	Configuring Cost Groups.....	179
14.2.2.3	Configuring Time Bands for Cost Groups	180
14.2.2.4	Assigning Cost Groups to Routing Rules.....	182
15	Enabling Applications.....	183
16	Control Network	185
16.1	Configuring SRD Table	185
16.2	Configuring SIP Interface Table	188
16.3	Configuring IP Groups.....	191
16.4	Configuring Proxy Sets Table	197
17	SIP Definitions	203
17.1	Configuring SIP Parameters	203
17.2	Configuring Account Table.....	203
17.3	Configuring Proxy and Registration Parameters.....	205
17.3.1	SIP Message Authentication Example	206
17.4	Configuring SIP Message Manipulation	208
17.5	Configuring SIP Message Policy Rules.....	212
17.6	Configuring RADIUS Accounting	215
18	Coders and Profiles	217
18.1	Configuring Coders	217
18.2	Configuring Coders Groups	220
18.3	Configuring IP Profiles	221
Session Border Controller Application.....		233

19 SBC Overview	235
19.1 SIP Network Definitions	236
19.2 SIP Dialog Initiation Process.....	236
19.3 User Registration and Internal Database	238
19.3.1 Initial Registration Request Processing.....	239
19.3.2 Internal Database	239
19.3.3 Routing using Internal Database	240
19.3.4 Registration Refreshes	240
19.3.5 Notification of Expired User Registration to SIP Proxy / Registrar	240
19.3.6 Registration Restriction Control.....	241
19.4 SBC Media Handling.....	241
19.4.1 Media Anchoring without Transcoding (Transparent)	243
19.4.2 Media Anchoring with Transcoding	243
19.4.3 No Media Anchoring	245
19.4.4 Transcoding Modes	247
19.4.5 Restricting Coders	248
19.4.6 Coder Transcoding	248
19.4.7 Prioritizing Coder List in SDP Offer	249
19.4.8 SRTP-RTP and SRTP-SRTP Transcoding	250
19.4.9 Multiple RTP Media Streams per Call Session	250
19.4.10 Interworking DTMF Methods	251
19.5 Fax Negotiation and Transcoding	251
19.6 Limiting SBC Call Duration.....	252
19.7 SIP Authentication Server for SBC Users	252
19.8 Interworking SIP Signaling.....	252
19.8.1 Interworking SIP 3xx Redirect Responses	253
19.8.1.1 Resultant INVITE Traversing Device	253
19.8.1.2 Local Handling of SIP 3xx	254
19.8.2 Interworking SIP Diversion and History-Info Headers	255
19.8.3 Interworking SIP REFER Messages.....	255
19.8.4 Interworking SIP PRACK Messages	256
19.8.5 Interworking SIP Session Timer	256
19.8.6 Interworking SIP Early Media	256
19.8.7 Interworking SIP re-INVITE Messages.....	258
19.8.8 Interworking SIP UPDATE Messages	258
19.8.9 Interworking SIP re-INVITE to UPDATE.....	259
19.8.10 Interworking Delayed Offer.....	259
19.8.11 Interworking Call Hold.....	259
19.9 Call Survivability	260
19.9.1 Auto-Provisioning of Subscriber-Specific Information for BroadWorks Server for Survivability.....	260
19.9.2 BroadSoft's Shared Phone Line Call Appearance for SBC Survivability.....	261
19.9.3 Call Survivability for Call Centers	262
19.9.4 Survivability Mode Display on Aastra IP Phones	264
19.10 Call Forking	265
19.10.1 Initiating SIP Call Forking	265
19.10.2 SIP Forking Initiated by SIP Proxy Server	265
19.11 Alternative Routing on Detection of Failed SIP Response	266
20 SBC Configuration	267
20.1 Configuring General Settings.....	267
20.2 Configuring Admission Control.....	268
20.3 Configuring Allowed Coder Groups.....	271

20.4	Routing SBC	272
20.4.1	Configuring Classification Rules	272
20.4.1.1	Classification Based on URI of Selected Header Example	277
20.4.2	Configuring Condition Rules	278
20.4.3	Configuring SBC IP-to-IP Routing	279
20.4.4	Configuring Alternative Routing Reasons	286
20.5	SBC Manipulations	287
20.5.1	Configuring IP-to-IP Inbound Manipulations	289
20.5.2	Configuring IP-to-IP Outbound Manipulations	292
Cloud Resilience Package Application		297
21	CRP Overview	299
22	CRP Configuration	301
22.1	Enabling the CRP Application	301
22.2	Configuring Call Survivability Mode	302
22.3	Pre-Configured IP Groups	303
22.4	Pre-Configured IP-to-IP Routing Rules	304
Stand-Alone Survivability Application		305
23	SAS Overview	307
23.1	SAS Operating Modes	307
23.1.1	SAS Outbound Mode	308
23.1.1.1	Normal State	308
23.1.1.2	Emergency State	308
23.1.2	SAS Redundant Mode	309
23.1.2.1	Normal State	309
23.1.2.2	Emergency State	310
23.1.2.3	Exiting Emergency and Returning to Normal State	310
23.2	SAS Routing	310
23.2.1	SAS Routing in Normal State	311
23.2.2	SAS Routing in Emergency State	313
24	SAS Configuration	315
24.1	General SAS Configuration	315
24.1.1	Enabling the SAS Application	315
24.1.2	Configuring Common SAS Parameters	315
24.2	Configuring SAS Outbound Mode	318
24.3	Configuring SAS Redundant Mode	318
24.4	Configuring Gateway Application with SAS	319
24.4.1	Gateway with SAS Outbound Mode	319
24.4.2	Gateway with SAS Redundant Mode	321
24.5	Advanced SAS Configuration	322
24.5.1	Manipulating URI user part of Incoming REGISTER	322
24.5.2	Manipulating Destination Number of Incoming INVITE	323
24.5.3	SAS Routing Based on IP-to-IP Routing Table	326
24.5.4	Blocking Calls from Unregistered SAS Users	331
24.5.5	Configuring SAS Emergency Calls	332
24.5.6	Adding SIP Record-Route Header to SIP INVITE	332
24.5.7	Re-using TCP Connections	333
24.5.8	Replacing Contact Header for SIP Messages	333

24.6 Viewing Registered SAS Users.....	334
25 SAS Cascading.....	335
High Availability System	337
26 HA Overview	339
26.1 Connectivity and Synchronization between Devices.....	339
26.2 Device Switchover upon Failure.....	340
26.3 HA Status	341
27 HA Configuration.....	343
27.1 Initial HA Configuration	343
27.1.1 Network Topology Types	343
27.1.1.1 Tx/Rx Port Settings	343
27.1.2 Configuring the Two HA Devices.....	345
27.1.2.1 Step 1: Configure the First Device	345
27.1.2.2 Step 2: Configure the Second Device	347
27.1.2.3 Step 3: Initialize HA on the Devices	348
27.2 Configuration while HA is Operational	348
27.3 Monitoring IP Entity and HA Switchover upon Ping Failure	349
27.4 Configuring Firewall Allowed Rules.....	351
28 HA Maintenance	353
28.1 Maintenance of Redundant Device	353
28.2 Replacing a Failed Device	353
28.3 Forcing a Switchover.....	353
28.4 Software Upgrade	353
Maintenance	355
29 Basic Maintenance.....	357
29.1 Resetting the Device	357
29.2 Remotely Resetting Device using SIP NOTIFY	358
29.3 Locking and Unlocking the Device	359
29.4 Saving Configuration.....	360
30 High Availability Maintenance.....	361
31 Software Upgrade.....	363
31.1 Loading Auxiliary Files	363
31.1.1 Call Progress Tones File	364
31.1.2 Dial Plan File.....	367
31.1.2.1 Creating a Dial Plan File.....	367
31.1.2.2 Obtaining IP Destination from Dial Plan File.....	367
31.1.3 User Information File	368
31.1.3.1 User Information File for SBC User Database	368
31.1.3.2 Enabling the User Info Table.....	369
31.2 Software License Key	370
31.2.1 Obtaining the Software License Key File.....	370
31.2.2 Installing the Software License Key.....	371

31.2.2.1	Installing Software License Key using Web Interface	371
31.3	Software Upgrade Wizard	373
31.4	Backing Up and Loading Configuration File	377
32	Automatic Update	379
32.1	Configuring Automatic Update	379
32.2	Automatic Configuration Methods	382
32.2.1	DHCP-based Configuration Server	382
32.2.2	HTTP-based Automatic Updates	382
32.2.3	Configuration using FTP or NFS	383
32.2.4	Configuration using AudioCodes EMS	383
32.3	Loading Files Securely (Disabling TFTP)	383
32.4	Remotely Triggering Auto Update using SIP NOTIFY	384
33	Restoring Factory Defaults	385
33.1	Restoring Defaults using CLI	385
33.2	Restoring Defaults using an ini File	386
Status, Performance Monitoring and Reporting		387
34	System Status	389
34.1	Viewing Device Information	389
34.2	Viewing Ethernet Port Information	390
35	Carrier-Grade Alarms	391
35.1	Viewing Active Alarms	391
35.2	Viewing Alarm History	392
36	Performance Monitoring	393
36.1	Viewing MOS per Media Realm	393
36.2	Viewing Quality of Experience	394
36.3	Viewing Average Call Duration	395
37	VoIP Status	397
37.1	Viewing Active IP Interfaces	397
37.2	Viewing Registered Users	397
38	Reporting Information to External Party	399
38.1	RTP Control Protocol Extended Reports (RTCP XR)	399
38.2	Generating Call Detail Records	402
38.2.1	Configuring CDR Reporting	402
38.2.2	CDR Field Description	403
38.2.2.1	CDR Fields for SBC Signaling	403
38.2.2.2	CDR Fields for SBC Media	405
38.2.3	Supported RADIUS Attributes	406
Diagnostics		409
39	Syslog and Debug Recordings	411
39.1	Syslog Message Format	411
39.1.1	Event Representation in Syslog Messages	412

39.1.2	Identifying AudioCodes Syslog Messages using Facility Levels	414
39.1.3	SNMP Alarms in Syslog Messages	414
39.2	Configuring Syslog Settings	415
39.3	Configuring Debug Recording	416
39.4	Filtering Syslog Messages and Debug Recordings	416
39.4.1	Filtering IP Network Traces	418
39.5	Viewing Syslog Messages	420
39.6	Collecting Debug Recording Messages	421
40	Testing SIP Signaling Calls	423
40.1	Configuring Test Call Endpoints.....	423
40.1.1	Starting, Stopping and Restarting Test Calls.....	426
40.1.2	Viewing Test Call Statistics.....	427
40.2	Configuring DTMF Tones for Test Calls.....	428
40.3	Configuring SBC Test Call with External Proxy	429
40.4	Test Call Configuration Examples.....	430
<hr/>		
Appendix		433
41	Dialing Plan Notation for Routing and Manipulation.....	435
42	Configuration Parameters Reference	437
42.1	Networking Parameters.....	437
42.1.1	Ethernet Parameters.....	437
42.1.2	Multiple VoIP Network Interfaces and VLAN Parameters	438
42.1.3	Routing Parameters.....	438
42.1.4	Quality of Service Parameters.....	439
42.1.5	NAT and STUN Parameters	440
42.1.6	NFS Parameters	441
42.1.7	DNS Parameters.....	441
42.1.8	DHCP Parameters	442
42.1.9	NTP and Daylight Saving Time Parameters.....	443
42.2	Management Parameters.....	444
42.2.1	General Parameters	444
42.2.2	Web Parameters	444
42.2.3	Telnet Parameters	447
42.2.4	SNMP Parameters.....	448
42.2.5	Serial Parameters	450
42.3	Debugging and Diagnostics Parameters.....	451
42.3.1	General Parameters	451
42.3.2	SIP Test Call Parameters	452
42.3.3	Syslog, CDR and Debug Parameters.....	453
42.3.4	HA Ping Parameters	456
42.3.5	Resource Allocation Indication Parameters.....	457
42.4	Security Parameters.....	458
42.4.1	General Parameters	458
42.4.2	HTTPS Parameters	458
42.4.3	SRTP Parameters.....	459
42.4.4	TLS Parameters.....	462
42.4.5	SSH Parameters.....	464
42.4.6	OCSP Parameters	465
42.4.7	IDS Parameters	465
42.5	RADIUS Parameters	467

42.6	SIP Media Realm Parameters.....	469
42.7	Control Network Parameters.....	471
42.7.1	IP Group, Proxy, Registration and Authentication Parameters	471
42.7.2	Network Application Parameters	478
42.8	General SIP Parameters	480
42.9	Coders and Profile Parameters.....	498
42.10	Channel Parameters	500
42.10.1	Voice Parameters	500
42.10.2	Coder Parameters	502
42.10.3	DTMF Parameters	503
42.10.4	RTP, RTCP and T.38 Parameters.....	504
42.11	Least Cost Routing Parameters	508
42.12	LDAP Parameters	509
42.13	SBC and CRP Parameters.....	512
42.14	Standalone Survivability Parameters	524
42.15	IP Media Parameters	528
42.16	Auxiliary and Configuration File Name Parameters	529
42.17	Automatic Update Parameters	530
43	DSP Templates	533
44	Technical Specifications	535

Notice

This document describes the AudioCodes Mediant 4000 SBC.

Information contained in this document is believed to be accurate and reliable at the time of printing. However, due to ongoing product improvements and revisions, AudioCodes cannot guarantee accuracy of printed material after the Date Published nor can it accept responsibility for errors or omissions. Before consulting this document, check the corresponding Release Notes regarding feature preconditions and/or specific support in this release. In cases where there are discrepancies between this document and the Release Notes, the information in the Release Notes supersedes that in this document. Updates to this document and other documents as well as software files can be downloaded by registered customers at <http://www.audiocodes.com/downloads>.

© Copyright 2013 AudioCodes Ltd. All rights reserved.

This document is subject to change without notice.

Date Published: March-03-2014

Trademarks

AudioCodes, AC, AudioCoded, Ardito, CTI2, CTI², CTI Squared, HD VoIP, HD VoIP Sounds Better, InTouch, IPmedia, Mediant, MediaPack, NetCoder, Netrake, Nuera, Open Solutions Network, OSN, Stretto, TrunkPack, VMAS, VoicePacketizer, VoIPerfect, VoIPerfectHD, What's Inside Matters, Your Gateway To VoIP and 3GX are trademarks or registered trademarks of AudioCodes Limited. All other products or trademarks are property of their respective owners. Product specifications are subject to change without notice.

WEEE EU Directive

Pursuant to the WEEE EU Directive, electronic and electrical waste must not be disposed of with unsorted waste. Please contact your local recycling authority for disposal of this product.

Customer Support

Customer technical support and services are provided by AudioCodes or by an authorized AudioCodes Service Partner. For more information on how to buy technical support for AudioCodes products and for contact information, please visit our Web site at www.audiocodes.com/support.

Abbreviations and Terminology

Each abbreviation, unless widely used, is spelled out in full when first used.

Throughout this manual, unless otherwise specified, the term *device* refers to the Mediant 4000 SBC.

Related Documentation

Manual Name
SIP CPE Release Notes
Mediant 4000 E-SBC Hardware Installation Manual



Note: The scope of this document does not fully cover security aspects for deploying the device in your environment. Security measures should be done in accordance with your organization's security policies. For basic security guidelines, you should refer to AudioCodes *Recommended Security Guidelines* document.



Note: Before configuring the device, ensure that it is installed correctly as instructed in the *Hardware Installation Manual*.



Note: This device is considered an INDOOR unit and therefore, must be installed only indoors.



Legal Notice:

- By default, the device supports export-grade (40-bit and 56-bit) encryption due to US government restrictions on the export of security technologies. To enable 128-bit and 256-bit encryption on your device, contact your AudioCodes sales representative.
- This device includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (<http://www.openssl.org/>).
- This device includes cryptographic software written by Eric Young (eay@cryptsoft.com).

Documentation Feedback

AudioCodes continually strives to produce high quality documentation. If you have any comments (suggestions or errors) regarding this document, please fill out the Documentation Feedback form on our Web site at <http://www.audiocodes.com/downloads>.

1 Overview

AudioCodes' Mediant 4000 Enterprise Session Border Controller (E-SBC), hereafter referred to as *device*, is a member of AudioCodes family of E-SBC products, enabling connectivity and security between small medium businesses (SMB) and service providers' VoIP networks. The device is a fully featured enterprise-class SBC provides a secured voice network deployment based on a Back-to-Back User Agent (B2BUA) implementation. The SBC functionality provides perimeter defense for protecting the enterprise from malicious VoIP attacks; mediation for allowing the connection of any IP PBX to any service provider; and service assurance for service quality and manageability.

The device also offers call "survivability" solutions using its Stand Alone Survivability (SAS) or Cloud Resilience Package applications, ensuring service continuity to enterprises served by a centralized SIP-based IP-Centrex server or branch offices of distributed enterprises. Call survivability enables internal office communication between SIP clients in the case of disconnection from the centralized SIP IP-Centrex server or IP-PBX.

The device is a single full-height (FH) AMC module running the session border controller application, consisting of the following:

- 1.25 GHz multi-core CPU
- Eight Ethernet 10/100/1000Base-T ports, grouped into pairs where one port is active while the other is standby (1+1), auto-negotiation, half- and full-duplex modes, and straight-through and crossover cable detection
- 1+1 power load-sharing and redundancy by providing two Power Supply modules
- 1+1 High Availability by deploying two Mediant 4000 devices, each connected to a different network (in the same subnet). In case of a network failure in one network, a switch over of traffic to the standby Mediant 4000 occurs.

The device supports an optional Media Processing Module (MPM), providing resources for additional IP-to-IP transcoding sessions.

The device allows full management through its HTTP/S-based embedded Web server. This user-friendly Web interface allows remote configuration using any standard Web browser (such as Microsoft™ Internet Explorer™). The device can also be managed through AudioCodes Element Management System (EMS), and SNMP v2/v3.

Reader's Notes

Part I

Getting Started with Initial Connectivity

2 Assigning the OAMP IP Address

The device is shipped with a factory default IP address for its operations, administration, maintenance, and provisioning (OAMP) interface, as shown in the table below:

Table 2-1: Default OAMP IP Address

IP Address	Value
IP Address	192.168.0.2 - this IP address is assigned to the first Ethernet Port Group (top-left ports 1 & 2)
Subnet Mask	255.255.255.0
Default Gateway IP Address	192.168.0.1

The default IP address can be used for initially accessing the device, using any of its management tools (i.e., embedded Web server, EMS, or Telnet). Once accessed, you can change this default IP address to correspond with your networking scheme in which the device is deployed. After changing the IP address, you can re-access the device with this new IP address and start configuring and managing the device as desired.

This section describes the different methods for changing the device's default IP address to suit your networking environment:

- Embedded command line interface (CLI) - see 'CLI' on page 21
- Embedded HTTP/S-based Web server - see 'Web Interface' on page 19



Note: If you are implementing the High Availability feature, see also 'HA Overview' on page 339 for initial setup.

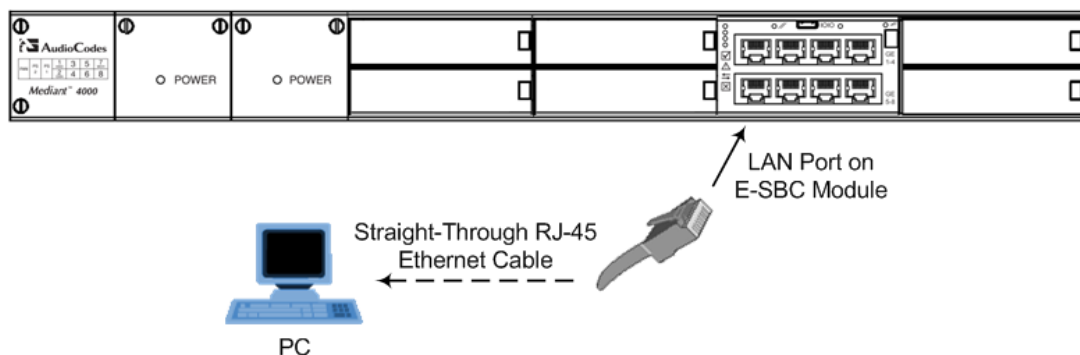
2.1 Web Interface

The procedure below describes how to assign a new OAMP IP address using the Web interface.

➤ **To assign an IP address using the Web interface:**

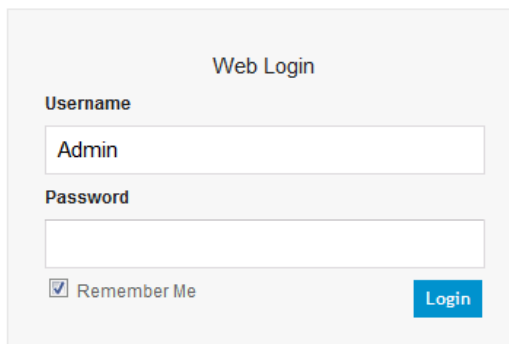
1. Disconnect the network cables (if connected) from the device.
2. Connect the first Ethernet Port Group (top-left ports 1 and 2) located on the front panel directly to the network interface of your computer, using a straight-through Ethernet cable.

Figure 2-1: LAN Connection to PC for Initial Connectivity



3. Change the IP address and subnet mask of your computer to correspond with the default IP address and subnet mask of the device.
4. Access the Web interface:
 - a. On your computer, start a Web browser and in the URL address field, enter the default IP address of the device; the Web interface's Login screen appears:

Figure 2-2: Web Login Screen



Web Login

Username

Admin

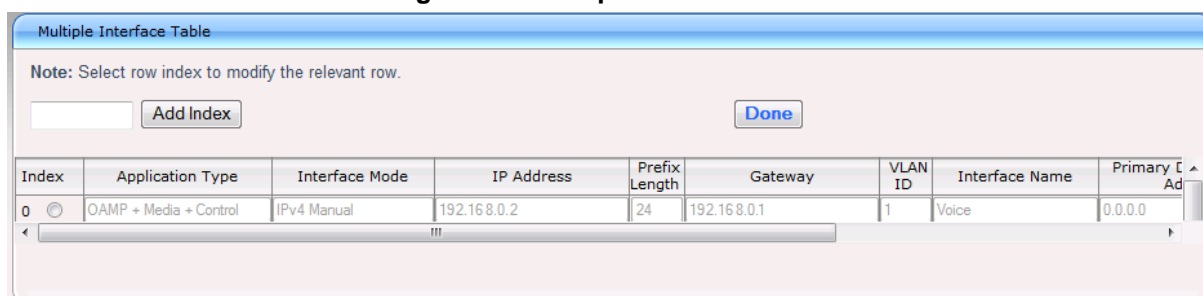
Password

☒ Remember Me

Login

- b. In the 'Username' and 'Password' fields, enter the default login user name ("Admin" - case-sensitive) and password ("Admin" - case-sensitive), and then click **Login**; the device's Web interface is accessed.
5. Open the Physical Ports Settings page (Configuration tab > VoIP menu > Network submenu > Physical Ports Settings) and then modify the device's physical Ethernet port-pair (group) that you want to later assign to the OAMP interface. For more information, see Configuring Physical Ethernet Ports on page 95.
6. Change the default IP address to one that corresponds with your network:
 - a. Open the Multiple Interface Table page (**Configuration** tab > **VoIP** menu > **Network** submenu > **IP Settings**).

Figure 2-3: Multiple Interface Table



Multiple Interface Table

Note: Select row index to modify the relevant row.

Add Index Done

Index	Application Type	Interface Mode	IP Address	Prefix Length	Gateway	VLAN ID	Interface Name	Primary IP Address
0	OAMP + Media + Control	IPv4 Manual	192.168.0.2	24	192.168.0.1	1	Voice	0.0.0.0

- b. Select the 'Index' radio button corresponding to the "OAMP + Media + Control" application type, and then click **Edit**.
 - c. Change the IP address, subnet mask, and Default Gateway IP address to correspond with your network IP addressing scheme.
 - d. From the 'Underlying Interface' drop-down list, select the physical Ethernet port-pair group that you want to assign to the interface.
 - e. Click **Apply**, and then click **Done** to validate your settings.
7. Save your settings to the flash memory with a device reset (see Resetting the Device on page 357).
8. Disconnect the computer from the device and then reconnect the device to your network.

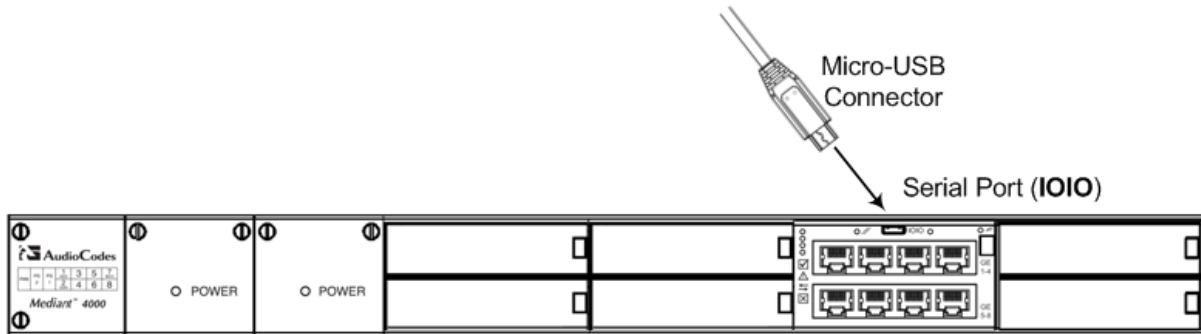
2.2 CLI

The procedure below describes how to assign the OAMP IP address, using CLI.

➤ **To assign the OAMP IP address using CLI:**

1. Connect the RS-232 port of the device to the serial communication port on your computer. For more information, refer to the *Hardware Installation Manual*.

Figure 2-4: Connecting the Serial Interface



2. Establish serial communication with the device using a terminal emulator program (such as HyperTerminal) with the following communication port settings:
 - Baud Rate: 115,200 bps
 - Data Bits: 8
 - Parity: None
 - Stop Bits: 1
 - Flow Control: None
3. At the CLI prompt, type the username (default is "Admin" - case sensitive), and then press Enter:


```
Username: Admin
```
4. At the prompt, type the password (default is "Admin" - case sensitive), and then press Enter:


```
Password: Admin
```
5. At the prompt, type the following, and then press Enter:


```
enable
```
6. At the prompt, type the password again, and then press Enter:


```
Password: Admin
```
7. At the prompt, type the following command to access the VoIP interface, and then press Enter:


```
# configure voip
```
8. At the prompt, type the following command to access the Multiple Interface table, and then press Enter:


```
(config-voip)# interface network-if 0
```
9. At the prompt, type the following command to set the IP address, and then press Enter:


```
(network-if-0)# set ip <IP address>
```
10. At the prompt, type the following command to set the prefix length, and then press Enter:


```
(network-if-0)# set prefix-length <prefix length, e.g., 16>
```

11. At the prompt, type the following command to set the Default Gateway, and then press Enter:
`(network-if-0)# set gateway <IP address>`
12. At the prompt, type the following command to exit the Multiple Interface table, and then press Enter:
`(network-if-0)# exit`
13. At the prompt, type the following command to exit the VoIP interface, and then press Enter:
`(config-voip)# exit`
14. At the prompt, type the following command to save your changes and to reset the device, and then press Enter:
`# reload now`

Part II

Management Tools

3 Introduction

This part provides an overview of the various management tools that can be used to configure the device. It also provides step-by-step procedures on how to configure the management settings.

The following management tools can be used to configure the device:

- Embedded HTTP/S-based Web server - see 'Web-based Management' on page [27](#)
- Command Line Interface (CLI) - see 'CLI-Based Management' on page [63](#)
- AudioCodes Element Management System - see EMS-Based Management on page [77](#)
- Simple Network Management Protocol (SNMP) browser software - see 'SNMP-Based Management' on page [71](#)
- Configuration *ini* file - see 'INI File-Based Management' on page [79](#)

**Notes:**

- Some configuration settings can only be done using a specific management tool. For example, some configuration can only be done using the Configuration ini file method.
- Throughout this manual, where a parameter is mentioned, its corresponding Web and ini file parameter is mentioned. The ini file parameters are enclosed in square brackets [...].
- For a list and description of all the configuration parameters, see 'Configuration Parameters Reference' on page [437](#).

Reader's Notes

4 Web-Based Management

The device provides an embedded Web server (hereafter referred to as *Web interface*), supporting fault management, configuration, accounting, performance, and security (FCAPS), including the following:

- Full configuration
- Software and configuration upgrades
- Loading auxiliary files, for example, the Call Progress Tones file
- Real-time, online monitoring of the device, including display of alarms and their severity
- Performance monitoring of voice calls and various traffic parameters

The Web interface provides a user-friendly, graphical user interface (GUI), which can be accessed using any standard Web browser (e.g., Microsoft™ Internet Explorer).

Access to the Web interface is controlled by various security mechanisms such as login user name and password, read-write privileges, and limiting access to specific IP addresses.



Notes:

- The Web interface allows you to configure most of the device's settings. However, additional configuration parameters may exist that are not available in the Web interface and which can only be configured using other management tools.
- Some Web interface pages and/or parameters are available only for certain hardware configurations or software features. The software features are determined by the installed Software License Key (see 'Software License Key' on page 370).

4.1 Getting Acquainted with the Web Interface

This section provides a description of the Web interface.

4.1.1 Computer Requirements

The client computer requires the following to work with the Web interface of the device:

- A network connection to the device
- One of the following Web browsers:
 - Microsoft™ Internet Explorer™ (Version 6.0 and later)
 - Mozilla Firefox® (Versions 5 through 9.0)
- Recommended screen resolutions: 1024 x 768 pixels, or 1280 x 1024 pixels



Note: Your Web browser must be JavaScript-enabled to access the Web interface.

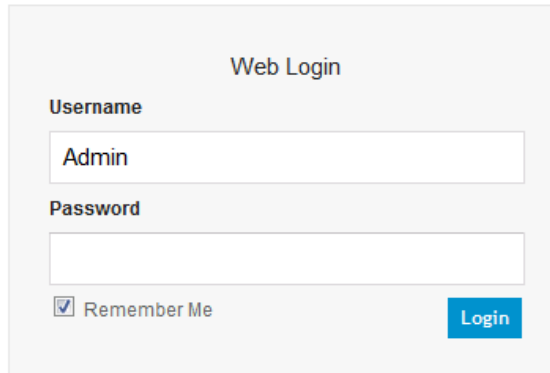
4.1.2 Accessing the Web Interface

The procedure below describes how to access the Web interface.

➤ **To access the Web interface:**

1. Open a standard Web browser (see 'Computer Requirements' on page 27).
2. In the Web browser, specify the IP address of the device (e.g., <http://10.1.10.10>); the Web interface's Login window appears, as shown below:

Figure 4-1: Web Login Screen



The image shows a web login form titled "Web Login". It contains two input fields: "Username" with the text "Admin" entered, and "Password" which is empty. Below the password field is a checkbox labeled "Remember Me" which is checked. To the right of the checkbox is a blue button labeled "Login".

3. In the 'Username' and 'Password' fields, enter the case-sensitive, user name and password respectively.
4. Click **Login**; the Web interface is accessed, displaying the Home page. For a detailed description of the Home page, see 'Viewing the Home Page' on page 49.

Notes:

- By default, Web access is only through the IP address of the OAMP interface. However, you can allow access from any of the device's IP network interfaces (i.e., OAMP, Control, or Media), by setting the EnableWebAccessFromAllInterfaces parameter to 1.
- The default username and password is "Admin". To change the login user name and password, see 'Configuring the Web User Accounts' on page 51.
- If you want the Web browser to remember your password, select the 'Remember Me' check box and then agree to the browser's prompt (depending on your browser) to save the password for future logins. On your next login attempt, simply press the Tab or Enter keys to auto-fill the 'Username' and 'Password' fields, and then click **Login**.



4.1.3 Areas of the GUI

The areas of the Web interface's GUI are shown in the figure below and described in the subsequent table.

Figure 4-2: Main Areas of the Web Interface GUI

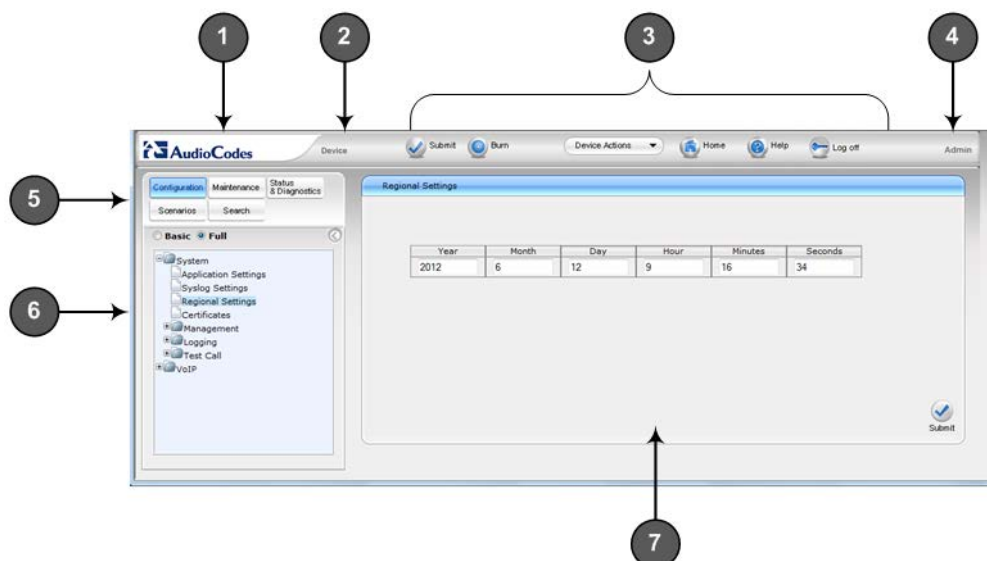




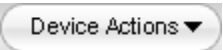



Table 4-1: Description of the Web GUI Areas

Item #	Description
1	Displays AudioCodes (corporate) logo image.
2	Displays the product name.
3	Toolbar, providing frequently required command buttons. For more information, see 'Toolbar Description' on page 30.
4	Displays the username of the Web user that is currently logged in.
5	Navigation bar, providing the following tabs for accessing various functionalities in the Navigation tree: <ul style="list-style-type: none"> ▪ Configuration, Maintenance, and Status & Diagnostics tabs: Access the configuration menus (see 'Working with Configuration Pages' on page 33) ▪ Scenarios tab: Creates configuration scenarios (see Working with Scenarios on page 40) ▪ Search tab: Enables a search engine for searching configuration parameters (see 'Searching for Configuration Parameters' on page 39)
6	Navigation tree, displaying a tree-like structure of elements (configuration menus, Scenario steps, or search engine) pertaining to the selected tab on the Navigation bar. For more information, see 'Navigation Tree' on page 31.
7	Work pane, displaying the configuration page of the selected menu in the Navigation tree. This is where configuration is done. For more information, see 'Working with Configuration Pages' on page 33.

4.1.4 Toolbar Description

The toolbar provides frequently required command buttons, described in the table below:

Table 4-2: Description of Toolbar Buttons

Icon	Button Name	Description
	Submit	Applies parameter settings to the device (see 'Saving Configuration' on page 360). Note: This icon is grayed out when not applicable to the currently opened page.
	Burn	Saves parameter settings to flash memory (see 'Saving Configuration' on page 360).
	Device Actions	Opens a drop-down list with frequently needed commands: <ul style="list-style-type: none"> ▪ Load Configuration File: Opens the Configuration File page for loading an <i>ini</i> file to the device (see 'Backing Up and Loading Configuration File' on page 377). ▪ Save Configuration File: Opens the Configuration File page for saving the <i>ini</i> file to a folder on a computer (see 'Backing Up and Loading Configuration File' on page 377). ▪ Reset: Opens the Maintenance Actions page for performing various maintenance procedures such as resetting the device (see 'Resetting the Device' on page 357). ▪ Software Upgrade Wizard: starts the Software Upgrade wizard for upgrading the device's software (see 'Software Upgrade Wizard' on page 373). ▪ Switch Over: Opens the High Availability Maintenance page for switching between Active and Redundant blades (see High Availability Maintenance on page 361). ▪ Reset Redundant: Opens the High Availability Maintenance page for resetting the Redundant blade (see High Availability Maintenance on page 361).
	Home	Opens the Home page (see 'Viewing the Home Page' on page 49).
	Help	Opens the Online Help topic of the currently opened configuration page (see 'Getting Help' on page 48).
	Log off	Logs off a session with the Web interface (see 'Logging Off the Web Interface' on page 48).



Note: If you modify a parameter that takes effect only after a device reset, after you click the **Submit** button in the configuration page, the toolbar displays "Reset", as shown in the figure below. This is a reminder that you need to later save your settings to flash memory and reset the device.

Figure 4-3: "Reset" Displayed on Toolbar



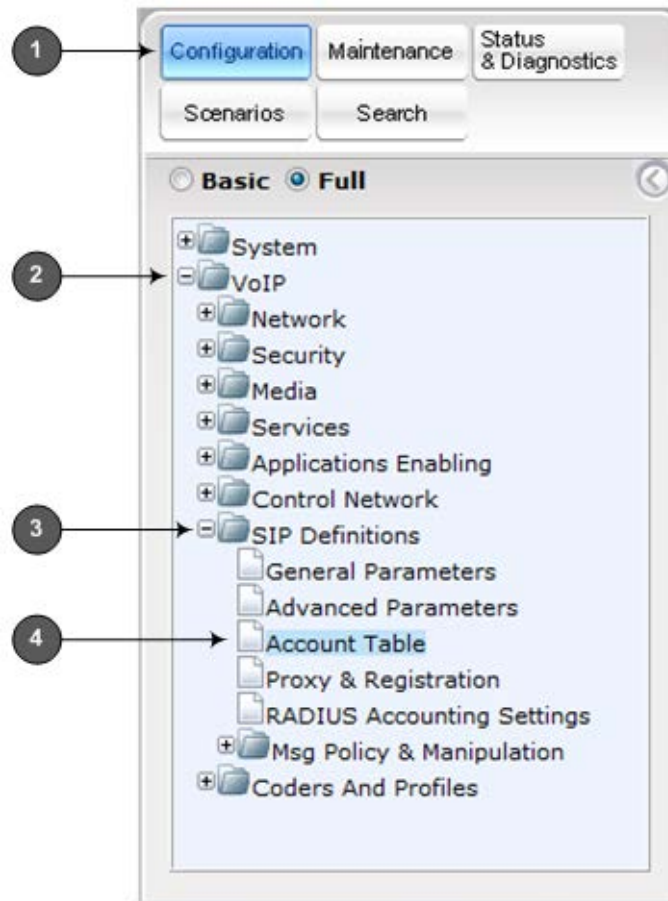
4.1.5 Navigation Tree

The Navigation tree is located in the Navigation pane and displays a tree-like structure of menus pertaining to the selected tab on the Navigation bar. You can drill-down to the required page item level to open its corresponding page in the Work pane.

The terminology used throughout this manual for referring to the hierarchical structure of the tree is as follows:

- *Menu*: first level (highest level)
- *Submenu*: second level - contained within a menu
- *Page item*: last level (lowest level in a menu) - contained within a menu or submenu

Figure 4-4: Navigating in Hierarchical Menu Tree (Example)



Note: The figure above is used only as an example. The displayed menus depend on supported features based on the Software License Key installed on your device.

4.1.5.1 Displaying Navigation Tree in Basic and Full View

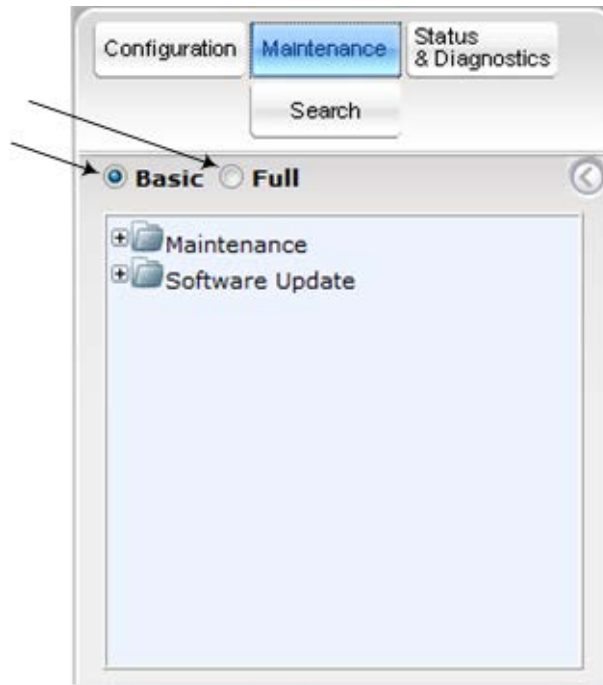
You can view an expanded or reduced display of the Navigation tree. This affects the number of displayed menus and submenus in the tree. The expanded (*Full*) view displays all the menus pertaining to the selected configuration tab; the reduced (*Basic*) view displays only commonly used menus. This is relevant when using the configuration tabs

(i.e., **Configuration**, **Maintenance**, and **Status & Diagnostics**) on the Navigation bar. The advantage of the Basic view is that it prevents "cluttering" of the Navigation tree with menus that may not be required.

➤ **To toggle between Full and Basic view:**

- To display a reduced menu tree, select the **Basic** option (default).
- To display all the menus and submenus in the Navigation tree, select the **Full** option.

Figure 4-5: Basic and Full View Options




Notes:

- After you reset the device, the Web GUI is displayed in Basic view.
- When in Scenario mode (see Scenarios on page 40), the Navigation tree is displayed in Full view.

4.1.5.2 Showing / Hiding the Navigation Pane

You can hide the Navigation pane to provide more space for elements displayed in the Work pane. This is especially useful when the Work pane displays a wide table. The arrow button located below the Navigation bar is used to hide and show the pane.

➤ **To hide and show the Navigation pane:**

- **To hide the Navigation pane:** Click the left-pointing arrow ; the pane is hidden and the button is replaced by the right-pointing arrow button.


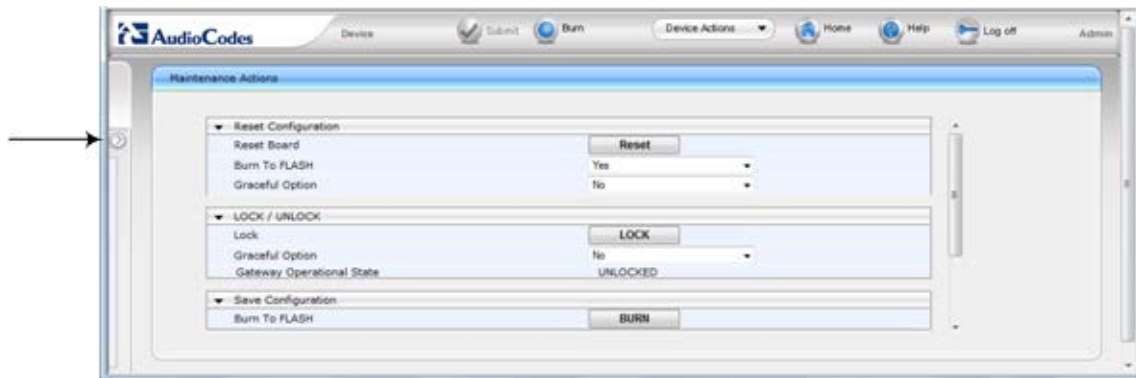
- **To show the Navigation pane:** Click the right-pointing arrow ; the pane is displayed and the button is replaced by the left-pointing arrow button.

Figure 4-6: Show and Hide Button (Navigation Pane in Hide View)





4.1.6 Working with Configuration Pages

The configuration pages contain the parameters for configuring the device and are displayed in the Work pane.

4.1.6.1 Accessing Pages

The configuration pages are accessed by clicking the required page item in the Navigation tree.

➤ To open a configuration page:

1. On the Navigation bar, click the required tab (**Configuration**, **Maintenance**, or **Status & Diagnostics**); the menus pertaining to the selected tab appear in the Navigation tree.
2. Navigate to the required page item, by performing the following:
 - Drill-down using the **plus**  sign to expand the menu and submenus.
 - Drill-up using the **minus**  sign to collapse the menu and submenus.
3. Click the required page item; the page opens in the Work pane.

You can also access previously opened pages by clicking the Web browser's **Back** button until you have reached the required page. This is useful if you want to view pages in which you have performed configurations in the current Web session.

Notes:

- You can also access certain pages from the **Device Actions** button located on the toolbar (see 'Toolbar Description' on page 30).
- To view all the menus in the Navigation tree, ensure that the Navigation tree is in Full view (see 'Displaying Navigation Tree in Basic and Full View' on page 31).
- To get Online Help for the currently displayed page, see 'Getting Help' on page 48.
- Certain pages may not be accessible or may be read-only, depending on the access level of your Web user account (see 'Configuring Web User Accounts' on page 51). If a page is read-only, "Read-Only Mode" is displayed at the bottom of the page.



4.1.6.2 Viewing Parameters

Some pages allow you to view a reduced or expanded display of parameters. The Web interface provides two methods for displaying page parameters:

- Displaying "basic" and "advanced" parameters - see 'Displaying Basic and Advanced Parameters' on page 34
- Displaying parameter groups - see 'Showing / Hiding Parameter Groups' on page 35

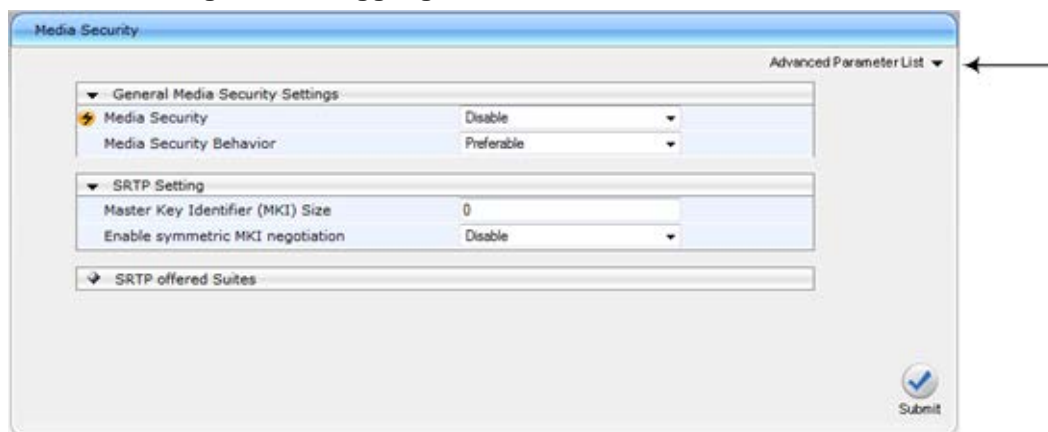
4.1.6.2.1 Displaying Basic and Advanced Parameters

Some pages provide a toggle button that allows you to show and hide parameters that typically are used only in certain deployments. This button is located on the top-right corner of the page and has two display states:

- **Advanced Parameter List** button with down-pointing arrow: click this button to display all parameters.
- **Basic Parameter List** button with up-pointing arrow: click this button to show only common (*basic*) parameters.

The figure below shows an example of a page displaying basic parameters only. If you click the **Advanced Parameter List** button (shown below), the page will also display the advanced parameters.

Figure 4-7: Toggling between Basic and Advanced View



Notes:

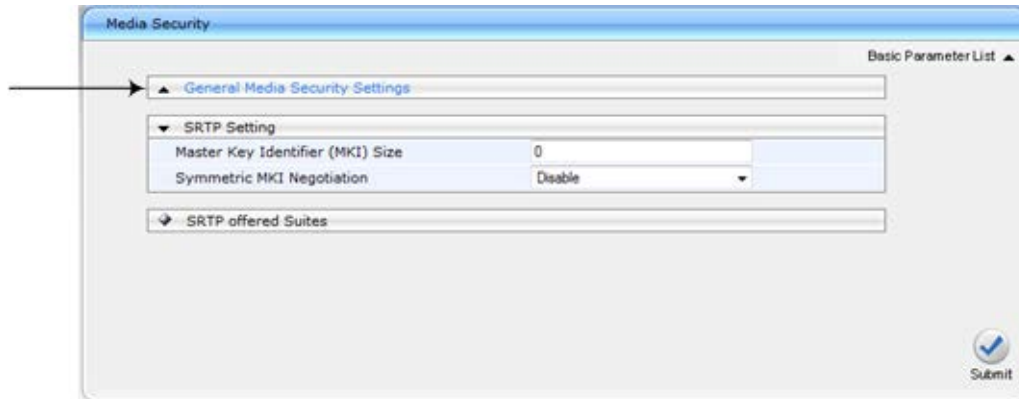
- When the Navigation tree is in Full mode (see 'Navigation Tree' on page 31), configuration pages display all their parameters.
- If a page contains only basic parameters, the **Basic Parameter List** button is not displayed.
- If you reset the device, the Web pages display only the basic parameters.
- The basic parameters are displayed in a dark blue background.



4.1.6.2.2 Showing / Hiding Parameter Groups

Some pages provide groups of parameters, which can be hidden or shown. To toggle between hiding and showing a group, simply click the group title button that appears above each group. The button appears with a down-pointing or up-pointing arrow, indicating that it can be collapsed or expanded when clicked, respectively.

Figure 4-8: Expanding and Collapsing Parameter Groups



4.1.6.3 Modifying and Saving Parameters



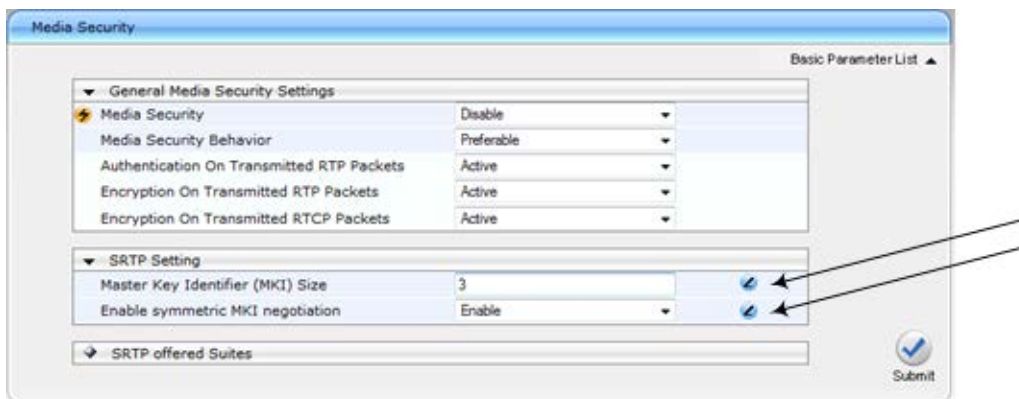



When you modify a parameter value on a page, the **Edit**  symbol appears to the right of the parameter. This indicates that the parameter has been modified, but has yet to be applied (submitted). After you apply your modifications, the  symbol disappears.

Figure 4-9: Edit Symbol after Modifying Parameter Value



- To save configuration changes on a page to the device's volatile memory (RAM), do one of the following:

- On the toolbar, click the **Submit**  button.
- At the bottom of the page, click the **Submit**  button.


When you click **Submit**, modifications to parameters with on-the-fly capabilities are immediately applied to the device and take effect. Parameters displayed on the page with the lightning bolt  symbol take effect only after a device reset. For resetting the device, see 'Resetting the Device' on page 357.



Note: Parameters saved to the volatile memory (by clicking **Submit**), revert to their previous settings after a hardware or software reset, or if the device is powered down. Therefore, to ensure parameter changes (whether on-the-fly or not) are retained, save ('burn') them to the device's non-volatile memory, i.e., flash (see 'Saving Configuration' on page 360).

If you enter an invalid parameter value (e.g., not in the range of permitted values) and then click **Submit**, a message box appears notifying you of the invalid value. In addition, the parameter value reverts to its previous value and is highlighted in red, as shown in the figure below:

Figure 4-10: Value Reverts to Previous Valid Value



4.1.6.4 Working with Tables

This section describes how to work with configuration tables, which are provided in basic or enhanced design, depending on the configuration page.

4.1.6.4.1 Basic Design Tables

A few of the tables in the Web interface are in basic design format. The figure below displays a typical table in the basic design format and the subsequent table describes its command buttons.

Figure 4-11: Adding an Index Entry to a Table




Table 4-3: Basic Table Design Description

Item #	Button / Field	
1	Add Index (or Add) button	Adds an index entry row to the table.
2	Edit	Edits the selected row.
3	Delete	Removes the selected row from the table.
4	'Add Index' field	Defines the index number. When adding a new row, enter the required index number in this field, and then click Add

Item #	Button / Field	
		Index.
5	Index radio button	Selects the row for editing and deleting.
-	Compact button	Organizes the index entries in ascending, consecutive order, starting from index 0. For example, assume you have three index entries, 0, 4 and 6. After you click Compact , index entry 4 is re-assigned to index 1 and index entry 6 is re-assigned to index 2.
-	Apply button	Saves the row configuration. Click this button after you add or edit each index entry.

4.1.6.4.2 Enhanced Design Tables

Most of the tables in the Web interface are designed in the enhanced table format. The figure below displays a typical table in the enhanced design format and the subsequent table describes its command buttons and areas.

Figure 4-12: Displayed Details Pane

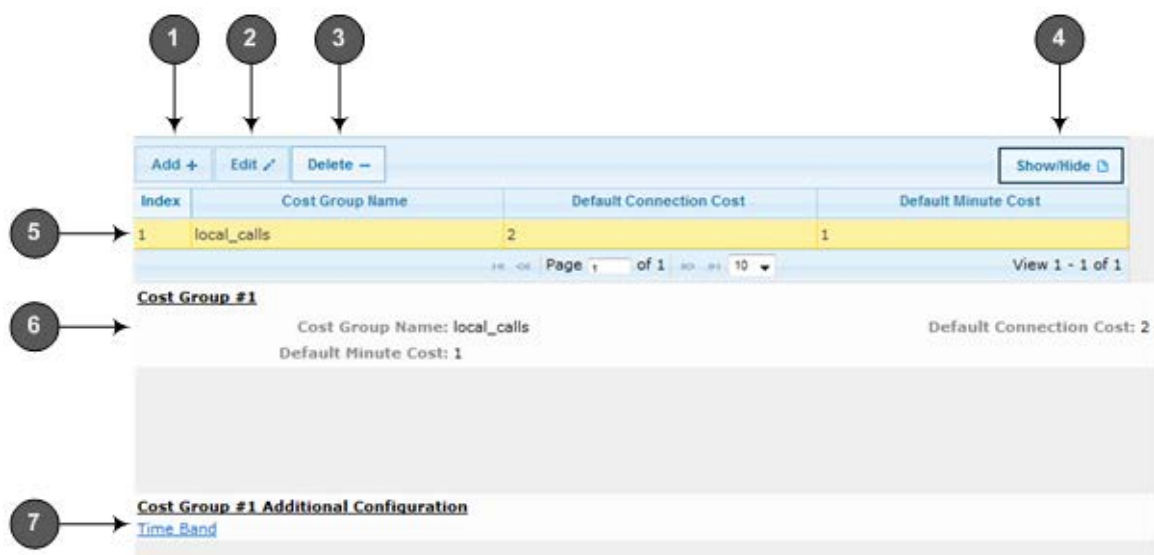
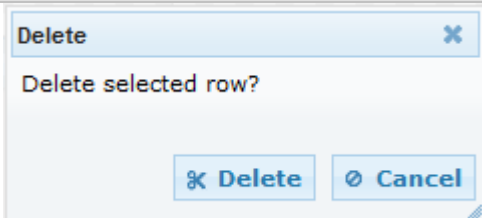


Table 4-4: Enhanced Table Design Description

Item #	Button	
1	Add	Adds a new index entry row to the table. When you click this button, a dialog box appears with parameters for configuring the new entry. When you have completed configuration, click the Submit button in the dialog box to add it to the table.
2	Edit	Edits the selected row.
3	Delete	Removes the selected row from the table. When you click this button, a confirmation box appears requesting you to confirm deletion. Click Delete to accept deletion.

Item #	Button	
		
4	Show/Hide	Toggles between displaying and hiding the full configuration of a selected row. This configuration is displayed below the table (see Item #6) and is useful for large tables that cannot display all its columns in the work pane.
5	-	Selected index row entry for editing, deleting and showing configuration.
6	-	Displays the full configuration of the selected row when you click the Show/Hide button.
7	-	Links to access additional configuration tables related to the current configuration.

If the configuration of an entry row is invalid, the index of the row is highlighted in red, as shown below:

Figure 4-13: Invalid Configuration with Index Highlighted in Red



The table also enables you to define the number of rows to display on the page and to navigate between pages displaying multiple rows. This is done using the page navigation area located below the table, as shown in the figure below:

Figure 4-14: Viewing Table Rows per Page

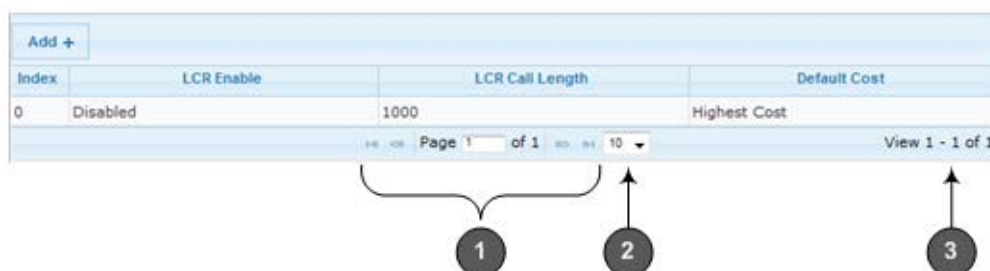


Table 4-5: Row Display and Page Navigation

Item #	Description
1	<p>Defines the page that you want to view. Enter the required page number or use the following page navigation buttons:</p> <ul style="list-style-type: none"> ➡ - Displays the next page ➡ - Displays the last page ⬅ - Displays the previous page ⬅ - Displays the first page
2	<p>Defines the number of rows to display per page. You can select 5 or 10, where the</p>

Item #	Description
	default is 10.
3	Displays the currently displayed page number.

4.1.7 Searching for Configuration Parameters

You can locate the exact Web page on which a specific parameter appears, by using the device's Search feature. The Web parameter's corresponding *ini* file parameter name is used as the search key. The search key can include the full parameter name (e.g., "EnableIPSec") or a substring of it (e.g., "sec"). If you search for a substring, all parameters containing the specified substring in their names are listed in the search result.

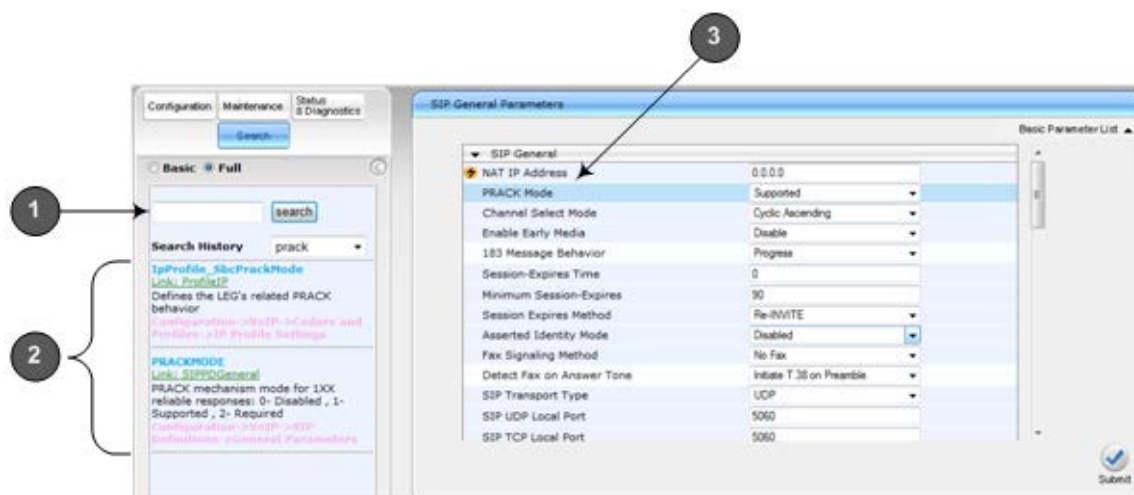


Note: If an *ini* file parameter is not configurable in the Web interface, the search fails.

➤ **To search for a parameter:**

1. On the Navigation bar, click the **Search** tab; the Search engine appears in the Navigation pane.
2. In the field alongside the **Search** button, enter the parameter name or a substring of the name for which you want to search. If you have done a previous search for such a parameter, instead of entering the required string, you can use the 'Search History' drop-down list to select the string saved from a previous search.
3. Click **Search**; a list of found parameters based on your search key appears in the Navigation pane. Each searched result displays the following:
 - *ini* file parameter name
 - Link (in green) to the Web page on which the parameter appears
 - Brief description of the parameter
 - Menu navigation path to the Web page on which the parameter appears

4. In the searched list, click the required parameter (green link) to open the page on which the parameter appears; the relevant page opens in the Work pane and the searched parameter is highlighted in the page for easy identification, as shown in the figure below:

Figure 4-15: Searched Result Screen

Table 4-6: Search Description

Item #	Description
1	Search field for entering search key and Search button for activating the search process.
2	Search results listed in Navigation pane.
3	Found parameter, highlighted on relevant Web page

4.1.8 Working with Scenarios

The Web interface allows you to create your own menu (*Scenario*) of up to 20 pages, selected from the menus in the Navigation tree (i.e., pertaining to the **Configuration**, **Maintenance**, and **Status & Diagnostics** tabs). Each page in the Scenario is referred to as a *Step*. For each Step, you can select up to 25 parameters on the page to include in the Scenario. Therefore, the Scenario feature is useful in that it allows you quick-and-easy access to commonly used configuration parameters specific to your network environment. When you log in to the Web interface, your Scenario is displayed in the Navigation tree.

Instead of creating a new Scenario, you can load a saved Scenario on a computer to the device (see 'Loading a Scenario to the Device' on page 45).

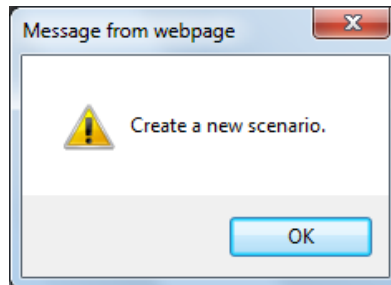
4.1.8.1 Creating a Scenario

The procedure below describes how to create a Scenario.

➤ **To create a Scenario:**

1. On the Navigation bar, click the **Scenarios** tab; a message box appears, requesting you to confirm creation of a Scenario:

Figure 4-16: Create Scenario Confirmation Message Box



Note: If a Scenario already exists, the Scenario Loading message box appears.

2. Click **OK**; the Scenario mode appears in the Navigation tree as well as the menus of the **Configuration** tab.
3. In the 'Scenario Name' field, enter an arbitrary name for the Scenario.
4. On the Navigation bar, click the **Configuration** or **Maintenance** tab to display their respective menus in the Navigation tree.
5. In the Navigation tree, select the required page item for the Step, and then in the page itself, select the required parameters by selecting the check boxes corresponding to the parameters.
6. In the 'Step Name' field, enter a name for the Step.
7. Click the **Next** button located at the bottom of the page; the Step is added to the Scenario and appears in the Scenario Step list.
8. Repeat steps 5 through 7 to add additional Steps (i.e., pages).
9. When you have added all the required Steps for your Scenario, click the **Save & Finish** button located at the bottom of the Navigation tree; a message box appears informing you that the Scenario has been successfully created.

10. Click **OK**; the Scenario mode is quit and the menu tree of the **Configuration** tab appears in the Navigation tree.

Figure 4-17: Creating a Scenario

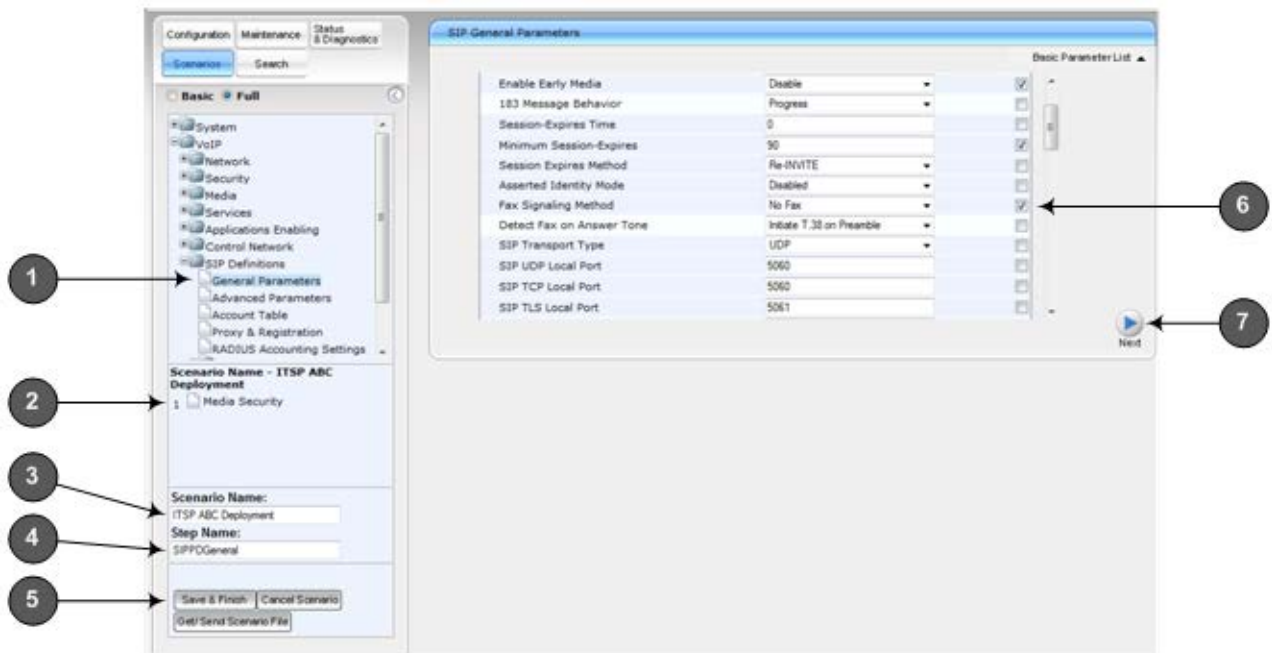


Table 4-7: Scenario Description

	Description
1	Selected page item in the Navigation tree whose page contains the parameter that you want to add to the Scenario Step.
2	Name of a Step that has been added to the Scenario.
3	'Scenario Name' field for defining a name for the Scenario.
4	'Step Name' field for defining a name for a Scenario Step.
5	Save & Finish button to save your Scenario.
6	Selected parameter(s) that you want added to a Scenario Step.
7	Next button to add the current Step to the Scenario and enables you to add additional Steps.

Notes:

- You can add up to 20 Steps per Scenario, where each Step can contain up to 25 parameters.
- When in Scenario mode, the Navigation tree is in 'Full' display (i.e., all menus are displayed in the Navigation tree) and the configuration pages are in 'Advanced Parameter List' display (i.e., all parameters are shown in the pages). This ensures accessibility to all parameters when creating a Scenario. For a description on the Navigation tree views, see 'Navigation Tree' on page 31.
- If you previously created a Scenario and you click the **Create Scenario** button, the previously created Scenario is deleted and replaced with the one you are creating.
- Only Security Administrator Web users can create Scenarios.



4.1.8.2 Accessing a Scenario

Once you have created the Scenario, you can access it by following the procedure below:

➤ **To access the Scenario:**

1. On the Navigation bar, select the **Scenario** tab; a message box appears, requesting you to confirm the loading of the Scenario.
2. Click **OK**; the Scenario and its Steps appear in the Navigation tree, as shown in the example below:

Figure 4-18: Scenario Example

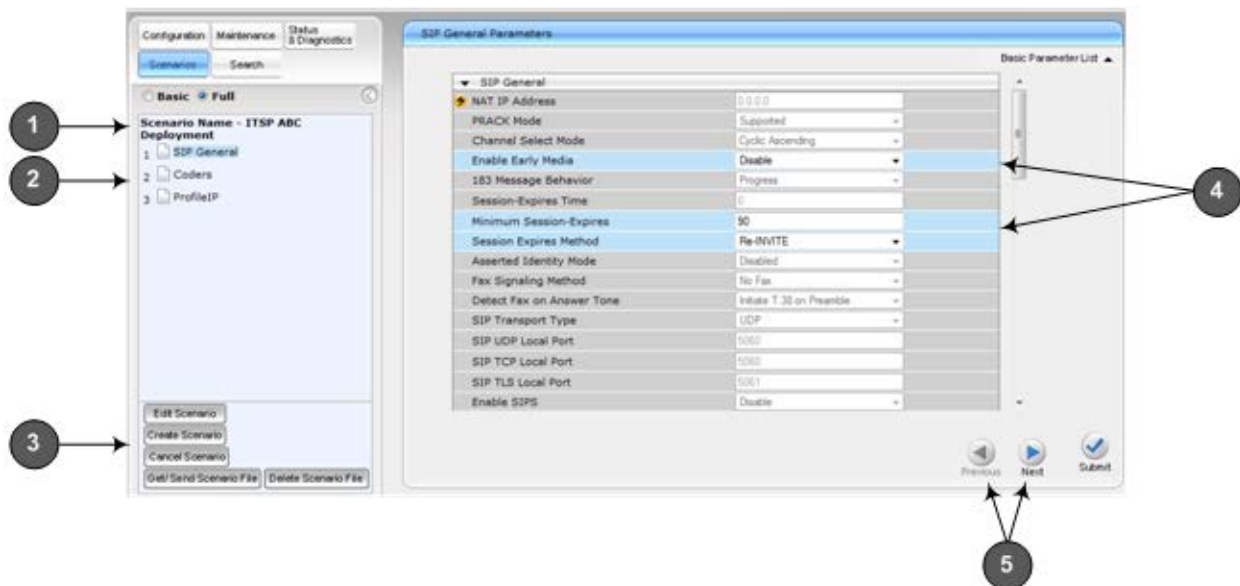




Table 4-8: Loaded Scenario Description

	Description
1	Scenario name.
2	Scenario Steps.
3	Scenario configuration command buttons.
4	Parameters available on a page for the selected Scenario Step. These are displayed in a blue background; unavailable parameters are displayed in a gray or light-blue background.
5	Navigation buttons for navigating between Scenario Steps: <ul style="list-style-type: none"> ▪ Next  button to open the next Step listed in the Scenario ▪ Previous  button to open the previous Step listed in the Scenario



Note: If you reset the device while in Scenario mode, after the device resets, you are returned once again to the Scenario mode.

4.1.8.3 Editing a Scenario

You can modify a Scenario as described in the procedure below.



Note: Only Security Administrator Web users can edit a Scenario.

➤ To edit a Scenario:

1. Open the Scenario.
2. Click the **Edit Scenario** button located at the bottom of the Navigation pane; the 'Scenario Name' and 'Step Name' fields appear.
3. You can perform the following edit operations:
 - **Add Steps:**
 - a. On the Navigation bar, select the desired tab (i.e., **Configuration** or **Maintenance**); the tab's menu appears in the Navigation tree.
 - b. In the Navigation tree, navigate to the desired page item; the corresponding page opens in the Work pane.
 - c. On the page, select the required parameters by marking their corresponding check boxes.
 - d. Click **Next**.
 - **Add or Remove Parameters:**
 - a. In the Navigation tree, select the required Step; the corresponding page opens in the Work pane.
 - b. To add parameters, select the check boxes corresponding to the desired parameters.
 - c. To remove parameters, clear the check boxes corresponding to the desired parameters.
 - d. Click **Next**.
 - **Edit Step Name:**
 - a. In the Navigation tree, select the required Step.
 - b. In the 'Step Name' field, modify the Step name.
 - c. On the page, click **Next**.
 - **Edit Scenario Name:**
 - a. In the 'Scenario Name' field, edit the Scenario name.
 - b. On the displayed page, click **Next**.
 - **Remove a Step:**
 - a. In the Navigation tree, select the required Step; the corresponding page opens in the Work pane.
 - b. On the page, clear all the check boxes corresponding to the parameters.
 - c. Click **Next**.
4. After clicking **Next**, a message box appears notifying you of the change. Click **OK**.
5. Click **Save & Finish**; a message box appears informing you that the Scenario has been successfully modified. The Scenario mode is exited and the menus of the **Configuration** tab appear in the Navigation tree.

4.1.8.4 Saving a Scenario to a PC

You can save a Scenario (as a *dat* file) to a folder on your computer. This is useful when you need multiple Scenarios to represent different deployments. Once you create a Scenario and save it to your computer, you can then keep on saving modifications to it under different Scenario file names. When you require a specific network environment setup, you can load the suitable Scenario file from your computer (see 'Loading a Scenario to the Device' on page 45).

➤ **To save a Scenario to a computer:**

1. On the Navigation bar, click the **Scenarios** tab; the Scenario appears in the Navigation tree.
2. Click the **Get/Send Scenario File** button, located at the bottom of the Navigation tree; the Scenario File page appears, as shown below:

Figure 4-19: Scenario File Page



3. Click the **Get Scenario File** button; the File Download window appears.
4. Click **Save**, and then in the Save As window navigate to the folder to where you want to save the Scenario file. When the file is successfully downloaded to your computer, the Download Complete window appears.
5. Click **Close** to close the window.

4.1.8.5 Loading a Scenario to the Device

The procedure below describes how to load a previously saved Scenario file (*data* file) from your computer to the device. For saving a Scenario, see 'Saving a Scenario to a PC' on page 45.

➤ **To load a Scenario to the device:**

1. On the Navigation bar, click the **Scenarios** tab; the Scenario appears in the Navigation tree.
2. Click the **Get/Send Scenario File** button, located at the bottom of the Navigation tree; the Scenario File page appears.
3. Click the **Browse** button, and then navigate to the Scenario file saved on your computer.
4. Click the **Send File** button.



Notes:

- You can only load a Scenario file to a device that has the same hardware configuration as the device on which it was created.
- The loaded Scenario replaces any existing Scenario.

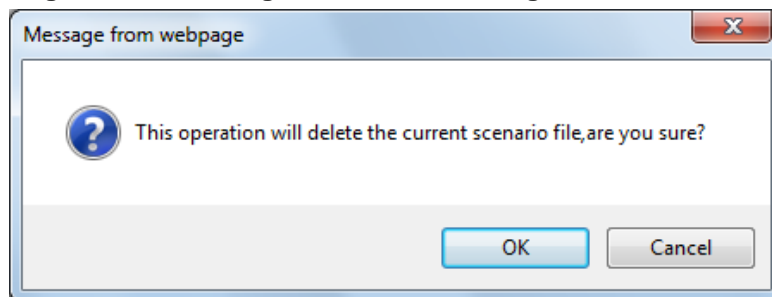
4.1.8.6 Deleting a Scenario

You can delete the Scenario, as described in the procedure below.

➤ **To delete the Scenario:**

1. On the Navigation bar, click the **Scenarios** tab; a message box appears, requesting you to confirm:
2. Click **OK**; the Scenario mode appears in the Navigation tree.
3. Click the **Delete Scenario File** button; a message box appears requesting confirmation for deletion.

Figure 4-20: Message Box for Confirming Scenario Deletion



4. Click **OK**; the Scenario is deleted and the Scenario mode closes.



Note: You can also delete a Scenario using the following alternative methods:

- Loading an empty *.dat* file (see 'Loading a Scenario to the Device' on page 45).
- Loading an *.ini* file with the ScenarioFileName parameter set to no value (i.e., ScenarioFileName = "").

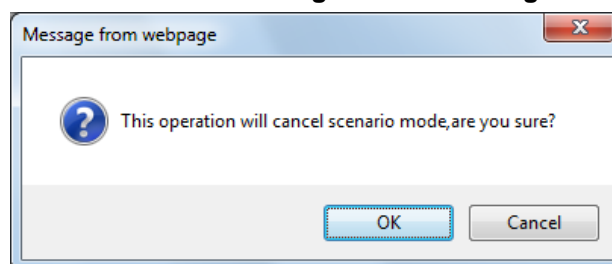
4.1.8.7 Quitting Scenario Mode

Follow the procedure below to quit the Scenario mode.

➤ **To quit the Scenario mode:**

1. On the Navigation bar, click any tab except the **Scenarios** tab, or click the **Cancel Scenarios** button located at the bottom of the Navigation tree; a message box appears, requesting you to confirm exiting Scenario mode, as shown below.

Figure 4-21: Confirmation Message Box for Exiting Scenario Mode



2. Click **OK** to exit.

4.1.9 Creating a Login Welcome Message

You can create a Welcome message box that is displayed on the Web Login page for logging in to the Web interface. The figure below displays an example of a Welcome message:

Figure 4-22: User-Defined Web Welcome Message after Login

The figure consists of two parts. The top part is a yellow rectangular box with a red 'Note' header. Inside the box, the text 'This is a Welcome message' is centered, flanked by asterisks. The bottom part is a 'Web Login' form. It has a title 'Web Login' at the top. Below the title are two input fields: 'Username' and 'Password'. Below the 'Password' field is a checkbox labeled 'Remember Me'. To the right of the checkbox is a blue 'Login' button.

To enable and create a Welcome message, use the WelcomeMessage table ini file parameter. If this parameter is not configured, no Welcome message is displayed.

Table 4-9: ini File Parameter for Welcome Login Message

Parameter	Description
[WelcomeMessage]	<p>Enables and defines a Welcome message that appears on the Web Login page for logging in to the Web interface.</p> <p>The format of this parameter is as follows:</p> <pre>[WelcomeMessage] FORMAT WelcomeMessage_Index = WelcomeMessage_Text; [WelcomeMessage]</pre> <p>For Example:</p> <pre>[WelcomeMessage] FORMAT WelcomeMessage_Index = WelcomeMessage_Text; WelcomeMessage 1 = "*****", WelcomeMessage 2 = "***** This is a Welcome message **", WelcomeMessage 3 = "*****", [WelcomeMessage]</pre> <p>Each index row represents a line of text in the Welcome message box. Up to 20 lines (or rows) of text can be defined.</p>

4.1.10 Getting Help





The Web interface provides you with context-sensitive Online Help. The Online Help provides brief descriptions of parameters pertaining to the currently opened page.

- **To view the Help topic of a currently opened page:**

1. On the toolbar, click the **Help**  button; the Help topic pertaining to the opened page appears, as shown below:

Figure 4-23: Help Topic for Current Page



2. To view a description of a parameter, click the **plus**  sign to expand the parameter. To collapse the description, click the **minus**  sign.
3. To close the Help topic, click the **close**  button located on the top-right corner of the Help topic window or simply click the **Help**  button.



Note: Instead of clicking the **Help** button for each page you open, you can open it once for a page and then simply leave it open. Each time you open a different page, the Help topic pertaining to that page is automatically displayed.

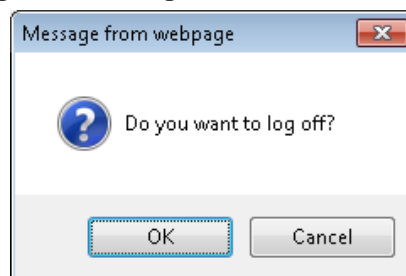
4.1.11 Logging Off the Web Interface

The procedure below describes how to log off the Web interface.

- **To log off the Web interface:**

1. On the toolbar, click the **Log Off**  icon; the following confirmation message box appears:

Figure 4-24: Log Off Confirmation Box



2. Click **OK**; you are logged off the Web session and the Web Login dialog box appears enabling you to re-login, if required.

4.2 Viewing the Home Page

The Home page is displayed when you access the device's Web interface. The Home page provides you with a graphical display of the device's front panel, showing color-coded status icons for various operations device.

➤ **To access the Home page:**


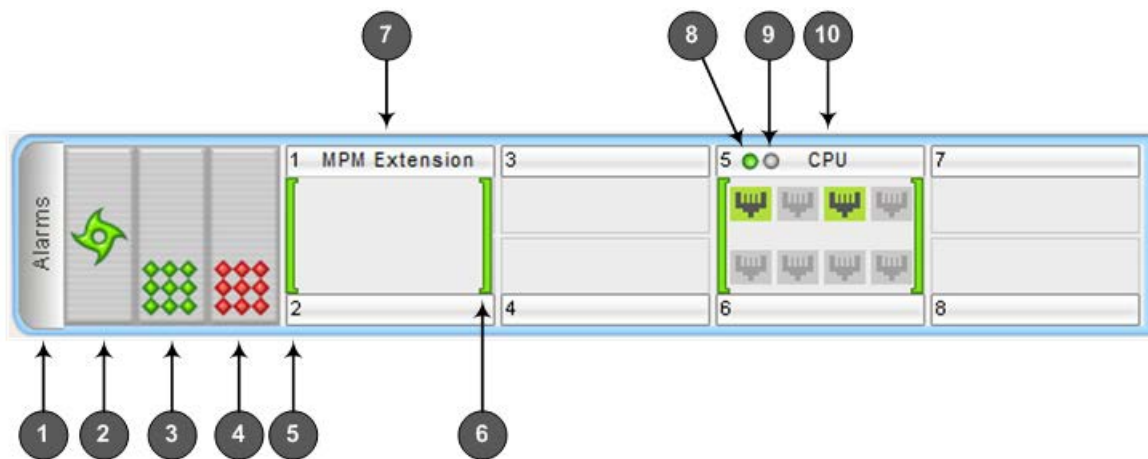
- On the toolbar, click the **Home**  icon.

Figure 4-25: Home Page



Notes:










- The displayed modules depend on your ordered hardware configuration.
- For a description of the Home page when the device is in High Availability (HA) mode, see 'HA Status on the Home Page' on page 341.






In addition to the color-coded status information depicted on the graphical display of the device, the Home page displays various read-only information in the General Information pane:

- **IP Address:** IP address of the device
 - **Subnet Mask:** Subnet mask address of the device
 - **Default Gateway Address:** Default gateway used by the device
 - **Firmware Version:** Software version running on the device
 - **Protocol Type:** Signaling protocol currently used by the device (i.e. SIP)
 - **Gateway Operational State:**
 - "LOCKED": device is locked (i.e. no new calls are accepted)
 - "UNLOCKED": device is not locked
 - "SHUTTING DOWN": device is currently shutting down
- To perform these operations, see 'Basic Maintenance' on page 357.
- **High Availability:** Status of the device's HA mode (for a description, see 'HA Status on the Home Page' on page 341)

The table below describes the areas of the Home page.

Table 4-10: Home Page Description

Item #	Description
1	<p>Displays the highest severity of an active alarm raised (if any) by the device:</p> <ul style="list-style-type: none"> Green = No alarms Red = Critical alarm Orange = Major alarm Yellow = Minor alarm <p>To view active alarms, click the Alarms area to open the Active Alarms page (see Viewing Active Alarms on page 391).</p>
2	<p>Fan tray unit status icon:</p> <ul style="list-style-type: none">  (green): Fan tray operating  (red): Fan tray failure <p>To view detailed information on the device's hardware components, click these icons to open the Components Status page (see Viewing Hardware Components Status).</p>
3	<p>Power Supply Unit 2 status icon:</p> <ul style="list-style-type: none">  (green): Power supply is operating  (red): Power supply failure or no power supply unit installed <p>To view detailed information on the device's hardware components, click these icons to open the Components Status page (see Viewing Hardware Components Status).</p>
4	Power Supply Unit 1 status indicator. See Item #3 for a description.
5	Chassis slot number.
6	<p>Module status icon:</p> <ul style="list-style-type: none">  (green): Module has been inserted or is correctly configured  (gray): Module was removed. "Reserved" is displayed alongside the module's name  (red): Module failure. "Failure" is displayed instead of the module's name
7	<p>Media Processing Module (MPM).</p> <p>Note: This is a customer-ordered module.</p>
8	<p>OK LED:</p> <ul style="list-style-type: none">  (red): Device undergoing startup  (green): Device startup has completed successfully

Item #	Description
9	<p>Active LED:</p> <ul style="list-style-type: none"> ■  (gray): No High Availability (HA). This is displayed if there is no Software License Key installed for HA support, HA is not configured, or upon HA initialization failure. ■  (green): Active device - "stand-alone" state (i.e., Redundant device is missing and HA is unavailable) or in "Active" state and HA is enabled ■  (yellow): <ul style="list-style-type: none"> ✓ Steady on: Redundant device while HA is unavailable (before HA synchronization with Active device has completed or failed connecting to Active device). ✓ Blinking: Redundant device while HA is enabled or device undergoing failure process (and reporting failure information to flash memory).
10	<p>SBC module, providing Ethernet port status icons:</p> <ul style="list-style-type: none"> ■  (green): Ethernet link is working ■  (gray): Ethernet link is not connected <p>To view detailed Ethernet port information, click these icons to open the Ethernet Port Information page (see Viewing Ethernet Port Information on page 390).</p>

4.3 Configuring Web User Accounts

You can create up to 10 Web user accounts for the device. Up to five Web users can simultaneously be logged in to the device's Web interface. Web user accounts prevent unauthorized access to the Web interface, enabling login access only to users with correct credentials (i.e., username and password). Each Web user account is composed of the following attributes:

- **Username and password:** Credentials that enable authorized login access to the Web interface.
- **Access level (user type):** Access privileges specifying what the user can view in the Web interface and its read/write privileges. The table below describes the different types of Web user account access levels:

Table 4-11: Access Levels of Web User Accounts

User Access Level	Numeric Representation*	Privileges
Master	220	Read / write privileges for all pages. Can create all user types, including additional Master users and Security Administrators. It can delete all users except the last Security Administrator.
Security Administrator	200	Read / write privileges for all pages. It can create all user types and is the only one that can create the first Master user. Note: There must be at least one Security Administrator.
Administrator	100	Read / write privileges for all pages except security-related pages, which are read-only.
Monitor	50	No access to security-related and file-loading pages; read-only access to other pages.

User Access Level	Numeric Representation*	Privileges
No Access	0	No access to any page. Note: This access level is not applicable when using advanced Web user account configuration in the Web Users table.
* The numeric representation of the access level is used only to define accounts in a RADIUS server (the access level ranges from 1 to 255).		

By default, the device is pre-configured with the following two Web user accounts:

Table 4-12: Pre-configured Web User Accounts

User Access Level	Username (Case-Sensitive)	Password (Case-Sensitive)
Security Administrator	Admin	Admin
Monitor	User	User

After you log in to the Web interface, the username is displayed on the toolbar.

If the Web session is idle (i.e., no actions are performed) for more than five minutes, the Web session expires and you are once again requested to login with your username and password. Users can be banned for a period of time upon a user-defined number of unsuccessful login attempts. Login information (such as how many login attempts were made and the last successful login time) can be presented to the user.

➤ **To prevent user access after a specific number of failed logins:**

1. From the 'Deny Access On Fail Count' drop-down list, select the number of failed logins after which the user is prevented access to the device for a user-defined time (see next step).
2. In the 'Deny Authentication Timer' field, enter the interval (in seconds) that the user needs to wait before a new login attempt from the same IP address can be done after reaching the number of failed login attempts (defined in the previous step).



Notes:

- For security, it's recommended that you change the default username and password.
- The Security Administrator user can change all attributes of all Web user accounts. Web users with access levels other than Security Administrator can change only their password and username.
- To restore the two Web user accounts to default settings (usernames and passwords), set the *ini* file parameter ResetWebPassword to 1.
- To log in to the Web interface with a different Web user, click the **Log off** button and then login with with a different username and password.
- You can set the entire Web interface to read-only (regardless of Web user access levels), by using the *ini* file parameter DisableWebConfig (see 'Web and Telnet Parameters' on page 444).
- You can define additional Web user accounts using a RADIUS server.

4.3.1 Basic User Accounts Configuration

This section describes basic Web user account configuration. This is relevant only if the two default, pre-configured Web user accounts - Security Administrator ("Admin") and Monitor ("User") - are sufficient for your management scheme.

For the Security Administrator, you can change only the username and password; not its access level. For the Monitor user, you can change username and password as well as access level (Administrator, Monitor, or No Access).



Notes:

- The access level of the Security Administrator cannot be modified.
- The access level of the second user account can be modified only by the Security Administrator.
- The username and password can be a string of up to 19 characters. When you log in to the Web interface, the username and password string values are case-sensitive, according to your configuration.
- Up to two users can be logged in to the Web interface at the same time, and they can be of the same user.

➤ To configure the two pre-configured Web user accounts:

1. Open the Web User Accounts page (**Configuration** tab > **System** menu > **Web User Accounts**). If you are logged in as Security Administrator, both Web user accounts are displayed (as shown below). If you are logged in with the second user account, only the details of this user account are displayed.

Figure 4-26: WEB User Accounts Page (for Users with 'Security Administrator' Privileges)

Current Logged User: Admin		
▼ Account Data for User: Admin		
User Name	Admin	Change User Name
Access Level	Security Administrator	
▼ Fill in the following 3 fields to change the password		
Current Password		
New Password		
Confirm New Password		Change Password
▼ Account Data for User: User		
User Name	User	Change User Name
Access Level	User Monitor	Change Access Level
▼ Fill in the following 3 fields to change the password		
Current Password		
New Password		
Confirm New Password		Change Password
▼ Web Users Table		
Create Web Users Table	Create Table	

2. To change the username of an account:
 - a. In the 'User Name' field, enter the new user name.
 - b. Click **Change User Name**; if you are currently logged in to the Web interface with this account, the 'Web Login' dialog box appears.

- c. Log in with your new user name.
3. To change the password of an account:
 - a. In the 'Current Password' field, enter the current password.
 - b. In the 'New Password' and 'Confirm New Password' fields, enter the new password.
 - c. Click **Change Password**; if you are currently logged in to the Web interface with this account, the 'Web Login' dialog box appears.
 - d. Log in with your new password.
4. To change the access level of the optional, second account:
 - a. Under the **Account Data for User: User** group, from the 'Access Level' drop-down list, select a new access level user.
 - b. Click **Change Access Level**; the new access level is applied immediately.

4.3.2 Advanced User Accounts Configuration

This section describes advanced Web user account configuration. This is relevant if you need the following management scheme:

- Enhanced security settings per Web user (e.g., limit session duration)
- More than two Web user accounts (up to 10 Web user accounts)
- Master users

This advanced Web user configuration is done in the Web Users table, which is initially accessed from the Web User Accounts page (see procedure below). Once this table is accessed, subsequent access immediately opens the Web Users table instead of the Web User Accounts page.



Notes:

- Only the Security Administrator user can **initially** access the Web Users table.
- Only Security Administrator and Master users can add, edit, or delete users.
- Admin users have read-only privileges in the Web Users table. Monitor users have no access to this page.
- If you delete a user who is currently in an active Web session, the user is immediately logged off by the device.
- All users can change their own passwords. This is done in the WEB Security Settings page (see 'Configuring Web Security Settings' on page 58).
- To remove the Web Users table and revert to the Web User Accounts page with the pre-configured, default Web user accounts, set the ResetWebPassword *ini* file parameter to 1. This also deletes all other Web users.
- Once the Web Users table is accessed, Monitor users and Admin users can only change their passwords in the Web Security Settings page (see 'Configuring Web Security Settings' on page 58). The new password must have at least four different characters than the previous password. (The Security Administrator users and Master users can change their passwords in the Web Users table and in the Web Security Settings page.)
- This table can only be configured using the Web interface.

➤ **To add Web user accounts with advanced settings:**

1. Open the Web Users Table page:
 - Upon initial access:
 - a. Open the Web User Accounts page (**Configuration** tab > **System** menu > **Web User Accounts**).
 - b. Under the **Web Users Table** group, click the **Create Table** button.
 - Subsequent access: **Configuration** tab > **System** menu > **Web User Accounts**.
- The Web Users table appears, listing the two default, pre-configured Web use accounts - Security Administrator ("Admin") and Monitor ("User"):

Figure 4-27: Web Users Table Page

Index	Username	Password	Status	Password Age	Session Limit	Session Timeout	Block Duration	User Level
0	Admin	*	Valid	0	2	60	60	SecAdmin
1	User	*	Valid	0	2	60	60	Monitor

Page 1 of 1 View 1 - 2 of 2

2. Click the **Add** button; the following dialog box is displayed:

Figure 4-28: Web Users Table - Add Record Dialog Box

Add Record

Index

0

Username

Password

Status

New

Password Age

90

Session Limit

2

Session Timeout

60

Block Duration

60

User Level

Monitor

Submit

Cancel

3. Add a user as required. For a description of the parameters, see the table below.
4. Click **Submit**.

Table 4-13: Web User Parameters Description

Parameter	Description
Web: Username	Defines the Web user's username. The valid value is a string of up to 40 alphanumeric characters, including the period ".", underscore "_", and hyphen "-" signs.
Web: Password	Defines the Web user's password. The valid value is a string of 8 to 40 ASCII characters, which must include the following: <ul style="list-style-type: none"> ▪ At least eight characters ▪ At least two letters that are upper case (e.g., "AA")

Parameter	Description
	<ul style="list-style-type: none"> At least two letters that are lower case (e.g., "aa") At least two numbers At least two signs (e.g., the dollar "\$" sign) No spaces in the string At least four characters different to the previous password
Web: Status	<p>Defines the status of the Web user.</p> <ul style="list-style-type: none"> New = (Default) User is required to change its password on the next login. When the user logs in to the Web interface, the user is immediately prompted to change the current password. Valid = User can log in to the Web interface as normal. Failed Access = This state is automatically set for users that exceed a user-defined number of failed login attempts, set by the 'Deny Access on Fail Count' parameter (see 'Configuring Web Security Settings' on page 58). These users can log in only after a user-defined timeout configured by the 'Block Duration' parameter (see below) or if their status is changed (to New or Valid) by a System Administrator or Master. Old Account = This state is automatically set for users that have not accessed the Web interface for a user-defined number of days, set by the 'User Inactivity Timer' (see 'Configuring Web Security Settings' on page 58). These users can only log in to the Web interface if their status is changed (to New or Valid) by a System Administrator or Master. <p>Notes:</p> <ul style="list-style-type: none"> The Old Account status is applicable only to Admin and Monitor users; System Administrator and Master users can be inactive indefinitely. For security, it is recommended to set the status of a newly added user to New in order to enforce password change.
Web: Password Age	<p>Defines the duration (in days) of the validity of the password. When this duration elapses, the user is prompted to change the password; otherwise, access to the Web interface is blocked.</p> <p>The valid value is 0 to 10000, where 0 means that the password is always valid. The default is 90.</p>
Web: Session Limit	<p>Defines the maximum number of Web interface sessions allowed for the user. In other words, this allows the same user account to log in to the device from different sources (i.e., IP addresses).</p> <p>The valid value is 0 to 5. The default is 2.</p> <p>Note: Up to 5 users can be logged in to the Web interface at any given.</p>
Web: Session Timeout	<p>Defines the duration (in minutes) of Web inactivity of a logged-in user, after which the user is automatically logged off the Web interface.</p> <p>The valid value is 0 to 100000. The default is according to the settings of the 'Session Timeout' global parameter (see 'Configuring Web Security Settings' on page 58).</p>

Parameter	Description
Web: Block Duration	<p>Defines the duration (in seconds) for which the user is blocked when the user exceeds a user-defined number of failed login attempts. This is configured by the 'Deny Access On Fail Count' parameter (see 'Configuring Web Security Settings' on page 58).</p> <p>The valid value is 0 to 100000, where 0 means that the user can do as many login failures without getting blocked. The default is according to the settings of the 'Deny Authentication Timer' parameter (see 'Configuring Web Security Settings' on page 58).</p> <p>Note: The 'Deny Authentication Timer' parameter relates to failed Web logins from specific IP addresses.</p>
Web: User Level	<p>Defines the user's access level.</p> <ul style="list-style-type: none"> Monitor = (Default) Read-only user. This user can only view Web pages and access to security-related pages is denied. Admin = Read/write privileges for all pages, except security-related pages including the Web Users table where this user has only read-only privileges. SecAdmin = Read/write privileges for all pages. This user is the Security Administrator. Master-User = Read/write privileges for all pages. This user also functions as a security administrator. <p>Notes:</p> <ul style="list-style-type: none"> At least one Security Administrator must exist. The last remaining Security Administrator cannot be deleted. The first Master user can be added only by a Security Administrator user. Additional Master users can be added, edited and deleted only by Master users. If only one Master user exists, it can be deleted only by itself. Master users can add, edit, and delete Security Administrators (but cannot delete the last Security Administrator). Only Security Administrator and Master users can add, edit, and delete Admin and Monitor users.

4.4 Displaying Login Information upon Login

The device can display login information immediately upon Web login.

➤ **To enable display of user login information upon a successful login:**

1. Open the WEB Security Settings page (**Configuration** tab > **System** menu > **Management** submenu > **WEB Security Settings**).
2. From the 'Display Login Information' drop-down list, select **Yes**.
3. Click **Submit** to apply your changes.

Once enabled, the Login Information window is displayed upon a successful login, as shown in the example below:

Figure 4-29: Login Information Window

Login Information	
Last Login Privilege	Security Administrator
Last Failed Login Time	15: 04: 19
Last Failed Login Date	10/06/2012
Last Failed Login IP	10.13.2.11
Login Attempts Since Last Success	2
Last Success Login Time	15: 03: 32
Last Success Login Date	10/06/2012
Last Success Login IP	10.13.2.11

Close

4.5 Configuring Web Security Settings

The WEB Security Settings page is used to define a secure Web access communication method. For a description of these parameters, see 'Web and Telnet Parameters' on page 444.

➤ **To define Web access security:**

1. Open the WEB Security Settings page (**Configuration** tab > **System** menu > **Management** submenu > **WEB Security Settings**).

General	
HTTP Authentication Mode	Web Based Authentication
Secured Web Connection (HTTPS)	HTTP and HTTPS
Requires Client Certificates for HTTPS connection	Disable
HTTPS Cipher String	RC4:EXP
Session	
Session Timeout (minutes)	15
Access Block Parameters	
Deny Authentication Timer	60
Deny Access On Fail Count	3
Display Login Information	No

2. Configure the parameters as required.
3. Click **Submit** to apply your changes.
4. To save the changes to flash memory, see 'Saving Configuration' on page 360.

4.6 Web Login Authentication using Smart Cards

You can enable Web login authentication using certificates from a third-party, common access card (CAC) with user identification. When a user attempts to access the device through the Web browser (HTTPS), the device retrieves the Web user's login username (and other information, if required) from the CAC. The user attempting to access the device is only required to provide the login password. Typically, a TLS connection is established between the CAC and the device's Web interface, and a RADIUS server is implemented to authenticate the password with the username. Therefore, this feature implements a two-factor authentication - what the user has (i.e., the physical card) and what the user knows (i.e., the login password).

This feature is enabled using the `EnableMgmtTwoFactorAuthentication` parameter.



Note: For specific integration requirements for implementing a third-party smart card for Web login authentication, contact your AudioCodes representative.

➤ To log in to the Web interface using CAC:

1. Insert the Common Access Card into the card reader.
2. Access the device using the following URL: `https://<host name or IP address>`; the device prompts for a username and password.
3. Enter the password only. As some browsers require that the username be provided, it's recommended to enter the username with an arbitrary value.

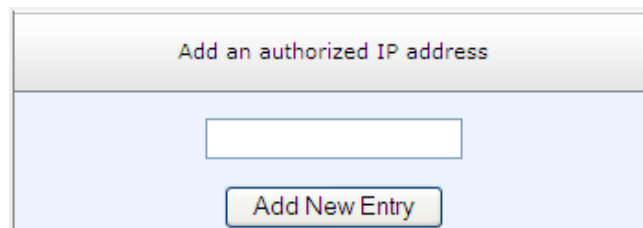
4.7 Configuring Web and Telnet Access List

The Web & Telnet Access List page is used to define IP addresses (up to ten) that are permitted to access the device's Web, Telnet, and SSH interfaces. Access from an undefined IP address is denied. If no IP addresses are defined, this security feature is inactive and the device can be accessed from any IP address. The Web and Telnet Access List can also be defined using the *ini* file parameter `WebAccessList_x` (see 'Web and Telnet Parameters' on page 444).

➤ **To add authorized IP addresses for Web, Telnet, and SSH interfaces access:**

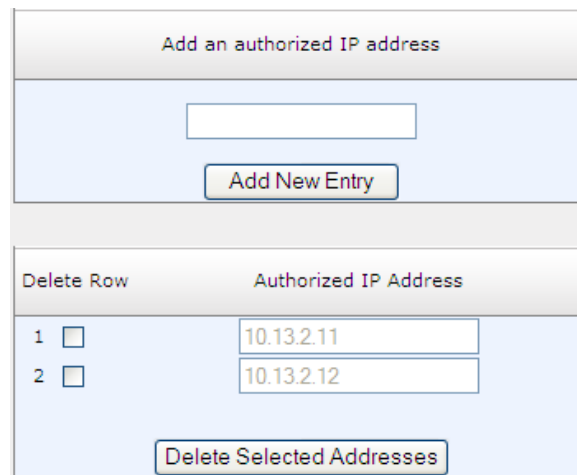
1. Open the Web & Telnet Access List page (**Configuration** tab > **System** menu > **Management** submenu > **Web & Telnet Access List**).

Figure 4-30: Web & Telnet Access List Page - Add New Entry



2. To add an authorized IP address, in the 'Add an authorized IP address' field, enter the required IP address, and then click **Add New Entry**; the IP address you entered is added as a new entry to the Web & Telnet Access List table.

Figure 4-31: Web & Telnet Access List Table



Delete Row	Authorized IP Address
<input type="checkbox"/>	10.13.2.11
<input type="checkbox"/>	10.13.2.12

3. To delete authorized IP addresses, select the Delete Row check boxes corresponding to the IP addresses that you want to delete, and then click **Delete Selected Addresses**; the IP addresses are removed from the table and these IP addresses can no longer access the Web and Telnet interfaces.
4. To save the changes to flash memory, see 'Saving Configuration' on page 360.



Notes:

- The first authorized IP address in the list must be your PC's (terminal) IP address; otherwise, access from your PC is denied.
- Delete your PC's IP address last from the 'Web & Telnet Access List' page. If it is deleted before the last, subsequent access to the device from your PC is denied.

4.8 Configuring RADIUS Settings

The RADIUS Settings page is used for configuring the Remote Authentication Dial In User Service (RADIUS) accounting parameters. For a description of these parameters, see 'Configuration Parameters Reference' on page 437.

➤ **To configure RADIUS:**

1. Open the RADIUS Settings page (**Configuration** tab > **System** menu > **Management** submenu > **RADIUS Settings**).

Figure 4-32: RADIUS Parameters Page

▼ General RADIUS Setting		
⚡ Enable RADIUS Access Control	Disable	▼
Use RADIUS for Web/Telnet Login	Disable	▼
⚡ RADIUS Authentication Server IP Address	0.0.0.0	
⚡ RADIUS Authentication Server Port	1645	
⚡ RADIUS Shared Secret	●●●●●●●●	
▼ General RADIUS Authentication		
Default Access Level	200	
⚡ Device Behavior Upon RADIUS Timeout	Verify Access Locally	▼
⚡ Local RADIUS Password Cache Mode	Reset Timer Upon Access	▼
Local RADIUS Password Cache Timeout [sec]	300	
RADIUS VSA Vendor ID	5003	
RADIUS VSA Access Level Attribute	35	

2. Configure the parameters as required.
3. Click **Submit** to apply your changes.
4. To save the changes to flash memory, see 'Saving Configuration' on page 360.

Reader's Notes

5 CLI-Based Management

This section provides an overview of the CLI-based management and configuration relating to CLI management. The device's CLI-based management interface can be accessed using the RS-232 serial port or by using Secure SHell (SSH) or Telnet through the Ethernet interface.



Notes:

- For security, CLI is disabled by default.
- For information on accessing the CLI interface through the RS-232 port interface, see 'CLI' on page 21.
- CLI is used only for debugging and mainly allows you to view various information regarding device configuration and performance.

5.1 Enabling CLI using Telnet

The device's CLI can be accessed using Telnet. Secure Telnet using Secure Socket Layer (SSL) can be configured whereby information is not transmitted in the clear. If SSL is used, a special Telnet client is required on your PC to connect to the Telnet interface over a secured connection; examples include C-Kermit for UNIX and Kermit-95 for Windows.

For security, some organizations require the display of a proprietary notice upon starting a Telnet session. You can use the configuration ini file parameter, WelcomeMessage to configure such a message (see Creating a Login Welcome Message on page 47).

➤ To enable Telnet:

1. Open the Telnet/SSH Settings page (**Configuration** tab > **System** menu > **Management** > **Telnet/SSH Settings**).

Figure 5-1: Telnet Settings on Telnet/SSH Settings Page

▼ Telnet Settings	
Embedded Telnet Server	Enable Unsecured ▼
Telnet Server TCP Port	23
⚡ Telnet Server Idle Timeout	0

2. Set the 'Embedded Telnet Server' parameter to **Enable Unsecured** or **Enable Secured** (i.e., SSL).
3. Configure the other Tenet parameters as required. For a description of these parameters, see Telnet Parameters on page 447.
4. Click **Submit**.
5. Save the changes to flash memory with a device reset.

5.2 Enabling CLI using SSH and RSA Public Key

The device's CLI can be accessed using Telnet. However, unless configured for TLS, Telnet is not secure as it requires passwords to be transmitted in clear text. To overcome this, Secure SHell (SSH) is used, which is the de-facto standard for secure CLI. SSH 2.0 is a protocol built above TCP, providing methods for key exchange, authentication, encryption, and authorization.

SSH requires appropriate client software for the management PC. Most Linux distributions have OpenSSH pre-installed; Windows-based PCs require an SSH client software such as

PuTTY, which can be downloaded from <http://www.chiark.greenend.org.uk/~sgtatham/putty/>.

By default, SSH uses the same username and password as the Telnet and Web server. SSH supports 1024/2048-bit RSA public keys, providing carrier-grade security. Follow the instructions below to configure the device with an administrator RSA key as a means of strong authentication.

➤ **To enable SSH and configure RSA public keys for Windows (using PuTTY SSH):**

1. Start the PuTTY Key Generator program, and then do the following:
 - a. Under the 'Parameters' group, do the following:
 - ◆ Select the **SSH-2 RSA** option.
 - ◆ In the 'Number of bits in a generated key' field, enter "1024" bits.
 - b. Under the 'Actions' group, click **Generate** and then follow the on-screen instructions.
 - c. Under the 'Actions' group, click **Save private key** to save the new private key to a file (*.ppk) on your PC.
 - d. Under the 'Key' group, select the displayed encoded text between "ssh-rsa" and "rsa-key-....", as shown in the example below:

Figure 5-2: Selecting Public RSA Key in PuTTY



2. Open the Telnet/SSH Settings page (**Configuration** tab > **System** menu > **Management** > **Telnet/SSH Settings**), and then do the following:
 - a. Set the 'Enable SSH Server' parameter to **Enable**.
 - b. Paste the public key that you copied in Step 1.d into the 'Admin Key' field, as shown below:

Figure 5-3: SSH Settings - Pasting Public RSA Key in 'Admin Key' Field

SSH Settings	
Enable SSH Server	Enable
Server Port	22
Admin Key	AAAAB3NzaC1yc2EAAAABJQAAAIBh4d5kzHckyRml3awnb2b/Cscd/d2FDolyGw/bqDYsLI8B9hCtu/GE5ADW8uf6FPxmM7ehhUcMptE2NzqZnz7S2/mE3xEWch451nre n7Jjs3kUcJH2Gdo1eXG2UgF82ag+y0pnu4wukD6k+m1ALMdMQNEVH5k6cw==
Require Public Key	Enable
Max Payload Size	32768
Max Binary Packet Size	35000
Enable Last Login Message	Enable
Max Login Attempts	3

- c. For additional security, you can set the 'Require Public Key' to **Enable**. This ensures that SSH access is only possible by using the RSA key and not by using user name and password.
 - d. Configure the other SSH parameters as required. For a description of these parameters, see SSH Parameters on page 464.
 - e. Click **Submit**.
3. Start the PuTTY Configuration program, and then do the following:
 - a. In the 'Category' tree, drill down to **Connection**, then **SSH**, and then **Auth**; the 'Options controlling SSH authentication' pane appears.
 - b. Under the 'Authentication parameters' group, click **Browse** and then locate the private key file that you created and saved in Step 4.
 4. Connect to the device with SSH using the username "Admin"; RSA key negotiation occurs automatically and no password is required.
- **To configure RSA public keys for Linux (using OpenSSH 4.3):**
1. Run the following command to create a new key in the admin.key file and to save the public portion to the admin.key.pub file:


```
ssh-keygen -f admin.key -N "" -b 1024
```
 2. Open the admin.key.pub file, and then copy the encoded string from "ssh-rsa" to the white space.
 3. Open the Telnet/SSH Settings page (**Configuration** tab > **System** menu > **Management** > **Telnet/SSH Settings**), and then paste the value copied in Step 2 into the 'Admin Key' field.
 4. Click **Submit**.
 5. Connect to the device with SSH, using the following command:


```
ssh -i admin.key xx.xx.xx.xx
```

 where xx.xx.xx.xx is the device's IP address. RSA-key negotiation occurs automatically and no password is required.

5.3 Establishing a CLI Session

The procedure below describes how to establish a CLI session with the device.



Notes:

- The default login username and password are both "Admin" (case-sensitive).
- Only the primary User Account, which has Security Administration access level (200) can access the device using Telnet. For configuring the username and password, see Configuring Web User Accounts on page 51.

- **To establish a CLI session with the device:**
1. Establish a Telnet or SSH session with the device using its OAMP IP address.
 2. Log in to the session using the username and password assigned to the Admin user of the Web interface:
 3. At the Username prompt, type the username, and then press Enter:


```
Username: Admin
```

4. At the Password prompt, type the password, and then press Enter:

```
Password: Admin
```

5. At the prompt, type the following, and then press Enter:

```
enable
```

6. At the prompt, type the password again, and then press Enter:

```
Password: Admin
```

5.4 CLI Commands

The CLI commands are used mainly to display current configuration and performance. These commands are organized in subdirectories. When the CLI session starts, you are located in the 'root' directory.

To access a subdirectory, type its name, and then press Enter. The CLI commands can be entered in an abbreviated format by typing only the letters shown in upper case (i.e., capital letters). For example, the **CHangePassWord** command can be entered by typing **chpw**. If you know the full path to a command inside one of the subdirectories, the short format can be used to run it directly. For example, the **PERformance** command in the **MGmt** subdirectory may be run directly by typing **/mg/perf**.

The following table summarizes the basic CLI commands:

Table 14: Basic CLI Commands

Purpose	Commands	Description
Help	h	Displays the help for a specific command, action, or parameter.
Navigation	cd	Enters another directory.
	cd root	Navigates to the root directory (/).
	..	Goes up one level.
	exit	Terminates the CLI session.

5.4.1 Status Commands

The following table summarizes the Show commands and their corresponding options.

Table 15: Show CLI Commands

Command	Short Format	Arguments	Description
SHow	sh	info dsp ip log	Displays operational data. <ul style="list-style-type: none"> info: Displays general device information tdm: Displays PSTN-related information dsp: Displays DSP resource information ip: Displays information about IP interfaces
SHow INFO	sh info	-	Displays device hardware information, versions, uptime, temperature reading, and the last reset reason.
SHow DSP	sh dsp	status perf	Displays status and version for each DSP device, along with overall performance statistics.

Command	Short Format	Arguments	Description
SHoW IP	sh ip	conf perf route	Displays IP interface status and configuration, along with performance statistics. Note: The display format may change according to the configuration.
SHoW LOG	sh log	[stop]	Displays (or stops displaying) Syslog messages in the CLI session.

Example:

```

/>sh info
Board type: gateway SDH, firmware version 6.60.000.020
Uptime: 0 days, 0 hours, 3 minutes, 54 seconds
Memory usage: 63%
Temperature reading: 39 C
Last reset reason:
Board was restarted due to issuing of a reset from Web interface
Reset Time : 7.1.2012 21.51.13

/>sh dsp status
DSP firmware: 491096AE8 Version:0660.03 Used=0 Free=480 Total=480
DSP device 0: Active Used=16 Free= 0 Total=16
DSP device 1: Active Used=16 Free= 0 Total=16
DSP device 2: Active Used=16 Free= 0 Total=16
DSP device 3: Active Used=16 Free= 0 Total=16
DSP device 4: Active Used=16 Free= 0 Total=16
DSP device 5: Active Used=16 Free= 0 Total=16
DSP device 6: Inactive
DSP device 7: Inactive
DSP device 8: Inactive
DSP device 9: Inactive
DSP device 10: Inactive
DSP device 11: Inactive
DSP device 12: Active Used=16 Free= 0 Total=16
DSP device 13: Active Used=16 Free= 0 Total=16
DSP device 14: Active Used=16 Free= 0 Total=16
DSP device 15: Active Used=16 Free= 0 Total=16
DSP device 16: Active Used=16 Free= 0 Total=16
DSP device 17: Active Used=16 Free= 0 Total=16
DSP device 18: Inactive
PSEC - DSP firmware: AC491IPSEC Version: 0660.03
CONFERENCE - DSP firmware: AC491256C Version: 0660.03

/>sh dsp perf
DSP Statistics (statistics for 968 seconds):
Active DSP resources: 480
Total DSP resources: 480
DSP usage %: 100

/>sh ip perf
Networking Statistics (statistics for 979 seconds):
IP KBytes TX: 25
IP KBytes RX: 330
IP KBytes TX per second: 0
IP KBytes RX per second: 1
IP Packets TX: 1171
IP Packets RX: 5273
IP Packets TX per second: 3
IP Packets RX per second: 12

```

```

Peak KByte/s TX in this interval: 18
Peak KByte/s RX in this interval: 4
Discarded packets: 186
DHCP requests sent: 0
IPSec Security Associations: 0
/>>/mg/perf reset
Done.

/>>sh ip perf
Networking Statistics (statistics for 2 seconds):
IP KBytes TX: 2
IP KBytes RX: 4
IP KBytes TX per second: 0
IP KBytes RX per second: 1
IP Packets TX: 24
IP Packets RX: 71
IP Packets TX per second: 3
IP Packets RX per second: 12
Peak KByte/s TX in this interval: 18
Peak KByte/s RX in this interval: 4
Discarded packets: 0
DHCP requests sent: 0
IPSec Security Associations: 0
/>>sh ip conf
Interface  IP Address          Subnet Mask          Default Gateway
-----
OAM        10.4.64.13          55.255.0.0           10.4.0.1
Media      10.4.64.13          255.255.0.0          10.4.0.1
Control    10.4.64.13          255.255.0.0          10.4.0.1
MAC address: 00-90-8f-04-5c-e9
/>>sh ip route
Destination      Mask                Gateway             Intf  Flags
-----
0.0.0.0          0.0.0.0            10.4.0.1            OAM   A S
10.4.0.0         255.255.0.0        10.4.64.13          OAM   A L
127.0.0.0        255.0.0.0          127.0.0.1           AR    S
127.0.0.1        255.255.255.255    127.0.0.1           A L   H
Flag legend: A=Active R=Reject L=Local S=Static E=rEDirect
M=Multicast
               B=Broadcast H=Host I=Invalid
End of routing table, 4 entries displayed.

```

5.4.2 Ping Command

The Ping command is described in the following table:

Table 16: Ping Command

Command	Short Format	Arguments	Description
PING	ping	[-n count] [-l size] [-w timeout] [-p cos] ip-address	Sends ICMP echo request packets to a specified IP address. <ul style="list-style-type: none"> count: number of packets to send. size: payload size in each packet. timeout: time (in seconds) to wait for a reply to each packet. cos: Class-of-Service (as per 802.1p) to use.

Example:

```

/>ping 10.31.2.10
Ping process started for address 10.31.2.10. Process ID - 27.
Reply from 10.31.2.10: bytes=0 time<0ms
Reply from 10.31.2.10: bytes=0 time<0ms
Reply from 10.31.2.10: bytes=0 time<0ms
Reply from 10.31.2.10: bytes=0 time<0ms
Ping statistics for 10.31.2.10:
Packets:Sent = 4, Received = 4, Lost 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 0ms, Average = 0ms

```

5.4.3 Management Commands

The commands under the **MGmt** directory, described in the table below, display current performance values.

Table 17: CLI Management Command

Command	Short Format	Arguments	Description
/MGmt/PERformance	/mg/perf	basic control dsp net ds1 reset	Displays performance statistics. The <i>reset</i> argument clears all statistics to zero.

5.4.4 Configuration Commands

The commands under the **CONFiguration** directory query and modify the current device configuration. The following commands are available:

Table 18: Configuration CLI Commands

Command	Short Format	Arguments	Description
SetConfigParam IP	/conf/scp ip	ip-addr subnet def-gw	Sets the IP address, subnet mask, and default gateway address of the device (on-the-fly). Note: This command may cause disruption of service. The CLI session may disconnect since the device changes its IP address.
RestoreFactorySettings	/conf/rfs		Restores all parameters to factory settings.
SaveAndRestart	/conf/sar		Saves all current configurations to the non-volatile memory and resets the device.
ConfigFile	/conf/cf	view get set	Retrieves the full <i>ini</i> file from the device and allows loading a new <i>ini</i> file directly in the CLI session. Note: The argument <i>view</i> displays the file, page by page. The argument <i>get</i> displays the file without breaks.

5.4.5 LDAP Commands

The commands under the **IPNetworking\OpenLdap** directory allow you to perform various Lightweight Directory Access Protocol (LDAP) actions.

Table 19: LDAP Commands

Sub-Command	Arguments	Description
LdapSTatus	-	Displays the LDAP connection status.
LdapSearch	<search key> <attribute1> [<attribute 2> ...<attribute 5>]	Searches an LDAP server. The parameters enclosed by [] are optional.
LDapOpen	-	Opens a connection to the LDAP server using parameters provided in the configuration file.
LDapSetDebugmode	<mode>	Sets the LdapDebugLevelMode parameter. Possible levels 0-3.
LDapGetDebugmode	-	Gets the LdapDebugLevelMode parameter value

6 SNMP-Based Management

The device provides an embedded SNMP Agent to operate with a third-party SNMP Manager (e.g., element management system or EMS) for operation, administration, maintenance, and provisioning (OAMP) of the device. The SNMP Agent supports standard Management Information Base (MIBs) and proprietary MIBs, enabling a deeper probe into the interworking of the device. The SNMP Agent can also send unsolicited events (SNMP traps) towards the SNMP Manager. All supported MIB files are supplied to customers as part of the release.

This section provides configuration relating to SNMP management.



Note: For more information on SNMP support such as SNMP traps, refer to the *SNMP User's Guide*.

6.1 Configuring SNMP Community Strings

The SNMP Community String page allows you to configure up to five read-only and up to five read-write SNMP community strings and to configure the community string that is used for sending traps.

For detailed descriptions of the SNMP parameters, see 'SNMP Parameters' on page 448.

➤ **To configure the SNMP community strings:**

1. Open the SNMP Community String page (**Configuration** tab > **System** menu > **Management** submenu > **SNMP** submenu > **SNMP Community String**).

Delete	Community String	Access Level
<input type="checkbox"/>		Read Only
<input type="checkbox"/>		Read Only
<input type="checkbox"/>		Read Only
<input type="checkbox"/>		Read Only
<input type="checkbox"/>		Read Only
<input type="checkbox"/>		Read / Write
<input type="checkbox"/>		Read / Write
<input type="checkbox"/>		Read / Write
<input type="checkbox"/>		Read / Write
<input type="checkbox"/>		Read / Write

Disable SNMP

Trap Community String

Trap Manager Host Name

2. Configure the SNMP community strings parameters according to the table below.
3. Click **Submit** to apply your changes.

4. To save the changes to flash memory, see 'Saving Configuration' on page 360.

To delete a community string, select the **Delete** check box corresponding to the community string that you want to delete, and then click **Submit**.

Table 6-1: SNMP Community String Parameters Description

Parameter	Description
Community String	<ul style="list-style-type: none"> ▪ Read Only [SNMPReadOnlyCommunityString_x]: Up to five read-only community strings (up to 19 characters each). The default string is 'public'. ▪ Read / Write [SNMPReadWriteCommunityString_x]: Up to five read / write community strings (up to 19 characters each). The default string is 'private'.
Trap Community String [SNMPTrapCommunityString]	Community string used in traps (up to 19 characters). The default string is 'trapuser'.

6.2 Configuring SNMP Trap Destinations

The SNMP Trap Destinations page allows you to configure up to five SNMP trap managers. You can associate a trap destination with SNMPv2 users and specific SNMPv3 users. Associating a trap destination with SNMPv3 users sends encrypted and authenticated traps to the SNMPv3 destination. By default, traps are sent unencrypted using SNMPv2.

- To configure SNMP trap destinations:

1. Open the SNMP Trap Destinations page (**Configuration** tab > **System** menu > **Management** submenu > **SNMP** > **SNMP Trap Destinations**).

Figure 6-1: SNMP Trap Destinations Page

		IP Address	Trap Port	Trap User	Trap Enable
<input checked="" type="checkbox"/>	SNMP Manager 1	0.0.0.0	162	v2cParams ▼	Enable ▼
<input checked="" type="checkbox"/>	SNMP Manager 2	0.0.0.0	162	hq-snmpv3 ▼	Enable ▼
<input type="checkbox"/>	SNMP Manager 3	0.0.0.0	162	v2cParams ▼	Enable ▼
<input type="checkbox"/>	SNMP Manager 4	0.0.0.0	162	v2cParams ▼	Enable ▼
<input type="checkbox"/>	SNMP Manager 5	0.0.0.0	18	v2cParams ▼	Enable ▼

2. Configure the SNMP trap manager parameters according to the table below.
3. Select the check box corresponding to the SNMP Manager that you wish to enable.
4. Click **Submit** to apply your changes.



Note: Only row entries whose corresponding check boxes are selected are applied when clicking **Submit**; otherwise, settings revert to their defaults.

Table 6-2: SNMP Trap Destinations Parameters Description

Parameter	Description
Web: SNMP Manager [SNMPManagerIsUsed_x]	Enables the SNMP Manager to receive traps and checks the validity of the configured destination (IP address and port number). <ul style="list-style-type: none"> ▪ [0] (check box cleared) = (Default) Disables SNMP Manager ▪ [1] (check box selected) = Enables SNMP Manager
Web: IP Address [SNMPManagerTableIP_x]	Defines the IP address (in dotted-decimal notation, e.g., 108.10.1.255) of the remote host used as the SNMP Manager. The device sends SNMP traps to this IP address.
Trap Port [SNMPManagerTrapPort_x]	Defines the port number of the remote SNMP Manager. The device sends SNMP traps to this port. The valid value range is 100 to 4000. The default is 162.
Web: Trap User [SNMPManagerTrapUser]	Associates a trap user with the trap destination. This determines the trap format, authentication level, and encryption level. <ul style="list-style-type: none"> ▪ v2cParams (default) = SNMPv2 user community string ▪ SNMPv3 user configured in 'Configuring SNMP V3 Users' on page 74
Trap Enable [SNMPManagerTrapSendingEnable_x]	Activates the sending of traps to the SNMP Manager. <ul style="list-style-type: none"> ▪ [0] Disable ▪ [1] Enable (Default)

6.3 Configuring SNMP Trusted Managers

The SNMP Trusted Managers page allows you to configure up to five SNMP Trusted Managers, based on IP addresses. By default, the SNMP agent accepts SNMP Get and Set requests from any IP address, as long as the correct community string is used in the request. Security can be enhanced by using Trusted Managers, which is an IP address from which the SNMP agent accepts and processes SNMP requests.



Notes: The SNMP Trusted Managers table can also be configured using the table ini file parameter, SNMPTrustedMgr_x (see 'SNMP Parameters' on page 448).

➤ **To configure SNMP Trusted Managers:**

1. Open the SNMP Trusted Managers page (**Configuration** tab > **System** menu > **Management** submenu > **SNMP** submenu > **SNMP Trusted Managers**).

Figure 6-2: SNMP Trusted Managers

Delete	Trusted Managers IP Address	
<input type="checkbox"/>	SNMP Trusted Manager 1	<input type="text" value="0.0.0.0"/>
<input type="checkbox"/>	SNMP Trusted Manager 2	<input type="text" value="0.0.0.0"/>
<input type="checkbox"/>	SNMP Trusted Manager 3	<input type="text" value="0.0.0.0"/>
<input type="checkbox"/>	SNMP Trusted Manager 4	<input type="text" value="0.0.0.0"/>
<input type="checkbox"/>	SNMP Trusted Manager 5	<input type="text" value="0.0.0.0"/>

2. Select the check box corresponding to the SNMP Trusted Manager that you want to enable and for whom you want to define an IP address.
3. Define an IP address in dotted-decimal notation.
4. Click **Submit** to apply your changes.
5. To save the changes, see 'Saving Configuration' on page 360.

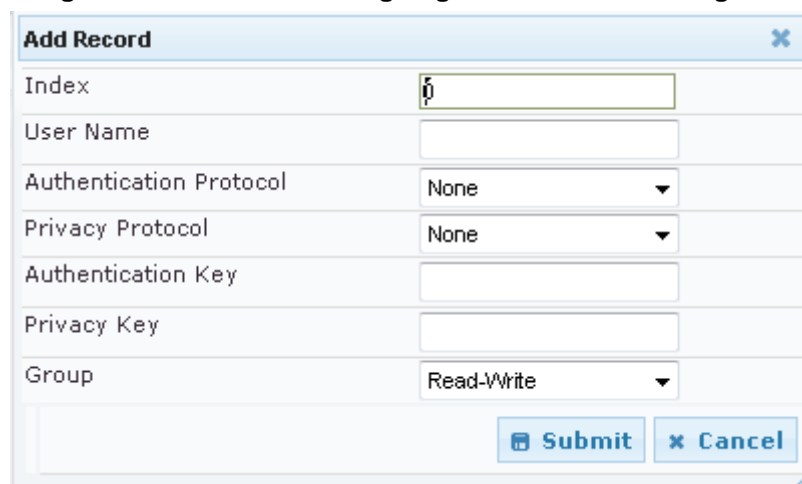
6.4 Configuring SNMP V3 Users

The SNMP v3 Users page allows you to configure authentication and privacy for up to 10 SNMP v3 users.

➤ **To configure SNMP v3 users:**

1. Open the SNMP v3 Users page (**Configuration** tab > **System** menu > **Management** submenu > **SNMP** submenu > **SNMP V3 Users**).
2. Click **Add**; the following dialog box appears:

Figure 6-3: SNMP V3 Setting Page - Add Record Dialog Box



The dialog box titled 'Add Record' contains the following fields and controls:

- Index:** A text input field with the value '5'.
- User Name:** An empty text input field.
- Authentication Protocol:** A dropdown menu with 'None' selected.
- Privacy Protocol:** A dropdown menu with 'None' selected.
- Authentication Key:** An empty text input field.
- Privacy Key:** An empty text input field.
- Group:** A dropdown menu with 'Read-Write' selected.
- Buttons:** 'Submit' and 'Cancel' buttons at the bottom right.

3. Configure the SNMP V3 Setting parameters according to the table below.
4. Click **Submit** to apply your settings.
5. To save the changes, see 'Saving Configuration' on page 360.

**Notes:**

- If you delete a user that is associated with a trap destination (in 'Configuring SNMP Trap Destinations' on page 72), the configured trap destination becomes disabled and the trap user reverts to default (i.e., SNMPv2).
- The SNMP v3 Users table can also be configured using the table ini file parameter, SNMPUsers (see 'SNMP Parameters' on page 448).

Table 6-3: SNMP V3 Users Parameters

Parameter	Description
Index [SNMPUsers_Index]	The table index. The valid range is 0 to 9.
User Name [SNMPUsers_Username]	Name of the SNMP v3 user. This name must be unique.
Authentication Protocol [SNMPUsers_AuthProtocol]	Authentication protocol of the SNMP v3 user. <ul style="list-style-type: none"> ▪ [0] None (default) ▪ [1] MD5 ▪ [2] SHA-1
Privacy Protocol [SNMPUsers_PrivProtocol]	Privacy protocol of the SNMP v3 user. <ul style="list-style-type: none"> ▪ [0] None (default) ▪ [1] DES ▪ [2] 3DES ▪ [3] AES-128 ▪ [4] AES-192 ▪ [5] AES-256
Authentication Key [SNMPUsers_AuthKey]	Authentication key. Keys can be entered in the form of a text password or long hex string. Keys are always persisted as long hex strings and keys are localized.
Privacy Key [SNMPUsers_PrivKey]	Privacy key. Keys can be entered in the form of a text password or long hex string. Keys are always persisted as long hex strings and keys are localized.
Group [SNMPUsers_Group]	The group with which the SNMP v3 user is associated. <ul style="list-style-type: none"> ▪ [0] Read-Only (default) ▪ [1] Read-Write ▪ [2] Trap Note: All groups can be used to send traps.

Reader's Notes

7 EMS-Based Management

AudioCodes Element Management System (EMS) is an advanced solution for standards-based management of E-SBCs within VoP networks, covering all areas vital for the efficient operation, administration, management and provisioning (OAM&P) of AudioCodes' families of E-SBCs. The EMS enables Network Equipment Providers (NEPs) and System Integrators (SIs) the ability to offer customers rapid time-to-market and inclusive, cost-effective management of next-generation networks. The standards-compliant EMS uses distributed SNMP-based management software, optimized to support day-to-day Network Operation Center (NOC) activities, offering a feature-rich management framework. It supports fault management, configuration and security.



Note: For more information on using the EMS tool, refer to the *EMS User's Manual* and *EMS Server IOM Manual*.

Reader's Notes

8 INI File-Based Management

The device can be configured using an ini file, which is a text-based file with an *ini* file extension name that can be created using any standard text-based editor such as Notepad. Each configuration element of the device has a corresponding ini file parameter that you can use in the ini file for configuring the device. When you have created the ini file with your ini file parameter settings, you apply these settings to the device by installing (loading) the ini file to the device.



Notes:

- For a list and description of the *ini* file parameters, see 'Configuration Parameters Reference' on page 437.
- To restore the device to default settings using the *ini* file, see 'Restoring Factory Defaults' on page 379.

8.1 INI File Format

The *ini* file can be configured with any number of parameters. These *ini* file parameters can be one of the following types:

- Individual parameters - see 'Configuring Individual ini File Parameters' on page 79
- Table parameters - see 'Configuring Table ini File Parameters' on page 79

8.1.1 Configuring Individual ini File Parameters

The syntax for configuring individual *ini* file parameters in the ini file is as follows:

- An optional, subsection name (or group name) enclosed in square brackets "[...]". **This** is used to conveniently group similar parameters by their functionality.
- Parameter name, followed by an equal "=" sign and then its value.
- Comments must be preceded by a semicolon ";".

```
[subsection name]
parameter name = value
parameter name = value
; this is a comment line
; for example:
[System Parameters]
SyslogServerIP = 10.13.2.69
EnableSyslog = 1
```

For general *ini* file formatting rules, see 'General ini File Formatting Rules' on page 81.

8.1.2 Configuring Table ini File Parameters

The table ini file parameters allow you to configure tables, which include multiple parameters (*columns*) and row entries (*indices*). When loading an *ini* file to the device, it's recommended to include only tables that belong to applications that are to be configured (dynamic tables of other applications are empty, but static tables are not).

The table ini file parameter is composed of the following elements:

- **Title of the table:** The name of the table in square brackets, e.g., [MY_TABLE_NAME].
- **Format line:** Specifies the columns of the table (by their string names) that are to be

configured.

- The first word of the Format line must be "FORMAT", followed by the Index field name and then an equal "=" sign. After the equal sign, the names of the columns are listed.
- Columns must be separated by a comma ",".
- The Format line must only include columns that can be modified (i.e., parameters that are not specified as read-only). An exception is Index fields, which are mandatory.
- The Format line must end with a semicolon ";".
- **Data line(s):** Contain the actual values of the columns (parameters). The values are interpreted according to the Format line.
 - The first word of the Data line must be the table's string name followed by the Index field.
 - Columns must be separated by a comma ",".
 - A Data line must end with a semicolon ";".
- **End-of-Table Mark:** Indicates the end of the table. The same string used for the table's title, preceded by a backslash "\", e.g., `[MY_TABLE_NAME]`.

The following displays an example of the structure of a table ini file parameter.

```
[Table_Title]
; This is the title of the table.
FORMAT Index = Column_Name1, Column_Name2, Column_Name3;
; This is the Format line.
Index 0 = value1, value2, value3;
Index 1 = value1, $$, value3;
; These are the Data lines.
[\\Table_Title]
; This is the end-of-the-table-mark.
```

The table ini file parameter formatting rules are listed below:

- Indices (in both the Format and the Data lines) must appear in the same order. The Index field must never be omitted.
- The Format line can include a subset of the configurable fields in a table. In this case, all other fields are assigned with the pre-defined default values for each configured line.
- The order of the fields in the Format line isn't significant (as opposed to the Index fields). The fields in the Data lines are interpreted according to the order specified in the Format line.
- The double dollar sign (\$\$) in a Data line indicates the default value for the parameter.
- The order of the Data lines is insignificant.
- Data lines must match the Format line, i.e., it must contain exactly the same number of Indices and Data fields and must be in exactly the same order.
- A row in a table is identified by its table name and Index field. Each such row may appear only once in the *ini* file.
- Table dependencies: Certain tables may depend on other tables. For example, one table may include a field that specifies an entry in another table. This method is used to specify additional attributes of an entity, or to specify that a given entity is part of a larger entity. The tables must appear in the order of their dependency (i.e., if Table X is referred to by Table Y, Table X must appear in the *ini* file before Table Y).

For general *ini* file formatting rules, see 'General ini File Formatting Rules' on page 81.

The table below displays an example of a table ini file parameter:

```
[ CodersGroup0 ]
FORMAT CodersGroup0_Index = CodersGroup0_Name, CodersGroup0_pTime,
CodersGroup0_rate, CodersGroup0_PayloadType, CodersGroup0_Sce;
```



```
CodersGroup0 0 = g711Alaw64k, 20, 0, 255, 0;  
CodersGroup0 1 = eg711Ulaw, 10, 0, 71, 0;  
[ \CodersGroup0 ]
```



Note: Do not include read-only parameters in the table ini file parameter as this can cause an error when attempting to load the file to the device.

8.1.3 General ini File Formatting Rules

The *ini* file must adhere to the following formatting rules:

- The *ini* file name must not include hyphens "-" or spaces; if necessary, use an underscore "_" instead.
- Lines beginning with a semi-colon ";" are ignored. These can be used for adding remarks in the *ini* file.
- A carriage return (i.e., Enter) must be done at the end of each line.
- The number of spaces before and after the equals sign "=" is irrelevant.
- Subsection names for grouping parameters are optional.
- If there is a syntax error in the parameter name, the value is ignored.
- Syntax errors in the parameter's value can cause unexpected errors (parameters may be set to the incorrect values).
- Parameter string values that denote file names (e.g., CallProgressTonesFileName) must be enclosed with inverted commas, e.g., CallProgressTonesFileName = 'cpt_usa.dat'.
- The parameter name is not case-sensitive.
- The parameter value is not case-sensitive, except for coder names.
- The *ini* file must end with at least one carriage return.

8.2 Loading an ini File

You can load an *ini* file to the device using the following methods:

- Web interface, using any of the following pages:
 - Configuration File - see 'Backing Up and Loading Configuration File' on page [377](#)
 - Load Auxiliary Files - see 'Loading Auxiliary Files' on page [363](#)

When loaded to the device, the configuration settings of the *ini* file are saved to the device's non-volatile memory. If a parameter is not included in the loaded *ini* file, the following occurs:

- Using the Load Auxiliary Files page: Current settings for parameters that were not included in the loaded ini file are retained.
- All other methods: The default is assigned to the parameters that were not included in the loaded ini file and thereby, overriding values previously configured for these parameters.


Notes:

- For a list and description of the *ini* file parameters, see 'Configuration Parameters Reference' on page 437.
- Some parameters are configurable only through the *ini* file (and not the Web interface).
- To restore the device to default settings using the *ini* file, see 'Restoring Factory Defaults' on page 379.

8.3 Modifying an ini File

You can modify an *ini* file currently used by the device. Modifying an *ini* file instead of loading an entirely new *ini* file preserves the device's current configuration.

➤ **To modify an *ini* file:**

1. Save the device's configuration as an *ini* file on your computer, using the Web interface (see 'Loading an ini File' on page 81).
2. Open the *ini* file using a text file editor such as Notepad, and then modify the *ini* file parameters as required.
3. Save the modified *ini* file, and then close the file.
4. Load the modified *ini* file to the device (see 'Loading an ini File' on page 81).



Tip: Before loading the *ini* file to the device, verify that the file extension of the file is *.ini*.

8.4 Secured Encoded ini File

The *ini* file contains sensitive information that is required for the functioning of the device. The file may be loaded to the device using HTTP. These protocols are not secure and are vulnerable to potential hackers. To overcome this security threat, the AudioCodes DConvert utility allows you to binary-encode (encrypt) the *ini* file before loading it to the device. For more information, refer to *DConvert Utility User's Guide*.


Notes:

- The procedure for loading an encoded *ini* file is identical to the procedure for loading an unencoded *ini* file (see 'Loading an ini File' on page 81).
- If you download from the device (to a folder on your computer) an *ini* file that was loaded encoded to the device, the file is saved as a regular *ini* file (i.e., unencoded).

Part III

General System Settings

9 Configuring Certificates

The Certificates page allows you to configure X.509 certificates, which are used for secure management of the device, secure SIP transactions, and other security applications.



Note: The device is shipped with an active TLS setup. Thus, configure certificates only if required.

9.1 Replacing the Device's Certificate

The device is supplied with a working TLS configuration consisting of a unique self-signed server certificate. If an organizational Public Key Infrastructure (PKI) is used, you may wish to replace this certificate with one provided by your security administrator.

➤ **To replace the device's certificate:**

1. Your network administrator should allocate a unique DNS name for the device (e.g., dns_name.corp.customer.com). This DNS name is used to access the device and therefore, must be listed in the server certificate.
2. If the device is operating in HTTPS mode, then set the 'Secured Web Connection (HTTPS)' parameter (HTTPSOnly) to **HTTP and HTTPS** (see 'Configuring Web Security Settings' on page 58). This ensures that you have a method for accessing the device in case the new certificate does not work. Restore the previous setting after testing the configuration.
3. Open the Certificates page (**Configuration** tab > **System** menu > **Certificates**).
4. Under the **Certificate Signing Request** group, do the following:
 - a. In the 'Subject Name [CN]' field, enter the DNS name.
 - b. Fill in the rest of the request fields according to your security provider's instructions.
 - c. Click the **Create CSR** button; a textual certificate signing request is displayed in the area below the button:

Figure 9-1: Certificate Signing Request Group

Certificate Signing Request	
Subject Name [CN]	audio.com
Organizational Unit [OU] (optional)	Headquarters
Company name [O] (optional)	Corporate
Locality or city name [L] (optional)	Poughkeepsie
State [ST] (optional)	New York
Country code [C] (optional)	US

After creating the CSR, copy the text below (including the BEGIN/END lines) and send it to your Certification Authority for signing.

```

-----BEGIN CERTIFICATE REQUEST-----
MIIBtjCCAR8CAQAwZjESMBAGA1UEAxMJKXVXkaW8uYzI9MRUwEwYDVQQLEwkiZWFK
cXVhcnRlcnMxZjA2QjNVBAoTCU9vcnBvcml0ZTEVMBMGA1UEBhMMXUg91Z2hrZWVw
c2llMRcwYwYDVQQIEWhozXcgW9yazELMAkGA1UEBhMCVVMwZ28wDQYJKoZIhvcN
AQEBBQADgY0AMIGJAoGBAPhpF2t4cLy3FRk5Bw7f1ZFWXQ7nvuocHtu7Nns071M
xL7Of8Yol63eeIK2eDo8nm6rJO677z/AHWUmf65pAK1CbO2PgOZNS0g6+5JAmJAA
1LnUnaoqjEsK7CF32uvolH//g/Fkhy5z1eNvobi+25Pn3JaJzEXc8DkgwZ19rRoqRZ
AgMBAAQgADANBgkqhkiG9w0BAQgFAAQBQgZDihdqbc1zkHdLFr+5BR3s8YgUX6MB
q7FgqFXA2zk1MgnBMc/MYfSGTBawrQ7P76dNj60DvmuCPf66zi5m2uq6Lqo1i
nLnqVPCmbdva/B1QyEpPbqhZgPULJ8CSesrrY3ru23A2eDUBvYh090IKRnAp//+3
ZvNZze5M5CBSLg==
-----END CERTIFICATE REQUEST-----

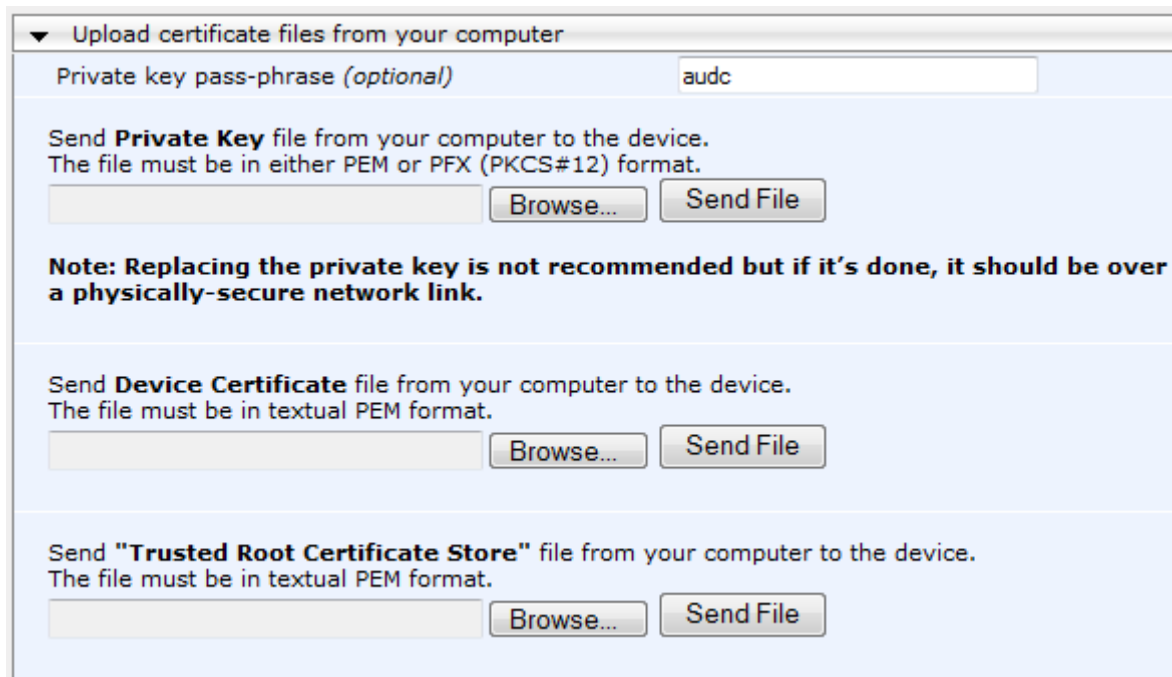
```


load them over a physically-secure connection such as a back-to-back Ethernet cable connected directly to the managing computer.

➤ **To replace the device's private key:**

1. Your security administrator should provide you with a private key in either textual PEM (PKCS #7) or PFX (PKCS #12) format. The file may be encrypted with a short pass-phrase, which should be provided by your security administrator.
2. If the device is operating in HTTPS mode, then set the 'Secured Web Connection (HTTPS)' field (HTTPSOnly) to **HTTP and HTTPS** (see 'Configuring Web Security Settings' on page 58). This ensures that you have a method for accessing the device in case the new configuration does not work. Restore the previous setting after testing the configuration.
3. Open the Certificates page (**Configuration** tab > **System** menu > **Certificates**) and scroll down to the **Upload certificate files from your computer** group.

Figure 9-3: Upload Certificate Files from your Computer Group



4. Fill in the 'Private key pass-phrase' field, if required.
5. Click the **Browse** button corresponding to the 'Send Private Key' field, navigate to the key file, and then click **Send File**.
6. If the security administrator has provided you with a device certificate file, load it using the 'Send Device Certificate' field.
7. After the files successfully load to the device, save the configuration with a device reset (see 'Saving Configuration' on page 360); the Web interface uses the new configuration.
8. Open the Certificates page again, and verify that under the **Certificate information** group (at the top of the page) the 'Private key' read-only field displays "OK"; otherwise, consult your security administrator.
9. If the device was originally operating in HTTPS mode and you disabled it in Step 2, then enable it by setting the 'Secured Web Connection (HTTPS)' field to **HTTPS Only**.

9.3 Mutual TLS Authentication

By default, servers using TLS provide one-way authentication. The client is certain that the identity of the server is authentic. When an organizational PKI is used, two-way authentication may be desired - both client and server should be authenticated using X.509 certificates. This is achieved by installing a client certificate on the managing PC and loading the root CA's certificate to the device's Trusted Root Certificate Store. The Trusted Root Certificate file may contain more than one CA certificate combined, using a text editor.

Since X.509 certificates have an expiration date and time, the device must be configured to use NTP (see 'Simple Network Time Protocol Support' on page 91) to obtain the current date and time. Without the correct date and time, client certificates cannot work.

➤ To enable mutual TLS authentication for HTTPS:

1. Set the 'Secured Web Connection (HTTPS)' field to **HTTPS Only** (see 'Configuring Web Security Settings' on page 58) to ensure you have a method for accessing the device in case the client certificate does not work. Restore the previous setting after testing the configuration.
2. Open the Certificates page (see 'Replacing the Device's Certificate' on page 85).
3. In the **Upload certificate files from your computer** group, click the **Browse** button corresponding to the 'Send Trusted Root Certificate Store ...' field, navigate to the file, and then click **Send File**.
4. When the operation is complete, set the 'Requires Client Certificates for HTTPS connection' field to **Enable** (see 'Configuring Web Security Settings' on page 58).
5. Save the configuration with a device reset (see 'Saving Configuration' on page 360).

When a user connects to the secured Web interface of the device:

- If the user has a client certificate from a CA that is listed in the Trusted Root Certificate file, the connection is accepted and the user is prompted for the system password.
- If both the CA certificate and the client certificate appear in the Trusted Root Certificate file, the user is not prompted for a password (thus, providing a single-sign-on experience - the authentication is performed using the X.509 digital signature).
- If the user does not have a client certificate from a listed CA or does not have a client certificate, the connection is rejected.



Notes:

- The process of installing a client certificate on your PC is beyond the scope of this document. For more information, refer to your operating system documentation, and/or consult your security administrator.
- The root certificate can also be loaded via the Automatic Update facility, using the HTTPSRootFileName *ini* file parameter.
- You can enable the device to check whether a peer's certificate has been revoked by an Online Certificate Status Protocol (OCSP) server (see Configuring Certificate Revocation Checking (OCSP) on page 89).

9.4 Configuring Certificate Revocation Checking (OCSP)

Some Public-Key Infrastructures (PKI) can revoke a certificate after it has been issued. You can configure the device to check whether a peer's certificate has been revoked, using the Online Certificate Status Protocol (OCSP). When OCSP is enabled, the device queries the OCSP server for revocation information whenever a peer certificate is received (IPSec, TLS client mode, or TLS server mode with mutual authentication).

➤ **To configure OCSP:**

1. Open the General Security Settings page (**Configuration** tab > **VoIP** menu > **Security** > **General Security Settings**).

Figure 9-4: OCSP Parameters

OCSP Settings	
Enable OCSP Server	Enable
Primary Server IP	212.10.5.6
Secondary Server IP	0.0.0.0
Server Port	2560
Default Response When Server Unreachable	Reject

2. Configure the OCSP parameters as required. For a description of these parameters, see OCSP Parameters on page 465.
3. Click **Submit**.



Notes:

- The device does not query OCSP for its own certificate.
- Some PKIs do not support OCSP but generate Certificate Revocation Lists (CRLs). For such cases, set up an OCSP server such as OCSPD.

9.5 Self-Signed Certificates

The device is shipped with an operational, self-signed server certificate. The subject name for this default certificate is 'ACL_nnnnnnn', where *nnnnnnn* denotes the serial number of the device. However, this subject name may not be appropriate for production and can be changed while still using self-signed certificates.

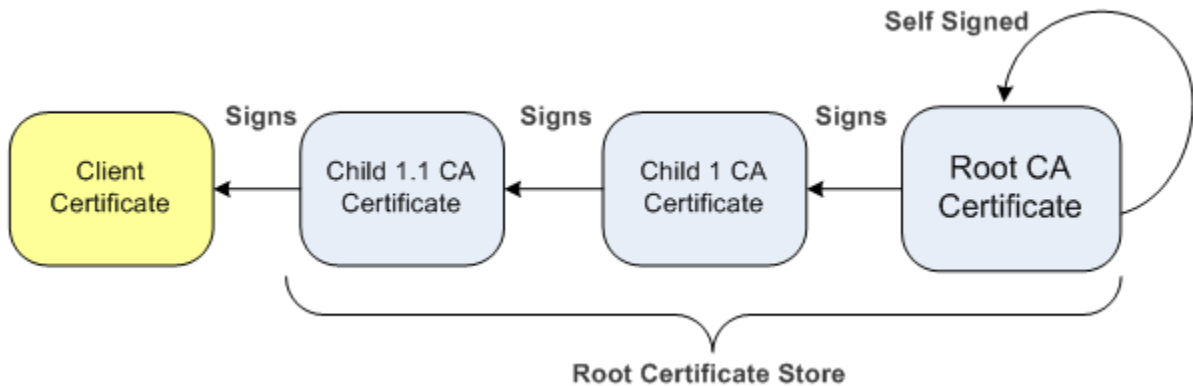
➤ **To change the subject name and regenerate the self-signed certificate:**

1. Before you begin, ensure the following:
 - You have a unique DNS name for the device (e.g., dns_name.corp.customer.com). This name is used to access the device and should therefore, be listed in the server certificate.
 - No traffic is running on the device. The certificate generation process is disruptive to traffic and should be executed during maintenance time.
2. Open the Certificates page (see 'Replacing the Device's Certificate' on page 85).
3. In the 'Subject Name **[CN]**' field, enter the fully-qualified DNS name (FQDN) as the certificate subject, select the desired private key size (in bits), and then click **Generate self-signed**; after a few seconds, a message appears displaying the new subject name.
4. Save the configuration with a device reset (see 'Saving Configuration' on page 360) for the new certificate to take effect.

9.6 Loading Certificate Chain for Trusted Root

A certificate chain is a sequence of certificates where each certificate in the chain is signed by the subsequent certificate. The last certificate in the list of certificates is the Root CA certificate, which is self-signed. The purpose of a certificate chain is to establish a chain of trust from a child certificate to the trusted root CA certificate. The CA vouches for the identity of the child certificate by signing it. A client certificate is considered trusted if one of the CA certificates up the certificate chain is found in the server certificate directory.

Figure 9-5: Certificate Chain Hierarchy



For the device to trust a whole chain of certificates, you need to combine the certificates into one text file (using a text editor). Once done, upload the file using the 'Trusted Root Certificate Store' field in the Certificates page.



Notes: The maximum supported size of the combined file of trusted chain of certificates is 100,000 bytes (including the certificate's headers).

10 Date and Time

The date and time of the device can be configured manually or it can be obtained automatically from a Simple Network Time Protocol (SNTP) server.

10.1 Configuring Date and Time Manually

The date and time of the device can be configured manually.

➤ **To manually configure the device's date and time, using the Web interface:**

1. Open the Regional Settings page (**Configuration** tab > **System** menu > **Regional Settings**).

Figure 10-1: Regional Settings Page

Year	Month	Day	Hour	Minutes	Seconds
2010	2	4	10	21	46

2. Enter the current date and time of the geographical location in which the device is installed.
3. Click the **Submit** button.



Notes:

- If the device is configured to obtain the date and time from an SNTP server, the fields on this page are read-only, displaying the received date and time.
- After performing a hardware reset, the date and time are returned to their defaults and thus, should be updated.

10.2 Automatic Date and Time through SNTP Server

The Simple Network Time Protocol (SNTP) client functionality generates requests and reacts to the resulting responses using the NTP version 3 protocol definitions (according to RFC 1305). Through these requests and responses, the NTP client synchronizes the system time to a time source within the network, thereby eliminating any potential issues should the local system clock 'drift' during operation. By synchronizing time to a network time source, traffic handling, maintenance, and debugging become simplified for the network administrator.

The NTP client follows a simple process in managing system time: the NTP client requests an NTP update, receives an NTP response, and then updates the local system clock based on a configured NTP server within the network.

The client requests a time update from a specified NTP server at a specified update interval. In most situations, this update interval is every 24 hours based on when the system was restarted. The NTP server identity (as an IP address or FQDN) and the update interval are user-defined, or an SNMP MIB object.

When the client receives a response to its request from the identified NTP server, it must be interpreted based on time zone or location offset that the system is to a standard point of reference called the Universal Time Coordinate (UTC). The time offset that the NTP client uses is configurable.

If required, the clock update is performed by the client as the final step of the update process. The update is performed in such a way as to be transparent to the end users. For instance, the response of the server may indicate that the clock is running too fast on the client. The client slowly robs bits from the clock counter to update the clock to the correct

time. If the clock is running too slow, then in an effort to catch the clock up, bits are added to the counter, causing the clock to update quicker and catch up to the correct time. The advantage of this method is that it does not introduce any disparity in the system time that is noticeable to an end user or that could corrupt call timeouts and timestamps.

The procedure below describes how to configure SNTP.

➤ **To configure SNTP using the Web interface:**

1. Open the Application Settings page (**Configuration** tab > **System** menu > **Application Settings**).

Figure 10-2: SNTP Configuration in Application Settings Page

▼ NTP Settings	
NTP Server DN/IP	212.13.4.5
NTP UTC Offset	Hours: 0 Minutes: 0
NTP Updated Interval	Hours: 24 Minutes: 0
NTP Secondary Server IP	
▼ Day Light Saving Time	
Day Light Saving Time	Enable
DST Mode	Day of month
Start Time	Sep 02 0 : 0
End Time	Apr 07 0 : 0
Offset [min]	60
Day of Month Start	Sep Sunday First 0 : 0
Day of Month End	Apr Sunday First 0 : 0

2. Configure the NTP parameters:
 - 'NTP Server DN/IP' (NTPServerIP) - defines the IP address or FQDN of the NTP server.
 - 'NTP UTC Offset' (NTPServerUTCOffset) - defines the time offset in relation to the UTC. For example, if your region is 2 hours ahead of the UTC, enter "2".
 - 'NTP Updated Interval' (NTPUpdateInterval) - defines the period after which the date and time of the device is updated.
 - 'NTP Secondary Server IP' (NTPSecondaryServerIP) - defines the secondary NTP server.
3. Configure daylight saving, if required:
 - 'Day Light Saving Time' (DayLightSavingTimeEnable) - enables daylight saving time.
 - 'DST Mode' - Determines the range type for configuring the start and end date for daylight saving:
 - ◆ **Day of Year:** The range is configured by date of month, for example, from January 4 to August 31.
 - ◆ **Day of month:** The range is configured by day of month, for example, from the second Sunday of May January to the last Sunday of August.
 - 'Start Time' (DayLightSavingTimeStart) and 'End Time' (DayLightSavingTimeEnd) - defines the period for which daylight saving time is relevant.
 - 'Offset' (DayLightSavingTimeOffset) - defines the offset in minutes to add to the time for daylight saving. For example, if your region has daylight saving of one hour, the time received from the NTP server is 11:00, and the UTC offset for your region is +2 (i.e., 13:00), you need to enter "60" to change the local time to 14:00.
4. Verify that the device is set to the correct date and time. You can do this by viewing the date and time in the Regional Settings page, as described in 'Configuring Date and Time Manually' on page 91.

Part IV

General VoIP Configuration

11 Network

This section describes the network-related configuration.

11.1 Configuring Physical Ethernet Ports

The device's physical Ethernet ports are grouped into pairs (termed *Group Members*), where each group consists of an active port and a standby port. This provides Ethernet port redundancy within a group, whereby if an active port disconnects the device switches over to the standby port. These port groups can be assigned to IP network interfaces in the Multiple Interface table (see 'Configuring IP Network Interfaces' on page 98). This enables physical separation of network interfaces, providing a higher level of segregation of sub-networks. Equipment connected to different physical ports is not accessible to one another. The only connection between them can be established by cross connecting them with media streams (a VoIP calls).

For each Ethernet port, you can configure the speed, duplex mode, native VLAN (PVID), and provide a brief description. Up to four port-pair redundancy groups are supported.



Notes:

- To configure the transmit (Tx) and receive (Rx) settings of each port group, see 'Configuring Tx/Rx for Ethernet Port-Pair Groups' on page 97.
- The Ethernet ports can also be configured using the table ini file parameter, PhysicalPortsTable.

➤ To configure the physical Ethernet ports:

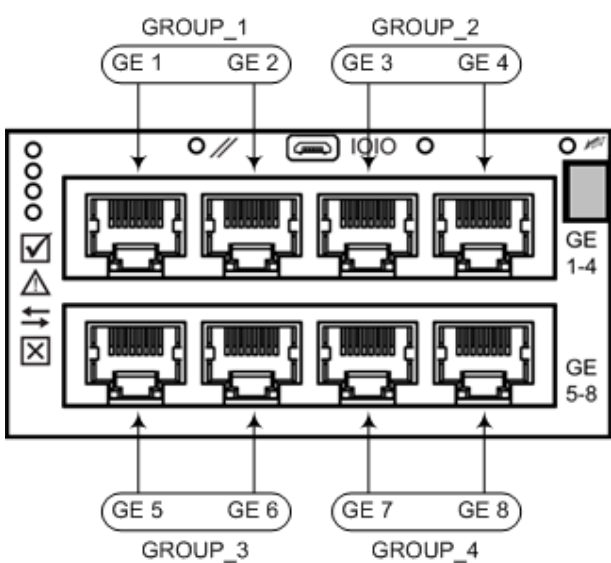
1. Open the Physical Ports Settings page (**Configuration** tab > **VoIP** menu > **Network** submenu > **Physical Ports Settings**).

Figure 11-1: Physical Ports Settings Page

Index	Port	Mode	Native Vlan	Speed&Duplex	Description	Group Member	Group Status
1 <input type="radio"/>	GE_1	Enable	1	Auto Negotiation	User Port #0	GROUP_1	Active
2 <input type="radio"/>	GE_2	Enable	1	Auto Negotiation	User Port #1	GROUP_1	Redundant
3 <input type="radio"/>	GE_3	Enable	2	Auto Negotiation	User Port #2	GROUP_2	Active
4 <input type="radio"/>	GE_4	Enable	2	Auto Negotiation	User Port #3	GROUP_2	Redundant
5 <input type="radio"/>	GE_5	Enable	1	Auto Negotiation	User Port #4	GROUP_3	Active
6 <input type="radio"/>	GE_6	Enable	1	Auto Negotiation	User Port #5	GROUP_3	Redundant
7 <input type="radio"/>	GE_7	Enable	1	Auto Negotiation	User Port #6	GROUP_4	Active
8 <input type="radio"/>	GE_8	Enable	1	Auto Negotiation	User Port #7	GROUP_4	Redundant

2. Select the 'Index' radio button corresponding to the port that you want to configure.
3. Click the **Edit** button.
4. Configure the ports (see the table below for a description of the parameters).
5. Click **Apply**.

Table 11-1: Physical Port Settings Parameters Description

Parameter	Description
Port [PhysicalPortsTable_Port]	<p>(Read-only) Displays the port number. The string values displayed on the Web page represent the physical ports, as shown below:</p> 
Mode [PhysicalPortsTable_Mode]	<p>(Read-only field) Displays the mode of the port:</p> <ul style="list-style-type: none"> [0] Disable [1] Enable (default)
Native Vlan [PhysicalPortsTable_NativeVlan]	<p>Defines the Native VLAN or PVID of the port. Incoming packets without a VLAN ID are tagged with this VLAN. For outgoing packets, if the VLAN ID as defined in the Multiple Interface table is the same as the Native VLAN ID, the device sends the packet without a VLAN; otherwise, the VLAN ID as defined in the Multiple Interface table takes precedence.</p> <p>The valid value range is 1 to 4096. The default is 1.</p>
Speed & Duplex [PhysicalPortsTable_SpeedDuplex]	<p>Defines the speed and duplex mode of the port.</p> <ul style="list-style-type: none"> [0] 10BaseT Half Duplex [1] 10BaseT Full Duplex [2] 100BaseT Half Duplex [3] 100BaseT Full Duplex [4] Auto Negotiation (default) [6] 1000BaseT Half Duplex [7] 1000BaseT Full Duplex
Description [PhysicalPortsTable_PortDescription]	<p>Defines an arbitrary description of the port.</p>
Group Member [PhysicalPortsTable_GroupMember]	<p>(Read-only field) Displays the group to which the port belongs.</p>
Group Status [PhysicalPortsTable_GroupStatus]	<p>(Read-only) Displays the status of the port:</p> <ul style="list-style-type: none"> "Active" - the active port "Redundant" - the standby (redundant) port

11.2 Configuring Tx/Rx for Ethernet Port-Pair Groups

The Ethernet Group Settings table allows you to configure the transmit (Tx) and receive (Rx) settings for the physical ports belonging to a port-pair group for 1+1 physical port redundancy.



Notes:

- The number of displayed Ethernet Port Groups depends on the ordered hardware configuration of your device.
- The displayed port name on the Web page represents the physical port on the device. For the mapping of these strings to the physical ports, see Configuring Physical Ethernet Ports on page 95.
- The Ethernet Group Settings table can also be configured using the table ini file parameter, EtherGroupTable.

➤ To configure the Ethernet port-pair groups:

1. Open the Ethernet Group Settings page (**Configuration** tab > **VoIP** menu > **Network** submenu > **Ethernet Group Settings**).

Figure 11-2: Ethernet Group Settings Page

Index	Group	Mode	Member 1	Member 2
1	<input checked="" type="radio"/> GROUP_1	2RX 1TX ▼	GE_3_1 ▼	GE_3_2 ▼
2	<input type="radio"/> GROUP_2	2RX 1TX	GE_3_3	GE_3_4
3	<input type="radio"/> GROUP_3	2RX 2TX	GE_4_1	GE_4_2
4	<input type="radio"/> GROUP_4	1RX 1TX	GE_4_3	GE_4_4

2. Select the 'Index' radio button corresponding to the port group that you want to configure.
3. Click the **Edit** button.
4. Configure the ports as required. For a description of the parameters, see the table below.
5. Click **Apply**.
6. Save your settings to flash memory with a device reset (see 'Saving Configuration' on page 360).

Table 11-2: Ethernet Group Settings Parameters Description

Parameter	Description
Group [EtherGroupTable_Group]	(Read-only) Displays the Ethernet port-pair group number.
Mode [EtherGroupTable_Mode]	Defines the mode of operation of the ports in the group: <ul style="list-style-type: none"> ▪ [2] 1RX/1TX = (Default) At any given time, only a single port in the group can transmit and receive packets. If a link exists on both ports, then the active one is either the first to have a link up or the lower-numbered port if both have the same link up from start. ▪ [3] 2RX/1TX = Both ports in the group can receive packets, but only one port can transmit. The transmitting port is determined

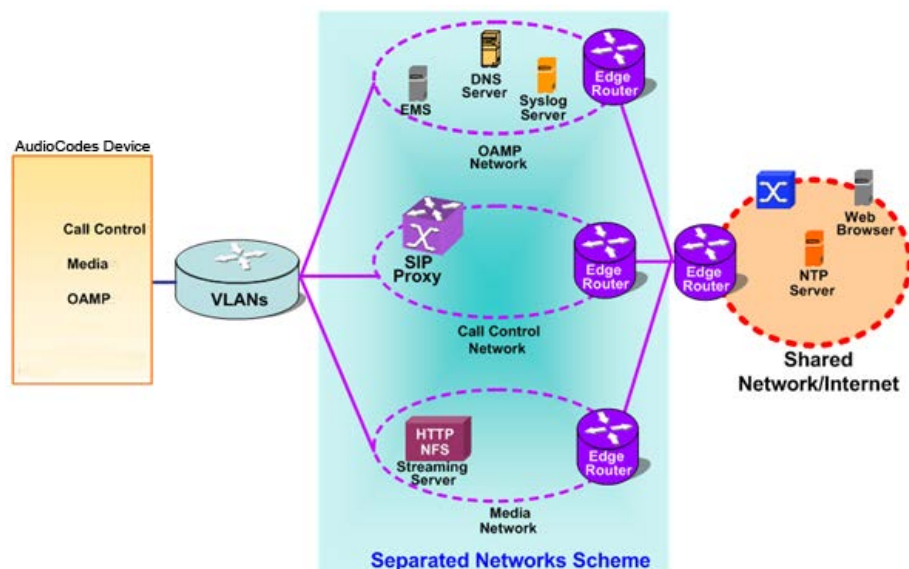
Parameter	Description
	<p>arbitrarily by the device. If the selected port fails at a later stage, a switchover to the redundant port is done, which begins to transmit as well as receive.</p> <ul style="list-style-type: none"> ▪ [4] 2RX/2TX = Both ports in the group can receive and transmit packets. <p>Notes:</p> <ul style="list-style-type: none"> ▪ It is recommended to use the 2RX/1TX option. In such a setup, the ports can be connected to the same LAN switch or each to a different switch where both are in the same subnet. If connecting each port to a different switch, the 2RX/2TX option can be used but only if the port group is associated with OAMP and/or Control application types, not media. ▪ For Ethernet port settings and connections of the Maintenance interface when implementing High Availability, see Initial HA Configuration on page 343.
Member 1 [EtherGroupTable_Member1]	(Read-only) Displays the first port in the Ethernet Group.
Member 2 [EtherGroupTable_Member2]	(Read-only) Displays the second port in the Ethernet Group.

11.3 Configuring IP Network Interfaces

You can configure a single VoIP network interface for all applications, which includes OAMP (management traffic), call control (SIP messages), and media (RTP traffic), or you can configure multiple logical, IP network interfaces for these applications. A need often arises to have logically separated network segments for these applications for administration and security. This can be achieved by employing Layer-2 VLANs and Layer-3 subnets.

The figure below illustrates a typical network architecture where the device is configured with three network interfaces for the OAMP, call control, and media applications. The device is connected to a VLAN-aware switch for directing traffic from and to the device to the three separated Layer-3 broadcast domains according to VLAN tags (middle pane).

Figure 11-3: Multiple Network Interfaces



The Multiple Interface Table page allows you to configure these network interfaces. Each row of the table defines a logical IP interface with the following attributes:

- Application type allowed on the interface:
 - Control - call control signaling traffic (i.e., SIP)
 - Media - RTP traffic
 - Operations, Administration, Maintenance and Provisioning (OAMP) - management (such as Web- and SNMP-based management)
 - Maintenance - Maintenance interface used in High Availability (HA) mode when two devices are deployed together for redundancy - this interface represents one of the LAN interfaces or Ethernet groups on each device used for the Ethernet connectivity between the two devices
- IP address and subnet mask represented by prefix length (IPv4 and IPv6 Internet Layer protocols)
- VLAN ID
- Default Gateway - traffic from this interface destined to a subnet that does not meet any of the routing rules, local or static routes, are forwarded to this gateway
- Primary and secondary DNS IP address (optional)
- Associated physical Ethernet port group (Underlying Device) used for the interface - useful for setting trusted and un-trusted networks on different physical ports

You can configure up to 48 interfaces, consisting of up to 47 Control and Media interfaces, including a Maintenance interface if implementing an HA system, and 1 OAMP interface.

The default VoIP interface is as follows:

- Application type: OAMP + Media + Control
- IP address: 192.168.0.2 with prefix length 24 (i.e., subnet mask 255.255.255.0)
- Default gateway: 192.168.0.1
- Name: "Voice"
- VLAN ID: 1

For configuring Quality of Service (QoS), see 'Configuring the QoS Settings' on page 111.

Complementing the Multiple Interface table is the IP Routing table, which allows you to define static routing rules for non-local hosts/subnets. For more information, see 'Configuring the IP Routing Table' on page 108.



Notes:

- For more information on HA and configuring the Maintenance interface, see Configuring High Availability on page 337.
- To configure firewall rules (access list) to allow or block packets received from specific IP network interfaces, see 'Configuring Firewall Settings' on page 125.
- The Multiple Interface table can also be configured using the table ini file parameter, InterfaceTable (see 'Networking Parameters' on page 437).

➤ **To configure IP network interfaces:**

1. Open the Multiple Interface Table page (**Configuration** tab > **VoIP** menu > **Network** submenu > **IP Settings**).

Figure 11-4: Multiple Interface Table

Index	Application Type	Interface Mode	IP Address	Prefix Length	Gateway	VLAN ID	Interface Name	Primary DNS Server IP Address	Secondary DNS Server IP Address	Underlying
0	OAMP + Media + Control	IPv4 Manual	10.8.244.80	16	10.8.0.1	1	Voice	0.0.0.0	0.0.0.0	GROUP_1
1	MAINTENANCE	IPv4 Manual	10.3.0.51	16	10.3.0.1	2	Unknown	0.0.0.0	0.0.0.0	GROUP_2

2. In the 'Add Index' field, enter the desired index number for the new interface, and then click **Add Index**; the index row is added to the table.
3. Configure the interface according to the table below.
4. Click the **Apply** button; the interface is added to the table and the **Done** button appears.
5. Click **Done** to validate the interface. If the interface is not valid (e.g., if it overlaps with another interface in the table or if it does not adhere to the other rules as summarized in 'Multiple Interface Table Configuration Summary and Guidelines' on page 102), a warning message is displayed.
6. Save the changes to flash memory and reset the device (see 'Saving Configuration' on page 360).


To view configured network interfaces that are currently active, click the IP Interface Status Table  button. For more information, see Viewing Active IP Interfaces on page 397.

Table 11-3: Multiple Interface Table Parameters Description

Parameter	Description
Table parameters	
Index [InterfaceTable_Index]	Table index row of the interface. The range is 0 to 47.
Web: Application Type EMS: Application Types [InterfaceTable_ApplicationTypes]	<p>Defines the applications allowed on the interface.</p> <ul style="list-style-type: none"> ▪ [0] OAMP = Operations, Administration, Maintenance and Provisioning (OAMP) applications (e.g., Web, Telnet, SSH, and SNMP). ▪ [1] Media = Media (i.e., RTP streams of voice). ▪ [2] Control = Call Control applications (e.g., SIP). ▪ [3] OAMP + Media = OAMP and Media applications. ▪ [4] OAMP + Control = OAMP and Call Control applications. ▪ [5] Media + Control = Media and Call Control applications. ▪ [6] OAMP + Media + Control = All application types are allowed on the interface. ▪ [99] MAINTENANCE = Only the Maintenance application for HA is allowed on this interface. <p>Note: For valid configuration, see Multiple Interface Table Configuration Rules on page 102.</p>
Web: Interface Mode [InterfaceTable_InterfaceMode]	<p>Defines the method that the interface uses to acquire its IP address.</p> <ul style="list-style-type: none"> ▪ [3] IPv6 Manual Prefix = IPv6 manual prefix IP address assignment. The IPv6 prefix (higher 64 bits) is set manually while the interface ID (the lower 64 bits) is derived from the device's MAC address.

Parameter	Description
	<ul style="list-style-type: none"> ▪ [4] IPv6 Manual = IPv6 manual IP address (128 bits) assignment. ▪ [10] IPv4 Manual = IPv4 manual IP address (32 bits) assignment. <p>Note: For valid configuration, see Multiple Interface Table Configuration Rules on page 102.</p>
Web/EMS: IP Address [InterfaceTable_IPAddress]	<p>Defines the IPv4/IPv6 IP address in dotted-decimal notation.</p> <p>Note: For valid configuration, see Multiple Interface Table Configuration Rules on page 102.</p>
Web/EMS: Prefix Length [InterfaceTable_PrefixLength]	<p>Defines the prefix length of the related IP address. This is a Classless Inter-Domain Routing (CIDR)-style representation of a dotted-decimal subnet notation. The CIDR-style representation uses a suffix indicating the number of bits which are set in the dotted-decimal format. For example, 192.168.0.0/16 is synonymous with 192.168.0.0 and subnet 255.255.0.0. This CIDR lists the number of '1' bits in the subnet mask (i.e., replaces the standard dotted-decimal representation of the subnet mask for IPv4 interfaces). For example, a subnet mask of 255.0.0.0 is represented by a prefix length of 8 (i.e., 11111111 00000000 00000000 00000000) and a subnet mask of 255.255.255.252 is represented by a prefix length of 30 (i.e., 11111111 11111111 11111111 11111100).</p> <p>The prefix length is a Classless Inter-Domain Routing (CIDR) style presentation of a dotted-decimal subnet notation. The CIDR-style presentation is the latest method for interpretation of IP addresses. Specifically, instead of using eight-bit address blocks, it uses the variable-length subnet masking technique to allow allocation on arbitrary-length prefixes.</p> <p>The prefix length for IPv4 can range from 0 to 30 and for IPv6 it must be set to 64.</p> <p>Note: For valid configuration, see Multiple Interface Table Configuration Rules on page 102.</p>
Web/EMS: Gateway [InterfaceTable_Gateway]	<p>Defines the IP address of the default gateway for the interface. When traffic is sent from this interface to an unknown destination (i.e., not in the same subnet and not defined for any static routing rule), it is forwarded to this default gateway.</p> <p>Note: For valid configuration, see Multiple Interface Table Configuration Rules on page 102.</p>
Web/EMS: VLAN ID [InterfaceTable_VlanID]	<p>Defines a VLAN ID for the interface.</p> <p>Note: For valid configuration, see Multiple Interface Table Configuration Rules on page 102.</p>
Web/EMS: Interface Name [InterfaceTable_InterfaceName]	<p>Defines a name for this interface. This name is used in various configuration tables to associate this network interface with other configuration entities such as Media Realms. It is also displayed in management interfaces (Web, CLI, and SNMP) for clarity where it has no functional use.</p> <p>The valid value is a string of up to 16 characters.</p> <p>Note: For valid configuration, see Multiple Interface Table Configuration Rules on page 102.</p>

Parameter	Description
Web/EMS: Primary DNS Server IP address [InterfaceTable_PrimaryDNSServerIPAddress]	(Optional) Defines the primary DNS server's IP address (in dotted-decimal notation), which is used for translating domain names into IP addresses for the interface. By default, no IP address is defined.
Web/EMS: Secondary DNS Server IP address [InterfaceTable_SecondaryDNSServerIPAddress]	(Optional) Defines the secondary DNS server's IP address (in dotted-decimal notation), which is used for translating domain names into IP addresses for the interface. By default, no IP address is defined.
Underlying Interface [InterfaceTable_UnderlyingInterface]	Assigns a physical Ethernet port-pair (Group Member) to this IP interface. This is useful for separating trusted networks from untrusted networks, by assigning each to different physical ports. To view the port groups and configure port settings, see Configuring Physical Ethernet Ports on page 95.

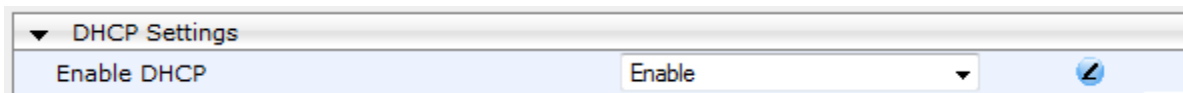
11.3.1 Booting using DHCP

You can configure the device to obtain an IP address from a DHCP server during bootup.

➤ To enable DHCP for obtaining an IP address:

1. Open the Application Settings page (**Configuration** tab > **System** menu > **Application Settings**).

Figure 11-5: Enabling DHCP - Application Settings Page



2. From the 'Enable DHCP' drop-down list, select **Enable**.
3. Click **Submit**.



Note: When using DHCP to acquire an IP address, the Multiple Interface table, VLANs and other advanced configuration options are disabled.

11.3.2 Assigning NTP Services to Application Types

You can associate the Network Time Protocol (NTP) application with the OAMP or Control application type. This is done using the EnableNTPasOAM ini file parameter.

11.3.3 Multiple Interface Table Configuration Rules

The Multiple Interface table configuration must adhere to the following rules:

- Multiple Control and Media interfaces can be configured with overlapping IP addresses and subnets.
- The prefix length replaces the dotted-decimal subnet mask presentation and must have a value of 0-30 for IPv4 addresses and a value of 64 for IPv6 addresses.
- Only one OAMP interface must be configured and this must be an IPv4 address. This OAMP interface can be combined with Media and Control.
- At least one Control interface must be configured with an IPv4 address.

- At least one Media interface must be configured with an IPv4 address.
- One or more Media and/or Control interfaces can be configured with an IPv6 address.
- The network interface types can be combined:
 - Example 1:
 - ◆ One combined OAMP-Media-Control interface with an IPv4 address
 - ◆ One combined Media-Control interface with an IPv6 address
 - Example 2:
 - ◆ One OAMP interface with an IPv4 address
 - ◆ One or more Control interfaces with IPv4 addresses
 - ◆ One or more Media interfaces with IPv4 interfaces
 - Example 3:
 - ◆ One combined OAMP-Media interface with an IPv4 address
 - ◆ One or more combined Media-Control interfaces with IPv4 addresses
 - ◆ None or additional combined Media-Control interfaces with IPv6 addresses
- Each network interface can be configured with a Default Gateway. The address of the Default Gateway must be in the same subnet as the associated interface. Additional static routing rules can be configured in the IP Routing table.
- The interface name must be configured (mandatory) and unique for each interface, and can include up to 16 characters.
- For IPv4 addresses, the 'Interface Mode' column must be set to IPv4 Manual (numeric value 10). For IPv6 addresses, the 'Interface Mode' can be set to either IPv6 Manual (numeric value 4) or IPv6 Manual Prefix (numeric value 3).
- The same VLAN ID may be shared between multiple IP interfaces, but adhering to the following guidelines:
 - The interfaces on the shared VLAN must be configured with IP addresses on the same subnet.
 - The first of these interfaces detected by the device with the VLAN ID is considered the "primary" interface.
 - Only the "primary" interface's Default Gateway is used.
 - Only the "primary" interface may be associated with a static route.
 - The interfaces on the shared VLAN must be configured with the same DNS servers.
- For network configuration to take effect, you must save the configuration to the device's flash memory (burn) with a device reset.

**Notes:**

- When configuring the network interfaces and VLANs in the Multiple Interface table using the Web interface, it is recommended to check that your configuration is valid, by clicking the Done button in the Multiple Interface Table page.
- Upon device start up, the Multiple Interface table is parsed and passes comprehensive validation tests. If any errors occur during this validation phase, the device sends an error message to the Syslog server and falls back to a "safe mode", using a single interface and no VLANs. Ensure that you view the Syslog messages that the device sends in system startup to see if any errors occurred.

11.3.4 Troubleshooting the Multiple Interface Table

If any of the Multiple Interface table guidelines are violated, the device falls back to a "safe mode" configuration, working temporarily with IP address 192.168.0.2. For more information on validation failures, consult the Syslog messages.

Validation failures may be caused by one of the following:

- One of the Application Types (OAMP, Control, or Media) are missing in the IPv4 interfaces.
- An IPv6 interface was defined with a prefix length other than 64.
- There are too many interfaces for Application Type, OAMP. There is only one interface defined, but the 'Application Types' column is not set to **OAMP + Media + Control** (numeric value 6).
- An IPv4 interface was defined with 'Interface Type' other than **IPv4 Manual** (10).
- An IPv6 interface was defined with 'Interface Type' other than IPv6 Manual (4) or IPv6 Manual Prefix (3).
- An IPv6 interface was defined with an IPv4 address in the 'Gateway' column (including "0.0.0.0").
- Two interfaces of the same address family have the same VLAN ID value.
- Two interfaces have the same name.

Apart from these validation errors, connectivity problems may be caused by one of the following:

- Trying to access the device with untagged traffic when VLANs are on and Native VLAN is not configured properly.
- The IP Routing table is not configured properly.

11.3.5 Networking Configuration Examples

This section provides configuration examples of networking interfaces.

11.3.5.1 One VoIP Interface for All Applications

This example describes the configuration of a single VoIP interface for all applications:

1. **Multiple Interface table:** Configured with a single interface for OAMP, Media and Control:

Table 11-4: Example of Single VoIP Interface in Multiple Interface Table

Index	Application Type	Interface Mode	IP Address	Prefix Length	Default	VLAN ID	Interface Name
0	OAMP, Media & Control	IPv4	192.168.0.2	16	192.168.0.1	1	myInterface

2. **IP Routing table:** Two routes are configured for directing traffic for subnet 201.201.0.0/16 to 192.168.11.10, and all traffic for subnet 202.202.0.0/16 to 192.168.11.1:

Table 11-5: Example of IP Routing Table

Destination IP Address	Prefix Length	Gateway IP Address	Metric	Interface Name
201.201.0.0	16	192.168.11.10	1	-

Destination IP Address	Prefix Length	Gateway IP Address	Metric	Interface Name
202.202.0.0	16	192.168.11.1	1	-

- The NTP applications remain with their default application types.

11.3.5.2 VoIP Interface per Application Type

This example describes the configuration of three VoIP interfaces; one for each application type:

- Multiple Interface table:** Configured with three interfaces, each for a different application type, i.e., one for OAMP, one for Call Control, and one for RTP Media, and each with a different VLAN ID and default gateway:

Table 11-6: Example of VoIP Interfaces per Application Type in Multiple Interface Table

Index	Application Type	Interface Mode	IP Address	Prefix Length	Gateway	VLAN ID	Interface Name	Underlying Interface
0	OAMP	IPv4 Manual	192.168.0.2	16	192.168.0.1	1	ManagementIF	-
1	Control	IPv4 Manual	200.200.85.14	24	200.200.85.1	200	myControlIF	-
2	Media	IPv4 Manual	211.211.85.14	24	211.211.85.1	211	myMediaIF	-

- IP Routing table:** A routing rule is required to allow remote management from a host in 176.85.49.0 / 24:

Table 11-7: Example IP Routing Table

Destination IP Address	Prefix Length	Gateway IP Address	Metric	Interface Name
176.85.49.0	24	192.168.11.1	1	-

- All other parameters are set to their respective default values. The NTP application remains with its default application types.

11.3.5.3 VoIP Interfaces for Combined Application Types

This example describes the configuration of multiple interfaces for the following applications:

- One interface for the OAMP application.
- Interfaces for Call Control and Media applications, where two of them are IPv4 interfaces and one is an IPv6 interface.

1. Multiple Interface table:

Table 11-8: Example of VoIP Interfaces of Combined Application Types in Multiple Interface Table

Index	Application Type	Interface Mode	IP Address	Prefix Length	Gateway	VLAN ID	Interface Name	Underlying Interface
0	OAMP	IPv4 Manual	192.168.0.2	16	192.168.0.1	1	Mgmt	
1	Media & Control	IPv4 Manual	200.200.85.14	24	200.200.85.1	201	MediaCntrl1	
2	Media & Control	IPv4 Manual	200.200.86.14	24	200.200.86.1	202	MediaCntrl2	
3	Media & Control	IPv6 Manual	2000::1:200:200:86:14	64	::	202	V6CntrlMedia2	

- #### 2. IP Routing table:
- A routing rule is required to allow remote management from a host in 176.85.49.0/24:

Table 11-9: Example of IP Routing Table

Destination IP Address	Prefix Length	Gateway IP Address	Metric	Interface Name
176.85.49.0	24	192.168.0.10	1	-

- #### 3. The NTP application is configured (using the ini file) to serve as OAMP applications:

```
EnableNTPasOAM = 1
```

- #### 4. DiffServ table:

- Layer-2 QoS values are assigned:
 - ◆ For packets sent with DiffServ value of 46, set VLAN priority to 6
 - ◆ For packets sent with DiffServ value of 40, set VLAN priority to 6
 - ◆ For packets sent with DiffServ value of 26, set VLAN priority to 4
 - ◆ For packets sent with DiffServ value of 10, set VLAN priority to 2
- Layer-3 QoS values are assigned:
 - ◆ For Media Service class, the default DiffServ value is set to 46
 - ◆ For Control Service class, the default DiffServ value is set to 40
 - ◆ For Gold Service class, the default DiffServ value is set to 26

- ◆ For Bronze Service class, the default DiffServ value is set to 10

Figure 11-6: Example of Layer-2 QoS in DiffServ Table

Index	Differentiated Services	VLAN Priority
1 <input type="radio"/>	0	7
2 <input type="radio"/>	46	6
3 <input type="radio"/>	40	6
4 <input type="radio"/>	26	4
5 <input type="radio"/>	10	2

▼ Differentiated Services	
Media Premium QoS	46
Control Premium QoS	40
Gold QoS	26
Bronze QoS	10

11.3.5.4 VoIP Interfaces with Multiple Default Gateways

Below is a configuration example using default gateways per IP network interface. In this example, the default gateway for OAMP is 192.168.0.1 and for Media and Control it is 200.200.85.1.

Table 11-10: Configured Default Gateway Example

Index	Application Type	Interface Mode	IP Address	Prefix Length	Gateway	VLAN ID	Interface Name
0	OAMP	IPv4 Manual	192.168.0.2	16	192.168.0.1	100	Mgmt
1	Media & Control	IPv4 Manual	200.200.85.14	24	200.200.85.1	200	CntrlMedia

A separate IP routing table enables you to configure static routing rules. Configuring the following static routing rules enables OAMP applications to access peers on subnet 17.17.0.0 through the gateway 192.168.10.1 (which is not the default gateway of the interface), and Media & Control applications to access peers on subnet 171.79.39.0 through the gateway 200.200.85.10 (which is not the default gateway of the interface).

Table 11-11: Separate Routing Table Example

Destination IP Address	Prefix Length	Gateway IP Address	Metric	Interface Name	Status
17.17.0.0	16	192.168.10.1	1	0	Active
171.79.39.0	24	200.200.85.10	1	1	Active

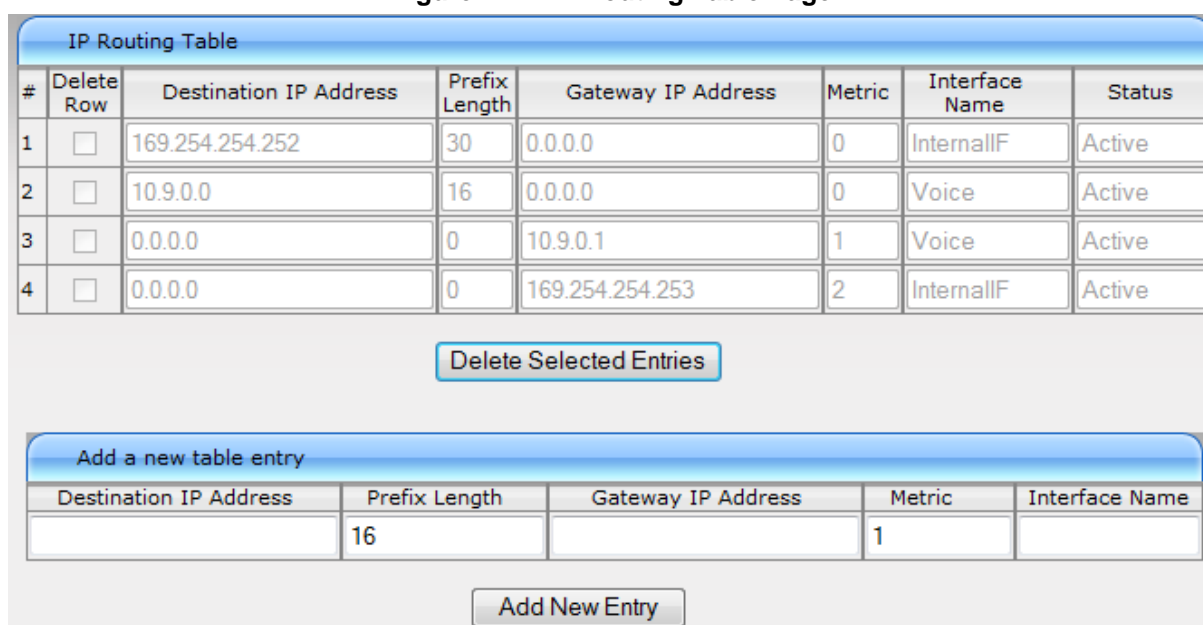
11.4 Configuring the IP Routing Table

The IP Routing Table page allows you to define up to 30 static IP routing rules for the device. These rules can be associated with a network interface (defined in the Multiple Interface table) and therefore, the routing decision is based on the source subnet/VLAN. If not associated with an IP interface, the static IP rule is based on destination IP address. Traffic destined to the subnet specified in the routing rule is re-directed to the defined gateway, reachable through the specified interface. Before sending an IP packet, the device searches this table for an entry that matches the requested destination host/network. If such an entry is found, the device sends the packet to the indicated router. If no explicit entry is found, the packet is sent to the default gateway.

➤ **To configure static IP routing:**

1. Open the IP Routing Table page (**Configuration** tab > **VoIP** menu > **Network** submenu > **IP Routing Table**).

Figure 11-7: IP Routing Table Page



#	Delete Row	Destination IP Address	Prefix Length	Gateway IP Address	Metric	Interface Name	Status
1	<input type="checkbox"/>	169.254.254.252	30	0.0.0.0	0	InternalIF	Active
2	<input type="checkbox"/>	10.9.0.0	16	0.0.0.0	0	Voice	Active
3	<input type="checkbox"/>	0.0.0.0	0	10.9.0.1	1	Voice	Active
4	<input type="checkbox"/>	0.0.0.0	0	169.254.254.253	2	InternalIF	Active

Delete Selected Entries

Add a new table entry				
Destination IP Address	Prefix Length	Gateway IP Address	Metric	Interface Name
	16		1	

Add New Entry

2. In the Add a new table entry table, add a new static routing rule according to the parameters described in the table below.
3. Click **Add New Entry**; the new routing rule is added to the IP routing table.

To delete a routing rule from the table, select the 'Delete Row' check box corresponding to the required routing rule, and then click **Delete Selected Entries**.



Notes:

- You can delete only inactive routing rules.
- The IP Routing table can also be configured using the table ini file parameter, StaticRouteTable.

Table 11-12: IP Routing Table Description

Parameter	Description
Destination IP Address [StaticRouteTable_Destination]	Defines the IP address of the destination host/network. The destination can be a single host or a whole subnet, depending on the Prefix Length configured for this routing rule.
Prefix Length [StaticRouteTable_PrefixLength]	Defines the Classless Inter-Domain Routing (CIDR)-style representation of a dotted-decimal subnet notation, of the destination host/network. The CIDR-style representation uses a suffix indicating the number of bits that are set in the dotted-decimal format. For example, 16 is synonymous with subnet 255.255.0.0.
The address of the host/network you want to reach is determined by an AND operation that is applied to the fields 'Destination IP Address' and 'Prefix Length'. For example, to reach the network 10.8.x.x, enter 10.8.0.0 in the 'Destination IP Address' field and 16 in the 'Prefix Length'. As a result of the AND operation, the value of the last two octets in the 'Destination IP Address' field is ignored. To reach a specific host, enter its IP address in the 'Destination IP Address' field and 32 in the 'Prefix Length' field.	
Gateway IP Address [StaticRouteTable_Gateway]	Defines the IP address of the router (next hop) used for traffic destined to the subnet/host as defined in the 'Destination IP Address' / 'Prefix Length' field. Note: The Gateway address must be in the same subnet as the IP address of the interface over which you configure this static routing rule.
Metric	Defines the number of hops needed to reach the specified destination. Note: The recommended value for this parameter is 1. This parameter must be set to a number greater than 0 for the routing rule to be valid. Routing entries with Hop Count equals 0 are local routes set automatically by the device.
Interface Name [StaticRouteTable_InterfaceName]	Assigns a network interface through which the 'Gateway IP Address' is reached. This is the string value as configured for the network interface in the 'Interface Name' field of the Multiple Interface table (see 'Configuring IP Network Interfaces' on page 98). Note: The IP address of the 'Gateway IP Address' field must be in the same subnet as this interface's IP address.
Status	Read-only field displaying the status of the static IP route: <ul style="list-style-type: none"> "Active" - routing rule is used by the device. "Inactive" - routing rule is not applied. When the destination IP address is not on the same segment with the next hop or the interface does not exist, the route state changes to "Inactive".

11.4.1 Interface Column

This example describes the configuration of static IP routing rules.



Note: The Interface Address family must be coherent with the Routing Rule Address family. IPv4 interfaces cannot be selected in an IPv6 routing rule, and vice versa.

1. Configure network interfaces in the Multiple Interface table, as shown below:

Table 11-13: Configured Network Interfaces in Multiple Interface Table

Index	Application Type	Interface Mode	IP Address	Prefix Length	Gateway	VLAN ID	Interface Name
0	OAMP	IPv4 Manual	192.168.0.2	16	192.168.0.1	501	Mng
1	Media & Control	IPv4 Manual	10.32.174.50	24	10.32.174.1	2012	MediaCntrl
2	Media	IPv4 Manual	10.33.174.50	24	10.33.174.1	2013	Media1
3	Control	IPv4 Manual	10.34.174.50	24	10.34.174.1	2014	Cntrl1

2. Configure static IP Routing rules in the IP Routing table, as shown below:

Table 11-14: Configured Static IP Routing Rules in IP Routing Table

Destination IP Address	Prefix Length	Gateway IP Address	Metric	Interface Name
10.31.174.0	24	192.168.11.1	1	Mng
174.96.151.15	24	10.32.174.12	1	MediaCntrl
10.35.174.0	24	10.34.174.240	1	Cntrl1

Note that the IP address configured in the 'Gateway IP Address' field (i.e., next hop) must reside on the same subnet as the IP address of the associated network interface that is specified in the 'Interface Name' field.

11.4.2 Routing Table Configuration Summary and Guidelines

The Routing table configurations must adhere to the following rules:

- Up to 30 different static routing rules can be configured.
- The 'Prefix Length' replaces the dotted-decimal subnet mask presentation. This column must have a value of 0-31 for IPv4 interfaces and a value of 64 for IPv6 interfaces.
- The 'Gateway IP Address' field must be on the same subnet as the IP address of the associated interface specified in the 'Interface Name' field.
- The 'Interface Name' selected for the routing rule must be of the same address family as the rule defined.

- The 'Metric' field must be set to 1.
- For the configuration settings to take effect, you must reset the device with a "burn" to flash memory.

11.4.3 Troubleshooting the Routing Table

When adding a new static routing rule, the added rule passes a validation test. If errors are found, the routing rule is rejected and is not added to the IP Routing table. Failed routing validations may result in limited connectivity (or no connectivity) to the destinations specified in the incorrect routing rule. For any error found in the Routing table or failure to configure a routing rule, the device sends a notification message to the Syslog server reporting the problem.

Common routing rule configuration errors may include the following:

- The IP address specified in the 'Gateway IP Address' field is unreachable from the interface specified in the 'Interface Name' field.
- The same destination is configured in two different routing rules.
- More than 30 routing rules have been configured.



Note: If an IP routing rule is required to access OAMP applications (for remote management, for example) and the route is not configured correctly, the route is not added and the device is not accessible remotely. To restore connectivity, the device must be accessed locally from the OAMP subnet and the required routes be configured.

11.5 Configuring Quality of Service

The Diff Serv Table page is used for configuring the Layer-2 and Layer-3 Quality of Service (QoS) parameters. Differentiated Services (DiffServ) is an architecture providing different types or levels of service for IP traffic. DiffServ (according to RFC 2474), prioritizes certain traffic types based on their priority, thereby, accomplishing a higher-level QoS at the expense of other traffic types. By prioritizing packets, DiffServ routers can minimize transmission delays for time-sensitive packets such as VoIP packets.

You can assign DiffServ to four classes of traffic and assign VLAN priorities (IEEE 802.1p) to various values of DiffServ:

- Premium Media service class – used for RTP media traffic
- Premium Control service class – used for call control (i.e., SIP) traffic
- Gold service class – used for streaming applications
- Bronze service class – used for OAMP applications

The Layer-3 QoS parameters define the values of the DiffServ field in the IP header of the frames related to a specific service class. The Layer-2 QoS parameters define the values for the 3 priority bits in the VLAN tag (according to the IEEE 802.1p standard) according to the value of the DiffServ field found in the packet IP header.

The DiffServ Table (DiffServToVlanPriority) allows you to configure up to 64 DiffServ-to-VLAN Priority mapping (Layer 2 class of service). For each packet sent to the LAN, the VLAN Priority of the packet is set according to the DiffServ value in the IP header of the packet.

The mapping of an application to its CoS and traffic type is shown in the table below:

Table 11-15: Traffic/Network Types and Priority

Application	Traffic / Network Types	Class-of-Service (Priority)
Debugging interface	Management	Bronze
Telnet	Management	Bronze
DHCP	Management	Network
Web server (HTTP)	Management	Bronze
SNMP GET/SET	Management	Bronze
Web server (HTTPS)	Management	Bronze
RTP traffic	Media	Premium media
RTCP traffic	Media	Premium media
T.38 traffic	Media	Premium media
SIP	Control	Premium control
SIP over TLS (SIPS)	Control	Premium control
Syslog	Management	Bronze
SNMP Traps	Management	Bronze
DNS client	Varies according to DNS settings: <ul style="list-style-type: none"> ▪ OAMP ▪ Control 	Depends on traffic type: <ul style="list-style-type: none"> ▪ Control: Premium Control ▪ Management: Bronze
NTP	Varies according to the interface type associated with NTP (see 'Assigning NTP Services to Application Types' on page 102): <ul style="list-style-type: none"> ▪ OAMP ▪ Control 	Depends on traffic type: <ul style="list-style-type: none"> ▪ Control: Premium control ▪ Management: Bronze


Notes:

- For the QoS settings to take effect, a device reset is required.
- You can also configure the DiffServ table using the table ini file parameter DiffServToVlanPriority.

➤ **To configure QoS:**

1. Open the Diff Serv Table page (**Configuration** tab > **VoIP** menu > **Network** submenu > **QoS Settings**).

Figure 11-8: DiffServ Table Page

Index	Differentiated Services	VLAN Priority
1 <input type="radio"/>	<input type="text" value="6"/>	<input type="text" value="1"/>

▼ Differentiated Services	
Media Premium QoS	<input type="text" value="46"/>
Control Premium QoS	<input type="text" value="40"/>
Gold QoS	<input type="text" value="26"/>
Bronze QoS	<input type="text" value="10"/>

2. Configure DiffServ to VLAN priority mapping (Layer-2 QoS):
 - a. Enter an index entry, and then click Add.
 - b. In the 'Differentiated Services' field, enter the DiffServ value (0-63) and its corresponding VLAN priority level (0-7).
 - c. Click Submit.
3. Configure the desired DiffServ (Layer-3 QoS) values for the following traffic classes:
 - Media Premium QoS: this affects Media RTP packets sent by the VoIP towards the LAN.
 - Control Premium QoS: this affects Control Protocol (SIP) packets sent by the VoIP towards the LAN.
 - Gold QoS: this affects HTTP Streaming packets sent by the VoIP towards the LAN.
 - Bronze QoS: this affects OAMP packets sent by the VoIP towards the LAN.
4. Click **Submit** to apply your changes.
5. Save the changes to flash memory and reset the device (see 'Saving Configuration' on page 360).

11.6 Disabling ICMP Redirect Messages

You can configure the device's handling of ICMP Redirect messages. These messages can either be rejected (ignored) or permitted.

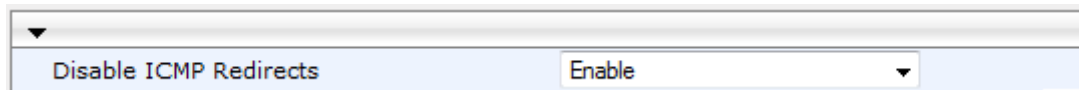


Note: You can also configure this feature using the ini file parameter `DisableICMPRedirects` (see 'Routing Parameters' on page 438).

➤ To configure the handling of ICMP Redirect messages:

1. Open the Network Settings page (**Configuration** tab > **VoIP** menu > **Network** submenu > **Network Settings**).

Figure 11-9: Disabling ICMP Redirect in Network Settings Page



The screenshot shows a web interface with a dropdown menu labeled 'Disable ICMP Redirects'. The selected option is 'Enable'.

2. From the 'Disable ICMP Redirects' drop-down list, select the required option.
3. Click **Submit** to apply your changes.

11.7 DNS

You can use the device's embedded domain name server (DNS) or an external, third-party DNS to translate domain names into IP addresses. This is useful if domain names are used as the destination in call routing. The device supports the configuration of the following DNS types:

- Internal DNS table - see 'Configuring the Internal DNS Table' on page 114
- Internal SRV table - see 'Configuring the Internal SRV Table' on page 116

11.7.1 Configuring the Internal DNS Table

The Internal DNS Table page, similar to a DNS resolution, translates up to 20 host (domain) names into IP addresses. This functionality can be used when a domain name (FQDN) is configured as an IP destination for IP-to-IP routing in the SBC IP-to-IP Routing table. Up to four different IP addresses can be assigned to the same host name. This is typically needed for alternative Tel-to-IP call routing.



Notes:

- The device initially attempts to resolve a domain name using the Internal DNS table. If the domain name isn't listed in the table, the device performs a DNS resolution using an external DNS server for the related IP network interface, configured in the Multiple Interface table (see 'Configuring IP Network Interfaces' on page 98).
- You can also configure the DNS table using the table ini file parameter, `DNS2IP` (see 'DNS Parameters' on page 441).

➤ **To configure the internal DNS table:**

1. Open the Internal DNS Table page (**Configuration** tab > **VoIP** menu > **Network** submenu > **DNS** submenu > **Internal DNS Table**).
2. Click **Add**; the following dialog box appears:

Figure 11-10: Internal DNS Table - Add Record Dialog Box

Add Record	
Index	0
Domain Name	dnscompany.com
First IP Address	10.8.2.15
Second IP Address	10.8.4.20
Third IP Address	10.8.16.17
Fourth IP Address	0.0.0.0
<input type="button" value="Submit"/> <input type="button" value="Cancel"/>	

3. Configure the DNS rule, as required. For a description of the parameters, see the table below.
4. Click **Submit**; the DNS rule is added to the table.

Table 11-16: Internal DNS Table Parameter Description

Parameter	Description
Domain Name [Dns2Ip_DomainName]	Defines the host name to be translated. The valid value is a string of up to 31 characters.
First IP Address [Dns2Ip_FirstIpAddress]	Defines the first IP address (in dotted-decimal format notation) to which the host name is translated.
Second IP Address [Dns2Ip_SecondIpAddress]	Defines the second IP address (in dotted-decimal format notation) to which the host name is translated.
Third IP Address [Dns2Ip_ThirdIpAddress]	Defines the third IP address (in dotted-decimal format notation) to which the host name is translated.
Fourth IP Address [Dns2Ip_FourthIpAddress]	Defines the fourth IP address (in dotted-decimal format notation) to which the host name is translated.

11.7.2 Configuring the Internal SRV Table

The Internal SRV Table page resolves host names to DNS A-Records. Three different A-Records can be assigned to each host name, where each A-Record contains the host name, priority, weight, and port.



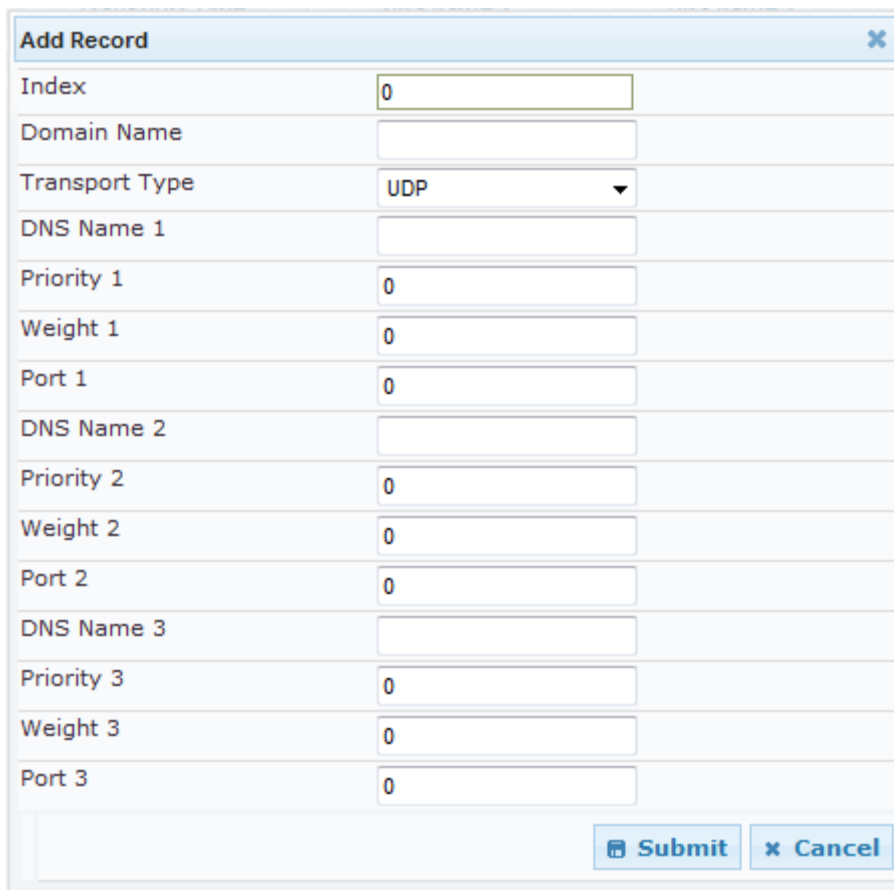
Notes:

- If the Internal SRV table is configured, the device initially attempts to resolve a domain name using this table. If the domain name isn't found, the device performs a Service Record (SRV) resolution using an external DNS server configured in the Multiple Interface table (see 'Configuring IP Network Interfaces' on page 98).
- The Internal SRV table can also be configured using the table ini file parameter, SRV2IP (see 'DNS Parameters' on page 441).

➤ To configure the Internal SRV table:

1. Open the Internal SRV Table page (**Configuration** tab > **VoIP** menu > **Network** submenu > **DNS** submenu > **Internal SRV Table**).
2. Click **Add**; the following dialog box appears:

Figure 11-11: Internal SRV Table Page



Index	0
Domain Name	
Transport Type	UDP
DNS Name 1	
Priority 1	0
Weight 1	0
Port 1	0
DNS Name 2	
Priority 2	0
Weight 2	0
Port 2	0
DNS Name 3	
Priority 3	0
Weight 3	0
Port 3	0

3. Configure the SRV rule, as required. For a description of the parameters, see the table below.
4. Click **Submit**; the SRV rule is added to the table.

Table 11-17: Internal SRV Table Parameter Description

Parameter	Description
Domain Name [Srv2lp_InternalDomain]	Defines the host name to be translated. The valid value is a string of up to 31 characters.
Transport Type [Srv2lp_TransportType]	Defines the transport type. <ul style="list-style-type: none"> ▪ [0] UDP (default) ▪ [1] TCP ▪ [2] TLS
DNS Name (1-3) [Srv2lp_Dns1/2/3]	Defines the first, second or third DNS A-Record to which the host name is translated.
Priority (1-3) [Srv2lp_Priority1/2/3]	Defines the priority of the target host. A lower value means that it is more preferred.
Weight (1-3) [Srv2lp_Weight1/2/3]	Defines a relative weight for records with the same priority.
Port (1-3) [Srv2lp_Port1/2/3]	Defines the TCP or UDP port on which the service is to be found.

11.8 Configuring NFS Settings

Network File System (NFS) enables the device to access a remote server's shared files and directories and to handle them as if they're located locally. The device can use NFS to load *cmp*, *ini*, and auxiliary files through the Automatic Update mechanism (see 'Automatic Update' on page 379).

You can configure up to 16 different NFS file systems. As a file system, the NFS is independent of machine types, operating systems and network architectures. Note that an NFS file server can share multiple file systems. There must be a separate row for each remote file system shared by the NFS file server that needs to be accessed by the device.

➤ To add remote NFS file systems:


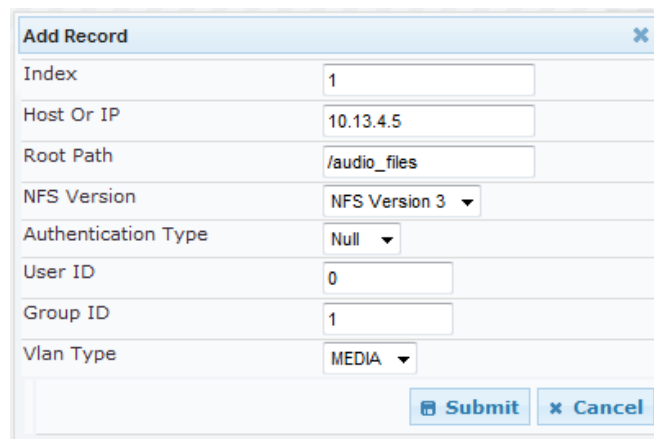
1. Open the Application Settings page (**Configuration** tab > **System** menu > **Application Settings**).
2. Under the 'NFS Settings' group, click the **NFS Table**  button; the NFS Table page appears.
3. Click the **Add** button; the Add Record dialog box appears:

Figure 11-12: Add Record Dialog Box for NFS



The 'Add Record' dialog box for NFS configuration contains the following fields and options:

- Index:** Text input field with value '1'.
- Host Or IP:** Text input field with value '10.13.4.5'.
- Root Path:** Text input field with value '/audio_files'.
- NFS Version:** Dropdown menu with 'NFS Version 3' selected.
- Authentication Type:** Dropdown menu with 'Null' selected.
- User ID:** Text input field with value '0'.
- Group ID:** Text input field with value '1'.
- Vlan Type:** Dropdown menu with 'MEDIA' selected.

At the bottom right, there are two buttons: **Submit** and **Cancel**.

4. Configure the NFS parameters according to the table below.
5. Click the **Submit** button; the remote NFS file system is immediately applied, which can be verified by the appearance of the 'NFS mount was successful' message in the Syslog server.
6. To save the changes to flash memory, see 'Saving Configuration' on page 360.


Notes:

- To avoid terminating current calls, a row must not be deleted or modified while the device is currently accessing files on that remote NFS file system.
- The combination of 'Host Or IP' and 'Root Path' must be unique for each row in the table. For example, the table must include only one row with a Host/IP of 192.168.1.1 and Root Path of /audio.
- The NFS table can also be configured using the table ini file parameter NFSServers (see 'NFS Parameters' on page 441)

Table 11-18: NFS Settings Parameters

Parameter	Description
Index	The row index of the remote file system. The valid range is 1 to 16.
Host Or IP [NFSServers_HostOrIP]	The domain name or IP address of the NFS server. If a domain name is provided, a DNS server must be configured.
Root Path [NFSServers_RootPath]	Path to the root of the remote file system in the format: /[path]. For example, '/audio'.
NFS Version [NFSServers_NfsVersion]	NFS version used to access the remote file system. <ul style="list-style-type: none"> ▪ [2] NFS Version 2 ▪ [3] NFS Version 3 (default)
Authentication Type [NFSServers_AuthType]	Authentication method used for accessing the remote file system. <ul style="list-style-type: none"> ▪ [0] Null ▪ [1] Unix (default)
User ID [NFSServers_UID]	User ID used in authentication when using Unix. The valid range is 0 to 65537. The default is 0.
Group ID [NFSServers_GID]	Group ID used in authentication when using Unix. The valid range is 0 to 65537. The default is 1.
VLAN Type [NFSServers_VlanType]	The VLAN type for accessing the remote file system. <ul style="list-style-type: none"> ▪ [0] OAM ▪ [1] MEDIA (default) <p>Note: This parameter applies only if VLANs are enabled or if Multiple IPs is configured (see 'Configuring IP Network Interfaces' on page 98).</p>

11.9 Network Address Translation Support

Network Address Translation (NAT) is a mechanism that maps internal IP addresses (and ports) used within a private network to global IP addresses and vice versa, providing transparent routing to end hosts. The primary advantages of NAT include (1) reduction in

the number of global IP addresses required in a private network (global IP addresses are only used to connect to the Internet) and (2) better network security by hiding the internal architecture.

The design of SIP creates a problem for VoIP traffic to pass through NAT. SIP uses IP addresses and port numbers in its message body. However, the NAT server is unable to modify the SIP messages and thus, can't change local addresses to global addresses.

This section discusses the device's solutions for overcoming NAT traversal issues.

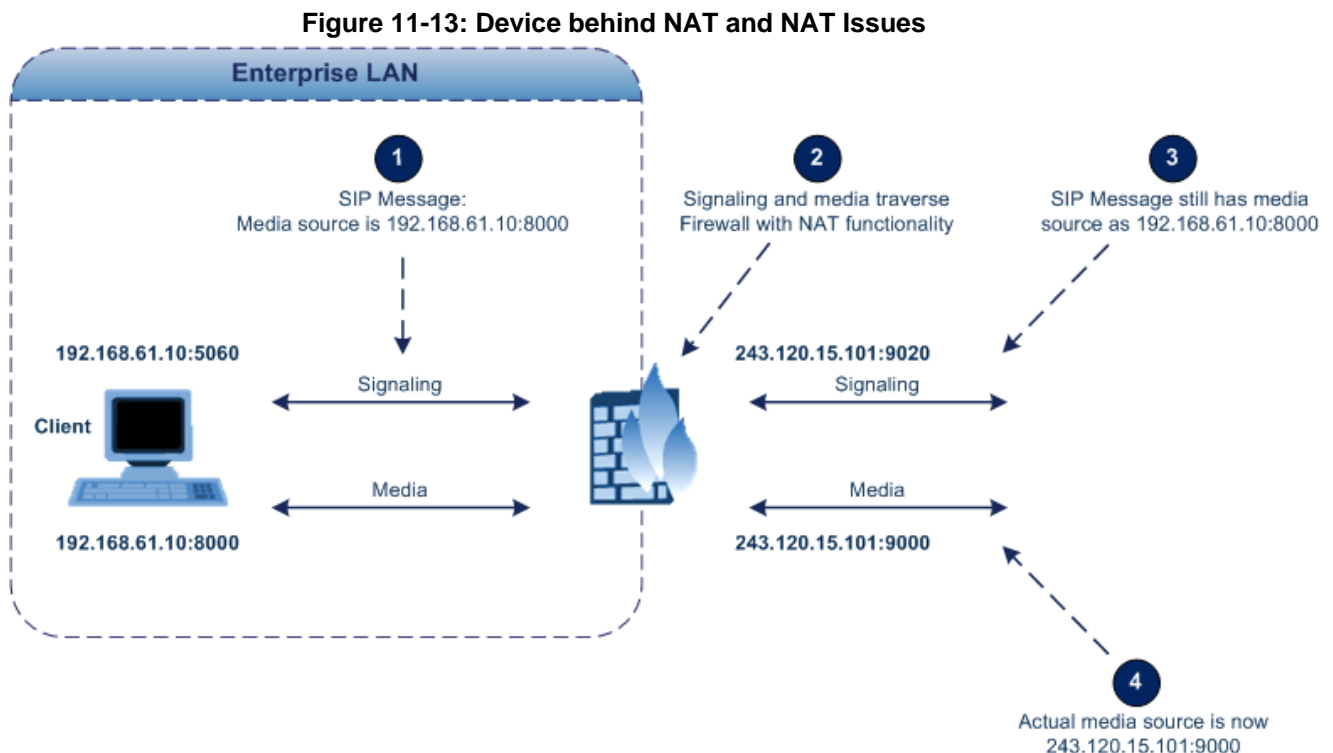
11.9.1 Device Located behind NAT

Two different streams traverse through NAT - signaling and media. A device located behind a NAT, that initiates a signaling path has problems receiving incoming signaling responses as they are blocked by the NAT server. Therefore, the initiating device must inform the receiving device where to send the media. To resolve this NAT problem, the following solutions are provided by the device, listed in priority of the selected method used by the device:

- a. If configured, uses the single Static NAT IP address for all interfaces - see 'Configuring a Static NAT IP Address for All Interfaces' on page 120.
- b. If configured, uses the NAT Translation table which configures NAT per interface - see 'Configuring NAT Translation per IP Interface' on page 120.

If NAT is not configured by any of the above-mentioned methods, the device sends the packet according to its IP address configured in the Multiple Interface table.

The figure below illustrates the NAT problem faced by the SIP networks where the device is located behind a NAT:



11.9.1.1 Configuring a Static NAT IP Address for All Interfaces

You can configure a global (public) IP address of the router to enable static NAT between the device and the Internet for all network interfaces. Thus, the device replaces the source IP address for media of all outgoing SIP messages sent on any of its network interfaces to this public IP address.

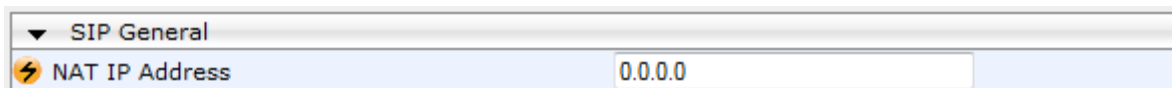


Note: The NAT IP address can also be configured using the ini file parameter, StaticNATIP.

➤ **To configure a single static NAT IP address for all interfaces:**

1. Open the SIP General Parameters page (**Configuration** tab > **VoIP** menu > **SIP Definitions** submenu > **General Parameters**).

Figure 11-14: Configuring Static NAT IP Address in SIP General Parameters Page



The screenshot shows a web interface for 'SIP General' parameters. Under the 'SIP General' section, there is a field labeled 'NAT IP Address' with a lightning bolt icon to its left. The field contains the value '0.0.0.0'.

2. In the 'NAT IP Address' field, enter the NAT IP address in dotted-decimal notation.
3. Click **Submit**.
4. Save the setting to the device's flash memory with a device reset (see 'Saving Configuration' on page 360).

11.9.1.2 Configuring NAT Translation per IP Interface

The NAT Translation table defines network address translation (NAT) rules for translating source IP addresses per VoIP interface (SIP control and RTP media traffic) into NAT IP addresses (*global* or *public*), when the device is located behind NAT. This allows, for example, the separation of VoIP traffic between different ITSP's, and topology hiding of internal IP addresses to the "public" network. Each IP interface (configured in the Multiple Interface table) can be associated with a NAT rule in this table, translating the source IP address and port of the outgoing packet into the NAT address (IP address and port range). The device's NAT traversal mechanism replaces the source IP address of SIP messages sent from a specified VoIP interface to a public IP address.



Note: The NAT Translation table can also be configured using the table ini file parameter, NATTranslation.

➤ **To configure NAT translation rules:**

1. Open the NAT Translation Table page (**Configuration** tab > **VoIP** menu > **Control Network** submenu > **NAT Translation Table**).
2. Click the **Add** button; the following dialog box appears:

Figure 11-15: NAT Translation Table Page

The screenshot shows a dialog box titled "Add Record" with a close button (X) in the top right corner. It contains seven input fields arranged vertically: "Index" with the value "0", "Source Interface Name" with the value "Voice", "Target IP Address" with the value "212.199.200.90", "Source Start Port" with the value "5070", "Source End Port" with the value "5070", "Target Start Port" with the value "5070", and "Target End Port" with the value "5070". At the bottom right of the dialog are two buttons: "Submit" and "Cancel".

3. Configure the parameters as required. For a description of the parameters, see the table below:
4. Click **Submit** to apply your changes.
5. To save the changes to flash memory, see 'Saving Configuration' on page 360.

Table 11-19: NAT Translation Table Parameters

Parameter	Description
Index [NATTranslation_Index]	Defines the table index entry. This table can include up to 32 entries.
Source Interface Name [NATTranslation_SourceInterfaceName]	Defines the name of the IP interface, as appears in the Multiple Interface table.
Target IP Address [NATTranslation_TargetIPAddress]	Defines the global IP address. This address is set in the SIP Via and Contact headers as well as in the o= and c= SDP fields.
Source Start Port [NATTranslation_SourceStartPort]	Defines the optional starting port range (1-65536) of the IP interface, used as matching criteria for this NAT rule. If not configured, the match is done on the entire port range. Only IP addresses and ports of matched source ports will be replaced.
Source End Port [NATTranslation_SourceEndPort]	Defines the optional ending port range (1-65536) of the IP interface, used as matching criteria for this NAT rule. If not configured, the match is done on the entire port range. Only IP addresses and ports of matched source ports will be replaced.
Target Start Port [NATTranslation_TargetStartPort]	Defines the optional, starting port range (1-65536) of the global address. If not configured, the ports are not replaced. Matching source ports are replaced with the target ports. This address is set in the SIP Via and Contact headers, as well as in the o= and c= SDP fields.

Parameter	Description
Target End Port [NATTranslation_TargetEndPoint]	Defines the optional, ending port range (1-65536) of the global address. If not configured, the ports are not replaced. Matching source ports are replaced with the target ports. This address is set in the SIP Via and Contact headers, as well as in the o= and c= SDP fields.

11.9.2 Remote UA behind NAT

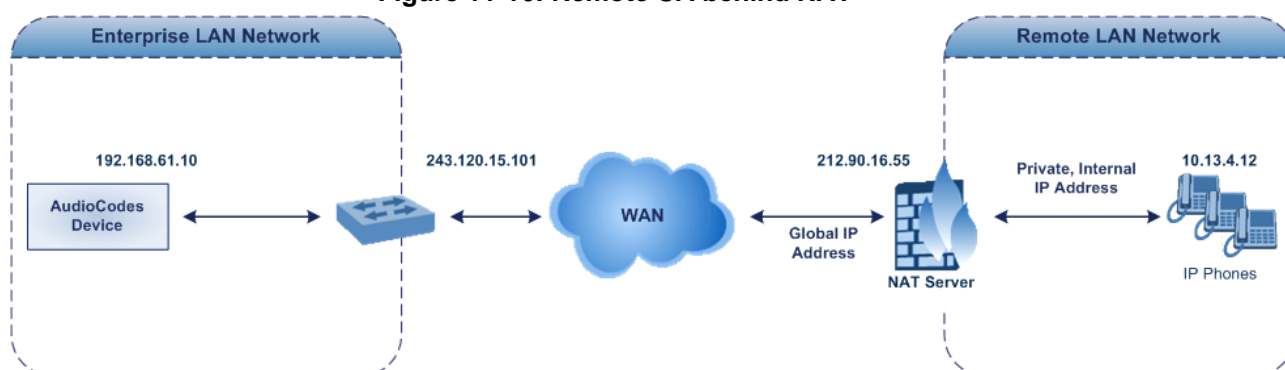
If the remote User Agent with which the device needs to communicate with is located behind NAT, the device can resolve the problem of activating the RTP/RTCP/T.38 streams to an invalid IP address / UDP port.

To resolve this NAT traversal issue, the device offers the following features:

- First Incoming Packet Mechanism - see 'First Incoming Packet Mechanism' on page 122
- RTP No-Op packets according to the avt-rtp-noop draft - see 'No-Op Packets' on page 123

The figure below illustrates a typical network architecture where the remote UA is located behind NAT:

Figure 11-16: Remote UA behind NAT



11.9.2.1 First Incoming Packet Mechanism

If the remote device resides behind a NAT device, it's possible that the device can activate the RTP/RTCP/T.38 streams to an invalid IP address / UDP port. To avoid such cases, the device automatically compares the source address of the first received incoming RTP/RTCP/T.38 stream with the IP address and UDP port of the remote device when the session was initially opened. If the two are not identical, the destination IP address of the outgoing RTP packets is set to the source IP address of the first incoming packet. The RTP, RTCP and T.38 can thus have independent destination IP addresses and UDP ports.

- **To enable NAT resolution using the First Incoming Packet mechanism:**
 1. Open the General Media Settings page (**Configuration** tab > **VoIP** menu > **Media** > **General Media Settings**).
 2. Set the 'NAT Traversal' parameter to **Enable**.
 3. Click **Submit**.

11.9.2.2 No-Op Packets

The device's No-Op packet support can be used to verify Real-Time Transport Protocol (RTP) and T.38 connectivity, and to keep NAT bindings and Firewall pinholes open. The No-Op packets are available for sending in RTP and T.38 formats.

You can control the activation of No-Op packets by using the *ini* file parameter NoOpEnable. If No-Op packet transmission is activated, you can control the time interval in which No-Op packets are sent in the case of silence (i.e., no RTP or T.38 traffic). This is done using the *ini* file parameter NoOpInterval. For a description of the RTP No-Op *ini* file parameters, see 'Networking Parameters' on page 437.

- **RTP No-Op:** The RTP No-Op support complies with IETF Internet-Draft draft-wing-avt-rtp-noop-03 ("A No-Op Payload Format for RTP"). This IETF document defines a No-Op payload format for RTP. The draft defines the RTP payload type as dynamic. You can control the payload type with which the No-Op packets are sent. This is performed using the RTPNoOpPayloadType *ini* parameter (see 'Networking Parameters' on page 437). The default payload type is 120.
- **T.38 No-Op:** T.38 No-Op packets are sent only while a T.38 session is activated. Sent packets are a duplication of the previously sent frame (including duplication of the sequence number).



Note: Receipt of No-Op packets is always supported.

11.10 Robust Receipt of Media Streams

The “robust-media” mechanism is an AudioCodes proprietary mechanism to filter out unwanted media (i.e., RTP, RTCP, and T.38) streams that are sent to the same port number on the device. In practise, the media RTP/RTCP ports may receive additional multiple unwanted media streams as result of traces of previous calls, call control errors, or deliberate attacks. When more than one media stream reaches the device on the same port number, the “robust-media” mechanism detects the valid media stream and ignores the rest.

The “robust-media” mechanism can be disabled by setting the `InboundMediaLatchMode` parameter to 0.

11.11 Multiple Routers Support

Multiple routers support is designed to assist the device when it operates in a multiple routers network. The device learns the network topology by responding to Internet Control Message Protocol (ICMP) redirections and caches them as routing rules (with expiration time).

When a set of routers operating within the same subnet serve as devices to that network and intercommunicate using a dynamic routing protocol, the routers can determine the shortest path to a certain destination and signal the remote host the existence of the better route. Using multiple router support, the device can utilize these router messages to change its next hop and establish the best path.



Note: Multiple Routers support is an integral feature that doesn't require configuration.

12 Security

This section describes the VoIP security-related configuration.

12.1 Configuring Firewall Settings

The device provides an internal firewall that enables you to configure network traffic filtering rules (*access list*). You can add up to 50 firewall rules. The access list offers the following firewall possibilities:

- Block traffic from known malicious sources
- Allow traffic only from known "friendly" sources, and block all other traffic
- Mix allowed and blocked network sources
- Limit traffic to a user-defined rate (blocking the excess)
- Limit traffic to specific protocols, and specific port ranges on the device

For each packet received on the network interface, the table is scanned from top to bottom until the first matching rule is found. This rule can either permit (*allow*) or deny (*block*) the packet. Once a rule in the table is located, subsequent rules further down the table are ignored. If the end of the table is reached without a match, the packet is accepted.

Notes:

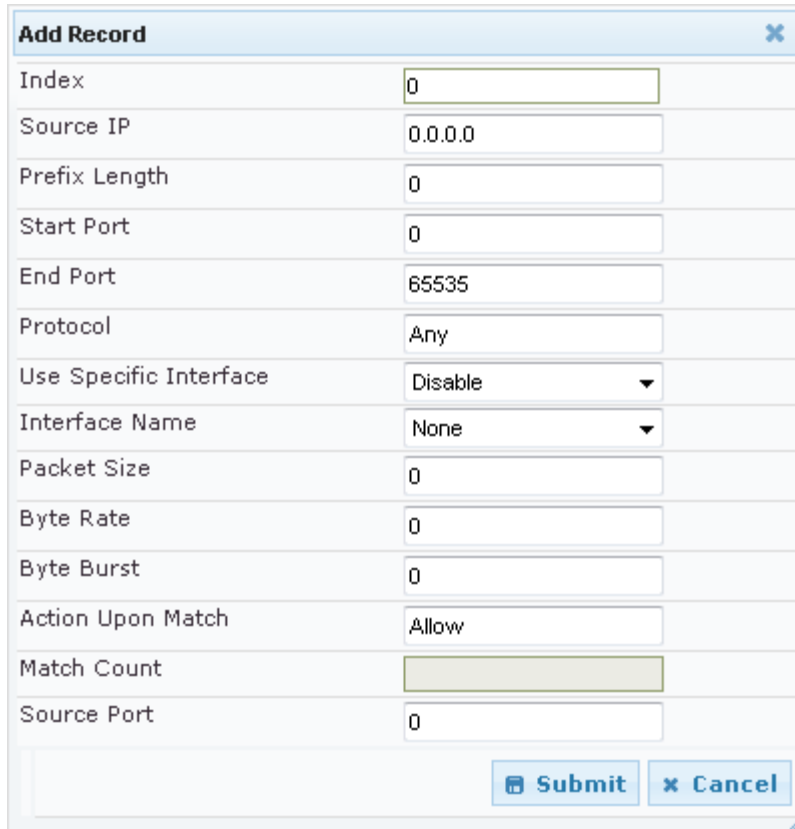
- This firewall applies to a very low-level network layer and overrides all your other security-related configuration. Thus, if you have configured higher-level security features (e.g., on the Application level), you must also configure firewall rules to permit this necessary traffic. For example, if you have configured IP addresses to access the Web and Telnet interfaces in the Web Access List (see 'Configuring Web and Telnet Access List' on page 60), you must configure a firewall rule that permits traffic from these IP addresses.
- Only Security Administrator users or Master users can configure firewall rules.
- Setting the 'Prefix Length' field to **0** means that the rule applies to **all** packets, regardless of the defined IP address in the 'Source IP' field. Therefore, it is highly recommended to set this parameter to a value other than 0.
- It is recommended to add a rule at the end of your table that blocks all traffic and to add firewall rules above it that allow required traffic (with bandwidth limitations). To block all traffic, use the following firewall rule:
 - Source IP: 0.0.0.0
 - Prefix Length: 0 (i.e., rule matches all IP addresses)
 - Start Port - End Port: 0-65535
 - Protocol: **Any**
 - Action Upon Match: **Block**
- If you are using the High Availability feature and you have configured "block" rules, then ensure that you also add "allow" rules for HA traffic. For more information, see Configuring Firewall Allowed Rules on page 354.
- You can also configure the firewall settings using the table ini file parameter, AccessList (see 'Security Parameters' on page 458).



➤ **To add firewall rules:**

1. Open the Firewall Settings page (**Configuration** tab > **VoIP** menu > **Security** submenu > **Firewall Settings**).
2. Click the **Add** button; the following dialog box appears:

Figure 12-1: Firewall Settings Page - Add Record



Add Record	
Index	0
Source IP	0.0.0.0
Prefix Length	0
Start Port	0
End Port	65535
Protocol	Any
Use Specific Interface	Disable
Interface Name	None
Packet Size	0
Byte Rate	0
Byte Burst	0
Action Upon Match	Allow
Match Count	
Source Port	0
<input type="button" value="Submit"/> <input type="button" value="Cancel"/>	

3. Configure the firewall parameters, as required. For a description of the parameters, see the table below.
4. Click **Submit** to add the new firewall rule to the table.
5. Reset the device to activate the rules.

The table below provides an example of configured firewall rules:

Table 12-1: Firewall Rule Examples

Parameter	Value per Rule				
	1	2	3	4	5
Source IP	12.194.231.76	12.194.230.7	0.0.0.0	192.0.0.0	0.0.0.0
Prefix Length	16	16	0	8	0
Start Port and End Port	0-65535	0-65535	0-65535	0-65535	0-65535
Protocol	Any	Any	icmp	Any	Any
Use Specific Interface	Enable	Enable	Disable	Enable	Disable
Interface Name	WAN	WAN	None	Voice-Lan	None
Byte Rate	0	0	40000	40000	0

Parameter	Value per Rule				
	1	2	3	4	5
Burst Bytes	0	0	50000	50000	0
Action Upon Match	Allow	Allow	Allow	Allow	Block

The firewall rules in the above configuration example do the following:

- **Rules 1 and 2:** Typical firewall rules that allow packets ONLY from specified IP addresses (e.g., proxy servers). Note that the prefix length is configured.
- **Rule 3:** A more "advanced" firewall rule - bandwidth rule for ICMP, which allows a maximum bandwidth of 40,000 bytes/sec with an additional allowance of 50,000 bytes. If, for example, the actual traffic rate is 45,000 bytes/sec, then this allowance would be consumed within 10 seconds, after which all traffic exceeding the allocated 40,000 bytes/sec is dropped. If the actual traffic rate then slowed to 30,000 bytes/sec, the allowance would be replenished within 5 seconds.
- **Rule 4:** Allows traffic from the LAN voice interface and limits bandwidth.
- **Rule 5:** Blocks all other traffic.

Table 12-2: Internal Firewall Parameters

Parameter	Description
Source IP [AccessList_Source_IP]	Defines the IP address (or DNS name) or a specific host name of the source network (i.e., from where the incoming packet is received).
Source Port [AccessList_Source_Port]	Defines the source UDP/TCP ports (of the remote host) from where packets are sent to the device. The valid range is 0 to 65535. Note: When set to 0, this field is ignored and any source port matches the rule.
Prefix Length [AccessList_PrefixLen]	(Mandatory) Defines the IP network mask - 32 for a single host or the appropriate value for the source IP addresses. <ul style="list-style-type: none"> ■ A value of 8 corresponds to IPv4 subnet class A (network mask of 255.0.0.0). ■ A value of 16 corresponds to IPv4 subnet class B (network mask of 255.255.0.0). ■ A value of 24 corresponds to IPv4 subnet class C (network mask of 255.255.255.0). The IP address of the sender of the incoming packet is trimmed in accordance with the prefix length (in bits) and then compared to the parameter 'Source IP'. The default is 0 (i.e., applies to all packets). You must change this value to any of the above options. Note: A value of 0 applies to all packets, regardless of the defined IP address. Therefore, you must set this parameter to a value other than 0.
Start Port [AccessList_Start_Port]	Defines the destination UDP/TCP start port (on this device) to where packets are sent. The valid range is 0 to 65535. Note: When the protocol type isn't TCP or UDP, the entire range must be provided.

Parameter	Description
End Port [AccessList_End_Port]	<p>Defines the destination UDP/TCP end port (on this device) to where packets are sent.</p> <p>The valid range is 0 to 65535.</p> <p>Note: When the protocol type isn't TCP or UDP, the entire range must be provided.</p>
Protocol [AccessList_Protocol]	<p>Defines the protocol type (e.g., UDP, TCP, ICMP, ESP or 'Any') or the IANA protocol number in the range of 0 (Any) to 255.</p> <p>Note: This field also accepts the abbreviated strings 'SIP' and 'HTTP'. Specifying these strings implies selection of the TCP or UDP protocols, and the appropriate port numbers as defined on the device.</p>
Use Specific Interface [AccessList_Use_Specific_Interface]	<p>Determines whether you want to apply the rule to a specific network interface defined in the Multiple Interface table (i.e., packets received from that defined in the Source IP field and received on this network interface):</p> <ul style="list-style-type: none"> ▪ [0] Disable (default) ▪ [1] Enable <p>Notes:</p> <ul style="list-style-type: none"> ▪ If enabled, then in the 'Interface Name' field (described below), select the interface to which the rule is applied. ▪ If disabled, then the rule applies to all interfaces.
Interface Name [AccessList_Interface_ID]	<p>Defines the network interface to which you want to apply the rule. This is applicable if you enabled the 'Use Specific Interface' field. The list displays interface names as defined in the Multiple Interface table in 'Configuring IP Network Interfaces' on page 98.</p>
Packet Size [AccessList_Packet_Size]	<p>Defines the maximum allowed packet size.</p> <p>The valid range is 0 to 65535.</p> <p>Note: When filtering fragmented IP packets, this field relates to the overall (re-assembled) packet size, and not to the size of each fragment.</p>
Byte Rate [AccessList_Byte_Rate]	<p>Defines the expected traffic rate (bytes per second), i.e., the allowed bandwidth for the specified protocol. In addition to this field, the 'Burst Bytes' field provides additional allowance such that momentary bursts of data may utilize more than the defined byte rate, without being interrupted.</p> <p>For example, if 'Byte Rate' is set to 40000 and 'Burst Bytes' to 50000, then this implies the following: the allowed bandwidth is 40000 bytes/sec with extra allowance of 50000 bytes; if, for example, the actual traffic rate is 45000 bytes/sec, then this allowance would be consumed within 10 seconds, after which all traffic exceeding the allocated 40000 bytes/sec is dropped. If the actual traffic rate then slowed to 30000 bytes/sec, then the allowance would be replenished within 5 seconds.</p>
Burst Bytes [AccessList_Byte_Burst]	<p>Defines the tolerance of traffic rate limit (number of bytes).</p> <p>The default is 0.</p>
Action Upon Match [AccessList_Allow_Type]	<p>Defines the firewall action to be performed upon rule match.</p> <ul style="list-style-type: none"> ▪ "Allow" = (Default) Permits these packets ▪ "Block" = Rejects these packets

Parameter	Description
Match Count [AccessList_MatchCount]	(Read-only) Displays the number of packets accepted or rejected by the rule.

12.2 Configuring General Security Settings

The General Security Settings page is used to configure various security features. For a description of the parameters appearing on this page, refer 'Configuration Parameters Reference' on page 437.

➤ **To configure the general security parameters:**

1. Open the General Security Settings page (**Configuration** tab > **VoIP** menu > **Security** submenu > **General Security Settings**).

▼ IPsec Setting		
Enable IP Security	Disable	▼
IKE Certificate Ext Validate	Disable	▼
▼ TLS Settings		
TLS Version	SSL 2.0-3.0 and TLS 1.0	▼
Strict Certificate Extension Validation	Disable	▼
FIPS140 Mode	Disable	▼
Client Cipher String	ALL:!ADH	
▼ SIP TLS Settings		
TLS Client Re-Handshake Interval	0	
TLS Mutual Authentication	Disable	▼
Peer Host Name Verification Mode	Disable	▼
TLS Client Verify Server Certificate	Disable	▼
TLS Remote Subject Name		
▼ OCSP Settings		
Enable OCSP Server	Disable	▼
Primary Server IP	0.0.0.0	
Secondary Server IP	0.0.0.0	
Server Port	2560	
Default Response When Server Unreachable	Reject	▼

2. Configure the parameters as required.
3. Click **Submit** to apply your changes.
4. To save the changes to flash memory, refer to 'Saving Configuration' on page 360.

12.3 Intrusion Detection System

The device can be configured to detect malicious attacks on its system and send SNMP traps if malicious activity is identified. The Intrusion Detection System (IDS) is an important feature for Enterprises to ensure legitimate calls are not being adversely affected by attacks and to prevent Theft of Service and unauthorized access. If, for example, you identify the source (IP address) of the attack, you can add that source to your blacklist to prevent it from accessing your device.

There are many types of malicious attacks, the most common being:

- **Denial of service:** This can be Denial of Service (DoS) where an attacker wishing to prevent a server from functioning correctly directs a large amount of requests – sometimes meaningless and sometimes legitimate, or it can be Distributed Denial of Service (DDoS) where the attacker controls a large group of systems to coordinate a large scale DoS attack against a system:
 - Message payload tampering: Attacker may inject harmful content into a message, e.g., by entering meaningless or wrong information, with the goal of exploiting a buffer overflow at the target. Such messages can be used to probe for vulnerabilities at the target.
 - Message flow tampering: This is a special case of DoS attacks. These attacks disturb the ongoing communication between users. An attacker can then target the connection by injecting fake signaling messages into the communication channel (such as CANCEL messages).
 - Message Flooding: The most common DoS attack is where an attacker sends a huge amount of messages (e.g., INVITEs) to a target. The goal is to overwhelm the target's processing capabilities, thereby rendering the target inoperable.
- **SPAM over Internet Telephony (SPIT):** VoIP spam is unwanted, automatically dialed, pre-recorded phone calls using VoIP. It is similar to e-mail spam.
- **Theft of Service (ToS):** Service theft can be exemplified by phreaking, which is a type of hacking that steals service (i.e., free calls) from a service provider, or uses a service while passing the cost to another person.

The IDS configuration is based on IDS Policies, where each policy can be configured with a set of IDS rules. Each rule defines a type of malicious attack to detect and the number of attacks during an interval (threshold) before an SNMP trap is sent. Each policy is then applied to a target under attack (SIP interface) and/or source of attack (Proxy Set and/or subnet address).

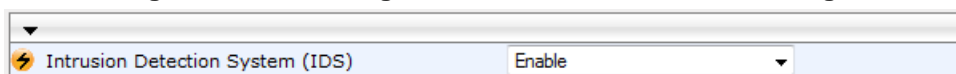
12.3.1 Enabling IDS

The procedure below describes how to enable IDS.

➤ To enable IDS:

1. Open the IDS Global Parameters page (**Configuration** tab > **VoIP** menu > **Security** > **Intrusion Detection and Prevention** > **Global Parameters**).

Figure 12-2: Enabling IDS on IDS Global Parameters Page



2. From the 'Intrusion Detection System' drop-down list, select **Enable**.
3. Reset the device with a burn-to-flash for the setting to take effect (see Saving Configuration).

12.3.2 Configuring IDS Policies

Configuring IDS policies is a two-stage process done in the following tables:

1. **IDS Policy table:** Defines a name and description for the policy. You can define up to 20 policies.
2. **IDS Rules table:** Defines the actual IDS rules per policy. Each policy can be configured with up to 20 rules.



Note: A maximum of 100 IDS rules can be configured (regardless of how many rules are assigned to each policy).

By default and for your convenience, the device provides three pre-configured IDS policies with rules that can be used in your deployment if they meet your requirements:

- "DEFAULT_FEU": Policy for far-end users in the WAN
- "DEFAULT_PROXY": Policy for proxy server
- "DEFAULT_GLOBAL": Policy with global thresholds

These default policies are read-only.

➤ **To configure IDS policies:**

1. Open the IDS Policy Table page (**Configuration** tab > **VoIP** menu > **Security** > **Intrusion Detection and Prevention** > **Policy Table**).

Figure 12-3: IDS Policy Table with Default Rules

Add + Edit ✎ Delete – Show/Hide ☰		
Index	Name	Description
0	DEFAULT_FEU	Default policy for FEU
1	DEFAULT_PROXY	Default policy for proxies
2	DEFAULT_GLOBAL	Default policy for global scope
Page 1 of 1 Show 10 records per page View 1 - 3 of 3		
IDS Policy Table #0 Additional Configuration IDS Rule Table		

2. Add a Policy name:
 - a. Click **Add**.

Figure 12-4: IDS Policy Table - Add Record

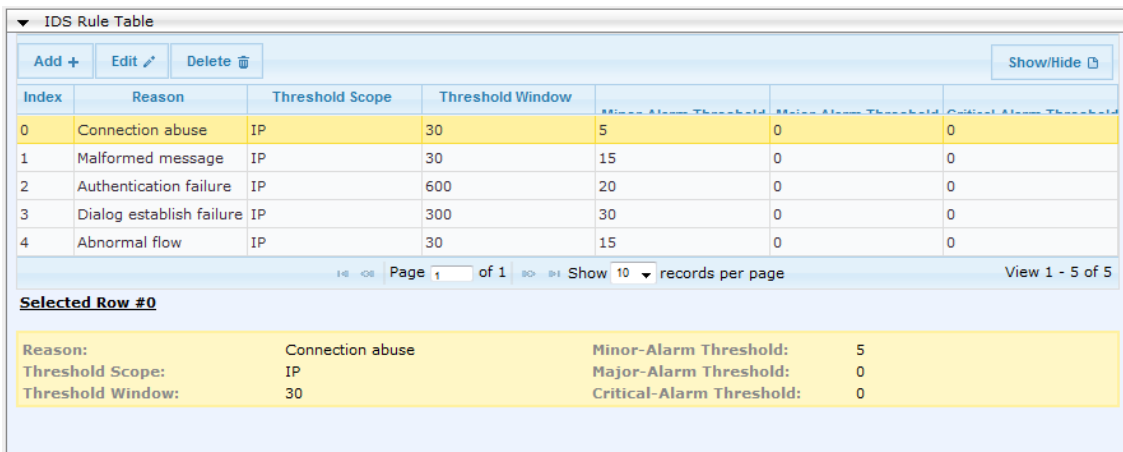
Add Record	
Index	3
Name	SIP-Trunk
Description	for attacks from SIP Trunk
<input type="button" value="Submit"/> <input type="button" value="Cancel"/>	

- b. Configure the parameters as described in the following table, and then click **Submit**.

Table 12-3: IDS Policy Table Parameters

Parameter	Description
Index CLI: policy [IDSPolicy_Index]	Defines the table row number for the policy.
Name CLI: rule [IDSPolicy_Description]	Defines a name for the policy. The valid value is a string of up to 20 characters.
Description [IDSPolicy_Name]	Defines an arbitrary description of the policy. The valid value is a string of up to 100 characters.

3. Add rules to the policy:
 - a. In the IDS Policy table, select the required policy and then click the **IDS Rule Table** link located below the table:

Figure 12-5: IDS Rule Table of Selected IDS Policy


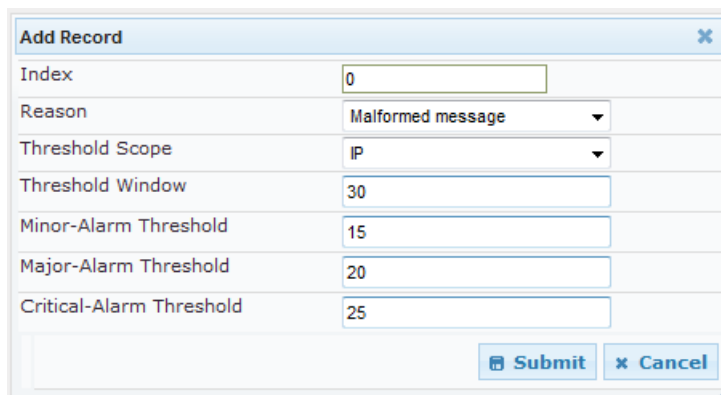
Index	Reason	Threshold Scope	Threshold Window	Minor Alarm Threshold	Major Alarm Threshold	Critical Alarm Threshold
0	Connection abuse	IP	30	5	0	0
1	Malformed message	IP	30	15	0	0
2	Authentication failure	IP	600	20	0	0
3	Dialog establish failure	IP	300	30	0	0
4	Abnormal flow	IP	30	15	0	0

Page 1 of 1 Show 10 records per page View 1 - 5 of 5

Selected Row #0

Reason:	Connection abuse	Minor-Alarm Threshold:	5
Threshold Scope:	IP	Major-Alarm Threshold:	0
Threshold Window:	30	Critical-Alarm Threshold:	0

- b. Click **Add**.

Figure 12-6: IDS Rule Table - Add Record


Index	0
Reason	Malformed message
Threshold Scope	IP
Threshold Window	30
Minor-Alarm Threshold	15
Major-Alarm Threshold	20
Critical-Alarm Threshold	25

Submit Cancel

- c. Configure the parameters as required, and then click **Submit**. For a description of these parameters, see the table below. The figure above shows an example configuration where if 15 malformed SIP messages are received within a period of 30 seconds, a minor alarm is sent. Every 30 seconds, the rule's counters are cleared.
 - d. To add more rules to the policy, repeat steps 1.b to 1.c.

Table 12-4: IDS Rule Table Parameters

Parameter	Description
Index CLI: rule-id [IDSRule_RuleID]	Defines the table row number for the rule.
Reason CLI: reason [IDSRule_Reason]	<p>Defines the type of intrusion attack (malicious event).</p> <ul style="list-style-type: none"> ▪ [0] Any = All events listed below are considered as attacks and are counted together. ▪ [1] Connection abuse (default) = TLS authentication failure. ▪ [2] Malformed message = <ul style="list-style-type: none"> ✓ Message exceeds a user-defined maximum message length (50K) ✓ Any SIP parser error ✓ Message Policy match (see Configuring SIP Message Policy Rules) ✓ Basic headers not present ✓ Content length header not present (for TCP) ✓ Header overflow ▪ [3] Authentication failure = <ul style="list-style-type: none"> ✓ Local authentication ("Bad digest" errors) ✓ Remote authentication (SIP 401/407 is sent if original message includes authentication) ▪ [4] Dialog establish failure = <ul style="list-style-type: none"> ✓ Classification failure (see Configuring Classification Rules) ✓ Routing failure ✓ Other local rejects (prior to SIP 180 response) ✓ Remote rejects (prior to SIP 180 response) ▪ [5] Abnormal flow = <ul style="list-style-type: none"> ✓ Requests and responses without a matching transaction user (except ACK requests) ✓ Requests and responses without a matching transaction (except ACK requests)
Threshold Scope CLI: threshold-scope [IDSRule_ThresholdScope]	<p>Defines the source of the attacker to consider in the device's detection count.</p> <ul style="list-style-type: none"> ▪ [0] Global = All attacks regardless of source are counted together during the threshold window. ▪ [2] IP = Attacks from each specific IP address are counted separately during the threshold window. ▪ [3] IP+Port = Attacks from each specific IP address:port are counted separately during the threshold window. This option is useful for NAT servers, where numerous remote machines use the same IP address but different ports. However, it is not recommended to use this option as it may degrade detection capabilities.
Threshold Window CLI: threshold-window [IDSRule_ThresholdWindow]	<p>Defines the threshold interval (in seconds) during which the device counts the attacks to check if a threshold is crossed. The counter is automatically reset at the end of the interval.</p> <p>The valid range is 1 to 1,000,000. The default is 1.</p>

Parameter	Description
Minor-Alarm Threshold CLI: minor-alm-thr [IDSRule_MinorAlarmThreshold]	Defines the threshold that if crossed a minor severity alarm is sent. The valid range is 1 to 1,000,000. A value of 0 or -1 means not defined.
Major-Alarm Threshold CLI: major-alm-thr [IDSRule_MajorAlarmThreshold]	Defines the threshold that if crossed a major severity alarm is sent. The valid range is 1 to 1,000,000. A value of 0 or -1 means not defined.
Critical-Alarm Threshold CLI: critical-alm-thr [IDSRule_CriticalAlarmThreshold]	Defines the threshold that if crossed a critical severity alarm is sent. The valid range is 1 to 1,000,000. A value of 0 or -1 means not defined.

12.3.3 Assigning IDS Policies

The IDS Match table enables you to use your configured IDS policies. This is done by assigning them to any or a combination of the following entities:

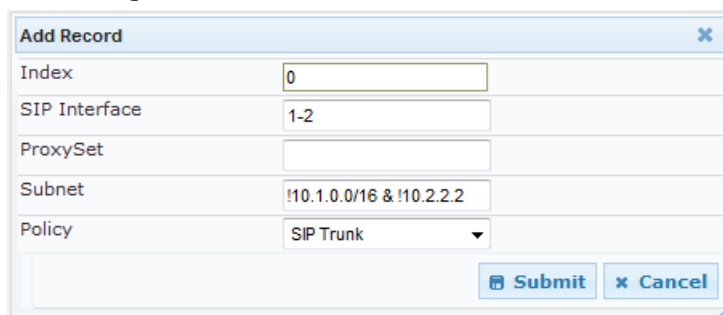
- **SIP Interface:** Detects malicious attacks (according to specified IDS Policy) on specific SIP Interface(s)
- **Proxy Sets:** Detects malicious attacks (according to specified IDS Policy) from specified Proxy Set(s)
- **Subnet addresses:** Detects malicious attacks (according to specified IDS Policy) from specified subnet address

Up to 20 IDS policy-matching rules can be configured.

➤ **To assign an IDS policy:**

1. Open the IDS Match Table page (**Configuration** tab > **VoIP** menu > **Security** > **Intrusion Detection and Prevention** > **Match Table**).
2. Click **Add**.

Figure 12-7: IDS Match Table - Add Record



The figure above shows a configuration example where the IDS Policy, "SIP Trunk" is applied to SIP Interfaces 1 and 2, and all source IP addresses outside of subnet 10.1.0.0/16 and IP address 10.2.2.2.

3. Configure the IDS matching parameters. For a description of these parameters, see the following table.
4. Click **Submit**.

Table 12-5: IDS Match Table Parameters

Parameter	Description
Index CLI: <code>!IDSMatch_Index</code>	Defines the table row number for the rule.
SIP Interface CLI: <code>!IDSMatch_SIPInterface</code>	<p>Defines the SIP Interface(s) to which you want to assign the IDS policy. This indicates the SIP Interfaces that are being attacked. The entered value must be the ID of the SIP Interface. The following syntax is supported:</p> <ul style="list-style-type: none"> A comma-separated list of SIP Interface IDs (e.g., 1,3,4) A hyphen "-" indicates a range of SIP Interfaces (e.g., 3,4-7 means IDs 3, and 4 through 7) A prefix of an exclamation mark "!" means negation of the set (e.g., !3,4-7 means all indexes excluding 3, and excluding 4 through 7)
ProxySet CLI: <code>!IDSMatch_ProxySet</code>	<p>Defines the Proxy Set(s) to which the IDS policy is assigned. This indicates the Proxy Sets from where the attacks are coming from. The following syntax is supported:</p> <ul style="list-style-type: none"> A comma-separated list of Proxy Set IDs (e.g., 1,3,4) A hyphen "-" indicates a range of Proxy Sets (e.g., 3,4-7 means IDs 3, and 4 through 7) A prefix of an exclamation mark "!" means negation of the set (e.g., !3,4-7 means all indexes excluding 3, and excluding 4 through 7) <p>Notes:</p> <ul style="list-style-type: none"> Only the IP address of the Proxy Set is considered (not the port). If a Proxy Set has multiple IP addresses, the device considers the Proxy Set as one entity and includes all its IP addresses in the same IDS count.
Subnet CLI: <code>!IDSMatch_Subnet</code>	<p>Defines the subnet(s) to which the IDS policy is assigned. This indicates the subnets from where the attacks are coming from. The following syntax can be used:</p> <ul style="list-style-type: none"> Basic syntax is a subnet in CIDR notation (e.g., 10.1.0.0/16 means all sources with IP address in the range 10.1.0.0–10.1.255.255) An IP address can be specified without the prefix length to refer to the specific IP address. Each subnet can be negated by prefixing it with "!", which means all IP addresses outside that subnet. Multiple subnets can be specified by separating them with "&" (and) or " " (or) operations. For example: <ul style="list-style-type: none"> ✓ 10.1.0.0/16 10.2.2.2: includes subnet 10.1.0.0/16 and IP address 10.2.2.2. ✓ !10.1.0.0/16 & !10.2.2.2: includes all addresses except those of subnet 10.1.0.0/16 and IP address 10.2.2.2. Note that the exclamation mark "!" appears before each subnet. ✓ 10.1.0.0/16 & !10.1.1.1: includes subnet 10.1.0.0/16, except IP address 10.1.1.1.
Policy CLI: <code>!IDSMatch_Policy</code>	Selects the IDS policy, configured in 'Configuring IDS Policies' on page 131.

12.3.4 Viewing IDS Alarms

The device uses SNMP (and Syslog) to notify the detection of malicious attacks. The trap displays the IDS Policy and Rule, and the Policy-Match index.

The device sends the SNMP alarm, `acIDSPolicyAlarm` whenever a threshold of a specific IDS Policy rule is crossed. For each scope that crosses this threshold, the device sends an additional SNMP event (trap) - `acIDSThresholdCrossNotification` - indicating the specific details (IP address or IP address:port). If the trap severity level is raised, the alarm of the former severity is cleared and the device then sends a new alarm with the new severity.

The SNMP alarm is cleared after a user-defined period (configured by the ini file parameter, `IDSAlarmClearPeriod`) during which no thresholds have been crossed. However, this "quiet" period must be at least twice the Threshold Window value (configured in 'Configuring IDS Policies' on page 131). For example, if `IDSAlarmClearPeriod` is set to 20 sec and the Threshold Window is set to 15 sec, the `IDSAlarmClearPeriod` parameter is ignored and the alarm is cleared only after 30 seconds (2 x 15 sec).

The figure below shows an example of IDS alarms in the Active Alarms table (Viewing Active Alarms), where a minor threshold alarm is cleared and replaced by a major threshold alarm:

Figure 12-8: IDS Alarms in Active Alarms Table

17	Minor	Board#1/IDSMatch#2/IDSRule#0	Policy 2 (Proxy): minor threshold (5) of signaling-msg cross in ip scope	24.10.2012 , 9:48:53
18	cleared	Board#1/IDSMatch#2/IDSRule#0	Alarm cleared: Policy 2 (Proxy): minor threshold (5) of signaling-msg cross in ip scope	24.10.2012 , 9:48:53
19	Major	Board#1/IDSMatch#2/IDSRule#0	Policy 2 (Proxy): major threshold (10) of signaling-msg cross in ip scope	24.10.2012 , 9:48:53

You can also view the IDS alarms in the CLI:

- To view active IDS alarms:

```
show voip security ids active-alarm all
```

- To view all IP addresses that crossed the threshold for an active IDS alarm:

```
show voip security ids active-alarm match * rule *
```

The device also sends IDS notifications in Syslog messages to a Syslog server (if enabled - see Configuring Syslog). The table below shows the Syslog text message per malicious event:

Table 12-6: Types of Malicious Events and Syslog Text String

Type	Description	Syslog String
Connection Abuse	TLS authentication failure	abuse-tls-auth-fail
Malformed Messages	<ul style="list-style-type: none"> Message exceeds a user-defined maximum message length (50K) Any SIP parser error Message policy match Basic headers not present Content length header not present (for TCP) Header overflow 	<ul style="list-style-type: none"> malformed-invalid-msg-len malformed-parse-error malformed-message-policy malformed-miss-header malformed-miss-content-len malformed-header-overflow
Authentication Failure	<ul style="list-style-type: none"> Local authentication ("Bad digest" errors) Remote authentication (SIP 401/407 is sent if original message includes authentication) 	<ul style="list-style-type: none"> auth-establish-fail auth-reject-response

Type	Description	Syslog String
Dialog Establishment Failure	<ul style="list-style-type: none">▪ Classification failure▪ Routing failure▪ Other local rejects (prior to SIP 180 response)▪ Remote rejects (prior to SIP 180 response)	<ul style="list-style-type: none">▪ establish-classify-fail▪ establish-route-fail▪ establish-local-reject▪ establish-remote-reject
Abnormal Flow	<ul style="list-style-type: none">▪ Requests and responses without a matching transaction user (except ACK requests)▪ Requests and responses without a matching transaction (except ACK requests)	<ul style="list-style-type: none">▪ flow-no-match-tu▪ flow-no-match-transaction

Reader's Notes

13 Media

This section describes the media-related configuration.

13.1 Configuring Voice Settings

The Voice Settings page configures various voice parameters such as voice volume, silence suppression, and DTMF transport type. For a detailed description of these parameters, see 'Configuration Parameters Reference' on page 437.

➤ **To configure the voice parameters:**

1. Open the Voice Settings page (**Configuration** tab > **VoIP** menu > **Media** submenu > **Voice Settings**).
2. Configure the Voice parameters as required.
3. Click **Submit** to apply your changes.
4. To save the changes to flash memory, see 'Saving Configuration' on page 360.

13.1.1 Configuring Voice Gain (Volume) Control

The device allows you to configure the level of the received (input gain) Tel-to-IP signal and the level of the transmitted (output gain) IP-to-Tel signal. The gain can be set between -32 and 31 decibels (dB).

The procedure below describes how to configure gain control using the Web interface:

➤ **To configure gain control using the Web interface:**

1. Open the Voice Settings page (**Configuration** tab > **VoIP** menu > **Media** submenu > **Voice Settings**).

Figure 13-1: Voice Volume Parameters in Voice Settings Page

Voice Volume (-32 to 31 dB)	<input type="text" value="0"/>
Input Gain (-32 to 31 dB)	<input type="text" value="0"/>

2. Configure the following parameters:
 - 'Voice Volume' (*VoiceVolume*) - Defines the voice gain control (in decibels) for IP-to-Tel
 - 'Input Gain' (*InputGain*) - Defines the PCM input gain control (in decibels) for Tel-to-IP
3. Click **Submit** to apply your settings.

13.1.2 Silence Suppression (Compression)

Silence suppression (compression) is a method for conserving bandwidth on VoIP calls by not sending packets when silence is detected. The device uses its VAD feature to detect periods of silence in the voice channel during an established call. When silence is detected, it stops sending packets in the channel.

The procedure below describes how to enable silence suppression using the Web interface.

➤ **To enable silence suppression using the Web interface:**

1. Open the Voice Settings page (**Configuration** tab > **VoIP** menu > **Media** submenu > **Voice Settings**).

Figure 13-2: Enabling Silence Suppression in Voice Settings Page

Silence Suppression	Enable	
---------------------	--------	---

2. Set the 'Silence Suppression' (*EnableSilenceCompression*) field to **Enable**.
3. Click **Submit** to apply your changes.

13.1.3 Echo Cancellation

The device supports adaptive linear (line) echo cancellation according to G.168-2002. Echo cancellation is a mechanism that removes echo from the voice channel. Echoes are reflections of the transmitted signal.

In this line echo, echoes are generated when two-wire telephone circuits (carrying both transmitted and received signals on the same wire pair) are converted to a four-wire circuit. Echoes are reflections of the transmitted signal, which result from impedance mismatch in the hybrid (bi-directional 2-wire to 4-wire converting device).





An estimated echo signal is built by feeding the decoder output signal to an RLS-like adaptive filter, which adapts itself to the characteristics of the echo path. The 'estimated echo signal' (the output of this filter) is then subtracted from the input signal (which is the sum of the desired input signal and the undesired echo) to provide a clean signal. To suppress the remaining residual echo, a Non Linear Processor (NLP) is used, as well as a double-talk (two people speak at the same time) detector that prevents false adaptation during near-end speech.

The device also supports acoustic echo cancellation for SBC calls. These echoes are composed of undesirable acoustical reflections (non-linear) of the received signal (i.e., from the speaker) which find their way from multiple reflections such as walls and windows into the transmitted signal (i.e., microphone). Therefore, the party at the far end hears his / her echo. The device removes these echoes and sends only the near-end's desired speech signal to the network (i.e., to the far-end party). The echo is composed of a linear part and a nonlinear part. However, in the Acoustic Echo Canceller, a substantial part of the echo is non-linear echo. To support this feature, the Forced Transcoding feature must be enabled so that the device uses DSPs.

The procedure below describes how to configure echo cancellation using the Web interface:

➤ **To configure echo cancellation using the Web interface:**

1. Configure line echo cancellation:
 - a. Open the Voice Settings page (**Configuration** tab > **VoIP** menu > **Media** submenu > **Voice Settings**).

Echo Canceller	Enable	
▼ Acoustic Echo Suppressor Settings		
 Network Echo Suppressor Enable	Disable	
Echo Canceller Type	Line echo canceller	
Attenuaion Intensity	0	
Max ERL Threshold - DB	0	
Min Reference Delay x10 msec	0	
Max Reference Delay x10 msec	40	

- b. Set the 'Echo Canceller' field (*EnableEchoCanceller*) to **Enable**.

2. Enable acoustic echo cancellation for SBC calls:
 - a. In the Voice Settings page, configure the following parameters:
 - ♦ 'Network Echo Suppressor Enable' (AcousticEchoSuppressorSupport) - enables the network Acoustic Echo Suppressor
 - ♦ 'Echo Canceller Type' (EchoCancellerType) - defines the echo canceller type
 - ♦ 'Attenuation Intensity' (AcousticEchoSuppAttenuationIntensity) - defines the acoustic echo suppressor signals identified as echo attenuation intensity
 - ♦ 'Max ERL Threshold' (AcousticEchoSuppMaxERLThreshold) - defines the acoustic echo suppressor maximum ratio between signal level and returned echo from the phone
 - ♦ 'Min Reference Delay' (AcousticEchoSuppMinRefDelayx10ms) - defines the acoustic echo suppressor minimum reference delay
 - ♦ 'Max Reference Delay' (AcousticEchoSuppMaxRefDelayx10ms) - defines the acoustic echo suppressor maximum reference delay
 - b. Open the IP Profile Settings page (Configuration tab > VoIP menu > Coders and Profiles submenu > IP Profile Settings), and set the 'Echo Canceller' field to Acoustic.
 - c. Enable the Forced Transcoding feature (using the TranscodingMode parameter) to allow the device to use DSP channels, which are required for acoustic echo cancellation.



Note: The following additional echo cancellation parameters are configurable only through the *ini* file:

- *ECHybridLoss* - defines the four-wire to two-wire worst-case Hybrid loss
- *ECNLPMODE* - defines the echo cancellation Non-Linear Processing (NLP) mode
- *EchoCancellerAggressiveNLP* - enables Aggressive NLP at the first 0.5 second of the call

13.2 Fax and Modem Capabilities

This section describes the device's fax and modem capabilities and corresponding configuration. The fax and modem configuration is done in the Fax/Modem/CID Settings page.



Notes:

- Unless otherwise specified, the configuration parameters mentioned in this section are available on this page.
- Some SIP parameters override these fax and modem parameters. For example, the *IsFaxUsed* parameter and V.152 parameters in Section 'V.152 Support' on page 151).
- For a detailed description of the parameters appearing on this page, see 'Configuration Parameters Reference' on page 437.

➤ **To access the fax and modem parameters:**

1. Open the Fax/Modem/CID Settings page (**Configuration** tab > **VoIP** menu > **Media** submenu > **Fax/Modem/CID Settings**).

Figure 13-3: Fax/Modem/CID Settings Page

▼ General Settings		
Fax Transport Mode	RelayEnable	▼
Caller ID Transport Type	Mute	▼
Caller ID Type	Standard Bellcore	▼
V.21 Modem Transport Type	Disable	▼
V.22 Modem Transport Type	Enable Bypass	▼
V.23 Modem Transport Type	Enable Bypass	▼
V.32 Modem Transport Type	Enable Bypass	▼
V.34 Modem Transport Type	Enable Bypass	▼
Fax CNG Mode	Disable	▼
CNG Detector Mode	Disable	▼

▼ Fax Relay Settings		
Fax Relay Redundancy Depth	0	
Fax Relay Enhanced Redundancy Depth	4	
Fax Relay ECM Enable	Enable	▼
Fax Relay Max Rate (bps)	33600bps	▼

▼ Bypass Settings		
Fax/Modem Bypass Coder Type	G711Alaw_64	▼
Fax/Modem Bypass Packing Factor	1	
Fax Bypass Output Gain	0	
Modem Bypass Output Gain	0	

2. Configure the parameters, as required.
3. Click **Submit** to apply your changes.

13.2.1 Fax/Modem Operating Modes

The device supports two modes of operation:

- Fax/modem negotiation that is not performed during the establishment of the call.
- Voice-band data (VBD) mode for V.152 implementation (see 'V.152 Support' on page 151): fax/modem capabilities are negotiated between the device and the remote endpoint at the establishment of the call. During a call, when a fax/modem signal is detected, transition from voice to VBD (or T.38) is automatically performed and no additional SIP signaling is required. If negotiation fails (i.e., no match is achieved for any of the transport capabilities), fallback to existing logic occurs (according to the parameter IsFaxUsed).

13.2.2 Fax/Modem Transport Modes

The device supports the following transport modes for fax per modem type (V.22/V.23/Bell/V.32/V.34):

- T.38 fax relay (see 'T.38 Fax Relay Mode' on page 143)
- G.711 Transport: switching to G.711 when fax/modem is detected (see 'G.711 Fax / Modem Transport Mode' on page 144)
- Fax fallback to G.711 if T.38 is not supported (see 'Fax Fallback' on page 145)
- Fax and modem bypass: a proprietary method that uses a high bit rate coder (see 'Fax/Modem Bypass Mode' on page 145)
- NSE Cisco's Pass-through bypass mode for fax and modem (see 'Fax / Modem NSE Mode' on page 147)
- Transparent with events: passing the fax / modem signal in the current voice coder with adaptations (see 'Fax / Modem Transparent with Events Mode' on page 148)
- Transparent: passing the fax / modem signal in the current voice coder (see 'Fax / Modem Transparent Mode' on page 148)
- RFC 2833 ANS Report upon Fax/Modem Detection (see 'RFC 2833 ANS Report upon Fax/Modem Detection' on page 149)

'Adaptations' refer to automatic reconfiguration of certain DSP features for handling fax/modem streams differently than voice.

13.2.2.1 T.38 Fax Relay Mode

In Fax Relay mode, fax signals are transferred using the T.38 protocol. T.38 is an ITU standard for sending fax across IP networks in real-time mode. The device currently supports only the T.38 UDP syntax.

T.38 can be configured in the following ways:

- Switching to T.38 mode using SIP Re-INVITE messages (see 'Switching to T.38 Mode using SIP Re-INVITE' on page 143)
- Automatically switching to T.38 mode without using SIP Re-INVITE messages (see 'Automatically Switching to T.38 Mode without SIP Re-INVITE' on page 144)

When fax transmission ends, the reverse switching from fax relay to voice is automatically performed at both the local and remote endpoints.

You can change the fax rate declared in the SDP, using the 'Fax Relay Max Rate' parameter (FaxRelayMaxRate). This parameter does not affect the actual transmission rate. You can also enable or disable Error Correction Mode (ECM) fax mode using the 'Fax Relay ECM Enable' parameter (FaxRelayECMEnable).

When using T.38 mode, you can define a redundancy feature to improve fax transmission over congested IP networks. This feature is activated using the 'Fax Relay Redundancy Depth' parameter (FaxRelayRedundancyDepth) and the 'Fax Relay Enhanced Redundancy Depth' parameter (FaxRelayEnhancedRedundancyDepth). Although this is a proprietary redundancy scheme, it should not create problems when working with other T.38 decoders.

13.2.2.1.1 Switching to T.38 Mode using SIP Re-INVITE

In the Switching to T.38 Mode using SIP Re-INVITE mode, upon detection of a fax signal the terminating device negotiates T.38 capabilities using a Re-INVITE message. If the far-end device doesn't support T.38, the fax fails. In this mode, the 'Fax Transport Mode' parameter (FaxTransportMode) is ignored.

➤ **To configure T.38 mode using SIP Re-INVITE messages:**

1. In the SIP General Parameters page (**Configuration** tab > **VoIP** menu > **SIP Definitions > General Parameters**), set the 'Fax Signaling Method' parameter to **T.38 Relay** (IsFaxUsed = 1).
2. In the Fax/Modem/CID Settings page, configure the following optional parameters:
 - 'Fax Relay Redundancy Depth' (FaxRelayRedundancyDepth)
 - 'Fax Relay Enhanced Redundancy Depth' (FaxRelayEnhancedRedundancyDepth)
 - 'Fax Relay ECM Enable' (FaxRelayECMEnable)
 - 'Fax Relay Max Rate' (FaxRelayMaxRate)



Note: The terminating gateway sends T.38 packets immediately after the T.38 capabilities are negotiated in SIP. However, the originating device by default, sends T.38 (assuming the T.38 capabilities are negotiated in SIP) only after it receives T.38 packets from the remote device. This default behavior cannot be used when the originating device is located behind a firewall that blocks incoming T.38 packets on ports that have not yet received T.38 packets from the internal network. To resolve this problem, the device should be configured to send CNG packets in T.38 upon CNG signal detection (CNGDetectorMode = 1).

13.2.2.1.2 Automatically Switching to T.38 Mode without SIP Re-INVITE

In the Automatically Switching to T.38 Mode without SIP Re-INVITE mode, when a fax signal is detected, the channel automatically switches from the current voice coder to answer tone mode and then to T.38-compliant fax relay mode.

➤ **To configure automatic T.38 mode:**

1. In the SIP General Parameters page (**Configuration** tab > **VoIP** menu > **SIP Definitions > General Parameters**), set the 'Fax Signaling Method' parameter to **No Fax** (IsFaxUsed = 0).
2. In the Fax/Modem/CID Settings page, set the 'Fax Transport Mode' parameter to **RelayEnable** (FaxTransportMode = 1).
3. Configure the following optional parameters:
 - 'Fax Relay Redundancy Depth' (FaxRelayRedundancyDepth)
 - 'Fax Relay Enhanced Redundancy Depth' (FaxRelayEnhancedRedundancyDepth)
 - 'Fax Relay ECM Enable' (FaxRelayECMEnable)
 - 'Fax Relay Max Rate' (FaxRelayMaxRate)

13.2.2.2 G.711 Fax / Modem Transport Mode

In this mode, when the terminating device detects fax or modem signals (CED or AnsAM), it sends a Re-INVITE message to the originating device, requesting it to re-open the channel in G.711 VBD with the following adaptations:

- Echo Celler = off
- Silence Compression = off
- Echo Celler Non-Linear Processor Mode = off
- Dynamic Jitter Buffer Minimum Delay = 40
- Dynamic Jitter Buffer Optimization Factor = 13

After a few seconds upon detection of fax V.21 preamble or super G3 fax signals, the device sends a second Re-INVITE enabling the echo canceller (the echo canceller is disabled only on modem transmission).

A 'gpmid' attribute is added to the SDP according to the following format:

- **For G.711 A-law:**

```
a=gpmid:0 vbd=yes;ecan=on (or off for modems)
```

- **For G.711 μ -law:**

```
a=gpmid:8 vbd=yes;ecan=on (or off for modems)
```

The following parameters are ignored and automatically set to **Events Only**:

- 'Fax Transport Mode' (FaxTransportMode)
- 'Vxx ModemTransportType' (VxxModemTransportType)

➤ **To configure fax / modem transparent mode:**

- In the SIP General Parameters page (**Configuration** tab > **VoIP** menu > **SIP Definitions** > **General Parameters**), set the 'Fax Signaling Method' parameter to **G.711 Transport** (IsFaxUsed = 2).

13.2.2.3 Fax Fallback

In this mode, when the terminating device detects a fax signal, it sends a Re-INVITE message to the originating device with T.38. If the remote device doesn't support T.38 (replies with SIP response 415 "Media Not Supported"), the device sends a new Re-INVITE with G.711 VBD with the following adaptations:

- Echo Canceller = on
- Silence Compression = off
- Echo Canceller Non-Linear Processor Mode = off
- Dynamic Jitter Buffer Minimum Delay = 40
- Dynamic Jitter Buffer Optimization Factor = 13

When the device initiates a fax session using G.711, a 'gpmid' attribute is added to the SDP according to the following format:

- **For G.711A-law:**

```
a=gpmid:0 vbd=yes;ecan=on
```

- **For G.711 μ -law:**

```
a=gpmid:8 vbd=yes;ecan=on
```

In this mode, the 'Fax Transport Mode' (FaxTransportMode) parameter is ignored and automatically set to **Disable** (transparent mode).

➤ **To configure fax fallback mode:**

- In the SIP General Parameters page (**Configuration** tab > **VoIP** menu > **SIP Definitions** > **General Parameters**), set the 'Fax Signaling Method' parameter to **Fax Fallback** (IsFaxUsed = 3).

13.2.2.4 Fax/Modem Bypass Mode

In this proprietary mode, when fax or modem signals are detected, the channel automatically switches from the current voice coder to a high bit-rate coder, according to the 'Fax/Modem Bypass Coder Type' parameter (FaxModemBypassCoderType). The channel is also automatically reconfigured with the following fax / modem adaptations:

- Disables silence suppression

- Enables echo cancellation for fax
- Disables echo cancellation for modem
- Performs certain jitter buffering optimizations

The network packets generated and received during the bypass period are regular voice RTP packets (per the selected bypass coder), but with a different RTP payload type according to the following parameters:

- 'Fax Bypass Payload Type' (FaxBypassPayloadType)
- ModemBypassPayloadType (ini file)

During the bypass period, the coder uses the packing factor, configured by the 'Fax/Modem Bypass Packing Factor' parameter (FaxModemBypassM). The packing factor determines the number of coder payloads (each the size of FaxModemBypassBasicRTPPacketInterval) that are used to generate a single fax/modem bypass packet. When fax/modem transmission ends, the reverse switching, from bypass coder to regular voice coder is performed.

➤ **To configure fax / modem bypass mode:**

1. In the SIP General Parameters page (**Configuration** tab > **VoIP** menu > **SIP Definitions** > **General Parameters**), set the 'Fax Signaling Method' parameter to **No Fax** (IsFaxUsed = 0).
2. In the Fax/Modem/CID Settings page, do the following:
 - a. Set the 'Fax Transport Mode' parameter to **Bypass** (FaxTransportMode = 2).
 - b. Set the 'V.21 Modem Transport Type' parameter to **Enable Bypass** (V21ModemTransportType = 2).
 - c. Set the 'V.22 Modem Transport Type' parameter to **Enable Bypass** (V22ModemTransportType = 2).
 - d. Set the 'V.23 Modem Transport Type' parameter to **Enable Bypass** (V23ModemTransportType = 2).
 - e. Set the 'V.32 Modem Transport Type' parameter to **Enable Bypass** (V32ModemTransportType = 2).
 - f. Set the 'V.34 Modem Transport Type' parameter to **Enable Bypass** (V34ModemTransportType = 2).
3. Set the ini file parameter, BellModemTransportType to 2 (Bypass).
4. Configure the following optional parameters:
 - 'Fax/Modem Bypass Coder Type' (FaxModemBypassCoderType).
 - 'Fax Bypass Payload Type' (FaxBypassPayloadType) - in the RTP/RTCP Settings page (**Configuration** tab > **VoIP** menu > **Media** > **RTP/RTCP Settings**).
 - ModemBypassPayloadType (ini file).
 - FaxModemBypassBasicRTPPacketInterval (ini file).
 - FaxModemBypassDJBufMinDelay (ini file).



Note: When the device is configured for modem bypass and T.38 fax, V.21 low-speed modems are not supported and fail as a result.



Tip: When the remote (non-AudioCodes) gateway uses the G.711 coder for voice and doesn't change the coder payload type for fax or modem transmission, it is recommended to use the Bypass mode with the following configuration:

- EnableFaxModemInbandNetworkDetection = 1.
- 'Fax/Modem Bypass Coder Type' = same coder used for voice.
- 'Fax/Modem Bypass Packing Factor'(FaxModemBypassM) = same interval as voice.
- ModemBypassPayloadType = 8 if voice coder is A-Law or 0 if voice coder is Mu-Law.

13.2.2.5 Fax / Modem NSE Mode

In this mode, fax and modem signals are transferred using Cisco-compatible Pass-through bypass mode. Upon detection of fax or modem answering tone signal, the terminating device sends three to six special NSE RTP packets (configured by the NSEpayloadType parameter; usually to 100). These packets signal the remote device to switch to G.711 coder, according to the 'Fax/Modem Bypass Packing Factor' parameter. After a few NSE packets are exchanged between the devices, both devices start using G.711 packets with standard payload type (8 for G.711 A-Law and 0 for G.711 Mu-Law). In this mode, no Re-INVITE messages are sent. The voice channel is optimized for fax/modem transmission (same as for usual bypass mode).

The parameters defining payload type for AudioCodes proprietary Bypass mode -- 'Fax Bypass Payload Type' (RTP/RTCP Settings page) and ModemBypassPayloadType (ini file) -- are not used with NSE Bypass.

When configured for NSE mode, the device includes in its SDP the following line:

```
a=rtpmap:100 X-NSE/8000
```

Where 100 is the NSE payload type.

The Cisco gateway must include the following definition:

```
modem passthrough nse payload-type 100 codec g711alaw
```

➤ To configure NSE mode:

1. In the SIP General Parameters page (**Configuration** tab > **VoIP** menu > **SIP Definitions** > **General Parameters**), set the 'Fax Signaling Method' parameter to **No Fax** (IsFaxUsed = 0).
2. In the Fax/Modem/CID Settings page, do the following:
 - a. Set the 'Fax Transport Mode' parameter to **Bypass** (FaxTransportMode = 2).
 - b. Set the 'V.21 Modem Transport Type' parameter to **Enable Bypass** (V21ModemTransportType = 2).
 - c. Set the 'V.22 Modem Transport Type' parameter to **Enable Bypass** (V22ModemTransportType = 2).
 - d. Set the 'V.23 Modem Transport Type' parameter to **Enable Bypass** (V23ModemTransportType = 2).
 - e. Set the 'V.32 Modem Transport Type' parameter to **Enable Bypass** (V32ModemTransportType = 2).
 - f. Set the 'V.34 Modem Transport Type' parameter to **Enable Bypass** (V34ModemTransportType = 2).
3. Set the ini file parameter, BellModemTransportType to 2 (Bypass).
4. Set the ini file parameter, NSEMode parameter to 1 (enables NSE).
5. Set the ini file parameter, NSEPayloadType parameter to 100.

13.2.2.6 Fax / Modem Transparent with Events Mode

In this mode, fax and modem signals are transferred using the current voice coder with the following automatic adaptations:

- Echo Canceller = on (or off for modems)
- Echo Canceller Non-Linear Processor Mode = off
- Jitter buffering optimizations

➤ **To configure fax / modem transparent with events mode:**

1. In the SIP General Parameters page (**Configuration** tab > **VoIP** menu > **SIP Definitions** > **General Parameters**), set the 'Fax Signaling Method' parameter to **No Fax** (IsFaxUsed = 0).
2. In the Fax/Modem/CID Settings page, do the following:
 - a. Set the 'Fax Transport Mode' parameter to **Events Only** (FaxTransportMode = 3).
 - b. Set the 'V.21 Modem Transport Type' parameter to **Events Only** (V21ModemTransportType = 3).
 - c. Set the 'V.22 Modem Transport Type' parameter to **Events Only** (V22ModemTransportType = 3).
 - d. Set the 'V.23 Modem Transport Type' parameter to **Events Only** (V23ModemTransportType = 3).
 - e. Set the 'V.32 Modem Transport Type' parameter to **Events Only** (V32ModemTransportType = 3).
 - f. Set the 'V.34 Modem Transport Type' parameter to **Events Only** (V34ModemTransportType = 3).
3. Set the ini file parameter, BellModemTransportType to 3 (transparent with events).

13.2.2.7 Fax / Modem Transparent Mode

In this mode, fax and modem signals are transferred using the current voice coder without notifications to the user and without automatic adaptations. It's possible to use Profiles (see 'Coders and Profiles' on page 217) to apply certain adaptations to the channel used for fax / modem. For example, to use the coder G.711, to set the jitter buffer optimization factor to 13, and to enable echo cancellation for fax and disable it for modem.

➤ **To configure fax / modem transparent mode:**

1. In the SIP General Parameters page (**Configuration** tab > **VoIP** menu > **SIP Definitions** > **General Parameters**), set the 'Fax Signaling Method' parameter to **No Fax** (IsFaxUsed = 0).
2. In the Fax/Modem/CID Settings page, do the following:
 - a. Set the 'Fax Transport Mode' parameter to **Disable** (FaxTransportMode = 0).
 - b. Set the 'V.21 Modem Transport Type' parameter to **Disable** (V21ModemTransportType = 0).
 - c. Set the 'V.22 Modem Transport Type' parameter to **Disable** (V22ModemTransportType = 0).
 - d. Set the 'V.23 Modem Transport Type' parameter to **Disable** (V23ModemTransportType = 0).
 - e. Set the 'V.32 Modem Transport Type' parameter to **Disable** (V32ModemTransportType = 0).
 - f. Set the 'V.34 Modem Transport Type' parameter to **Disable** (V34ModemTransportType = 0).
3. Set the ini file parameter, BellModemTransportType to 0 (transparent mode).

4. Configure the following optional parameters:
 - a. Coders table - (**Configuration** tab > **VoIP** menu > **Coders and Profiles** > **Coders**).
 - b. 'Dynamic Jitter Buffer Optimization Factor' (DJBufOptFactor) - RTP/RTCP Settings page (**Configuration** tab > **VoIP** menu > **Media** > **RTP/RTCP Settings**).
 - c. 'Silence Suppression' (EnableSilenceCompression) - Voice Settings page (**Configuration** tab > **VoIP** menu > **Media** > **Voice Settings**).
 - d. 'Echo Cancellor' (EnableEchoCancellor) - Voice Settings page.



Note: This mode can be used for fax, but is not recommended for modem transmission. Instead, use the Bypass (see 'Fax/Modem Bypass Mode' on page 145) or Transparent with Events modes (see 'Fax / Modem Transparent with Events Mode' on page 148) for modem.

13.2.2.8 RFC 2833 ANS Report upon Fax/Modem Detection

The device (terminator gateway) sends RFC 2833 ANS/ANSam events upon detection of fax and/or modem answer tones (i.e., CED tone). This causes the originator to switch to fax/modem. This parameter is applicable only when the fax or modem transport type is set to bypass, Transparent-with-Events, V.152 VBD, or G.711 transport. When the device is located on the originator side, it ignores these RFC 2833 events

➤ To configure RFC 2833 ANS Report upon fax/modem detection:

1. In the SIP General Parameters page (**Configuration** tab > **VoIP** menu > **SIP Definitions** > **General Parameters**), set the 'Fax Signaling Method' parameter to **No Fax** or **Fax Fallback** (IsFaxUsed = 0 or 3).
2. In the Fax/Modem/CID Settings page, do the following:
 - a. Set the 'Fax Transport Mode' parameter to **Bypass** (FaxTransportMode = 2).
 - b. Set the 'V.xx Modem Transport Type' parameters to **Enable Bypass** (VxxModemTransportType = 2).
3. Set the ini file parameter, FaxModemNTMode to 1 (enables this feature).

13.2.3 V.34 Fax Support

V.34 fax machines can transmit data over IP to the remote side using various methods. The device supports the following modes for transporting V.34 fax data over IP:

- Bypass mechanism for V.34 fax transmission (see 'Bypass Mechanism for V.34 Fax Transmission' on page 150)
- T38 Version 0 relay mode, i.e., fallback to T.38 (see 'Relay Mode for T.30 and V.34 Faxes' on page 150)



Note: The CNG detector is disabled in all the subsequent examples. To disable the CNG detector, set the 'CNG Detector Mode' parameter (CNGDetectorMode) to **Disable**.

13.2.3.1 Bypass Mechanism for V.34 Fax Transmission

In this proprietary scenario, the device uses bypass (or NSE) mode to transmit V.34 faxes, enabling the full utilization of its speed.

➤ **To use bypass mode for T.30 and V.34 faxes:**

1. In the Fax/Modem/CID Settings page, do the following:
 - a. Set the 'Fax Transport Mode' parameter to **Bypass** (FaxTransportMode = 2).
 - b. Set the 'V.22 Modem Transport Type' parameter to **Enable Bypass** (V22ModemTransportType = 2).
 - c. Set the 'V.23 Modem Transport Type' parameter to **Enable Bypass** (V23ModemTransportType = 2).
 - d. Set the 'V.32 Modem Transport Type' parameter to **Enable Bypass** (V32ModemTransportType = 2).
 - e. Set the 'V.34 Modem Transport Type' parameter to **Enable Bypass** (V34ModemTransportType = 2).

➤ **To use bypass mode for V.34 faxes, and T.38 for T.30 faxes:**

1. In the Fax/Modem/CID Settings page, do the following:
 - a. Set the 'Fax Transport Mode' parameter to **Relay** (FaxTransportMode = 1).
 - b. Set the 'V.22 Modem Transport Type' parameter to **Enable Bypass** (V22ModemTransportType = 2).
 - c. Set the 'V.23 Modem Transport Type' parameter to **Enable Bypass** (V23ModemTransportType = 2).
 - d. Set the 'V.32 Modem Transport Type' parameter to **Enable Bypass** (V32ModemTransportType = 2).
 - e. Set the 'V.34 Modem Transport Type' parameter to **Enable Bypass** (V34ModemTransportType = 2).

13.2.3.2 Relay Mode for T.30 and V.34 Faxes

In this scenario, V.34 fax machines are forced to use their backward compatibility with T.30 faxes and operate in the slower T.30 mode.

➤ **To use T.38 mode for V.34 and T.30 faxes:**

1. In the Fax/Modem/CID Settings page, do the following:
 - a. Set the 'Fax Transport Mode' parameter to **Relay** (FaxTransportMode = 1).
 - b. Set the 'V.22 Modem Transport Type' parameter to **Disable** (V22ModemTransportType = 0).
 - c. Set the 'V.23 Modem Transport Type' parameter to **Disable** (V23ModemTransportType = 0).
 - d. Set the 'V.32 Modem Transport Type' parameter to **Disable** (V32ModemTransportType = 0).
 - e. Set the 'V.34 Modem Transport Type' parameter to **Disable** (V34ModemTransportType = 0).

13.2.4 V.152 Support

The device supports the ITU-T recommendation V.152 (Procedures for Supporting Voice-Band Data over IP Networks). Voice-band data (VBD) is the transport of modem, facsimile, and text telephony signals over a voice channel of a packet network with a codec appropriate for such signals.

For V.152 capability, the device supports T.38 as well as VBD codecs (i.e., G.711 A-law and G.711 μ -law). The selection of capabilities is performed using the coders table (see 'Configuring Coders' on page 217).

When in VBD mode for V.152 implementation, support is negotiated between the device and the remote endpoint at the establishment of the call. During this time, initial exchange of call capabilities is exchanged in the outgoing SDP. These capabilities include whether VBD is supported and associated RTP payload types ('gpmid' SDP attribute), supported codecs, and packetization periods for all codec payload types ('ptime' SDP attribute). After this initial negotiation, no Re-INVITE messages are necessary as both endpoints are synchronized in terms of the other side's capabilities. If negotiation fails (i.e., no match was achieved for any of the transport capabilities), fallback to existing logic occurs (according to the parameter IsFaxUsed).

Below is an example of media descriptions of an SDP indicating support for V.152. In the example, V.152 implementation is supported (using the dynamic payload type 96 and G.711 u-law as the VBD codec) as well as the voice codecs G.711 μ -law and G.729.

```
v=0
o=- 0 0 IN IPV4 <IPAddressA>
s=-
t=0 0
p=+1
c=IN IP4 <IPAddressA>
m=audio <udpPort A> RTP/AVP 18 0
a=ptime:10
a=rtpmap:96 PCMU/8000
a=gpmid: 96 vbd=yes
```

Instead of using VBD transport mode, the V.152 implementation can use alternative relay fax transport methods (e.g., fax relay over IP using T.38). The preferred V.152 transport method is indicated by the SDP 'pmft' attribute. Omission of this attribute in the SDP content means that VBD mode is the preferred transport mechanism for voice-band data. To configure T.38 mode, use the CodersGroup parameter.



Note: You can also configure the device to handle G.711 coders received in INVITE SDP offers as VBD coders, using the HandleG711asVBD parameter. For example, if the device is configured with G.729 and G.711 VBD coders and it receives an INVITE with an SDP offer containing G.729 and "regular" G.711 coders, it sends an SDP answer containing G.729 and G.711 VBD coders, allowing subsequent bypass (passthrough) sessions if fax / modem signals are detected during the call.

13.3 Configuring RTP/RTCP Settings

This section describes configuration relating to Real-Time Transport Protocol (RTP) and RTP Control Protocol (RTCP).

13.3.1 Configuring the Dynamic Jitter Buffer

Voice frames are transmitted at a fixed rate. If the frames arrive at the other end at the same rate, voice quality is perceived as good. However, some frames may arrive slightly faster or slower than the other frames. This is called jitter (delay variation) and degrades the perceived voice quality. To minimize this problem, the device uses a jitter buffer. The jitter buffer collects voice packets, stores them and sends them to the voice processor in evenly spaced intervals.

The device uses a dynamic jitter buffer that can be configured with the following:

- **Minimum delay:** Defines the starting jitter capacity of the buffer. For example, at 0 msec, there is no buffering at the start. At the default level of 10 msec, the device always buffers incoming packets by at least 10 msec worth of voice frames.
- **Optimization Factor:** Defines how the jitter buffer tracks to changing network conditions. When set at its maximum value of 12, the dynamic buffer aggressively tracks changes in delay (based on packet loss statistics) to increase the size of the buffer and doesn't decay back down. This results in the best packet error performance, but at the cost of extra delay. At the minimum value of 0, the buffer tracks delays only to compensate for clock drift and quickly decays back to the minimum level. This optimizes the delay performance but at the expense of a higher error rate.

The default settings of 10 msec Minimum delay and 10 Optimization Factor should provide a good compromise between delay and error rate. The jitter buffer 'holds' incoming packets for 10 msec before making them available for decoding into voice. The coder polls frames from the buffer at regular intervals in order to produce continuous speech. As long as delays in the network do not change (jitter) by more than 10 msec from one packet to the next, there is always a sample in the buffer for the coder to use. If there is more than 10 msec of delay at any time during the call, the packet arrives too late. The coder tries to access a frame and is not able to find one. The coder must produce a voice sample even if a frame is not available. It therefore compensates for the missing packet by adding a Bad-Frame-Interpolation (BFI) packet. This loss is then flagged as the buffer being too small. The dynamic algorithm then causes the size of the buffer to increase for the next voice session. The size of the buffer may decrease again if the device notices that the buffer is not filling up as much as expected. At no time does the buffer decrease to less than the minimum size configured by the Minimum delay parameter.

In certain scenarios, the **Optimization Factor is set to 13**: One of the purposes of the Jitter Buffer mechanism is to compensate for clock drift. If the two sides of the VoIP call are not synchronized to the same clock source, one RTP source generates packets at a lower rate, causing under-runs at the remote Jitter Buffer. In normal operation (optimization factor 0 to 12), the Jitter Buffer mechanism detects and compensates for the clock drift by occasionally dropping a voice packet or by adding a BFI packet.

Fax and modem devices are sensitive to small packet losses or to added BFI packets. Therefore, to achieve better performance during modem and fax calls, the Optimization Factor should be set to 13. In this special mode the clock drift correction is performed less frequently - only when the Jitter Buffer is completely empty or completely full. When such condition occurs, the correction is performed by dropping several voice packets simultaneously or by adding several BFI packets simultaneously, so that the Jitter Buffer returns to its normal condition.

The procedure below describes how to configure the jitter buffer using the Web interface.

➤ **To configure jitter buffer using the Web interface:**

1. Open the RTP/RTCP Settings page (**Configuration** tab > **VoIP** menu > **Media** submenu > **RTP/RTCP Settings**). The relevant parameters are listed under the 'General Settings' group, as shown below:

Figure 13-4: Jitter Buffer Parameters in the RTP/RTCP Settings Page

General Settings	
Dynamic Jitter Buffer Minimum Delay	10
Dynamic Jitter Buffer Optimization Factor	10

2. Set the 'Dynamic Jitter Buffer Minimum Delay' parameter (DJBufMinDelay) to the minimum delay (in msec) for the Dynamic Jitter Buffer.
3. Set the 'Dynamic Jitter Buffer Optimization Factor' parameter (DJBufOptFactor) to the Dynamic Jitter Buffer frame error/delay optimization factor.
4. Click **Submit** to apply your settings.

13.3.2 Comfort Noise Generation

The device can generate artificial background noise, called *comfort* noise, in the voice channel during periods of silence (i.e. when no call party is speaking). This is useful in that it reassures the call parties that the call is still connected. The device detects silence using its Voice Activity Detection (VAD) mechanism. When the Calling Tone (CNG) is enabled and silence is detected, the device transmits Silence Identifier Descriptors (SIDs) parameters to reproduce the local background noise at the remote (receiving) side.

The Comfort Noise Generation (CNG) support also depends on the silence suppression (SCE) setting for the coder used in the voice channel. For more information, see the description of the CNG-related parameters.

The procedure below describes how to configure CNG using the Web interface.

➤ **To configure CNG using the Web interface:**

1. Open the RTP/RTCP Settings page (**Configuration** tab > **VoIP** menu > **Media** submenu > **RTP/RTCP Settings**). The relevant parameters are listed under the 'General Settings' group, as shown below:

Figure 13-5: Comfort Noise Parameter in RTP/RTCP Settings Page

Comfort Noise Generation Negotiation	Enable
--------------------------------------	--------

2. Set the 'Comfort Noise Generation Negotiation' parameter (ComfortNoiseNegotiation) to **Enable**.
3. Click **Submit** to apply your changes.

13.3.3 Dual-Tone Multi-Frequency Signaling

This section describes the configuration of Dual-Tone Multi-Frequency (DTMF) signaling.

13.3.3.1 Configuring DTMF Transport Types

The device supports various methods for transporting DTMF digits over the IP network to the remote endpoint. These methods and their configuration are configured in the DTMF & Dialing page (**Configuration** tab > **VoIP** menu > **GW and IP to IP** > **DTMF and Supplementary** > **DTMF & Dialing**):

- **Using INFO message according to Nortel IETF draft:** DTMF digits are sent to the remote side in INFO messages. To enable this mode, define the following:
 - a. Set the 'Declare RFC 2833 in SDP' parameter to **No** (RxDTMFOption = 0).
 - b. Set the '1st Tx DTMF Option' parameter to **INFO (Nortel)** (TxDTMFOption = 1).

Note: In this mode, DTMF digits are removed from the audio stream (and the 'DTMF Transport Type' parameter is automatically set to **Mute DTMF**).
- **Using INFO message according to Cisco's mode:** DTMF digits are sent to the remote side in INFO messages. To enable this mode, define the following:
 - a. Set the 'Declare RFC 2833 in SDP' parameter to **No** (RxDTMFOption = 0).
 - b. Set the '1st Tx DTMF Option' parameter to **INFO (Cisco)** (TxDTMFOption = 3).

Note: In this mode, DTMF digits are removed from the audio stream (and the 'DTMF Transport Type' parameter is automatically set to **Mute DTMF**).
- **Using NOTIFY messages according to IETF Internet-Draft draft-mahy-sipping-signaled-digits-01:** DTMF digits are sent to the remote side using NOTIFY messages. To enable this mode, define the following:
 - a. Set the 'Declare RFC 2833 in SDP' parameter to **No** (RxDTMFOption = 0).
 - b. Set the '1st Tx DTMF Option' parameter to **NOTIFY** (TxDTMFOption = 2).

Note: In this mode, DTMF digits are removed from the audio stream (and the 'DTMF Transport Type' parameter is automatically set to **Mute DTMF**).
- **Using RFC 2833 relay with Payload type negotiation:** DTMF digits are sent to the remote side as part of the RTP stream according to RFC 2833. To enable this mode, define the following:
 - a. Set the 'Declare RFC 2833 in SDP' parameter to **Yes** (RxDTMFOption = 3).
 - b. Set the '1st Tx DTMF Option' parameter to **RFC 2833** (TxDTMFOption = 4).

Note: To set the RFC 2833 payload type with a value other than its default, use the RFC2833PayloadType parameter. The device negotiates the RFC 2833 payload type using local and remote SDP and sends packets using the payload type from the received SDP. The device expects to receive RFC 2833 packets with the same payload type as configured by this parameter. If the remote side doesn't include 'telephony-event' in its SDP, the device sends DTMF digits in transparent mode (as part of the voice stream).
- **Sending DTMF digits (in RTP packets) as part of the audio stream (DTMF Relay is disabled):** This method is typically used with G.711 coders. With other low-bit rate (LBR) coders, the quality of the DTMF digits is reduced. To enable this mode, define the following:
 - a. Set the 'Declare RFC 2833 in SDP' parameter to **No** (RxDTMFOption = 0).
 - b. Set the '1st Tx DTMF Option' parameter to **Not Supported** (TxDTMFOption = 0).
 - c. Set the ini file parameter, DTMFTransportType to 2 (i.e., transparent).
- **Using INFO message according to Korea mode:** DTMF digits are sent to the remote side in INFO messages. To enable this mode, define the following:
 - a. Set the 'Declare RFC 2833 in SDP' parameter to **No** (RxDTMFOption = 0).

- b. Set the '1st Tx DTMF Option' parameter to **INFO (Cisco)** (TxDTMFOption = 3).

Note: In this mode, DTMF digits are removed from the audio stream (and the 'DTMF Transport Type' parameter is automatically set to **Mute DTMF**).



Notes:

- The device is always ready to receive DTMF packets over IP in all possible transport modes: INFO messages, NOTIFY, and RFC 2833 (in proper payload type) or as part of the audio stream.
- To exclude RFC 2833 Telephony event parameter from the device's SDP, set the 'Declare RFC 2833 in SDP' parameter to **No**.

The following parameters affect the way the device handles the DTMF digits:

- TxDTMFOption, RxDTMFOption, RFC2833TxPayloadType, and RFC2833RxPayloadType
- MGCPDTMFDetectionPoint, DTMFVolume, DTMFTransportType, DTMFDigitLength, and DTMFInterDigitInterval

13.3.3.2 Configuring RFC 2833 Payload

The procedure below describes how to configure the RFC 2833 payload using the Web interface:

➤ **To configure RFC 2833 payload using the Web interface:**

1. Open the RTP/RTCP Settings page (**Configuration** tab > **VoIP** menu > **Media** submenu > **RTP/RTCP Settings**). The relevant parameters are listed under the 'General Settings' group, as shown below:

Figure 13-6: RFC 2833 Payload Parameters in RTP/RTCP Settings Page

RTP Redundancy Depth	0
Packing Factor	1
Basic RTP Packet Interval	Default ▼
RFC 2833 TX Payload Type	96
RFC 2833 RX Payload Type	96
RFC 2198 Payload Type	104
Fax Bypass Payload Type	102
Enable RFC 3389 CN Payload Type	Enable ▼

2. Configure the following parameters:
 - 'RTP Redundancy Depth' (RTPRedundancyDepth) - enables the device to generate RFC 2198 redundant packets.
 - 'Enable RTP Redundancy Negotiation' (EnableRTPRedundancyNegotiation) - enables the device to include the RTP redundancy dynamic payload type in the SDP, according to RFC 2198.
 - 'RFC 2833 TX Payload Type' (RFC2833TxPayloadType) - defines the Tx RFC 2833 DTMF relay dynamic payload type.
 - 'RFC 2833 RX Payload Type' (RFC2833RxPayloadType) - defines the Rx RFC 2833 DTMF relay dynamic payload type.
 - 'RFC 2198 Payload Type' (RFC2198PayloadType) - defines the RTP redundancy packet payload type according to RFC 2198.
3. Click **Submit** to apply your settings.

13.3.4 Configuring RTP Base UDP Port

You can configure the range of UDP ports for RTP, RTCP, and T.38. The UDP port range can be configured using media realms in the Media Realm table, allowing you to assign different port ranges (media realms) to different interfaces. However, if you do not use media realms, you can configure the lower boundary of the UDP port used for RTP, RTCP (RTP port + 1) and T.38 (RTP port + 2), using the 'RTP Base UDP Port' (BaseUDPPort) parameter. For example, if the BaseUDPPort is set to 6000, then one channel may use the ports RTP 6000, RTCP 6001, and T.38 6002, while another channel may use RTP 6010, RTCP 6011, and T.38 6012.

The range of possible UDP ports is 6,000 to 64,000 (default base UDP port is 6000). The port range is calculated using the BaseUDPPort parameter as follows: **BaseUDPPort to (BaseUDPPort + <channels -1> * 10)**

The default local UDP ports for audio and fax media streams is calculated using the following formula: **BaseUDPPort + (Channel ID * 10) + Port Offset**

Where the port offsets are as follows:

- **Audio RTP:** 0
- **Audio RTCP:** 1
- **Fax T.38:** 2

For example, the local T.38 UDP port for channel 30 is calculated as follows: **6000 + (30*10) + 2 = 6302**

The maximum (when all channels are required) UDP port range is calculated as follows:

- BaseUDPPort to (BaseUDPPort + 4030*10)



Notes:

- The device allocates the UDP ports randomly to the channels.
- To configure the device to use the same port for both RTP and T.38 packets, set the T38UseRTPPort parameter to 1.
- If you are using Media Realms (see 'Configuring Media Realms' on page 157), the port range configured for the Media Realm must be within this range defined by the BaseUDPPort parameter.

The procedure below describes how to configure the RTP base UDP port using the Web interface.

➤ To configure the RTP base UDP port:

1. Open the RTP/RTCP Settings page (**Configuration** tab > **VoIP** menu > **Media** submenu > **RTP/RTCP Settings**). The relevant parameter is listed under the 'General Settings' group, as shown below:

Figure 13-7: RTP Based UDP Port in RTP/RTCP Settings Page

⚡ RTP Base UDP Port	6000
---------------------	------

2. Set the 'RTP Base UDP Port' parameter to the required value.
3. Click **Submit**.
4. Reset the device for the settings to take effect.

13.4 Configuring Various Codec Attributes

The following codec attributes can be configured in the General Media Settings page:

- AMR coder:
 - 'Payload Format': Defines the AMR payload format type.
- SILK coder (Skype's default audio codec):
 - 'Silk Tx Inband FEC': Enables forward error correction (FEC) for the SILK coder.
 - 'Silk Max Average Bit Rate': Defines the maximum average bit rate for the SILK coder.

For a detailed description of these parameters and for additional codec parameters, see 'Coder Parameters' on page 502.

➤ **To configure general media parameters:**

1. Open the General Media Settings page (**Configuration** tab > **VoIP** menu > **Media** submenu > **General Media Settings**).

Figure 13-8: Coder Attribute Configuration in General Media Settings Page

▲ General Settings	
▼ SILK Coders Settings	
Silk Tx Inband FEC	Disable
Silk Max Average Bit Rate	16000
▼ AMR Bandwidth Efficient Configuration	
AMR Payload Format	Octet Aligned

2. Configure the parameters as required, and then click **Submit**.
3. To save the changes to flash memory, see 'Saving Configuration' on page 360.

13.5 Configuring Media Realms

The Media Realm Table page allows you to define a pool of up to 64 SIP media interfaces, termed *Media Realms*. Media Realms allow you to divide a Media-type interface, which is configured in the Multiple Interface table, into several realms, where each realm is specified by a UDP port range. You can also define the maximum number of sessions per Media Realm. Once configured, Media Realms can be assigned to IP Groups (see 'Configuring IP Groups' on page 191) or SRDs (see 'Configuring SRD Table' on page 185).

Once you have configured a Media Realm, you can configure it with the following:

- Quality of Experience parameters for reporting to AudioCodes SEM server used for monitoring the quality of calls (see 'Configuring Quality of Experience Parameters per Media Realm' on page 159)
- Bandwidth management (see 'Configuring Bandwidth Management per Media Realm' on page 162)



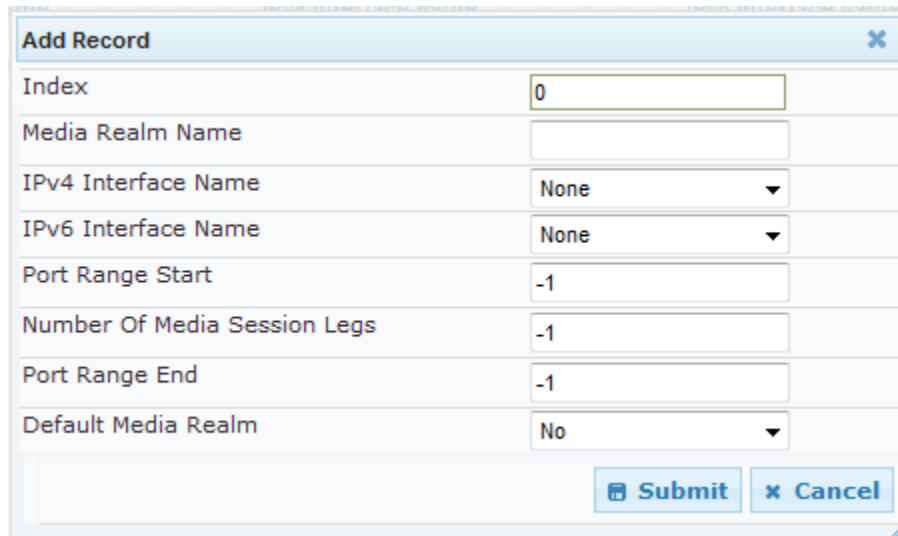
Notes:

- If different Media Realms are assigned to an IP Group and to an SRD, the IP Group's Media Realm takes precedence.
- For this setting to take effect, a device reset is required.
- The Media Realm table can also be configured using the table ini file parameter, CpMediaRealm.

➤ **To define a Media Realm:**

1. Open the Media Realm Table page (**Configuration** tab > **VoIP** menu > **Media** submenu > **Media Realm Configuration**).
2. Click the **Add** button; the following appears:

Figure 13-9: Media Realm Page - Add Record Dialog Box



3. Configure the parameters as required. See the table below for a description of each parameter
4. Click **Submit** to apply your settings.
5. Reset the device to save the changes to flash memory (see 'Saving Configuration' on page 360).

Table 13-1: Media Realm Table Parameter Descriptions

Parameter	Description
Index [CpMediaRealm_Index]	Defines the required table index number.
Media Realm Name [CpMediaRealm_MediaRealmName]	Defines an arbitrary, identifiable name for the Media Realm. The valid value is a string of up to 40 characters. Notes: <ul style="list-style-type: none"> ▪ This parameter is mandatory. ▪ The name assigned to the Media Realm must be unique. ▪ This Media Realm name is used in the SRD and IP Groups table.
IPv4 Interface Name [CpMediaRealm_IPv4IF]	Assigns an IPv4 interface to the Media Realm. This is name of the interface as configured for the Interface Name field in the Multiple Interface table.
IPv6 Interface Name [CpMediaRealm_IPv6IF]	Assigns an IPv6 interface to the Media Realm. This is name of the interface as configured for the Interface Name field in the Multiple Interface table.
Port Range Start [CpMediaRealm_PortRangeStart]	Defines the starting port for the range of Media interface UDP ports. Notes: <ul style="list-style-type: none"> ▪ You must either configure all media realms with port ranges or all without; not some with and some without. ▪ The available UDP port range is calculated using the

Parameter	Description
	BaseUDPport parameter: <ul style="list-style-type: none"> ✓ BaseUDPport to BaseUDPport + 4030*10 ▪ Port ranges over 60,000 must not be used.
Number of Media Session Legs [CpMediaRealm_MediaSessionLeg]	Defines the number of media sessions associated with the range of ports. This is the number of media sessions available in the port range. For example, 100 ports correspond to 10 media sessions, since ports are allocated in chunks of 10.
Port Range End [CpMediaRealm_PortRangeEnd]	Read-only field displaying the ending port for the range of Media interface UDP ports. This field is calculated by adding the 'Media Session Leg' field (multiplied by the port chunk size) to the 'Port Range Start' field. A value appears once a row has been successfully added to the table.
Is Default [CpMediaRealm_IsDefault]	Defines the Media Realm as the default Media Realm. This default Media Realm is used when no Media Realm is configured for an IP Group or SRD for a specific call. <ul style="list-style-type: none"> ▪ [0] No (default) ▪ [1] Yes Notes: <ul style="list-style-type: none"> ▪ This parameter can be set to Yes for only one defined Media Realm. ▪ If this parameter is not configured, then the first Media Realm in the table is used as the default. ▪ If the table is not configured, then the default Media Realm includes all the configured media interfaces.

13.5.1 Configuring Quality of Experience per Media Realm

You can configure Quality of Experience (QoE) per Media Realm. This enables you to monitor and analyze media and signaling traffic, allowing you to detect problems causing service degradation. The device can save call information and statistics at call start, at call end, or at specific changes in the call. The information is stored as call records on an external server. The device connects, as a client, to the server using TLS over TCP.

You can specify the call parameters to monitor and configure their upper and lower thresholds. If these thresholds are exceeded, the device can be configured to do the following:

- Reports the change in the monitored parameter to the monitoring server (default).
- Sends RFC 2198 RTP redundancy packets on the call leg that crossed the threshold. This enables the device to adapt to the changed network status. In this option, you can also configure the redundancy depth. The channel configuration is unchanged if the change requires channel reopening. Currently, this option is applicable only when the monitored parameter is remote packet loss.

The device can be configured to monitor the following parameters on the local (i.e., at the device) or remote side:

- Packet loss
- Mean Opinion Score (MOS)
- Jitter
- Packet delay
- Residual Echo Return Loss (RERL)

At any given time during a call, each of these parameters can be in one of the following states according to its value in the last RTCP / RTCP XR packet:

- Gray - indicates that the value is unknown
- Green - indicates good call quality
- Yellow - indicates medium call quality
- Red - indicates poor call quality

The mapping between the values of the parameters and the color is according to the configured threshold of these parameters, per Media Realm. The call itself also has a state (color), which is the worst-state color of all the monitored parameters. Each time a color of a parameter changes, the device sends a report to the external server. A report is also sent at the end of each call.



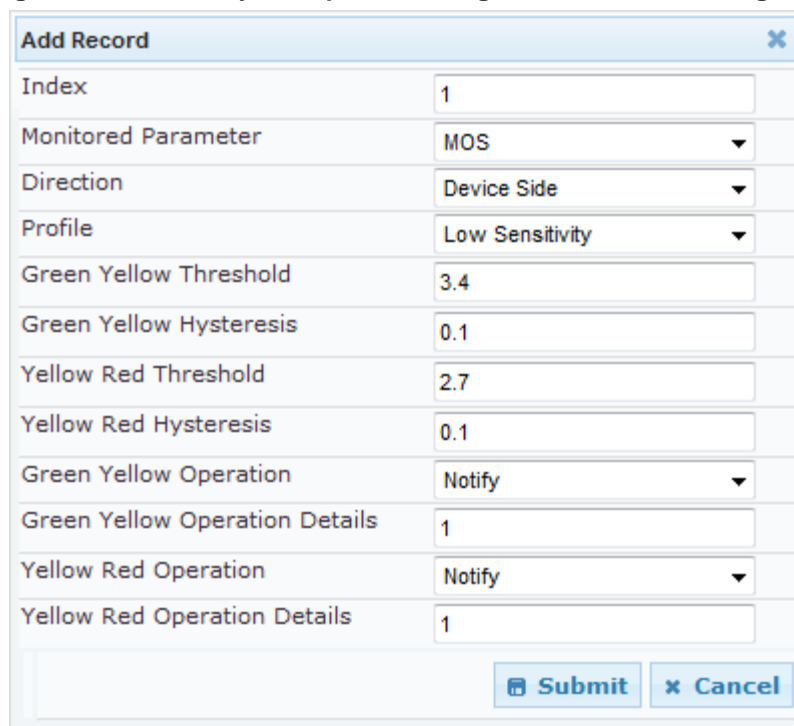
Notes:

- The QoE feature is available only if the device is installed with the relevant Software License Key.
- To configure the address of the AudioCodes Session Experience Manager (SEM) server to where the device reports the QoE, see 'Configuring SEM Server for Media Quality of Experience' on page 164.
- You can also configure QoE per Media Realm using the table *ini* file parameter QOERules.

➤ To configure QoE per Media Realm:

1. Open the Media Realm Table page (**Configuration** tab > **VoIP** menu > **Media** submenu > **Media Realm Configuration**).
2. Select the Media Realm for which you want to configure Quality of Experience, and then click the **Quality Of Experience** link; the Quality Of Experience page appears.
3. Click the **Add** button; the following dialog box appears:

Figure 13-10: Quality of Experience Page - Add Record Dialog Box



Index	1
Monitored Parameter	MOS
Direction	Device Side
Profile	Low Sensitivity
Green Yellow Threshold	3.4
Green Yellow Hysteresis	0.1
Yellow Red Threshold	2.7
Yellow Red Hysteresis	0.1
Green Yellow Operation	Notify
Green Yellow Operation Details	1
Yellow Red Operation	Notify
Yellow Red Operation Details	1

Submit Cancel

The figure above shows value thresholds for the MOS parameter, which are assigned using pre-configured values of the Low Sensitivity profile. In this example setting, if the MOS value changes by 0.1 (hysteresis) to 3.3 or 3.5, the device sends a report to the SEM indicating this change. If the value changes to 3.3, it sends a yellow state (i.e., medium quality); if the value changes to 3.5, it sends a green state.

4. Configure the parameters as required. See the table below for a description of each parameter.
5. Click **Submit** to apply your settings.

Table 13-2: Quality of Experience Parameter Descriptions

Parameter	Description
Index [QOERules_RuleIndex]	Defines the table index entry. Up to four table row entries can be configured per Media Realm.
Monitored Parameter [QOERules_MonitoredParam]	Defines the parameter to monitor and report. <ul style="list-style-type: none"> ▪ [0] MOS (default) ▪ [1] Delay ▪ [2] Packet Loss ▪ [3] Jitter ▪ [4] RERL
Direction [QOERules_Direction]	Defines the monitoring direction. <ul style="list-style-type: none"> ▪ [0] Device Side (default) ▪ [1] Remote Side
Profile [QOERules_Profile]	Defines the pre-configured threshold profile to use. <ul style="list-style-type: none"> ▪ [0] No Profile = No profile is used and you need to define the thresholds in the parameters described below. ▪ [1] Low Sensitivity = Automatically sets the thresholds to low sensitivity values. Therefore, reporting is done only if changes in parameters' values is significant. ▪ [2] Default Sensitivity = Automatically sets the thresholds to a medium sensitivity. ▪ [3] High Sensitivity = Automatically sets the thresholds to high sensitivity values. Therefore, reporting is done for small fluctuations in parameters' values.
Green Yellow Threshold [QOERules_GreenYellowThreshold]	Defines the parameter threshold values between green (good quality) and yellow (medium quality) states.
Green Yellow Hysteresis [QOERules_GreenYellowHysteresis]	Defines the hysteresis (fluctuation) for the green-yellow threshold. When the threshold is exceeded by this hysteresis value, the device sends a report to the SEM indicating this change.
Yellow Red Threshold [QOERules_YellowRedThreshold]	Defines the parameter threshold values between yellow (medium quality) and red (poor quality). When this threshold is exceeded, the device sends a report to the SEM indicating this change.
Yellow Red Hysteresis [QOERules_YellowRedHysteresis]	Defines the hysteresis (fluctuation) for the yellow-red threshold. When the threshold is exceeded by this hysteresis value, the device sends a report to the SEM indicating this change.

Parameter	Description
Green Yellow Operation [QOERules_GreenYellowOperation]	<p>Defines the action that is done if the green-yellow threshold is crossed.</p> <ul style="list-style-type: none"> [1] Notify = (Default) Device sends a report to the SEM server. [2] Activate 2198 = RTP redundancy packets are sent to the relevant call leg. <p>Note: This field is applicable only if the monitored parameter is remote packet loss.</p>
Green Yellow Operation Details [QOERules_GreenYellowOperationDetails]	<p>Note: This field is currently not supported.</p> <p>Defines the desired RTP redundancy depth. The actual redundancy depth on the relevant call leg is the minimum between the desired depth and the maximum supported depth on that call leg.</p> <p>Note: This field is applicable only if the 'Green Yellow Operation' field is set to Activate 2198.</p>
Yellow Red Operation [QOERules_YellowRedOperation]	<p>Note: This field is currently not supported.</p> <p>Defines the action that is done if the yellow-red threshold is crossed.</p> <ul style="list-style-type: none"> [1] Notify = (Default) Device sends a report to the SEM server. [2] Activate 2198 = RTP redundancy packets are sent to the relevant call leg. Note: This field is applicable only if the monitored parameter is remote packet loss.
Yellow Red Operation Details [QOERules_YellowRedOperationDetails]	<p>Note: This field is currently not supported.</p> <p>Defines the desired RTP redundancy depth. The actual redundancy depth on the relevant call leg is the minimum between the desired depth and the maximum supported depth on that call leg.</p> <p>Note: This field is applicable only if the 'Yellow Red Operation' field is set to Activate 2198.</p>

13.5.2 Configuring Bandwidth Management per Media Realm

Bandwidth management enables you to configure bandwidth utilization thresholds per Media Realm which when exceeded, the device can do one of the following:

- Generate an appropriate SNMP alarm, which is cleared when the bandwidth utilization returns to normal.
- Block any additional calls on the Media Realm.

Bandwidth management includes the following bandwidth utilization states:

- Normal
- High threshold
- Critical threshold

When a transition occurs between two bandwidth threshold states, based on threshold and hysteresis values, the device executes the configured action. The transition possibilities include Normal-High threshold state changes and High-Critical threshold state changes. Thus, up to two thresholds can be configured per Media Realm; one for each state transition.

**Notes:**

- This feature is available only if the device is installed with the relevant Software License Key.
- For your bandwidth management settings to take effect, you must reset the device.
- You can also use the BWManagement *ini* file parameter to configure bandwidth management per Media Realm.

➤ **To configure bandwidth management rules per Media Realm:**

1. Open the Media Realm Table page (**Configuration** tab > **VoIP** menu > **Media** submenu > **Media Realm Configuration**).
2. Select the Media Realm for which you want to configure bandwidth management rules, and then click the **Bandwidth Management** link; the Bandwidth Management page appears.
3. Click the **Add** button; the following dialog box appears:

Figure 13-11: Bandwidth Management Page - Add record Dialog Box

The figure above shows an example where if the bandwidth for this Media Realm reaches 41,000 Bps (i.e., 40,000 plus 1,000 hysteresis), the device blocks any additional calls. If the bandwidth later decreases to 39,000 Bps (i.e., 40,000 minus 1,000 hysteresis), the device allows additional calls.

4. Configure the parameters as required. See the table below for a description of each parameter.
5. Click **Submit** to apply your settings.
6. Reset the device for your settings to take effect.

Table 13-3: Bandwidth Management Parameter Descriptions

Parameter	Description
Index BWManagement_ThresholdIndex]	Defines the index of the table row entry. This index determines the bandwidth threshold type for the rule: <ul style="list-style-type: none"> ▪ [0] High Threshold Rule ▪ [1] Critical Threshold Rule
Rule Action [BWManagement_RuleAction]	Defines the action that the device performs when the configured threshold is exceeded: <ul style="list-style-type: none"> ▪ [0] Report Only (default) ▪ [1] No more calls
Threshold [BWManagement_Threshold]	Defines the bandwidth threshold in bytes per second (Bps). The default is 0.

Parameter	Description
Hysteresis [BWManagement_Hysteresis]	Defines the bandwidth fluctuation (change) from the threshold value at which the device performs the configured action. The default is 0.

13.6 Configuring Server for Media Quality of Experience

The device can be configured to report voice (media) quality of experience to AudioCodes Session Experience Manager (SEM) server, a plug-in for AudioCodes EMS. The reports include real-time metrics of the quality of the actual call experience and processed by the SEM.



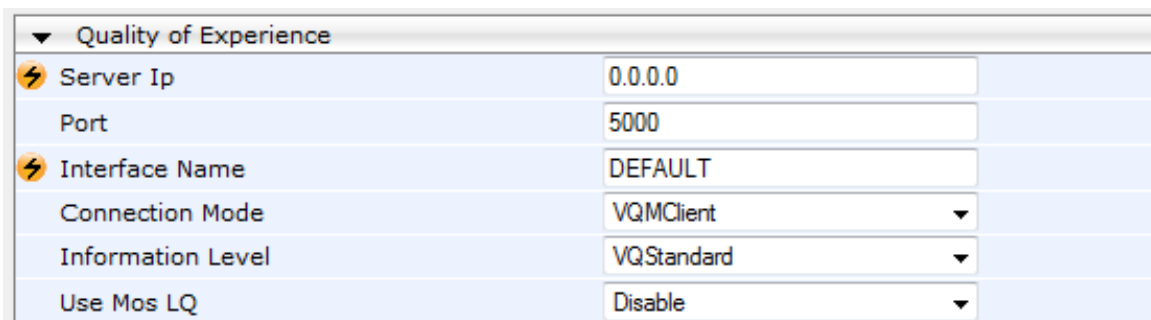
Notes:

- To support this feature, the device must be installed with the relevant Software License Key.
- To configure the parameters to report and their thresholds per Media Realm, see 'Configuring Quality of Experience per Media Realm' on page 159.
- For information on the SEM server, refer to the *EMS User's Manual*.

➤ To configure QoE reporting of media:

1. Open the Media Quality of Experience page (**Configuration** tab > **VoIP** menu > **Media** submenu > **Media Quality of Experience**).

Figure 13-12: Media Quality of Experience Page



Quality of Experience	
⚡ Server Ip	0.0.0.0
Port	5000
⚡ Interface Name	DEFAULT
Connection Mode	VQMClient
Information Level	VQStandard
Use Mos LQ	Disable

2. Configure the parameters as required
 - 'Server Ip' (QOEServerIP) - defines the IP address of the SEM server
 - 'Port' (QOEPort) - defines the port of the SEM server
 - 'Interface Name' (QOEInterfaceName) - defines the device's IP network interface on which the SEM reports are sent
 - 'Use Mos LQ' (QOEUseMosLQ) - defines the reported MOS type (listening or conversational)
3. Click **Submit** to apply your changes.
4. To save the changes to flash memory, see 'Saving Configuration' on page 360.

13.7 Configuring Media Security

The device supports Secured RTP (SRTP) according to RFC 3711. SRTP is used to encrypt RTP and RTCP transport for protecting VoIP traffic. SRTP requires a key exchange mechanism that is performed according to RFC 4568 – “Session Description Protocol (SDP) Security Descriptions for Media Streams”. The key exchange is done by adding a 'crypto' attribute to the SDP. This attribute is used (by both sides) to declare the various supported cipher suites and to attach the encryption key. If negotiation of the encryption data is successful, the call is established.

SRTP supports the following cipher suites (all other suites are ignored):

- AES_CM_128_HMAC_SHA1_32
- AES_CM_128_HMAC_SHA1_80

When the device is the offering side, it generates an MKI of a size configured by the 'Master Key Identifier (MKI) Size' parameter. The length of the MKI is limited to four bytes. If the remote side sends a longer MKI, the key is ignored. The key lifetime field is not supported. However, if it is included in the key it is ignored and the call does not fail.

The device supports the following session parameters (as defined in RFC 4568, SDP Security Descriptions for Media Streams):

- UNENCRYPTED_SRTP
- UNENCRYPTED_SRTCP
- UNAUTHENTICATED_SRTP

Session parameters should be the same for the local and remote sides. When the device is the offering side, the session parameters are configured by the following parameter - 'Authentication On Transmitted RTP Packets', 'Encryption On Transmitted RTP Packets', and 'Encryption On Transmitted RTCP Packets'. When the device is the answering side, the device adjusts these parameters according to the remote offering. Unsupported session parameters are ignored, and do not cause a call failure.

Below is an example of crypto attributes usage:

```
a=crypto:1 AES_CM_128_HMAC_SHA1_80
inline:PsKoMpHlCg+b5X0YLuSvNrImEh/dAe
a=crypto:2 AES_CM_128_HMAC_SHA1_32
inline:IsPtLoGkBf9a+c6XVzRuMqHlDnEiAd
```

The device also supports symmetric MKI negotiation, whereby it can be configured to forward the MKI size received in the SDP offer crypto line in the SDP answer crypto line.

To configure the device's mode of operation if negotiation of the cipher suite fails, use the 'Media Security Behavior' parameter. This parameter can be set to enforce SRTP, whereby incoming calls that don't include encryption information are rejected.

**Notes:**

- For a detailed description of the SRTP parameters, see SRTP Parameters on page 459.
- When SRTP is used, the channel capacity may be reduced.

➤ **To configure media security:**

1. Open the Media Security page (**Configuration** tab > **VoIP** menu > **Media** submenu > **Media Security**).

▼ General Media Security Settings	
Media Security	Disable ▼
Media Security Behavior	Preferable ▼
Authentication On Transmitted RTP Packets	Active ▼
Encryption On Transmitted RTP Packets	Active ▼
Encryption On Transmitted RTCP Packets	Active ▼
▼ SRTP Setting	
Master Key Identifier (MKI) Size	0
Enable symmetric MKI negotiation	Disable ▼
◆ SRTP offered Suites	
CIPHER SUITES AES CM 128 HMAC SHA1 80	<input checked="" type="checkbox"/>
CIPHER SUITES AES CM 128 HMAC SHA1 32	<input checked="" type="checkbox"/>
CIPHER SUITES ARIA CM 128 HMAC SHA1 80	<input checked="" type="checkbox"/>
CIPHER SUITES ARIA CM 192 HMAC SHA1 80	<input checked="" type="checkbox"/>

2. Configure the parameters as required.
3. Click **Submit** to apply your changes.
4. To save the changes to flash memory, see 'Saving Configuration' on page 360.

14 Services

This section describes configuration for various supported services.

14.1 Routing Based on LDAP Active Directory Queries

The device supports Lightweight Directory Access Protocol (LDAP), enabling call routing decisions based on information stored on a third-party LDAP server (or Microsoft's Active Directory™ enterprise directory server). This feature enables the usage of a single common, popular database to manage and maintain information regarding user's availability, presence, and location.

14.1.1 Configuring the LDAP Server

The basic LDAP mechanism is described below:

- **Connection:** The device connects and binds to the remote LDAP server either during the service's initialization (at device start-up) or whenever the LDAP server's IP address and port is changed. Service makes 10 attempts to connect and bind to the remote LDAP server with a timeout of 20 seconds between attempts. If connection fails, the service remains in disconnected state until either the LDAP server's IP address or port is changed.

If connection to the LDAP server later fails, the service attempts to reconnect, as described previously. The SNMP alarm `acLDAPLostConnection` is sent when connection is broken. Upon successful reconnection, the alarm is cleared.

Binding to the LDAP server can be anonymous or not. For anonymous binding, the `LDAPBindDN` and `LDAPPassword` parameters must not be defined or set to an empty string.

The address of the LDAP server can be a DNS name / FQDN configured by the `LDAPServerDomainName` parameter, or an IP address configured by the `LDAPServerIP` parameter.



Note: If you configure an FQDN, make sure that the `LDAPServerIP` parameter is left empty.

- **Search:** For the device to run a search using the LDAP service, the path to the directory's subtree (or DN) where the search is to be done must be configured using the `LDAPSearchDN` parameter. Up to three DNs can be configured. The search key, or *filter* in LDAP references, which defines the exact DN to be found and one or more attributes whose values should be returned, must also be defined.

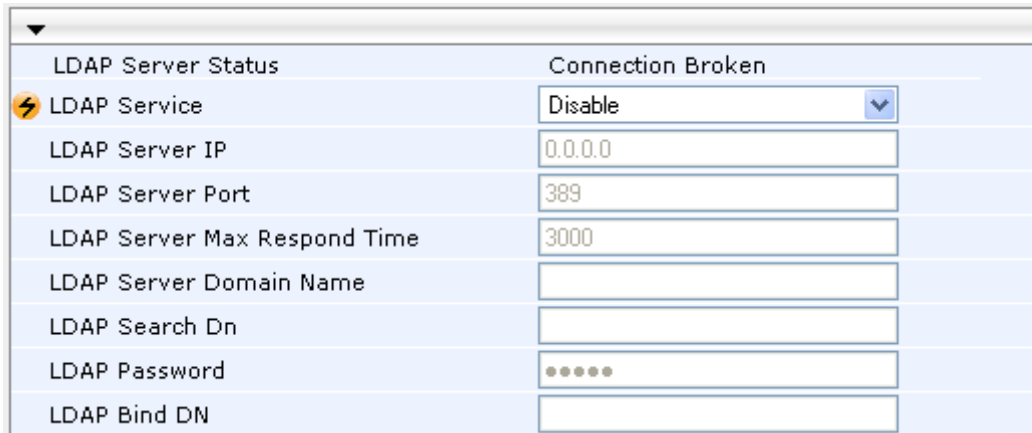
If connection to the LDAP server is disrupted during the search, all search requests are dropped and an alarm indicating a failed status is sent to client applications.

The LDAP Settings page is used for configuring the LDAP server parameters. For a full description of these parameters, see 'Configuration Parameters Reference' on page [437](#).

➤ **To configure the LDAP server parameters:**

1. Open the LDAP Settings page (**Configuration** tab > **VoIP** menu > **Services** submenu > **LDAP Settings**).

Figure 14-1: LDAP Settings Page



LDAP Server Status	Connection Broken
⚡ LDAP Service	Disable
LDAP Server IP	0.0.0.0
LDAP Server Port	389
LDAP Server Max Respond Time	3000
LDAP Server Domain Name	
LDAP Search Dn	
LDAP Password	•••••
LDAP Bind DN	

The read-only 'LDAP Server Status' field displays one of the following possibilities:

- "Not Applicable"
 - "Connection Broken"
 - "Connecting"
 - "Connected"
2. Configure the parameters as required.
 3. Click **Submit** to apply your changes.
 4. To save the changes to flash memory, see 'Saving Configuration' on page 360.

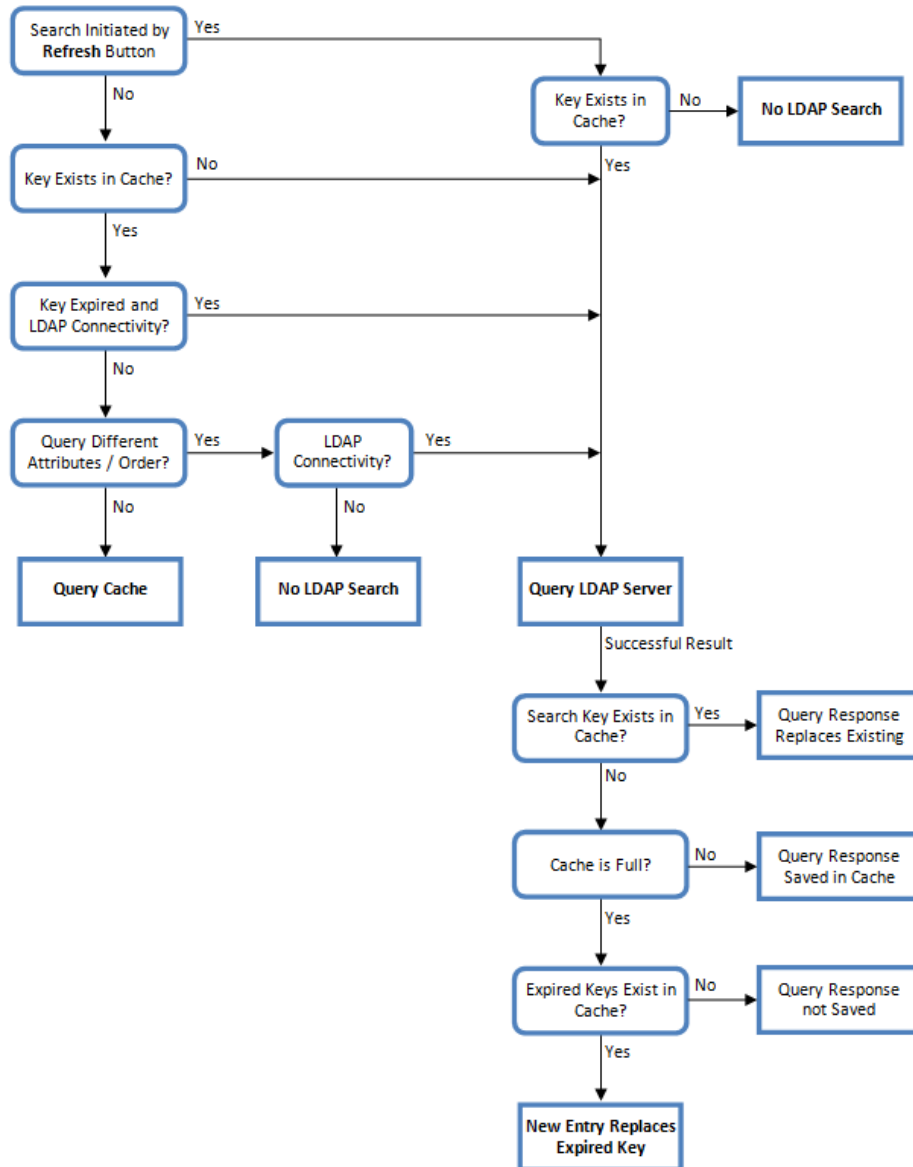
14.1.2 Configuring the Device's LDAP Cache

The device provides an option for storing recent LDAP queries and responses in its local cache. The cache is used for subsequent queries and/or in case of LDAP server failure. The advantage of enabling this feature includes the following:

- Improves routing decision performance by using local cache for subsequent LDAP queries
- Reduces number of queries performed on an LDAP server and corresponding bandwidth consumption
- Provides partial survivability in case of intermittent LDAP server failure (or network isolation)

The handling of LDAP queries with the LDAP cache is shown in the flowchart below:

Figure 14-2: LDAP Query Process with Local LDAP Cache



The LDAP Settings page is used for configuring the LDAP cache parameters.



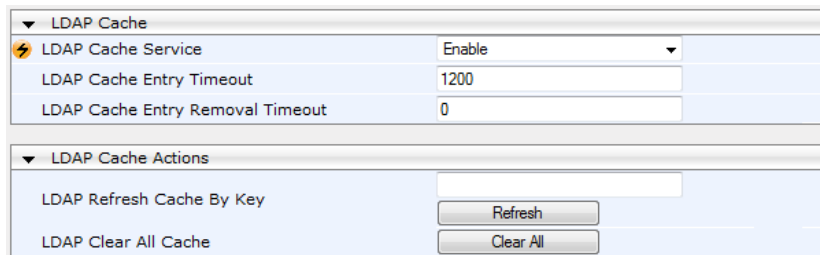
Notes:

- The LDAP cache parameters are available only if you have enabled the LDAP service (see 'Configuring the LDAP Server' on page 167).
- If on the first LDAP query, the result fails for at least one attribute and is successful for at least one, the partial result is cached. However, for subsequent queries, the device does not use the partially cached result, but does a new query with the LDAP server again.
- For a full description of the cache parameters, see 'Configuration Parameters Reference' on page 437.

➤ **To configure the LDAP cache parameters:**

1. Open the LDAP Settings page (**Configuration** tab > **VoIP** menu > **Services** submenu > **LDAP Settings**).

Figure 14-3: LDAP Settings Page - Cache Parameters



LDAP Cache	
LDAP Cache Service	Enable
LDAP Cache Entry Timeout	1200
LDAP Cache Entry Removal Timeout	0

LDAP Cache Actions	
LDAP Refresh Cache By Key	Refresh
LDAP Clear All Cache	Clear All

2. Configure the parameters as required.
3. Click **Submit** to apply your changes.
4. To save the changes to flash memory, see 'Saving Configuration' on page 360.

The LDAP Settings page also provides you with the following buttons:

- **LDAP Refresh Cache By Key:** Refreshes a saved LDAP entry response in the cache of a specified LDAP search key. If a request with the specified key exists in the cache, the request is resent to the LDAP server.
- **LDAP Clear All Cache:** Removes all LDAP entries in the cache.

14.1.3 Active Directory based Tel-to-IP Routing for Microsoft Lync

Typically, enterprises wishing to deploy Microsoft® Lync™ Server 2010 (formerly known as Office Communication Server 2007) are faced with a complex, call routing dial plan when migrating users from their existing PBX or IP PBX to the Lync Server 2010 platform. As more and more end-users migrate to the new voice system, dialing plan management and PBX link capacity can be adversely impacted. To resolve this issue, enterprises can employ Microsoft's Active Directory (AD), which provides a central database to manage and maintain information regarding user's availability, presence, and location.

The device supports Tel-to-IP call routing decisions based on information stored on the AD. Based on queries sent to the AD, the device can route the Tel call to one of the following IP domains:

- Lync client (formally OCS) - users connected to Lync Server 2010 through the Mediation Server
- PBX or IP PBX - users not yet migrated to Lync Server 2010
- Mobile - mobile number
- Private - private telephone line for Lync users (in addition to the primary telephone line)

14.1.3.1 Querying the AD and Routing Priority

The device queries the AD using the initial destination number (i.e., called number). The query can return up to four user phone numbers, each pertaining to one of the IP domains (i.e., private number, Lync / OCS number, PBX / IP PBX number, and mobile number). The configuration parameters listed in the table below are used to configure the query attribute keys that defines the AD attribute that you wish to query in the AD:

Table 14-1: Parameters for Configuring Query Attribute Key

Parameter	Queried User Domain (Attribute) in AD	Query or Query Result Example
MSLDAPPBXNumAttributeName	PBX or IP PBX number (e.g., "telephoneNumber" - default)	telephoneNumber=+3233554447
MSLDAPOCSNumAttributeName	Mediation Server / Lync client number (e.g., "msRTCSIP-line")	msRTCSIP-line=john.smith@company.com
MSLDAPMobileNumAttributeName	Mobile number (e.g., "mobile")	mobile=+3247647156
MSLDAPPrivateNumAttributeName	Any attribute (e.g., "msRTCSIP-PrivateLine") Note: Used only if set to same value as Primary or Secondary key.	msRTCSIP-PrivateLine=+3233554480
MSLDAPPrimaryKey	Primary Key query search instead of PBX key - can be any AD attribute	msRTCSIP-PrivateLine=+3233554480
MSLDAPSecondaryKey	Secondary Key query key search if Primary Key fails - can be any attribute	-

The process for querying the AD and subsequent routing based on the query results is as follows:

1. If the Primary Key is configured, it uses the defined string as a primary key instead of the one defined in MSLDAPPBXNumAttributeName. It requests the attributes which are described below.
2. If the primary query is not found in the AD and the Secondary Key is configured, it does a second query for the destination number using a second AD attribute key name, configured by the MSLDAPSecondaryKey parameter.
3. If none of the queries are successful, it routes the call to the original dialed destination number according to the routing rule matching the "LDAP_ERR" destination prefix number value, or rejects the call with a SIP 404 "Not Found" response.
4. For each query (primary or secondary), it requests to query the following attributes (if they're not configured as an empty string):
 - MSLDAPPBXNumAttributeName
 - MSLDAPOCSNumAttributeName
 - MSLDAPMobileNumAttributeName
 In addition, it queries the special attribute defined in MSLDAPPrivateNumAttributeName, only if the query key (primary or secondary) is equal to its value.
5. If the query is found: The AD returns up to four attributes - Lync / OCS, PBX / IP PBX, private (only if it equals Primary or Secondary key), and mobile.

6. The device adds unique prefix keywords to the query results in order to identify the query type (i.e., IP domain). These prefixes are used as the prefix destination number value in the Outbound IP Routing table to denote the IP domains:
 - "PRIVATE" (PRIVATE:<private_number>): used to match a routing rule based on query results of the private number (MSLDAPPrivateNumAttributeName)
 - "OCS" (OCS:<Lync_number>): used to match a routing rule based on query results of the Lync client number (MSLDAPOCSNumAttributeName)
 - "PBX" (PBX:<PBX_number>): used to match a routing rule based on query results of the PBX / IP PBX number (MSLDAPPBXNumAttributeName)
 - "MOBILE" (MOBILE:<mobile_number>): used to match a routing rule based on query results of the mobile number (MSLDAPMobileNumAttributeName)
 - "LDAP_ERR": used to match a routing rule based on a failed query result when no attribute is found in the AD



Note: These prefixes are involved only in the routing and manipulation processes; they are not used as the final destination number.

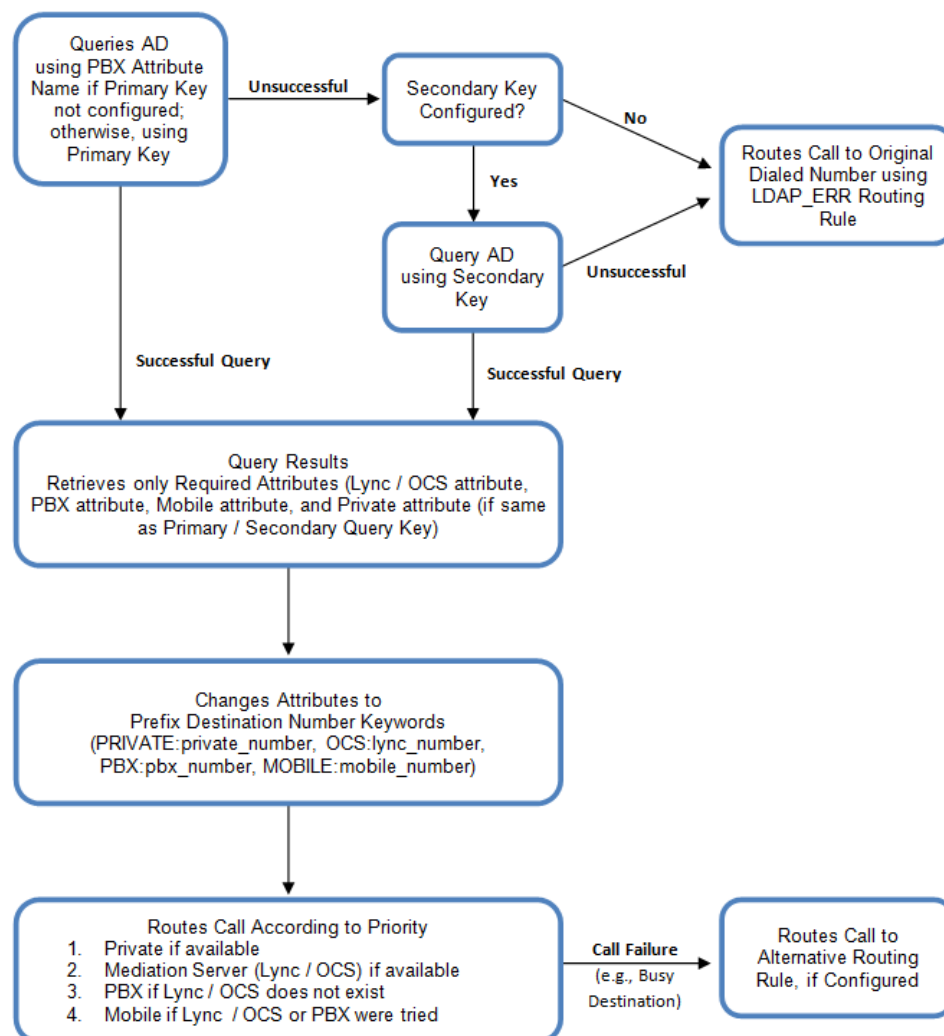
7. The device uses the Outbound IP Routing table to route the call based on the LDAP query result. The device routes the call according to the following priority:
 1. **Private line:** If the query is done for the private attribute and it's found, then the device routes the call according to this attribute.
 2. **Mediation Server SIP address (Lync / OCS):** If the private attribute does not exist or is not queried, then the device routes the call to the Mediation Server (which then routes the call to the Lync client).
 3. **PBX / IP PBX:** If the Lync / OCS client is not found in the AD, it routes the call to the PBX / IP PBX.
 4. **Mobile number:** If the Lync / OCS client (or Mediation Server) is unavailable (e.g., SIP response 404 "Not Found" upon INVITE sent to Lync client), and the PBX / IP PBX is also unavailable, then the device routes the call to the user's mobile number (if exists in the AD).
 5. **Alternative route:** If the call routing to all the above fails (e.g., due to unavailable destination - call busy), the device can route the call to an alternative destination if an alternative routing rule is configured.
 6. **"Redundant" route:** If the query failed (i.e., no attribute found in the AD), the device uses the routing rule matching the "LDAP_ERR" prefix destination number value.



Note: For Enterprises implementing a PBX / IP PBX system, but yet to migrate to Lync Server 2010, if the PBX / IP PBX system is unavailable or has failed, the device uses the AD query result for the user's mobile phone number, routing the call through the PSTN to the mobile destination.

The flowchart below summarizes the device's process for querying the AD and routing the call based on the query results:

Figure 14-4: LDAP Query Flowchart



Note: If you are using the device's local LDAP cache, see 'Configuring the Device's LDAP Cache' on page 168 for the LDAP query process.

14.1.3.2 Configuring AD-Based Routing Rules

The procedure below describes how to configure Tel-to-IP routing based on LDAP queries.

➤ **To configure LDAP-based Tel-to-IP routing for Lync Server 2010:**

1. Configure the LDAP server parameters, as described in 'Configuring the LDAP Server' on page 167.
2. Configure the AD attribute names used in the LDAP query:
 - a. Open the Advanced Parameters page (**Configuration** tab > **VoIP** menu > **SIP Definitions** submenu > **Advanced Parameters**).

Figure 14-5: LDAP Parameters for Microsoft Lync Server 2010

MS LDAP Settings	
MS LDAP OCS Number attribute name	msRTCSIP-PrimaryUserAddress
MS LDAP PBX Number attribute name	telephoneNumber
MS LDAP MOBILE Number attribute name	mobile

- b. Configure the LDAP attribute names as desired.
3. Configure AD-based IP-to-IP routing rules:
 - a. Open the IP-to-IP Routing Table page (Configuration tab > VoIP menu > SBC submenu > Routing SBC > IP to IP Routing Table). For more information, see Configuring SBC IP-to-IP Routing on page 279.
 - b. Configure query-result routing rules for each IP domain (private, PBX / IP PBX, Lync / OCS clients, and mobile), using the LDAP keywords (case-sensitive) in the Destination Username Prefix field:
 - ◆ PRIVATE: Private number
 - ◆ OCS: Lync / OCS client number
 - ◆ PBX: PBX / IP PBX number
 - ◆ MOBILE: Mobile number
 - ◆ LDAP_ERR: LDAP query failure
 - c. Configure a routing rule for routing the initial call (LDAP query) to the LDAP server, by setting the 'Destination Type' field to LDAP for denoting the IP address of the LDAP server.
 - d. For alternative routing, enable the alternative routing mechanism and configure corresponding SIP reasons for alternative routing. For this feature, alternative routing starts from the table row located under the LDAP query row.

The table below shows an example for configuring AD-based SBC routing rules in the IP-to-IP Routing Table:

Table 14-2: AD-Based SBC IP-to-IP Routing Rule Configuration Examples

Index	Destination Username Prefix	Destination Type	Destination Address
1	PRIVATE:	Dest Address	10.33.45.60
2	PBX:	Dest Address	10.33.45.65
3	OCS:	Dest Address	10.33.45.68
4	MOBILE:	Dest Address	10.33.45.100
5	LDAP_ERR	Dest Address	10.33.45.80
6	*	LDAP	
7	*	Dest Address	10.33.45.72

The configured routing rule example is explained below:

- **Rule 1:** Sends call to private telephone line (at 10.33.45.60) upon successful AD query result for the private attribute.
- **Rule 2:** Sends call to IP PBX (at 10.33.45.65) upon successful AD query result for the PBX attribute.
- **Rule 3:** Sends call to Lync client (i.e., Mediation Server at 10.33.45.68) upon successful AD query result for the Lync attribute.
- **Rule 4:** Sends call to user's mobile phone number (to PSTN through the device's IP address at 10.33.45.100) upon successful AD query result for the Mobile attribute.
- **Rule 5:** Sends call to IP address of device (10.33.45.80) if AD query failure (e.g., no response from LDAP server or attribute not found).
- **Rule 6:** Sends query for original destination number of received call to the LDAP server.
- **Rule 7:** Alternative routing rule that sends the call of original dialed number to IP destination 10.33.45.72. This rule is applied in any of the following cases
 - LDAP functionality is disabled.
 - LDAP query is successful but call fails (due to, for example, busy line) to all the relevant attribute destinations (private, Lync, PBX, and mobile), and a relevant SBC Alternative Routing Reason (see Configuring Alternative Routing Reasons on page 286) has been configured.

Once the device receives the original incoming call, the first rule that it uses is Rule 6, which queries the AD server. When the AD replies, the device searches the table, from the first rule down, for the matching destination phone prefix (i.e., "PRIVATE:", "PBX:", "OCS:", "MOBILE:", and "LDAP_ERR:"), and then sends the call to the appropriate destination.

14.2 Least Cost Routing

This section provides a description of the device's least cost routing (LCR) feature and how to configure it.

14.2.1 Overview

The LCR feature enables the device to choose the outbound IP destination routing rule based on lowest call cost. This is useful in that it enables service providers to optimize routing costs for customers. For example, you may wish to define different call costs for local and international calls, or different call costs for weekends and weekdays (specifying even the time of call). The device sends the calculated cost of the call to a Syslog server (as Information messages), thereby enabling billing by third-party vendors.

LCR is implemented by defining Cost Groups and assigning them to routing rules in the Outbound IP Routing table. The device searches this routing table for matching routing rules, and then selects the rule with the lowest call cost. If two routing rules have identical costs, then the rule appearing higher up in the table is used (i.e., first-matched rule). If a selected route is unavailable, the device selects the next least-cost routing rule. However, even if a matched rule is not assigned a Cost Group, the device can select it as the preferred route over other matched rules with Cost Groups. This is determined according to the settings of the Default Cost parameter in the Routing Rule Groups table.

The Cost Group defines a fixed connection cost (*connection cost*) and a charge per minute (*minute cost*). Cost Groups can also be configured with time segments (*time bands*), which define connection cost and minute cost based on specific days of the week and time of day (e.g., from Saturday through Sunday, between 6:00 and 18:00). If multiple time bands are configured per Cost Group and a call spans multiple time bands, the call cost is calculated using only the time band in which the call was initially established.

In addition to Cost Groups, the device can calculate the call cost using an optional, user-defined average call duration value. The logic in using this option is that a Cost Group may be cheap if the call duration is short, but due to its high minute cost, may prove very expensive if the duration is lengthy. Thus, together with Cost Groups, the device can use this option to determine least cost routing. The device calculates the Cost Group call cost as follows: $\text{Total Call Cost} = \text{Connection Cost} + (\text{Minute Cost} * \text{Average Call Duration})$.

The below table shows an example of call cost when taking into consideration call duration. This example shows four defined Cost Groups and the total call cost if the average call duration is 10 minutes:

Table 14-3: Call Cost Comparison between Cost Groups for different Call Durations

Cost Group	Connection Cost	Minute Cost	Total Call Cost per Duration	
			1 Minute	10 Minutes
A	1	6	7	61
B	0	10	10	100
C	0.3	8	8.3	80.3
D	6	1	7	16

If four matching routing rules are located in the routing table and each one is assigned a different Cost Group as listed in the table above, then the rule assigned Cost Group "D" is selected. Note that for one minute, Cost Groups "A" and "D" are identical, but due to the average call duration, Cost Group "D" is cheaper. Therefore, average call duration is an important factor in determining the cheapest routing role.

Below are a few examples of how you can implement LCR:

- **Example 1:** This example uses two different Cost Groups for routing local calls and international calls:

Two Cost Groups are configured as shown below:

Cost Group	Connection Cost	Minute Cost
1. "Local Calls"	2	1
2. "International Calls"	6	3

The Cost Groups are assigned to routing rules for local and international calls in the Outbound IP Routing table:

Routing Index	Dest Phone Prefix	Destination IP	Cost Group ID
1	2000	x.x.x.x	1 "Local Calls"
2	00	x.x.x.x	2 "International Calls"

- **Example 2:** This example shows how the device determines the cheapest routing rule in the Outbound IP Routing table:

The Default Cost parameter (global) in the Routing Rule Groups table is set to **Min**, meaning that if the device locates other matching LCR routing rules (with Cost Groups assigned), the routing rule without a Cost Group is considered the lowest cost route.

- The following Cost Groups are configured:

Cost Group	Connection Cost	Minute Cost
1. "A"	2	1
2. "B"	6	3

- The Cost Groups are assigned to routing rules in the Outbound IP Routing table:

Routing Index	Dest Phone Prefix	Destination IP	Cost Group ID
1	201	x.x.x.x	"A"
2	201	x.x.x.x	"B"
3	201	x.x.x.x	0
4	201	x.x.x.x	"B"

The device calculates the optimal route in the following index order: 3, 1, 2, and then 4, due to the following logic:

- Index 1 - Cost Group "A" has the lowest connection cost and minute cost
- Index 2 - Cost Group "B" takes precedence over Index 4 entry based on the first-matched method rule
- Index 3 - no Cost Group is assigned, but as the Default Cost parameter is set to **Min**, it is selected as the cheapest route
- Index 4 - Cost Group "B" is only second-matched rule (Index 1 is the first)

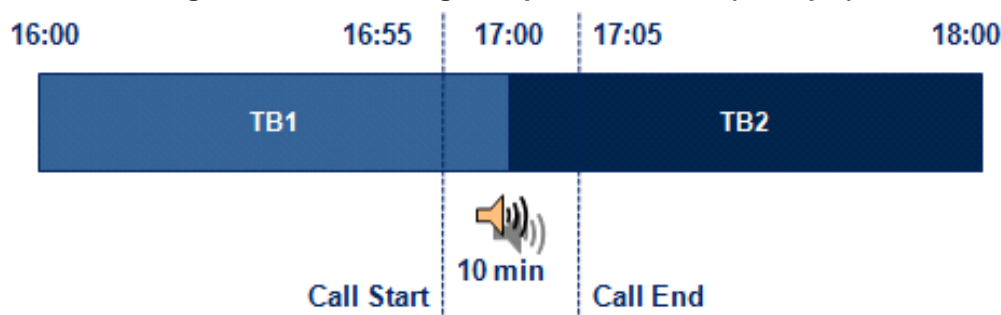
- **Example 3:** This example shows how the cost of a call is calculated if the call spans over multiple time bands:

Assume a Cost Group, "CG Local" is configured with two time bands, as shown below:

Cost Group	Time Band	Start Time	End Time	Connection Cost	Minute Cost
CG Local	TB1	16:00	17:00	2	1
	TB2	17:00	18:00	7	2

Assume that the call duration is 10 minutes, occurring between 16:55 and 17:05. In other words, the first 5 minutes occurs in time band "TB1" and the next 5 minutes occurs in "TB2", as shown below:

Figure 14-6: LCR using Multiple Time Bands (Example)



The device calculates the call using the time band in which the call was initially established, regardless of whether the call spans over additional time bands:

Total call cost = "TB1" Connection Cost + ("TB1" Minute Cost x call duration) = 2 + 1 x 10 min = 12

14.2.2 Configuring LCR

The following main steps need to be done to configure LCR:

1. Enable the LCR feature and configure the average call duration and default call connection cost - see 'Enabling LCR and Configuring Default LCR' on page 178.
2. Configure Cost Groups - see 'Configuring Cost Groups' on page 179.
3. Configure Time Bands for a Cost Group - see 'Configuring Time Bands for Cost Groups' on page 180.
4. Assign Cost Groups to outbound IP routing rules - see 'Assigning Cost Groups to Routing Rules' on page 182.

14.2.2.1 Enabling the LCR Feature

The procedure below describes how to enable the LCR feature. This also includes configuring the average call duration and default call cost for routing rules that are not assigned Cost Groups in the Outbound IP Routing table.

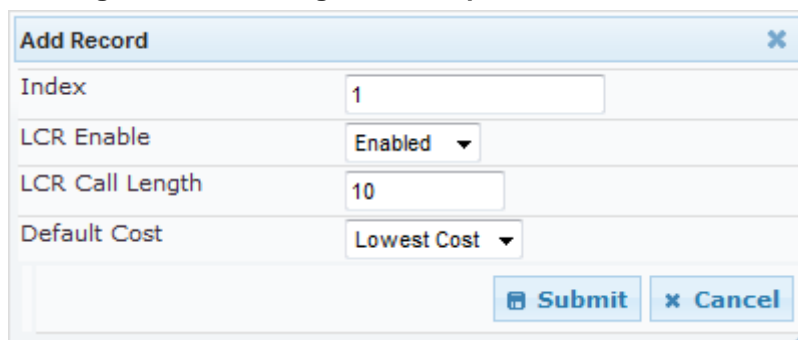


Note: The Routing Rule Groups table can also be configured using the table ini file parameter, RoutingRuleGroups.

➤ To enable LCR:

1. Open the Routing Rule Groups Table page (**Configuration** tab > **VoIP** menu > **Services** submenu > **Least Cost Routing** > **Routing Rule Groups Table**).
2. Click the **Add** button; the Add Record dialog box appears:

Figure 14-7: Routing Rule Groups Table - Add Record



3. Configure the parameters as required. For a description of the parameters, see the table below.
4. Click **Submit**; the entry is added to the Routing Rule Groups table.

Table 14-4: Routing Rule Groups Table Description

Parameter	Description
Index [RoutingRuleGroups_Index]	Defines the table index entry. Note: Only one index entry can be configured.
LCR Enable [RoutingRuleGroups_LCREnable]	Enables the LCR feature: <ul style="list-style-type: none"> ▪ [0] Disabled (default) ▪ [1] Enabled

Parameter	Description
LCR Call Length [RoutingRuleGroups_LCRAverageCallLength]	<p>Defines the average call duration (in minutes) and is used to calculate the variable portion of the call cost. This is useful, for example, when the average call duration spans over multiple time bands. The LCR is calculated as follows: cost = call connect cost + (minute cost * average call duration)</p> <p>The valid value range is 0-65533. the default is 1.</p> <p>For example, assume the following Cost Groups:</p> <ul style="list-style-type: none"> ▪ "Weekend A": call connection cost is 1 and charge per minute is 6. Therefore, a call of 1 minute cost 7 units. ▪ "Weekend_ B": call connection cost is 6 and charge per minute is 1. Therefore, a call of 1 minute cost 7 units. <p>Therefore, for calls under one minute, "Weekend A" carries the lower cost. However, if the average call duration is more than one minute, then "Weekend B" carries the lower cost.</p>
Default Cost [RoutingRuleGroups_LCRDefaultCost]	<p>Determines whether routing rules in the Outbound IP Routing table without an assigned Cost Group are considered a higher cost or lower cost route compared to other matched routing rules that are assigned Cost Groups.</p> <ul style="list-style-type: none"> ▪ [0] Lowest Cost = If the device locates other matching LCR routing rules, this routing rule is considered the lowest cost route and therefore, it is selected as the route to use (default.) ▪ [1] Highest Cost = If the device locates other matching LCR routing rules, this routing rule is considered as the highest cost route and therefore, is not used or used only if the other cheaper routes are unavailable. <p>Note: If more than one valid routing rule without a defined Cost Group exists, the device selects the first-matched rule.</p>

14.2.2.2 Configuring Cost Groups

The procedure below describes how to configure Cost Groups. Cost Groups are defined with a fixed call connection cost and a call rate (charge per minute). Once configured, you can configure Time Bands for each Cost Group. Up to 10 Cost Groups can be configured.

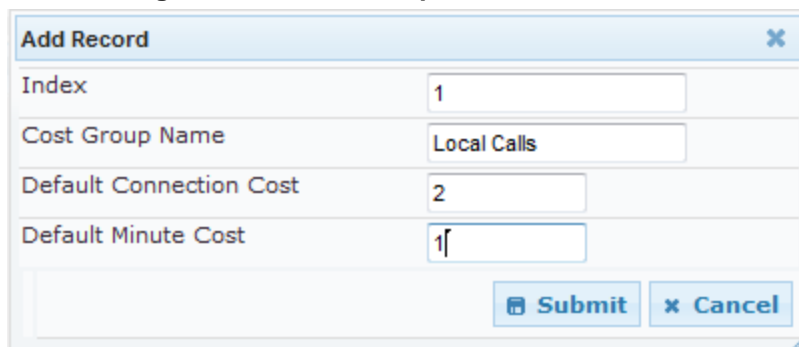


Note: The Cost Group table can also be configured using the table ini file parameter, CostGroupTable.

➤ **To configure Cost Groups:**

1. Open the Cost Group Table page (**Configuration** tab > **VoIP** menu > **Services** submenu > **Least Cost Routing** > **Cost Group Table**).
2. Click the **Add** button; the Add Record dialog box appears:

Figure 14-8: Cost Group Table - Add Record



3. Configure the parameters as required. For a description of the parameters, see the table below.
4. Click **Submit**; the entry is added to the Cost Group table.

Table 14-5: Cost Group Table Description

Parameter	Description
Index [CostGroupTable_Index]	Defines the table index entry.
Cost Group Name [CostGroupTable_CostGroupName]	Defines an arbitrary name for the Cost Group. The valid value is a string of up to 30 characters. Note: Each Cost Group must have a unique name.
Default Connect Cost [CostGroupTable_DefaultConnectionCost]	Defines the call connection cost (added as a fixed charge to the call) for a call outside the time bands. The valid value range is 0-65533. The default is 0. Note: When calculating the cost of a call, if the current time of the call is not within a time band configured for the Cost Group, then this default connection cost is used.
Default Time Cost [CostGroupTable_DefaultMinuteCost]	Defines the call charge per minute for a call outside the time bands. The valid value range is 0-65533. The default is 0. Note: When calculating the cost of a call, if the current time of the call is not within a time band configured for the Cost Group, then this default charge per minute is used.

14.2.2.3 Configuring Time Bands for Cost Groups

The procedure below describes how to configure Time Bands for a Cost Group. The time band defines the day and time range for which the time band is applicable (e.g., from Saturday 05:00 to Sunday 24:00) as well as the fixed call connection charge and call rate per minute for this interval. Up to 70 time bands can be configured, and up to 21 time bands can be assigned to each Cost Group.


Notes:

- You cannot define overlapping time bands.
- The Time Band table can also be configured using the table ini file parameter, CostGroupTimebands.

➤ **To configure Time Bands for a Cost Group:**

1. Open the Cost Group Table page (**Configuration** tab > **VoIP** menu > **Services** submenu > **Least Cost Routing** > **Cost Group Table**).
2. Select a Cost Group for which you want to assign Time Bands, and then click the **Time Band** link located below the table; the Time Band table for the selected Cost Group appears.
3. Click the **Add** button; the Add Record dialog box appears:

Figure 14-9: Time Band Table - Add Record

4. Configure the parameters as required. For a description of the parameters, see the table below.
5. Click **Submit**; the entry is added to the Time Band table for the relevant Cost Group.

Table 14-6: Time Band Table Description

Parameter	Description
Index [CostGroupTimebands_TimebandIndex]	Defines the table index entry.
Start Time [CostGroupTimebands_StartTime]	Defines the day and time of day from when this time band is applicable. The format is ddd:hh:mm (e.g., sun:06:00), where: <ul style="list-style-type: none"> ddd is the day (i.e., sun, mon, tue, wed, thu, fri, or sat) hh and mm denote the time of day, where hh is the hour (00-23) and mm the minutes (00-59)
End Time [CostGroupTimebands_EndTime]	Defines the day and time of day until when this time band is applicable. For a description of the valid values, see the parameter above.
Connection Cost [CostGroupTimebands_ConnectionCost]	Defines the call connection cost during this time band. This is added as a fixed charge to the call. The valid value range is 0-65533. The default is 0. Note: The entered value must be a whole number (i.e., not a decimal).
Minute Cost [CostGroupTimebands_MinuteCost]	Defines the call cost per minute charge during this timeband. The valid value range is 0-65533. The default is 0. Note: The entered value must be a whole number (i.e., not a decimal).

14.2.2.4 Assigning Cost Groups to Routing Rules

Once you have configured your Cost Groups, you need to assign them to routing rules in the IP-to-IP Routing table - see Configuring SBC IP-to-IP Routing on page [279](#).

15 Enabling Applications

The device supports the following main applications:

- Stand-Alone Survivability (SAS) application (see SAS Overview on page 307)
- Session Border Control (SBC) application (see SBC Overview on page 235)
- Cloud Resilience Package (see CRP Overview on page 299)

The procedure below describes how to enable these applications. Once an application is enabled, the Web GUI provides menus and parameter fields relevant to the application.



Notes:

- This page displays the application only if the device is installed with the relevant Software License Key supporting the application (see 'Software License Key' on page 370).
- For enabling an application, a device reset is required.

➤ To enable an application:

1. Open the Applications Enabling page (**Configuration** tab > **VoIP** menu > **Applications Enabling** submenu > **Applications Enabling**).

Figure 15-1: Applications Enabling Page

⚡ SAS Application	Disable
⚡ SBC Application	Enable

2. From the relevant application drop-down list, select **Enable**.
3. Save (burn) the changes to the device's flash memory with a device reset (see 'Saving Configuration' on page 360).

Reader's Notes

16 Control Network

This section describes configuration of the network at the SIP control level.

16.1 Configuring SRD Table

The SRD Settings page allows you to configure up to 32 signaling routing domains (SRD). An SRD is configured with a unique name and assigned a Media Realm. Additional SBC attributes such as media anchoring and user registration can also be configured.

An SRD is a set of definitions together creating multiple, virtual multi-service IP gateways:

- Multiple and different SIP signaling interfaces (SRD associated with a SIP Interface) and RTP media (associated with a Media Realm) for multiple Layer-3 networks. Due to the B2BUA nature of the SBC application, different interfaces can be assigned to each leg of the call.
- Can operate with multiple gateway customers that may reside either in the same or in different Layer-3 networks as the device. This allows separation of signaling traffic between different customers. In such a scenario, the device is configured with multiple SRD's.

Typically, one SRD is defined for each group of SIP UAs (e.g. proxies, IP phones, application servers, gateways, and softswitches) that communicate with each other. This provides these entities with VoIP services that reside on the same Layer-3 network (must be able to communicate without traversing NAT devices and must not have overlapping IP addresses). Routing from one SRD to another is possible, whereby each routing destination (IP Group or destination address) indicates the SRD to which it belongs.

Once configured, you can use the SRD as follows:

- Associate it with a SIP Interface (see 'Configuring SIP Interface Table' on page 188)
- Associate it with an IP Group (see 'Configuring IP Groups' on page 191)
- Associate it with a Proxy Set (see 'Configuring Proxy Sets Table' on page 197)
- Apply an Admission Control rule to it (see 'Configuring Admission Control Table' on page 268)
- Define it as a Classification rule for the incoming SIP request (see 'Configuring Classification Rules' on page 272)
- Use it as a destination IP-to-IP routing rule (see 'Configuring SBC IP-to-IP Routing' on page 279)

The SRD Settings page also displays the IP Groups, Proxy Sets, and SIP Interfaces associated with a selected SRD index.



Notes:

- On the SRD Settings page, you can also configure a SIP Interface in the SIP Interface table, instead of navigating to the SIP Interface Table page as described in 'Configuring SIP Interface Table' on page 188.
- The SRD table can also be configured using the table ini file parameter, SRD.

➤ **To configure SRDs:**

1. Open the SRD Settings page (**Configuration** tab > **VoIP** menu > **Control Network** submenu > **SRD Table**).

Figure 16-1: SRD Settings Page

▼	
SRD Index	0 - Not Exist ▼
▼ Common Parameters	
SRD Name	<input type="text"/>
Media Realm	<input type="text"/>
▼ SBC Parameters	
Internal SRD Media Anchoring	Anchor Media ▼
Block Unregistered Users	No ▼
Max Number Of Registered Users	-1
Enable Un-Authenticated Registrations	Yes ▼

2. From the 'SRD Index' drop-down list, select an index for the SRD, and then configure it according to the table below.
3. Click **Submit** to apply your changes.
4. To save the changes to flash memory, see 'Saving Configuration' on page 360.

Table 16-1: SRD Table Parameters

Parameter	Description
SRD Name [SRD_Name]	Mandatory descriptive name of the SRD. The valid value can be a string of up to 21 characters.
Media Realm [SRD_MediaRealm]	Defines the Media Realm associated with the SRD. The entered string value must be identical (and case-sensitive) to the Media Realm name configured in the Media Realm table (see 'Configuring Media Realms' on page 157). The valid value is a string of up to 40 characters. Notes: <ul style="list-style-type: none"> ▪ If the Media Realm is later deleted from the Media Realm table, then this value becomes invalid in the SRD table. ▪ For configuring Media Realms, see 'Configuring Media Realms' on page 157.
SBC Parameters	
Internal SRD Media Anchoring [SRD_IntraSRDMediaAnchoring]	Determines whether the device performs media anchoring or not on media for the SRD. <ul style="list-style-type: none"> ▪ [0] Anchor Media = (Default) RTP traverses the device and each leg uses a different coder or coder parameters. ▪ [1] Don't Anchor Media = The RTP packet flow does not traverse the device; instead, the two SIP UA's establish a direct RTP/SRTP (media) flow between one another. Notes: <ul style="list-style-type: none"> ▪ When No Media Anchoring is enabled: <ul style="list-style-type: none"> ✓ The device does not perform manipulation on SDP data (offer/answer transactions) such as ports, IP address, and coders. ✓ Opening voice channels and allocation of IP media ports

Parameter	Description
	<p>are not required.</p> <ul style="list-style-type: none"> When two UA's pertain to the same SRD and this parameter is set to [1], and one of the UA's is defined as a foreign user (example, "follow me service") located on the WAN while the other UA is located on the LAN, then calls between these two UA's can't be established until this parameter is set to 0, as the device doesn't interfere in the SIP signaling. In other words, parameters such as IP addresses are not manipulated for calls between LAN and WAN (although required). When the global parameter SBCDirectMedia is disabled, you cannot enable No Media Anchoring for two UA's pertaining to separate SRDs; No Media Anchoring can only be enable for two UA's pertaining to the same SRD. For more information on media handling, see SBC Media Handling on page 241.
Block Unregistered Users [SRD_BlockUnRegUsers]	<p>Determines whether the device blocks (rejects) incoming calls (INVITE requests) from unregistered users (pertaining to User-type IP Groups) for the SRD.</p> <ul style="list-style-type: none"> [0] No = Calls from unregistered users are not blocked (default). [1] Yes = Blocks calls from unregistered users. <p>Note: When the call is blocked, the device sends a SIP 500 "Server Internal Error" response to the remote end.</p>
Max Number of Registered Users [SRD_MaxNumOfRegUsers]	<p>Maximum number of users belonging to this SRD that can register with the device. By default, no limitation exists for registered users</p>
Enable Un-Authenticated Registrations [SRD_EnableUnAuthenticatedRegistrations]	<p>Determines whether the device blocks REGISTER requests from new users (i.e., users not registered in the device's registration database) when the destination is a User-type IP Group.</p> <ul style="list-style-type: none"> [0] No = The device sends REGISTER requests to the SIP proxy server and only if authenticated by the server does the device add the user registration to its database. [1] Yes = The device adds REGISTER requests to its database even if the requests are not authenticated by a SIP proxy (default).

16.2 Configuring SIP Interface Table

The SIP Interface table allows you to configure up to 32 SIP Interfaces. The SIP Interface defines a listening port and type (UDP, TCP, or TLS) for SIP signaling traffic on a specific logical IP network interface configured for the device (in the Multiple Interface table).

The SIP Interface is configured for a specific application (i.e., SAS and SBC) and associated with an SRD. For each SIP Interface, you can assign a SIP message policy, enable TLS mutual authentication, enable TCP keepalive, and determine the SIP response sent upon classification failure.

SIP Interfaces can be used, for example, for the following:

- Using SIP signaling interfaces per call leg (i.e., each SIP entity communicates with a specific SRD).
- Using different SIP listening ports for a single or for multiple IP network interfaces.
- Differentiating between applications by creating SIP Interfaces per application.
- Separating signaling traffic between networks (e.g., different customers) to use different routing tables, manipulations, SIP definitions, and so on.

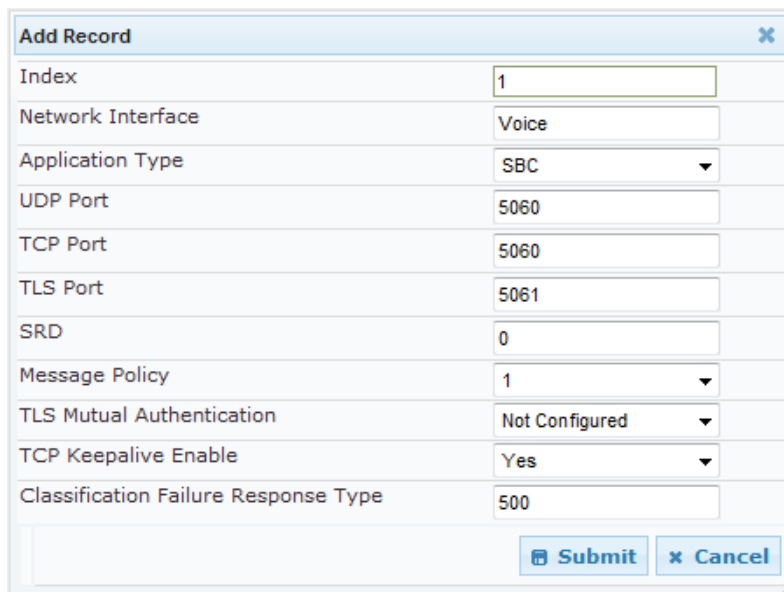


Note: The SIP Interface table can also be configured using the table *ini* file parameter, SIPInterface.

➤ To configure the SIP Interface table:

1. Open the SIP Interface Table page (**Configuration** tab > **VoIP** menu > **Control Network** submenu > **SIP Interface Table**).
2. Click the **Add** button; the following dialog box appears:

Figure 16-2: SIP Interface Table – Add Record



Index	1
Network Interface	Voice
Application Type	SBC
UDP Port	5060
TCP Port	5060
TLS Port	5061
SRD	0
Message Policy	1
TLS Mutual Authentication	Not Configured
TCP Keepalive Enable	Yes
Classification Failure Response Type	500

Submit Cancel

3. Click **Submit** to apply your settings.

Table 16-2: SIP Interface Table Parameters

Parameter	Description
Network Interface [SIPInterface_NetworkInterface]	<p>Defines the Control-type IP network interface that you want to associate with the SIP Interface. This value string must be identical (including case-sensitive) to that configured in the 'Interface Name' field of the Multiple Interface table (see 'Configuring IP Network Interfaces' on page 98).</p> <p>The default is not configured.</p> <p>Note: SIP Interfaces that are assigned to a specific SRD must be defined with the same network interface. For example, if you define three SIP Interfaces for SRD ID #8, all these SIP Interfaces must be defined with the same network interface (e.g., "SIP1").</p>
Application Type [SIPInterface_ApplicationType]	<p>Defines the application type associated with the SIP Interface.</p> <ul style="list-style-type: none"> ▪ [1] SAS = Stand-Alone Survivability (SAS) application. ▪ [2] SBC = SBC application.
UDP Port [SIPInterface_UDPPort]	<p>Defines the listening and source UDP port.</p> <p>The valid range is 1 to 65534. The default is 5060.</p> <p>Notes:</p> <ul style="list-style-type: none"> ▪ This port must be outside of the RTP port range. ▪ Each SIP Interface must have a unique signaling port (i.e., no two SIP Interfaces can share the same port - no port overlapping).
TCP Port [SIPInterface_TCPPort]	<p>Defines the listening TCP port.</p> <p>The valid range is 1 to 65534. The default is 5060.</p> <p>Notes:</p> <ul style="list-style-type: none"> ▪ This port must be outside of the RTP port range. ▪ Each SIP Interface must have a unique signaling port (i.e., no two SIP Interfaces can share the same port - no port overlapping).
TLS Port [SIPInterface_TLSPort]	<p>Defines the listening TLS port.</p> <p>The valid range is 1 to 65534. The default is 5061.</p> <p>Notes:</p> <ul style="list-style-type: none"> ▪ This port must be outside of the RTP port range. ▪ Each SIP Interface must have a unique signaling port (i.e., no two SIP Interfaces can share the same port - no port overlapping).
SRD [SIPInterface_SRD]	<p>Assigns an SRD ID to the SIP Interface.</p> <p>The default is 0.</p> <p>Notes:</p> <ul style="list-style-type: none"> ▪ Each SRD can be associated with up to three SIP Interfaces, where each SIP Interface pertains to a different Application Type (SAS, and SBC). ▪ SIP Interfaces that are assigned to a specific SRD must be defined with the same network interface. For example, if you define three SIP Interfaces for SRD ID #8, all these SIP Interfaces must be defined with the same network interface (e.g., "SIP1").

Parameter	Description
	<ul style="list-style-type: none"> To configure SRDs, see 'Configuring SRD Table' on page 185.
Message Policy [SIPInterface_MessagePolicy]	<p>Assigns a SIP message policy to the SIP interface.</p> <p>Note: To configure SIP message policies, see 'Configuring SIP Message Policy Rules'.</p>
TLS Mutual Authentication [SIPInterface_TLSMutualAuthentication]	<p>Enables TLS mutual authentication per SIP Interface.</p> <ul style="list-style-type: none"> [-1] Not Configured = (Default) The SIPRequireClientCertificate global parameter setting is applied. [0] Disable = Device does not request the client certificate for TLS connection. [1] Enable = Device requires receipt and verification of the client certificate to establish the TLS connection.
TCP Keepalive Enable [SIPInterface_TCPKeepaliveEnable]	<p>Enables the TCP Keep-Alive mechanism with the IP entity on this SIP interface. TCP keepalive can be used, for example, to keep a NAT entry open for clients located behind a NAT server or simply to check that the connection to the IP entity is available.</p> <ul style="list-style-type: none"> [0] No (default) [1] Yes <p>Note: For configuring TCP keepalive, use the following ini file parameters: TCP TCPKeepAliveTime, TCPKeepAliveInterval, and TCPKeepAliveRetry.</p>
Classification Failure Response Type [SIPInterface_ClassificationFailureResponseType]	<p>Defines the SIP response code that the device sends if a received SIP request (OPTIONS, REGISTER, or INVITE) has failed the SBC classification process.</p> <p>The valid value can be a SIP response code from 400 through 699, or it can be set to 0 to not send any response at all. The default response code is 500 (Server Internal Error).</p> <p>This feature is important for preventing Denial of Service (DoS) attacks, typically initiated from the WAN. Malicious attackers can use SIP scanners to detect ports used by SIP devices. These scanners scan devices by sending UDP packets containing a SIP request to a range of specified IP addresses, listing those that return a valid SIP response. Once the scanner finds a device that supports SIP, it extracts information from the response and identifies the type of device (IP address and name) and can execute DoS attacks. A way to defend the device against such attacks is to not send a SIP reject response to these unclassified "calls" so that the attacker assumes that no device exists at such an IP address and port.</p> <p>Note: This parameter is applicable only if the device is set to reject unclassified calls. This is configured using the 'Unclassified Calls' parameter on the General Settings page (Configuration tab > VoIP menu > SBC > General Settings).</p>

16.3 Configuring IP Groups

The IP Group Table page allows you to create up to 32 IP Groups. The IP Group represents a SIP entity on the network with which the device communicates. This can be a server (e.g., IP PBX or ITSP) or it can be a group of users (e.g., LAN IP phones). For servers, the IP Group is typically used to define the server's IP address by associating it with a Proxy Set (see 'Configuring Proxy Sets Table' on page 197).

For the SBC application, this table can also be used to classify incoming SIP dialog-initiating requests (e.g., INVITE messages) to specific IP Groups based on the associated Proxy Set ID. However, it is highly recommended to use the Classification table for classifying incoming SIP dialogs to the IP Groups (see 'Configuring Classification Rules' on page 272). See the 'Classify by Proxy Set' parameter below for a detailed description of this feature and for important recommendations.

IP Groups are used for IP-to-IP routing rules where they represent the source and destination of the call (see 'Configuring SBC IP-to-IP Routing' on page 279).



Notes:

- IP Group ID 0 cannot be used. This IP Group is set to default values and is used by the device when IP Groups are not implemented.
- When operating with multiple IP Groups, the default Proxy server must not be used (i.e., the parameter `IsProxyUsed` must be set to 0).
- If different SRDs are configured in the IP Group and Proxy Set tables, the SRD defined for the Proxy Set takes precedence.
- You can also configure the IP Groups table using the table ini file parameter, `IPGroup` (see 'Configuration Parameters Reference' on page 437).

➤ To configure IP Groups:

1. Open the IP Group Table page (**Configuration** tab > **VoIP** menu > **Control Network** submenu > **IP Group Table**).
2. Click the **Add** button: the following dialog box appears:

Figure 16-3: IP Group Table – Add

Common		SBC
Index	0	
Type	Server	
Description		
Proxy Set ID	-1	
SIP Group Name		
Contact User		
Local Host Name		
SRD	0	
Media Realm Name		
IP Profile ID	0	
<input type="button" value="Submit"/> <input type="button" value="Cancel"/>		

3. Configure the IP Group parameters according to the table below.
4. Click **Submit**.
5. To save the changes to flash memory, see 'Saving Configuration' on page 360.

Table 16-3: IP Group Parameters

Parameter	Description
Common Parameters	
Type [IPGroup_Type]	<p>Defines the type of IP Group:</p> <ul style="list-style-type: none"> ▪ [0] Server = Used when the destination address, configured by the Proxy Set, of the IP Group (e.g., ITSP, Proxy, IP-PBX, or Application server) is known. ▪ [1] User = Represents a group of users such as IP phones and softphones where their location is dynamically obtained by the device when REGISTER requests and responses traverse (or are terminated) by the device. These users are considered remote (far-end) users. <p>Typically, this IP Group is configured with a Serving IP Group that represents an IP-PBX, Application or Proxy server that serves this User-type IP Group. Each SIP request sent by a user of this IP Group is proxied to the Serving IP Group. For registrations, the device updates its internal database with the AOR and contacts of the users.</p> <p>Digest authentication using SIP 401/407 responses (if needed) is performed by the Serving IP Group. The device forwards these responses directly to the SIP users.</p> <p>To route a call to a registered user, a rule must be configured in the SBC IP-to-IP Routing table. The device searches the dynamic database (by using the request URI) for an entry that matches a registered AOR or Contact. Once an entry is found, the IP destination is obtained from this entry, and a SIP request is sent to the destination.</p> <p>The device also supports NAT traversal for the SIP clients located behind NAT. In this case, the device must be defined with a global IP address.</p> <ul style="list-style-type: none"> ▪ [2] Gateway = This is applicable only to the SBC application in scenarios where the device receives requests to and from a gateway representing multiple users. This IP Group type is necessary as the other IP Group types are not suitable: <ul style="list-style-type: none"> ✓ The IP Group cannot be defined as a Server since its destination address is unknown during configuration. ✓ The IP Group cannot be defined as a User since the SIP Contact header of the incoming REGISTER does not represent a specific user. The Request-URI user part can change and therefore, the device is unable to identify an already registered user and therefore, adds an additional record to the database. <p>The IP address of the Gateway IP Group is obtained dynamically from the host part of the Contact header in the REGISTER request received from the IP Group. Therefore, routing to this IP Group is possible only once a REGISTER request is received. If a REGISTER refresh request arrives, the device updates the new location (i.e., IP address) of the IP Group. If the REGISTER fails, no update is performed. If an UN-REGISTER request arrives, the IP address associated with the IP Group is deleted and therefore, no routing to the IP Group is</p>

Parameter	Description
	done.
Description [IPGroup_Description]	Defines a brief description for the IP Group. The valid value is a string of up to 29 characters. The default is an empty field.
Proxy Set ID [IPGroup_ProxySetId]	Assigns a Proxy Set ID to the IP Group. All INVITE messages destined to this IP Group are sent to the IP address configured for the Proxy Set. Notes: <ul style="list-style-type: none"> Proxy Set ID 0 must not be used; this is the device's default Proxy. The Proxy Set is applicable only to Server-type IP Groups. The SRD configured for this Proxy Set in the Proxy Set table is automatically assigned to this IP Group (see the 'SRD' field below). To configure Proxy Sets, see 'Configuring Proxy Sets Table' on page 197.
SIP Group Name [IPGroup_SIPGroupName]	Defines the SIP Request-URI host name used in INVITE and REGISTER messages sent to this IP Group, or the host name in the From header of INVITE messages received from this IP Group. The valid value is a string of up to 100 characters. The default is an empty field. Notes: <ul style="list-style-type: none"> If this parameter is not configured, the value of the global parameter, ProxyName is used instead (see 'Configuring Proxy and Registration Parameters' on page 205). If the IP Group is of User type, this parameter is used internally as a host name in the Request-URI for Tel-to-IP initiated calls. For example, if an incoming call from the device's T1 trunk is routed to a User-type IP Group, the device first creates the Request-URI (<destination_number>@<SIP Group Name>), and then it searches the internal database for a match.
Contact User [IPGroup_ContactUser]	Defines the user part of the From, To, and Contact headers of SIP REGISTER messages, and the user part of the Contact header of INVITE messages received from this IP Group and forwarded by the device to another IP Group. Notes: <ul style="list-style-type: none"> This parameter is applicable only to Server-type IP Groups. This parameter is overridden by the 'Contact User' parameter in the 'Account' table (see 'Configuring Account Table' on page 203).
Local Host Name [IPGroup_ContactName]	Defines the host name (string) that the device uses in the SIP message's Via and Contact headers. This is typically used to define an FQDN as the host name. The device uses this string for Via and Contact headers in outgoing INVITE messages to a specific IP Group, and the Contact header in SIP 18x and 200 OK responses for incoming INVITE messages from a specific IP Group. The Inbound IP Routing table can be used to identify the source IP Group from where the INVITE message was received. If this parameter is not configured (default), these headers are populated with the device's dotted-decimal IP address of the

Parameter	Description
	network interface on which the message is sent. Note: To ensure proper device handling, this parameter should be a valid FQDN.
SRD [IPGroup_SRD]	Assigns an SRD to the IP Group. The default is 0. Notes: <ul style="list-style-type: none"> For this parameter to take effect, a device reset is required. To configure SRDs, see Configuring SRD Table on page 185. For Server-type IP Groups, if you assign the IP Group with a Proxy Set ID (in the 'Proxy Set ID' field), the SRD field is automatically set to the SRD value assigned to the Proxy Set in the Proxy Set table.
Media Realm Name [IPGroup_MediaRealm]	Assigns a Media Realm to the IP Group. The string value must be identical (including case-sensitive) to the Media Realm name defined in the Media Realm table. Notes: <ul style="list-style-type: none"> For this parameter to take effect, a device reset is required. If the Media Realm is later deleted from the Media Realm table, then this value becomes invalid. For configuring Media Realms, see Configuring Media Realms on page 157.
IP Profile ID [IPGroup_ProfileId]	Assigns an IP Profile to the IP Group. The default is 0. Note: To configure IP Profiles, see 'Configuring IP Profiles' on page 221.
SBC Parameters	
Classify By Proxy Set [IPGroup_ClassifyByProxySet]	Defines whether the incoming INVITE is classified to an IP Group according to its associated Proxy Set ID. <ul style="list-style-type: none"> [0] Disable [1] Enable (default) <p>This classification occurs only if classification according to the device's database fails to (i.e., received INVITE is not from a registered user). The classification proceeds with checking whether the INVITE's IP address (if host names, then according to the dynamically resolved IP address list) is defined for a Proxy Set ID (in the Proxy Set table). If a Proxy Set ID has such an IP address, the device classifies the INVITE as belonging to the IP Group associated with this Proxy Set. The Proxy Set ID is assigned to the IP Group using the IP Group table's 'Proxy Set ID' parameter (see above).</p> Notes: <ul style="list-style-type: none"> In cases where multiple IP Groups are associated with the same Proxy Set ID, do not enable this feature. If enabled, the device is unable to correctly classify the incoming INVITEs to the appropriate IP Groups. To enhance security, it is highly recommended to disable this parameter so that the device can use the Classification table rules to classify the call. If this parameter is enabled, the Classification table is not used if an associated Proxy Set is found.

Parameter	Description
	<ul style="list-style-type: none"> This parameter is applicable only to Server-type IP Groups.
Max Number Of Registered Users [IPGroup_MaxNumOfRegUsers]	<p>Defines the maximum number of users in this IP Group that can register with the device. By default, no limitation exists for registered users.</p> <p>Note: This field is applicable only to User-type IP Groups.</p>
Source URI Input [IPGroup_SourceUriInput]	<p>Defines the SIP header in the incoming INVITE to use for call matching characteristics based on source URIs.</p> <ul style="list-style-type: none"> [-1] Not Configured (default) [0] From [1] To [2] Request-URI [3] P-Asserted - First Header [4] P-Asserted - Second Header [5] P-Preferred [6] Route [7] Diversion [8] P-Associated-URI [9] P-Called-Party-ID [10] Contact <p>Notes:</p> <ul style="list-style-type: none"> This parameter is applicable only when classification is done according to the Classification table. If the configured SIP header does not exist in the incoming INVITE message, the classification of the message to a source IP Group fails. If the device receives an INVITE as a result of a REFER request or a 3xx response, then the incoming INVITE is routed according to the Request-URI. The device identifies such INVITES according to a specific prefix in the Request-URI header, configured by the SBCXferPrefix parameter. Therefore, in this scenario, the device ignores this parameter setting.
Destination URI Input [IPGroup_DestUriInput]	<p>Defines the SIP header in the incoming INVITE to use for call matching characteristics based on destination URIs.</p> <ul style="list-style-type: none"> [-1] Not Configured (default) [0] From [1] To [2] Request-URI [3] P-Asserted - First Header [4] P-Asserted - Second Header [5] P-Preferred [6] Route [7] Diversion [8] P-Associated-URI [9] P-Called-Party-ID [10] Contact <p>Notes:</p> <ul style="list-style-type: none"> This parameter is applicable only when classification is done according to the Classification table.

Parameter	Description
	<ul style="list-style-type: none"> If the configured SIP header does not exist in the incoming INVITE message, the classification of the message to a source IP Group fails. If the device receives an INVITE as a result of a REFER request or a 3xx response, then the incoming INVITE is routed according to the Request-URI. The device identifies such INVITEs according to a specific prefix in the Request-URI header, configured by the SBCXferPrefix parameter. Therefore, in this scenario, the device ignores this parameter setting.
Inbound Message Manipulation Set [IPGroup_InboundManSet]	Assigns a Message Manipulation Set (rule) to the IP Group for SIP message manipulation on the inbound message. The Message Manipulation rules are configured using the MessageManipulations parameter (see Configuring SIP Message Manipulation on page 208).
Outbound Message Manipulation Set [IPGroup_OutboundManSet]	Message Manipulation Set (rule) that you want to assign to this IP Group for SIP message manipulation on the outbound message. The Message Manipulation rules are configured using the MessageManipulations parameter (see Configuring SIP Message Manipulation on page 208).
Registration Mode [IPGroup_RegistrationMode]	<p>Defines the registration mode for the IP Group:</p> <ul style="list-style-type: none"> [0] User initiates registrations (default) [1] SBC initiate registrations = Used when the device serves as a client (e.g., with an IP PBX). This functions only with User Info file. [2] No registrations needed = The device adds users to its database in active state.
Authentication Mode [IPGroup_AuthenticationMode]	<p>Defines the authentication mode.</p> <ul style="list-style-type: none"> [0] User Authenticates = (Default) The device does not handle the authentication, but simply passes the authentication messages between the SIP user agents. [1] SBC as client = The device authenticates as a client. It receives the 401/407 response from the proxy requesting for authentication. The device sends the proxy the authorization credentials (i.e., user name and password) according to one of the following: 1) account defined in the Account table (only if authenticating Server-type IP Group), 2) global username and password parameters (only if authenticating Server-type IP Group), 3) User Information file, or 4) sends request to users requesting credentials (only if authenticating User-type IP Group). [2] SBC as Server = The device authenticates as a server (using the User Information file).
Authentication Method List [IPGroup_MethodList]	<p>Defines SIP methods that the device must challenge. Multiple entries are separated by the backslash "\". If you set this parameter to an empty value, no methods are challenged.</p> <p>The default value is "INVITE\REGISTER".</p> <p>Note: This parameter is applicable only if the 'Authentication Mode' parameter is set to SBC as Server [2].</p>

Parameter	Description
SBC Client Forking Mode CLI: enable-sbc-client-forking [IPGroup_EnableSBCClientForking]	<p>Defines call forking of INVITE messages to up to five separate SIP outgoing legs for User-type IP Groups. This occurs if multiple contacts are registered under the same AoR in the device's registration database.</p> <ul style="list-style-type: none"> ▪ [0] Sequential = (Default) Sequentially sends the INVITE to each contact. If there is no answer from the first contact, it sends the INVITE to the second contact, and so on until a contact answers. If no contact answers, the call fails or is routed to an alternative destination, if configured. ▪ [1] Parallel = Sends the INVITE simultaneously to all contacts. The call is established with the first contact that answers. ▪ [2] Sequential Available Only = Sequentially sends the INVITE only to available contacts (i.e., not busy). If there is no answer from the first available contact, it sends the INVITE to the second contact, and so on until a contact answers. If no contact answers, the call fails or is routed to an alternative destination, if configured. <p>Note: The device can also fork INVITE messages received for a Request-URI of a specific contact (user) registered in the database to all other users located under the same AoR as the specific contact. This is configured using the SBCSendInviteToAllContacts parameter.</p>

16.4 Configuring Proxy Sets Table

The Proxy Sets Table page allows you to define *Proxy Sets*. A Proxy Set is a group of Proxy servers defined by IP address or fully qualified domain name (FQDN). You can define up to 32 Proxy Sets, each with up to five Proxy server addresses. For each Proxy server address you can define the transport type (i.e., UDP, TCP, or TLS). In addition, Proxy load balancing and redundancy mechanisms can be applied per Proxy Set if it contains more than one Proxy address.

Proxy Sets can later be assigned to Server-type IP Groups (see 'Configuring IP Groups' on page 191). When the device sends an INVITE message to an IP Group, it is sent to the IP address or domain name defined for the Proxy Set that is associated with the IP Group. In other words, the Proxy Set represents the **destination** of the call. Typically, for IP-to-IP call routing, at least two Proxy Sets are defined for call destination – one for each leg (IP Group) of the call (i.e., both directions). For example, one Proxy Set for the Internet Telephony Service provider (ITSP) interfacing with one 'leg' of the device and another Proxy Set for the second SIP entity (e.g., ITSP) interfacing with the other 'leg' of the device.



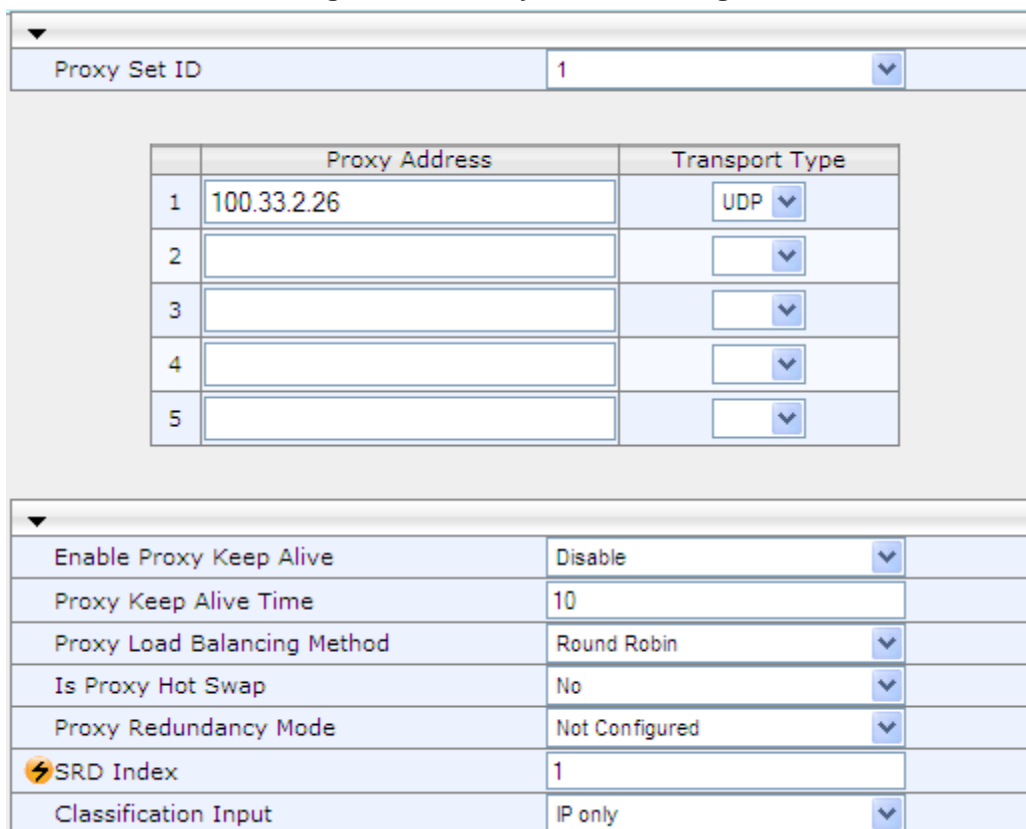
Notes:

- Proxy Sets can be assigned only to Server-type IP Groups.
- To enable classification of IP Groups according to Proxy Set ID, in the IP Group table, set the 'Classify By Proxy Set' parameter to Enable.
- The Proxy Set table can also be configured using two complementary tables:
 - ✓ Proxy Set ID with IP addresses: Table ini file parameter, ProxyIP.
 - ✓ Attributes for the Proxy Set: Table ini file parameter, ProxySet.

➤ **To configure Proxy Sets:**

1. Open the Proxy Sets Table page (**Configuration** tab > **VoIP** menu > **Control Network** submenu > **Proxy Sets Table**).

Figure 16-4: Proxy Sets Table Page



Proxy Set ID: 1

	Proxy Address	Transport Type
1	100.33.2.26	UDP
2		
3		
4		
5		

Enable Proxy Keep Alive: Disable
 Proxy Keep Alive Time: 10
 Proxy Load Balancing Method: Round Robin
 Is Proxy Hot Swap: No
 Proxy Redundancy Mode: Not Configured
 SRD Index: 1
 Classification Input: IP only

2. From the 'Proxy Set ID' drop-down list, select an ID for the desired group.
3. Configure the Proxy parameters, as required. For a description of the parameters, see the table below.
4. Click **Submit**.
5. To save the changes to flash memory, see 'Saving Configuration' on page 360.

Table 16-4: Proxy Sets Table Parameters

Parameter	Description
Web: Proxy Set ID EMS: Index [ProxySet_Index]	<p>Defines the Proxy Set identification number.</p> <p>The valid value is 0 to 31. Proxy Set ID 0 is used as the default Proxy Set.</p> <p>Typically, when IP Groups are used, there is no need to use the default Proxy and all routing and registration rules can be configured using IP Groups and the Account tables (see 'Configuring Account Table' on page 203).</p>
Proxy Address [ProxyIp_IPAddress]	<p>Defines the address (and optionally, port number) of the Proxy server. Up to five addresses can be configured per Proxy Set.</p> <p>The address can be defined as an IP address in dotted-decimal notation (e.g., 201.10.8.1) or FQDN. You can also specify the port in the format:</p> <ul style="list-style-type: none"> For IPv4 address: <IP address>:<port> (e.g., 201.10.8.1:5060) For IPv6 address: <[IPV6 address]>:<port> (e.g.,

Parameter	Description
	<p>[2000::1:200:200:86:14]:5060)</p> <p>If you enable Proxy Redundancy (by setting the parameter EnableProxyKeepAlive to 1 or 2), the device can operate with multiple Proxy servers. If there is no response from the first (<i>primary</i>) Proxy defined in the list, the device attempts to communicate with the other (<i>redundant</i>) Proxies in the list. When a redundant Proxy is located, the device either continues operating with it until the next failure occurs or reverts to the primary Proxy (refer to the parameter ProxyRedundancyMode). If none of the Proxy servers respond, the device goes over the list again.</p> <p>The device also provides real-time switching (Hot-Swap mode) between the primary and redundant proxies (refer to the parameter IsProxyHotSwap). If the first Proxy doesn't respond to the INVITE message, the same INVITE message is immediately sent to the next Proxy in the list. The same logic applies to REGISTER messages (if RegistrarIP is not defined).</p> <p>Notes:</p> <ul style="list-style-type: none"> ▪ If EnableProxyKeepAlive is set to 1 or 2, the device monitors the connection with the Proxies by using keep-alive messages (OPTIONS or REGISTER). ▪ To use Proxy Redundancy, you must specify one or more redundant Proxies. ▪ When a port number is specified (e.g., domain.com:5080), DNS NAPTR/SRV queries aren't performed, even if ProxyDNSQueryType is set to 1 or 2.
Transport Type [ProxyIp_TransportType]	<p>Defines the transport type of the proxy server.</p> <ul style="list-style-type: none"> ▪ [0] UDP ▪ [1] TCP ▪ [2] TLS ▪ [-1] = Undefined <p>Note: If no transport type is selected, the value of the global parameter SIPTransportType is used.</p>
Web/EMS: Enable Proxy Keep Alive [ProxySet_EnableProxyKeepAlive]	<p>Enables the Keep-Alive mechanism with the Proxy server(s).</p> <ul style="list-style-type: none"> ▪ [0] Disable (default). ▪ [1] Using Options = Enables Keep-Alive with Proxy using SIP OPTIONS messages. ▪ [2] Using Register = Enables Keep-Alive with Proxy using SIP REGISTER messages. <p>If set to 'Using Options', the SIP OPTIONS message is sent every user-defined interval (configured by the parameter ProxyKeepAliveTime). If set to 'Using Register', the SIP REGISTER message is sent every user-defined interval (configured by the SBCProxyRegistrationTime parameter for SBC application). Any response from the Proxy, either success (200 OK) or failure (4xx response) is considered as if the Proxy is communicating correctly.</p> <p>Notes:</p> <ul style="list-style-type: none"> ▪ For Survivability mode for User-type IP Groups, this parameter must be enabled (1 or 2). ▪ This parameter must be set to 'Using Options' when Proxy redundancy is used.

Parameter	Description
	<ul style="list-style-type: none"> When this parameter is set to 'Using Register', the homing redundancy mode is disabled. When the active proxy doesn't respond to INVITE messages sent by the device, the proxy is tagged as 'offline'. The behavior is similar to a Keep-Alive (OPTIONS or REGISTER) failure. If this parameter is enabled and the proxy uses the TCP/TLS transport type, you can enable CRLF Keep-Alive mechanism, using the UsePingPongKeepAlive parameter.
Web: Proxy Keep Alive Time EMS: Keep Alive Time [ProxySet_ProxyKeepAliveTime]	<p>Defines the Proxy keep-alive time interval (in seconds) between Keep-Alive messages.</p> <p>The valid range is 5 to 2,000,000. The default is 60.</p> <p>Note: This parameter is applicable only if the parameter EnableProxyKeepAlive is set to 1 (OPTIONS). When the parameter EnableProxyKeepAlive is set to 2 (REGISTER), the time interval between Keep-Alive messages is determined by the SBCProxyRegistrationTime parameter for the SBC application.</p>
Web: Proxy Load Balancing Method EMS: Load Balancing Method [ProxySet_ProxyLoadBalancingMethod]	<p>Enables the Proxy Load Balancing mechanism per Proxy Set ID.</p> <ul style="list-style-type: none"> [0] Disable = Load Balancing is disabled (default) [1] Round Robin [2] Random Weights <p>When the Round Robin algorithm is used, a list of all possible Proxy IP addresses is compiled. This list includes all IP addresses per Proxy Set, after necessary DNS resolutions (including NAPTR and SRV, if configured). After this list is compiled, the Proxy Keep-Alive mechanism (according to parameters EnableProxyKeepAlive and ProxyKeepAliveTime) tags each entry as 'offline' or 'online'. Load balancing is only performed on Proxy servers that are tagged as 'online'.</p> <p>All outgoing messages are equally distributed across the list of IP addresses. REGISTER messages are also distributed unless a RegistrarIP is configured.</p> <p>The IP addresses list is refreshed according to ProxyIPListRefreshTime. If a change in the order of the entries in the list occurs, all load statistics are erased and balancing starts over again.</p> <p>When the Random Weights algorithm is used, the outgoing requests are not distributed equally among the Proxies. The weights are received from the DNS server by using SRV records. The device sends the requests in such a fashion that each Proxy receives a percentage of the requests according to its assigned weight. A single FQDN should be configured as a Proxy IP address. The Random Weights Load Balancing is not used in the following scenarios:</p> <ul style="list-style-type: none"> The Proxy Set includes more than one Proxy IP address. The only Proxy defined is an IP address and not an FQDN. SRV is not enabled (DNSQueryType). The SRV response includes several records with a different Priority value.
Web/EMS: Is Proxy Hot-Swap [ProxySet_IsProxyHotSwap]	<p>Enables the Proxy Hot-Swap redundancy mode.</p> <ul style="list-style-type: none"> [0] No (default) [1] Yes <p>If Proxy Hot-Swap is enabled, the SIP INVITE/REGISTER</p>

Parameter	Description
	message is initially sent to the first Proxy/Registrar server. If there is no response from the first Proxy/Registrar server after a specific number of retransmissions (configured by the parameter HotSwapRtx), the message is resent to the next redundant Proxy/Registrar server.
Web/EMS: Redundancy Mode [ProxySet_ProxyRedundancy Mode]	<p>Determines whether the device switches back to the primary Proxy after using a redundant Proxy.</p> <ul style="list-style-type: none"> ▪ [-1] Not configured = (Default) The global parameter, ProxyRedundancyMode applies. ▪ [0] Parking = The device continues operating with a redundant (now active) Proxy until the next failure, after which it operates with the next redundant Proxy. ▪ [1] Homing = The device always attempts to operate with the primary Proxy server (i.e., switches back to the primary Proxy whenever it's available). <p>Notes:</p> <ul style="list-style-type: none"> ▪ To use the Proxy Redundancy mechanism, you need to enable the keep-alive with Proxy option, by setting the parameter EnableProxyKeepAlive to 1 or 2. ▪ If this parameter is configured, then the global parameter is ignored.
Web/EMS: SRD Index [ProxySet_ProxySet_SRD]	<p>Defines the SRD associated with the Proxy Set ID. The default is SRD 0.</p> <p>Notes:</p> <ul style="list-style-type: none"> ▪ For this parameter to take effect, a device reset is required. ▪ To configure SRDs, see Configuring SRD Table on page 185.
Web/EMS: Classification Input [ClassificationInput]	<p>Defines how the device classifies an IP call to the Proxy Set.</p> <ul style="list-style-type: none"> ▪ [0] Compare only IP = (Default) The call is classified to the Proxy Set according to its IP address only. ▪ [1] Compare IP, port and transport type = The call is classified to the Proxy Set according to its IP address, port, and transport type. <p>Note: This parameter is applicable only if the IP Group table's parameter, 'Classify by Proxy Set' is set to Enable.</p>

Reader's Notes

17 SIP Definitions

This section describes configuration of SIP parameters.

17.1 Configuring SIP Parameters

Many of the stand-alone SIP parameters associated with various features can be configured in the following pages:

- **SIP General Parameters page:** Provides SIP parameters for configuring general SIP features. To access this page, use the following path: **Configuration** tab > **VoIP** menu > **SIP Definitions** > **General Parameters**.
- **SIP Advanced Parameters page:** Provides SIP parameters for configuring advanced SIP features. To access this page, use the following path: **Configuration** tab > **VoIP** menu > **SIP Definitions** > **Advanced Parameters**.

For a description of these parameters, refer to the section corresponding to the feature or see 'Configuration Parameters Reference' on page 437.

17.2 Configuring Account Table

The Account Table page lets you define up to 32 Accounts per source ("served") IP Group. Accounts are used to register and/or digest authenticate a served IP Group, using a username and password, to a destination ("serving") IP Group. For example, the device can use the Account table to register an IP PBX, which is connected to the device, to an ITSP. The device sends the registration requests to the Proxy Set ID (see 'Configuring Proxy Sets Table' on page 197) that is associated with the serving IP Group.

A served IP Group can register to more than one serving IP Group (e.g., multiple ITSPs). This is done by configuring multiple entries in the Account table for the same served IP Group, but with different serving IP Groups, user name/password, host name, and contact user values.



Note: The Account table can also be configured using the table ini file parameter, Account.

➤ To configure Accounts:

1. Open the Account Table page (**Configuration** tab > **VoIP** menu > **SIP Definitions** submenu > **Account Table**).

Figure 17-1: Account Table Page

Index	Served Trunk Group	Serving IP Group	User Name	Password	Host Name	Register	Contact User	Application Type
1	-1	1		•		No ▼		SBC ▼

2. In the 'Add' field, enter the desired table row index, and then click **Add**. A new row appears.
3. Configure the Account parameters according to the table below.
4. Click the **Apply** button to save your changes.
5. To save the changes, see 'Saving Configuration' on page 360.
6. To perform registration, click the **Register** button; to unregister, click **Unregister**. The registration method for each Trunk Group is according to the setting of the 'Registration Mode' parameter in the Trunk Group Settings page.

Table 17-1: Account Table Parameters Description

Parameter	Description
Served IP Group [Account_ServedIPGroup]	Defines the Source IP Group (e.g., IP-PBX) for which registration and/or authentication is done.
Serving IP Group [Account_ServingIPGroup]	<p>Defines the destination IP Group ID to where the SIP REGISTER requests, if enabled, are sent and authentication is done. The actual destination to where the REGISTER requests are sent is the IP address configured for the Proxy Set ID that is associated with the IP Group.</p> <p>Registration occurs only if the 'Register' parameter in the Account table is set to Yes.</p> <p>Note: If no match is found in this table for incoming or outgoing calls, the username and password is taken from the UserName and Password parameters on the Proxy & Registration page.</p>
Username [Account_Username]	<p>Defines the digest MD5 Authentication user name.</p> <p>The valid value is a string of up to 50 characters.</p>
Password [Account_Password]	<p>Defines the digest MD5 Authentication password.</p> <p>The valid value is a string of up to 50 characters.</p> <p>Note: After you click the Apply button, this password is displayed as an asterisk (*).</p>
Host Name [Account_HostName]	<p>Defines the Address of Record (AOR) host name. It appears in REGISTER From/To headers as ContactUser@HostName. For successful registrations, this host name is also included in the INVITE request's From header URI.</p> <p>This parameter can be up to 49 characters.</p> <p>Note: If this parameter is not configured or if registration fails, the 'SIP Group Name' parameter configured in the IP Group table is used instead.</p>
Register [Account_Register]	<p>Enables registration.</p> <ul style="list-style-type: none"> ▪ [0] No (Default) ▪ [1] Yes <p>When enabled, the device sends REGISTER requests to the Serving IP Group. The host name (i.e., host name in SIP From/To headers) and Contact User (user in From/To and Contact headers) are taken from this table upon successful registration. See the example below:</p> <pre>REGISTER sip:xyz SIP/2.0 Via: SIP/2.0/UDP 10.33.37.78;branch=z9hG4bKac1397582418 From: <sip:ContactUser@HostName>;tag=1c1397576231 To: <sip: ContactUser@HostName > Call-ID: 1397568957261200022256@10.33.37.78 CSeq: 1 REGISTER Contact: <sip:ContactUser@10.33.37.78>;expires=3600 Expires: 3600 User-Agent: Sip-Gateway/v.6.00A.008.002 Content-Length: 0</pre>
Contact User [Account_ContactUser]	<p>Defines the AOR user name. This appears in REGISTER From/To headers as ContactUser@HostName, and in INVITE/200 OK Contact headers as ContactUser@<device's IP address>.</p> <p>Notes:</p>

Parameter	Description
	<ul style="list-style-type: none"> If this parameter is not configured, the 'Contact User' parameter in the IP Group table is used instead. If registration fails, then the user part in the INVITE Contact header contains the source party number.
Application Type [Account_ApplicationType]	Defines the application type: <ul style="list-style-type: none"> [2] SBC = SBC application.


17.3 Configuring Proxy and Registration Parameters

The Proxy & Registration page allows you to configure the Proxy server and registration parameters. For a description of the parameters appearing on this page, see 'Configuration Parameters Reference' on page 437.

➤ To configure the Proxy and registration parameters:

1. Open the Proxy & Registration page (**Configuration** tab > **VoIP** menu > **SIP Definitions** submenu > **Proxy & Registration**).

Figure 17-2: Proxy & Registration Page

Use Default Proxy	Yes
Proxy Set Table	
Proxy Name	
Redundancy Mode	Parking
Proxy IP List Refresh Time	60
Enable Fallback to Routing Table	Disable
Prefer Routing Table	No
Use Routing Table for Host Names and Profiles	Disable
Always Use Proxy	Disable
Redundant Routing Mode	Routing Table
SIP ReRouting Mode	Standard Mode
Enable Registration	Disable
Gateway Name	
Gateway Registration Name	
DNS Query Type	A-Record
Proxy DNS Query Type	A-Record
Subscription Mode	Per Endpoint
Number of RTX Before Hot-Swap	3
Use Gateway Name for OPTIONS	No
User Name	joe
Password	mikey
Cnonce	Default_Cnonce
Registration Mode	Per Endpoint
Set Out-Of-Service On Registration Failure	Disable
Challenge Caching Mode	None
Mutual Authentication Mode	Optional


2. Configure the parameters as required.
3. Click **Submit** to apply your changes.

➤ **To register or un-register the device to a Proxy/Registrar:**

- Click the **Register** button to register.
- Click **Un-Register** button to un-register.

Instead of registering the entire device, you can register specific entities as listed below by using the **Register** button located on the page in which these entities are configured:

- Accounts - Account table (see 'Configuring Account Table' on page 203)

Click the **Proxy Set Table**  button to Open the Proxy Sets Table page to configure groups of proxy addresses. Alternatively, you can open this page from the **Proxy Sets Table** page item (see 'Configuring Proxy Sets Table' on page 197 for a description of this page).

17.3.1 SIP Message Authentication Example

The device supports basic and digest (MD5) authentication types, according to SIP RFC 3261 standard. A proxy server might require authentication before forwarding an INVITE message. A Registrar/Proxy server may also require authentication for client registration. A proxy replies to an unauthenticated INVITE with a 407 Proxy Authorization Required response, containing a Proxy-Authenticate header with the form of the challenge. After sending an ACK for the 407, the user agent can then re-send the INVITE with a Proxy-Authorization header containing the credentials.

User agents, Redirect or Registrar servers typically use the SIP 401 Unauthorized response to challenge authentication containing a WWW-Authenticate header, and expect the re-INVITE to contain an Authorization header.

The following example shows the Digest Authentication procedure, including computation of user agent credentials:

1. The REGISTER request is sent to a Registrar/Proxy server for registration:

```
REGISTER sip:10.2.2.222 SIP/2.0
Via: SIP/2.0/UDP 10.1.1.200
From: <sip: 122@10.1.1.200>;tag=1c17940
To: <sip: 122@10.1.1.200>
Call-ID: 634293194@10.1.1.200
User-Agent: Sip-Gateway/Mediant 4000 E-SBC/v.6.60.010.006
CSeq: 1 REGISTER
Contact: sip:122@10.1.1.200:
Expires:3600
```

2. Upon receipt of this request, the Registrar/Proxy returns a 401 Unauthorized response:

```
SIP/2.0 401 Unauthorized
Via: SIP/2.0/UDP 10.2.1.200
From: <sip:122@10.2.2.222 >;tag=1c17940
To: <sip:122@10.2.2.222 >
Call-ID: 634293194@10.1.1.200
Cseq: 1 REGISTER
Date: Mon, 30 Jul 2012 15:33:54 GMT
Server: Columbia-SIP-Server/1.17
Content-Length: 0
WWW-Authenticate: Digest realm="audiocodes.com",
nonce="11432d6bce58ddf02e3b5e1c77c010d2",
stale=FALSE,
algorithm=MD5
```

3. According to the sub-header present in the WWW-Authenticate header, the correct REGISTER request is created.
4. Since the algorithm is MD5:
 - The username is equal to the endpoint phone number "122".
 - The realm return by the proxy is "audiocodes.com".
 - The password from the *ini* file is "AudioCodes".
 - The equation to be evaluated is "122:audiocodes.com:AudioCodes". According to the RFC, this part is called A1.
 - The MD5 algorithm is run on this equation and stored for future usage.
 - The result is "a8f17d4b41ab8dab6c95d3c14e34a9e1".
5. The par called A2 needs to be evaluated:
 - The method type is "REGISTER".
 - Using SIP protocol "sip".
 - Proxy IP from *ini* file is "10.2.2.222".
 - The equation to be evaluated is "REGISTER:sip:10.2.2.222".
 - The MD5 algorithm is run on this equation and stored for future usage.
 - The result is "a9a031cfddcb10d91c8e7b4926086f7e".
6. Final stage:
 - A1 result: The nonce from the proxy response is "11432d6bce58ddf02e3b5e1c77c010d2".
 - A2 result: The equation to be evaluated is "A1:11432d6bce58ddf02e3b5e1c77c010d2:A2".
 - The MD5 algorithm is run on this equation. The outcome of the calculation is the response needed by the device to register with the Proxy.
 - The response is "b9c45d0234a5abf5ddf5c704029b38cf".

At this time, a new REGISTER request is issued with the following response:

```
REGISTER sip:10.2.2.222 SIP/2.0
Via: SIP/2.0/UDP 10.1.1.200
From: <sip: 122@10.1.1.200>;tag=1c23940
To: <sip: 122@10.1.1.200>
Call-ID: 654982194@10.1.1.200
Server: Audiocodes-Sip-Gateway/Mediant 4000 E-
SBC/v.6.60.010.006
CSeq: 1 REGISTER
Contact: sip:122@10.1.1.200:
Expires:3600
Authorization: Digest, username: 122,
realm="audiocodes.com",
nonce="11432d6bce58ddf02e3b5e1c77c010d2",
uri="10.2.2.222",
response="b9c45d0234a5abf5ddf5c704029b38cf"
```

7. Upon receiving this request and if accepted by the Proxy, the Proxy returns a 200 OK response, completing the registration transaction:

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP 10.1.1.200
From: <sip: 122@10.1.1.200>;tag=1c23940
To: <sip: 122@10.1.1.200>
Call-ID: 654982194@10.1.1.200
Cseq: 1 REGISTER
Date: Thu, 26 Jul 2012 09:34:42 GMT
```

```
Server: Columbia-SIP-Server/1.17
Content-Length: 0
Contact: <sip:122@10.1.1.200>; expires="Thu, 26 Jul 2012
10:34:42 GMT"; action=proxy; q=1.00
Contact: <122@10.1.1.200:>; expires="Tue, 19 Jan 2038 03:14:07
GMT"; action=proxy; q=0.00
Expires: Thu, 26 Jul 2012 10:34:42 GMT
```

17.4 Configuring SIP Message Manipulation

The Message Manipulations page allows you to define up to 100 SIP message manipulation rules. Each manipulation rule can be assigned any Manipulation Set ID (0 to 19), enabling you to create groups (sets) of manipulation rules whereby rules of a group are configured with the same Manipulation Set ID number. To use these Manipulation Sets, you need to assign them to IP Groups in the IP Group table (see 'Configuring IP Groups' on page 191) where they can be applied to inbound and/or outbound SIP messages.

SIP message manipulation enables the normalization of SIP messaging fields between communicating network segments. For example, it allows service providers to design their own policies on the SIP messaging fields that must be present before a SIP call enters their network. Similarly, enterprises and small businesses may have policies for the information that can enter or leave their networks for policy or security reasons from a service provider. The manipulations can also be implemented to resolve incompatibilities between SIP devices inside the enterprise network.

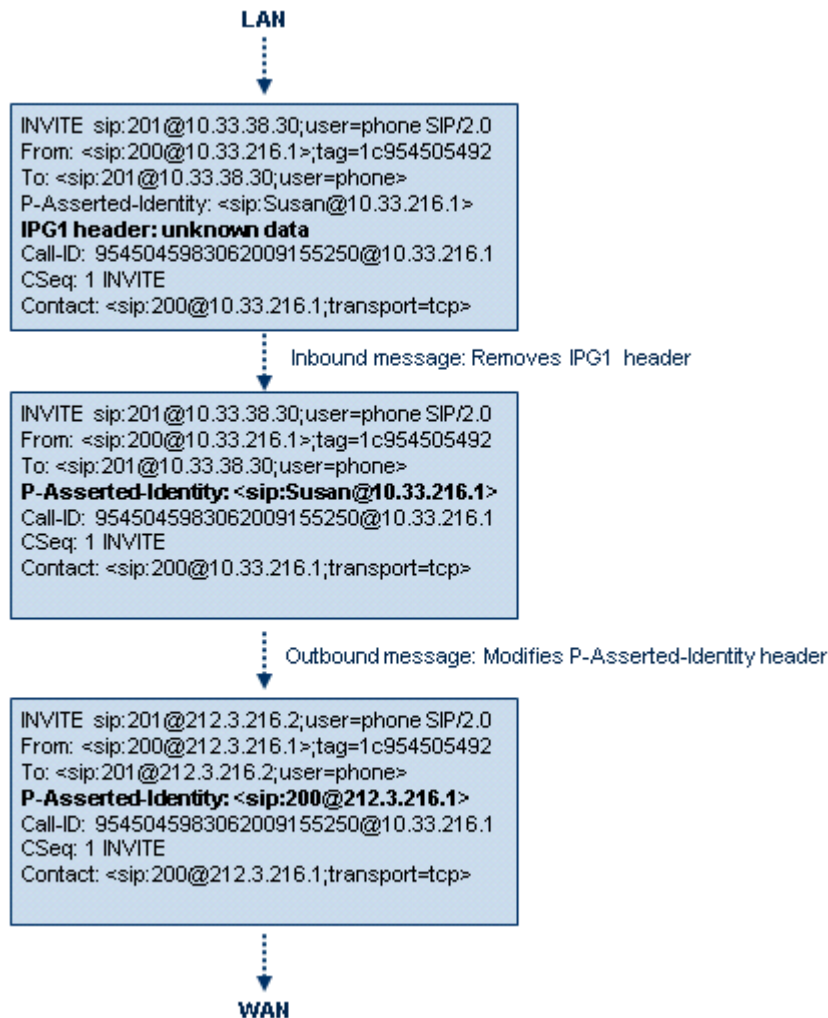
SIP messaging manipulation supports the following:

- Addition of new headers.
- Removal of headers ("Black list").
- Modification of header components - value, header value (e.g., URI value of the P-Asserted-Identity header can be copied to the From header), call's parameter values.
- Deletion of SIP body (e.g., if a message body isn't supported at the destination network this body is removed).
- Translating one SIP response code to another.
- Topology hiding (generally present in SIP headers such as Via, Record Route, Route and Service-Route).
- Configurable identity hiding (information related to identity of subscribers for example, P-Asserted-Identity, Referred-By, Identity and Identity-Info).
- Apply conditions per rule - the condition can be on parts of the message or call's parameters.
- Multiple manipulation rules on the same SIP message.

The manipulation is performed on SIP messages according to the Classification table (source/destination of username/host prefixes, and source IP address). The manipulation can be performed on message type (Method, Request/Response, and Response type) and multiple manipulation rules can be configured for the same SIP message.

The figure below illustrates a SIP message manipulation example:

Figure 17-3: SIP Header Manipulation Example



Notes:

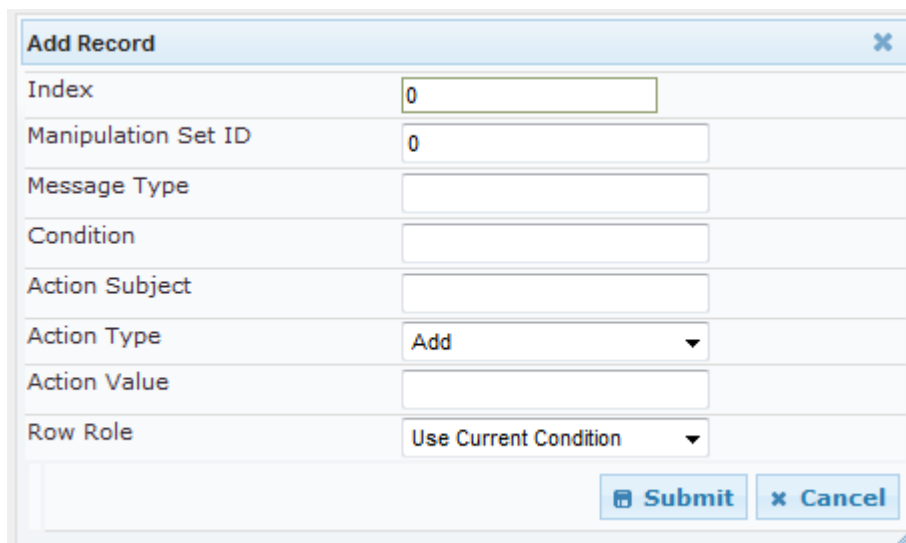
- For a detailed description of the syntax for configuring SIP message manipulation rules, refer to *SIP Message Manipulations Quick Reference Guide*.
- The values entered in the table are not case-sensitive.
- For the SBC application, SIP message manipulation is done only after the Classification, inbound/outbound number manipulations, and routing processes.
- Each message can be manipulated twice - on the source leg and on the destination leg (i.e., source and destination IP Groups).
- Unknown SIP parts can only be added or removed.
- SIP manipulations do not allow you to remove or add mandatory SIP headers. They can only be modified and only on requests that initiate new dialogs. Mandatory SIP headers include To, From, Via, CSeq, Call-Id, and Max-Forwards.
- The Message Manipulations table can also be configured using the table *ini* file parameter, MessageManipulations.



➤ **To configure SIP message manipulation rules:**

1. Open the Message Manipulations page (**Configuration** tab > **VoIP** menu > **SIP Definitions** submenu > **Msg Policy & Manipulation** > **Message Manipulations**).
2. Click the **Add** button; the following dialog box appears:

Figure 17-4: Message Manipulations Table - Add Record Dialog Box



3. Configure the SIP message manipulation rule as required. See the table below for a description of each parameter.
4. Click **Submit** to apply your changes.

The figure below displays an example of configured message manipulation rules:

- Index 0 - adds the suffix ".com" to the host part of the To header.
- Index 1 - changes the user part of the From header to the user part of the P-Asserted-ID.
- Index 2 - changes the user part of the SIP From header to "200".
- Index 3 - if the user part of the From header equals "unknown", then it is changed according to the srcIPGroup call's parameter.
- Index 4 - removes the Priority header from an incoming INVITE message.

Figure 17-5: Message Manipulations Page

Index	Manipulation Set ID	Message Type	Condition	Action Subject	Action Type	Action Value
0	0	invite.response.200		header.to.url.user	Add Prefix	'.com'
1	1	invite.response.200		header.from.url.user	Modify	header.p-asserted-id.url.user
2	2	invite.request		header.from.url.user	Modify	'200'
3	3	invite.request	header.from.url.user='Unkn	header.from.url.user	Modify	param.ipq.src.user
4	4	invite.request		header.priority	Remove	

Table 17-2: Message Manipulations Parameters

Parameter	Description
Index [MessageManipulations_Index]	Defines the table row index for the rule. The valid value is 0 to 99. The default is 0. Note: Each rule must be configured with a unique index.
Manipulation Set ID CLI: manipulation-set-id [MessageManipulations]	Defines a Manipulation Set ID for the rule. You can define the same Manipulation Set ID for multiple rules to create a group of rules. The Manipulation Set ID is used to assign the manipulation rules to an IP

Parameter	Description
_ManSetID]	Group (in the IP Group table) for inbound and/or outbound messages. The valid value is 0 to 19. The default is 0.
Matching Characteristics	
Message Type [MessageManipulations _MessageType]	Defines the SIP message type that you want to manipulate. The valid value is a string denoting the SIP message. For example: <ul style="list-style-type: none"> ▪ Empty = rule applies to all messages ▪ Invite = rule applies to all INVITE requests and responses ▪ Invite.Request = rule applies to INVITE requests ▪ Invite.Response = rule applies to INVITE responses ▪ subscribe.response.2xx = rule applies to SUBSCRIBE confirmation responses Note: Currently, SIP 100 Trying messages cannot be manipulated.
Condition [MessageManipulations _Condition]	Defines the condition that must exist for the rule to apply. The valid value is a string. For example: <ul style="list-style-type: none"> ▪ header.from.url.user== '100' (indicates that the user part of the From header must have the value "100") ▪ header.contact.param.expires > '3600' ▪ header.to.url.host contains 'domain' ▪ param.call.dst.user != '100'
Operation	
Action Subject [MessageManipulations _ActionSubject]	Defines the SIP header upon which the manipulation is performed.
Action Type [MessageManipulations _ActionType]	Defines the type of manipulation. <ul style="list-style-type: none"> ▪ [0] Add (default) = adds new header/param/body (header or parameter elements). ▪ [1] Remove = removes header/param/body (header or parameter elements). ▪ [2] Modify = sets element to the new value (all element types). ▪ [3] Add Prefix = adds value at the beginning of the string (string element only). ▪ [4] Add Suffix = adds value at the end of the string (string element only). ▪ [5] Remove Suffix = removes value from the end of the string (string element only). ▪ [6] Remove Prefix = removes value from the beginning of the string (string element only).
Action Value [MessageManipulations _ActionValue]	Defines a value (string) that you want to use in the manipulation. The syntax is as follows: <ul style="list-style-type: none"> ▪ string/<message-element>/<call-param> + ▪ string/<message-element>/<call-param> For example: <ul style="list-style-type: none"> ▪ 'itsp.com' ▪ header.from.url.user

Parameter	Description
	<ul style="list-style-type: none"> param.call.dst.user param.call.dst.host + '.com' param.call.src.user + '<' + header.from.url.user + '@' + header.p-asserted-id.url.host + '>' <p>Note: Only single quotation marks must be used.</p>
Row Role [MessageManipulations _RowRole]	<p>Determines which condition must be used for the rule of this table row.</p> <ul style="list-style-type: none"> [0] Use Current Condition = The condition entered in this row must be matched in order to perform the defined action (default). [1] Use Previous Condition = The condition of the rule configured directly above this row must be used in order to perform the defined action. This option allows you to configure multiple actions for the same condition. <p>Note: When multiple manipulations rules apply to the same header, the next rule applies to the result string of the previous rule.</p>

17.5 Configuring SIP Message Policy Rules

You can configure SIP message policy rules for blocking (blacklist) unwanted incoming SIP messages or allowing (whitelist) receipt of desired messages. This feature allows you to define legal and illegal characteristics of a SIP message. Message policies can be applied globally (default) or per signaling domain by assigning it to a SIP interface in the SIP Interface table (see 'Configuring SIP Interface Table' on page 188).

This feature is helpful against VoIP fuzzing (also known as robustness testing), which sends different types of packets to its "victims" for finding bugs and vulnerabilities. For example, the attacker might try sending a SIP message containing either an over-sized parameter or too many occurrences of a parameter.

Each message policy rule can be configured with the following:

- Maximum message length
- Maximum SIP header length
- Maximum message body length
- Maximum number of headers
- Maximum number of bodies
- Option to send 400 "Bad Request" response if message request is rejected
- Blacklist and whitelist for defined SIP methods (e.g., INVITE)
- Blacklist and whitelist for defined SIP bodies



Note: The Message Policy table can also be configured using the table ini file parameter, MessagePolicy.

➤ **To configure SIP message policy rules:**

1. Open the Message Policy Table page (**Configuration** tab > **VoIP** menu > **SIP Definitions** > **Msg Policy & Manipulation** > **Message Policy Table**).
2. Click the **Add** button; the Add Record dialog box appears:

Figure 17-6: Message Policy Table - Add Record Dialog Box

Index	1
Max Message Length	1400
Max Header Length	300
Max Body Length	300
Max Num Headers	20
Max Num Bodies	5
Send Rejection	Policy Reject
Method List	INVITEREFER
Method List Type	Policy Blacklist
Body List	
Body List Type	Policy Blacklist

The policy defined above limits SIP messages to 32,768 characters, headers to 256 characters, bodies to 512 characters, number of headers to 16, and only permits two bodies. Invalid requests are rejected. Only INVITE and BYE requests are permitted and there are no restrictions on bodies.

3. Configure the SIP message policy rule as required. See the table below for a description of each parameter.
4. Click **Submit** to apply your changes.
5. To save the changes to flash memory, see 'Saving Configuration' on page 360.

Table 17-3: SIP Message Policy Parameters

Parameter	Description
Index [MessagePolicy_Index]	Defines the table index entry.
Max Message Length [MessagePolicy_MaxMessageLength]	Defines the maximum SIP message length. The valid value is up to 32,768 characters. The default is 32,768.
Max Header Length [MessagePolicy_MaxHeaderLength]	Defines the maximum SIP header length. The valid value is up to 512 characters. The default is 512.
Max Body Length [MessagePolicy_MaxBodyLength]	Defines the maximum SIP message body length. This is the value of the Content-Length header. The valid value is up to 1,024 characters. The default is 1,024.

Parameter	Description
Max Num Headers [MessagePolicy_MaxNumHeaders]	<p>Defines the maximum number of SIP headers.</p> <p>The valid value is any number up to 32. The default is 32.</p> <p>Note: The device supports up to 20 SIP Record-Route headers that can be received in a SIP INVITE request or 200 OK response. If it receives more than this, it responds with a SIP 513 'Message Too Large' response.</p>
Max Num Bodies [MessagePolicy_MaxNumBodies]	<p>Defines the maximum number of bodies (e.g., SDP) in the SIP message.</p> <p>The valid value is any number up to 8. The default is 8.</p>
Send Rejection [MessagePolicy_SendRejection]	<p>Determines whether the device sends a 400 "Bad Request" response if a message request is rejected.</p> <ul style="list-style-type: none"> ▪ [0] Policy Reject = (Default) If the message is a request, then the device sends a response to reject the request. ▪ [1] Policy Drop = The device ignores the message without sending any response.
Method List [MessagePolicy_MethodList]	<p>Defines the SIP methods (e.g., INVITE\BYE) to which the rule applies. The syntax for entering the methods is as follows:</p> <ul style="list-style-type: none"> ▪ Methods must be separated by a backslash (\). ▪ The entered value is not case sensitive.
Method List Type [MessagePolicy_MethodListType]	<p>Determines the policy for the SIP methods.</p> <ul style="list-style-type: none"> ▪ [0] Policy Blacklist = The specified methods (in the 'Method List' field) are rejected by the policy. ▪ [1] Policy Whitelist = (Default) The specified methods (in the 'Method List' field) are allowed by the policy.
Body List [MessagePolicy_BodyList]	<p>Defines the SIP body (i.e., value of the Content-Type header) to which the rule applies.</p>
Body List Type [MessagePolicy_BodyListType]	<p>Determines the policy for the defined SIP body.</p> <ul style="list-style-type: none"> ▪ [0] Policy Blacklist = The specified SIP body (in the 'Body List' field) is rejected by the policy. ▪ [1] Policy Whitelist = (Default) The specified SIP body (in the 'Body List' field) is allowed by the policy.

17.6 Configuring RADIUS Accounting

The the RADIUS Parameters page allows you to enable RADIUS accounting of SIP calls by a RADIUS accounting server. The device can send the accounting messages to the RADIUS server upon call release, call connection and release, or call setup and release.

**Notes:**

- For RADIUS accounting settings to take effect, you must save the settings to flash memory with a device reset.
- For a description of the RADIUS accounting parameters, see 'RADIUS Parameters' on page 467.

➤ **To configure RADIUS accounting:**

1. Open the RADIUS Parameters page (**Configuration** tab > **VoIP** menu > **SIP Definitions** submenu > **RADIUS Parameters Settings**).

Figure 17-7: RADIUS Accounting Parameters Page

▼	
⚡ Enable RADIUS Access Control	Enable ▼
Accounting Server IP Address	0.0.0.0
Accounting Port	1646
RADIUS Accounting Type	At Call Release ▼
AAA Indications	None ▼

2. Configure the parameters as required.
3. Click **Submit**.

Reader's Notes

18 Coders and Profiles

This section describes configuration of the coders and SIP profiles parameters.

18.1 Configuring Coders

The Coders page allows you to configure up to 10 voice coders for the device. Each coder can be configured with packetization time (ptime), bit rate, payload type, and silence suppression. The first coder configured in the table has the highest priority and is used by the device whenever possible. If the remote side cannot use the first coder, the device attempts to use the next coder in the table, and so on.



Notes:

- A specific coder can only be configured once in the table.
- If packetization time and/or rate are not specified, the default is applied.
- Only the packetization time of the first coder in the coder list is declared in INVITE/200 OK SDP, even if multiple coders are defined.
- The device always uses the packetization time requested by the remote side for sending RTP packets. If not specified, the packetization time is assigned the default value.
- The value of several fields is hard-coded according to common standards (e.g., payload type of G.711 U-law is always 0). Other values can be set dynamically. If no value is specified for a dynamic field, a default is assigned. If a value is specified for a hard-coded field, the value is ignored.
- If silence suppression is not configured for a coder, the settings of the EnableSilenceCompression parameter is used.
- The G.722 coder provides Packet Loss Concealment (PLC) capabilities, ensuring higher voice quality.
- For G.729, it's also possible to select silence suppression without adaptations.
- If G.729 is selected and silence suppression is disabled, the device includes 'annexb=no' in the SDP of the relevant SIP messages. If silence suppression is enabled or set to 'Enable w/o Adaptations', 'annexb=yes' is included. An exception to this logic is when the remote gateway is a Cisco device (IsCiscoSCEMode).
- For defining groups of coders, which can be assigned to Tel and IP Profiles, see 'Configuring Coder Groups' on page 220.
- For information on V.152 and implementation of T.38 and VBD coders, see 'Supporting V.152 Implementation' on page 151.
- The Coders table can also be configured using the table *ini* file parameter, CodersGroup.

➤ **To configure the device's coders:**

1. Open the Coders page (**Configuration** tab > **VoIP** menu > **Coders and Profiles** submenu > **Coders**).

Figure 18-1: Coders Table Page

Coder Name	Packetization Time	Rate	Payload Type	Silence Suppression
G.723.1	30	5.3	4	Disabled

2. From the 'Coder Name' drop-down list, select the required coder.
3. From the 'Packetization Time' drop-down list, select the packetization time (in msec) for the selected coder. The packetization time determines how many coder payloads are combined into a single RTP packet.
4. From the 'Rate' drop-down list, select the bit rate (in kbps) for the selected coder.
5. In the 'Payload Type' field, if the payload type (i.e., format of the RTP payload) for the selected coder is dynamic, enter a value from 0 to 120 (payload types of 'well-known' coders cannot be modified).
6. From the 'Silence Suppression' drop-down list, enable or disable the silence suppression option for the selected coder.
7. Repeat steps 2 through 6 for the next optional coders.
8. Click **Submit**.
9. To save the changes to flash memory, see 'Saving Configuration' on page 360.

The table below lists the supported coders:

Table 18-1: Supported Coders

Coder Name	Packetization Time (msec)	Rate (kbps)	Payload Type	Silence Suppression
G.711 A-law [g711Alaw64k]	10, 20 (default), 30, 40, 50, 60, 80, 100, 120	64	8	<ul style="list-style-type: none"> ▪ [0] Disable ▪ [1] Enable
G.711 U-law [g711Ulaw64k]	10, 20 (default), 30, 40, 50, 60, 80, 100, 120	64	0	<ul style="list-style-type: none"> ▪ [0] Disable ▪ [1] Enable
G.711A-law_VBD [g711AlawVbd]	10, 20 (default), 30, 40, 50, 60, 80, 100, 120	64	Dynamic (0-127) Default is 180	N/A
G.711U-law_VBD [g711UlawVbd]	10, 20 (default), 30, 40, 50, 60, 80, 100, 120	64	Dynamic (0-127) Default is 120	N/A
G.722 [g722]	20 (default), 40, 60, 80, 100, 120	64 (default)	9	N/A

Coder Name	Packetization Time (msec)	Rate (kbps)	Payload Type	Silence Suppression
G.723.1 [g7231]	30 (default), 60, 90, 120, 150	<ul style="list-style-type: none"> ▪ [0] 5.3 (default) ▪ [1] 6.3 	4	<ul style="list-style-type: none"> ▪ [0] Disable ▪ [1] Enable
G.726 [g726]	10, 20 (default), 30, 40, 50, 60, 80	<ul style="list-style-type: none"> ▪ [0] 16 ▪ [1] 24 ▪ [2] 32 (default) ▪ [3] 40 	Dynamic (0-127) Default is 23	<ul style="list-style-type: none"> ▪ [0] Disable ▪ [1] Enable
G.729 [g729]	10, 20 (default), 30, 40, 50, 60, 80, 100	8	18	<ul style="list-style-type: none"> ▪ [0] Disable ▪ [1] Enable ▪ [2] Enable w/o Adaptations
AMR [Amr]	20 (default)	<ul style="list-style-type: none"> ▪ [0] 4.75 ▪ [1] 5.15 ▪ [2] 5.90 ▪ [3] 6.70 ▪ [4] 7.40 ▪ [5] 7.95 ▪ [6] 10.2 ▪ [7] 12.2 (default) 	Dynamic (0-127)	<ul style="list-style-type: none"> ▪ [0] Disable ▪ [1] Enable
AMR-WB [Amr-WB]	20 (default)	<ul style="list-style-type: none"> ▪ [0] 6.6 ▪ [1] 8.85 ▪ [2] 12.65 ▪ [3] 14.25 ▪ [4] 15.85 ▪ [5] 18.25 ▪ [6] 19.85 ▪ [7] 23.05 ▪ [8] 23.85 (default) 	Dynamic (0-127)	<ul style="list-style-type: none"> ▪ [0] Disable ▪ [1] Enable
silk-nb [Silk-8KHz]	20 (default), 40, 60, 80, and 100	8	Dynamic (default is 76)	N/A
silk-wb [Silk-16KHz]	20 (default), 40, 60, 80, and 100	16	Dynamic (default is 77)	N/A
T.38 [t38fax]	N/A	N/A	N/A	N/A
T.38 Version 3 [t38fax]	-	-	-	-

18.2 Configuring Coder Groups

The Coder Group Settings page allows you to define up to 10 groups of coders (termed *Coder Groups*). For each Coder Group, you can define up to 10 coders configured with packetization time (ptime), rate, payload type, and silence suppression. The first coder in the Coder Group table has the highest priority and is used by the device whenever possible. If the remote side cannot use the first coder, the device attempts to use the next coder, and so on.

Coder Groups can be used as follows: Assigned to IP Profiles in the IP Profiles table (see 'Configuring IP Profiles' on page 221).



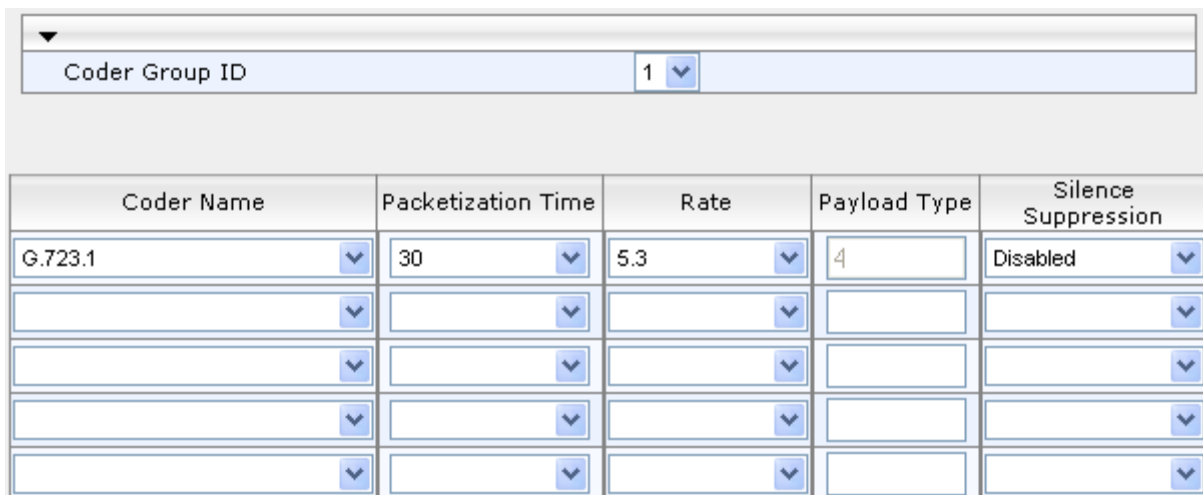
Notes:

- A specific coder can be selected only once per Coder Group.
- For a list of supported coders, see 'Configuring Coders' on page 217.
- The Coder Group Settings table can also be configured using the table *ini* file parameter, CodersGroup.

➤ To configure Coder Groups:

1. Open the Coder Group Settings page (**Configuration** tab > **VoIP** menu > **Coders and Profiles** submenu > **Coders Group Settings**).

Figure 18-2: Coder Group Settings Page



Coder Name	Packetization Time	Rate	Payload Type	Silence Suppression
G.723.1	30	5.3	4	Disabled

2. From the 'Coder Group ID' drop-down list, select a Coder Group ID.
3. From the 'Coder Name' drop-down list, select the first coder for the Coder Group.
4. From the 'Packetization Time' drop-down list, select the packetization time (in msec) for the coder. The packetization time determines how many coder payloads are combined into a single RTP packet.
5. From the 'Rate' drop-down list, select the bit rate (in kbps) for the coder you selected.
6. In the 'Payload Type' field, if the payload type (i.e., format of the RTP payload) for the coder you selected is dynamic, enter a value from 0 to 120 (payload types of common coders cannot be modified).
7. From the 'Silence Suppression' drop-down list, enable or disable the silence suppression option for the coder you selected.
8. Repeat steps 3 through 7 for the next coders (optional).
9. Repeat steps 2 through 8 for the next coder group (optional).
10. Click **Submit** to apply your changes.

18.3 Configuring IP Profiles

The IP Profile Settings table allows you to define up to nine *IP Profiles*. An IP Profile is a set of special call configuration behaviors relating to signaling (e.g., SIP message terminations such as REFER) and media (e.g., coder used) applied to specific IP calls (inbound and/or outbound). Therefore, IP Profiles provide high-level adaptation when the device interworks between different IP entities (for Tel and IP sides), each of which may require different handling by the device. For example, if a specific IP entity uses the G.711 coder only, you can configure an IP Profile with G.711 for this IP entity.

Many of the parameters in the IP Profile Settings table have a corresponding "global" parameter. If an IP Profile is not associated with specific calls, the settings of the global parameters are applied to these calls.

IP Profiles are assigned to IP Groups - see Configuring IP Groups on page [191](#).

**Notes:**

- IP Profiles can also be implemented when using a Proxy server (when the AlwaysUseRouteTable parameter is set to 1).
- RxDTMFOption configures the received DTMF negotiation method: [-1] not configured, use the global parameter; [0] don't declare RFC 2833; [1] declare RFC 2833 payload type is SDP.
- You can also configure IP Profiles using the table ini file parameter, IPProfile (see Configuration Parameters Reference on page [437](#)).

➤ **To configure IP Profiles:**

1. Open the IP Profile Settings page (**Configuration** tab > **VoIP** menu > **Coders and Profiles** submenu > **IP Profile Settings**).

Figure 18-3: IP Profile Settings Page

▼	
Profile ID	1 ▼
Profile Name	
▲ Common Parameters	
▲ Gateway Parameters	
▼ SBC	
Transcoding Mode	Force ▼
Extension Coders Group ID	None ▼
Allowed Coders Group ID	None ▼
Allowed Coders Mode	Restriction ▼
Diversion Mode	Don't Care ▼
History Info Mode	Don't Care ▼
Media Security Behavior	As Is ▼
RFC 2833 Behavior	As Is ▼
Alternative DTMF Method	Don't Care ▼
P-Asserted-Identity	Don't Care ▼
SBC Fax Coders Group ID	None ▼
SBC Fax Behavior	0
SBC Fax Offer Mode	0
SBC Fax Answer Mode	1
SBC Session Expires Mode	Transparent ▼
SBC Remote Early Media RTP	Immediate ▼
SBC Remote Can Play Ringback	Yes ▼
SBC Remote Supports RFC 3960	Not Supported ▼
SBC Multiple 18x Support	supported ▼
SBC Early Media Response Type	Transparent ▼
SBC Remote Update Support	Supported ▼
SBC Remote Re-Invite Support	Supported ▼
SBC Remote Refer Behavior	Transparent ▼
SBC Remote Early Media Support	supported ▼
SBC Remote 3xx Behavior	Transparent ▼
SBC Remote Delayed Offer Support	Supported ▼
SBC PRACK Mode	Transparent ▼
SBC Enforce MKI Size	do-not-enforce ▼
SBC User Registration Time	60
SBC Remote Hold Format	transparent ▼

2. From the 'Profile ID' drop-down list, select the IP Profile index.
3. In the 'Profile Name' field, enter an arbitrary name that allows you to easily identify the IP Profile.
4. Configure the parameters as required.
5. Click **Submit** to apply your changes.

Table 18-2: IP Profile Parameters Description

Parameter	Description
Web: Profile ID [IpProfile_Index]	Defines a unique index number for the IP Profile.
Web: Profile Name [IpProfile_ProfileName]	(Optional) Defines a descriptive name for the IP Profile.
Common Parameters	
Web: RTP IP DiffServ [IpProfile_IPDiffServ]	For a description, see the global parameter PremiumServiceClassMediaDiffServ.
Web: Signaling DiffServ [IpProfile_SigIPDiffServ]	For a description, see the global parameter PremiumServiceClassControlDiffServ.
Web: Disconnect on Broken Connection [IpProfile_DisconnectOnBrokenConnection]	For a description, see the global parameter DisconnectOnBrokenConnection.
Web: Media IP Version Preference [IpProfile_MediaIPVersionPreference]	For a description, see the global parameter MediaIPVersionPreference.
Web: Dynamic Jitter Buffer Minimum Delay [IpProfile_JitterBufMinDelay]	For a description, see the global parameter DJBufMinDelay.
Web: Dynamic Jitter Buffer Optimization Factor [IpProfile_JitterBufOptFactor]	For a description, see the global parameter DJBufOptFactor.
Web: RTP Redundancy Depth [IpProfile_RTPRedundancyDepth]	For a description, see the global parameter RTPRedundancyDepth.
Web: Echo Celler [IpProfile_EnableEchoCeller]	For a description, see the global parameter EnableEchoCeller.
Web: Input Gain [IpProfile_InputGain]	For a description, see the global parameter InputGain.
Web: Voice Volume [IpProfile_VoiceVolume]	For a description, see the global parameter VoiceVolume.
Web: Symmetric MKI Negotiation [IpProfile_EnableSymmetricMKI]	For a description, see the global parameter EnableSymmetricMKI.
Web: MKI Size [IpProfile_MKISize]	For a description, see the global parameter SRTPTxPacketMKISize.
Gateway Parameters	
Web: Fax Signaling Method [IpProfile_IsFaxUsed]	For a description, see the global parameter IsFaxUsed.
Web: Play Ringback Tone to IP [IpProfile_PlayRBTone2IP]	For a description, see the global parameter PlayRBTone2IP.

Parameter	Description
Web: Enable Early Media [IpProfile_EnableEarlyMedia]	For a description, see the global parameter EnableEarlyMedia.
Web: Copy Destination Number to Redirect Number [IpProfile_CopyDest2RedirectNumber]	For a description, see the global parameter CopyDest2RedirectNumber.
Web: Media Security Behavior [IpProfile_MediaSecurityBehaviour]	For a description, see the global parameter MediaSecurityBehaviour.
Web: Number of Calls Limit [IpProfile_CallLimit]	<p>Defines the maximum number of concurrent calls (incoming and outgoing). If the number of concurrent calls reaches this limit, the device rejects any new incoming and outgoing calls belonging to this IP Profile.</p> <p>This parameter can also be set to the following:</p> <ul style="list-style-type: none"> ▪ [-1] = (Default) No limitation on calls. ▪ [0] = Calls are rejected. <p>Note: For IP-to-IP calls, you can configure the device to route calls to an alternative IP Group when this maximum number of concurrent calls is reached. To do so, you need to add an alternative routing rule in the Outbound IP Routing table that reroutes the call to an alternative IP Group. You also need to add a rule to the Reason for Alternative Routing table to initiate an alternative rule for Tel-to-IP calls using cause 805.</p>
Web: Progress Indicator to IP [IpProfile_ProgressIndicator2IP]	For a description, see the global parameter ProgressIndicator2IP.
Web: Coder Group [IpProfile_CodersGroupID]	For a description, see the global parameter CodersGroup.
Web: Remote RTP Base UDP Port [IpProfile_RemoteBaseUDPPort]	For a description, see the global parameter RemoteBaseUDPPort.
Web: First Tx DTMF Option [IpProfile_FirstTxDtmfOption]	For a description, see the global parameter TxDTMFOption.
Web: Second Tx DTMF Option [IpProfile_SecondTxDtmfOption]	For a description, see the global parameter TxDTMFOption.
Web: Declare RFC 2833 in SDP [IpProfile_RxDTMFOption]	For a description, see the global parameter RxDTMFOption.
Web: Enable Hold [IpProfile_EnableHold]	For a description, see the global parameter EnableHold.
SBC Parameters	
Web: Transcoding Mode [IpProfile_TranscodingMode]	For a description, see the global parameter TranscodingMode.
Web: Extension Coders Group ID [IpProfile_SBCExtensionCodersGroupID]	Defines the Coder Group ID used for Extended (additional) coders added to the outgoing leg for this profile. This is used when transcoding is required between two IP entities (i.e., the SDP answer from one doesn't include any coder included in the offer previously sent by the other). Therefore, to allow IP

Parameter	Description
	<p>entities to communicate with each other regardless of their capabilities, an Extended coders table with at least one coder that is supported by each IP entity needs to be assigned to each IP Group. Therefore, each offer destined to specific IP Groups includes this coder.</p> <p>Note: To configure Coders Groups, see Configuring Coders Groups.</p>
Web: Allowed Coders Group ID [IpProfile_SBCAllowedCodersGroupID]	<p>Associates a Coders Group ID for defining the coders that can be used for this IP entity.</p> <p>Notes:</p> <ul style="list-style-type: none"> For a description of the Allowed Coders feature, see Restricting Coders. To configure Allowed Coders Groups, see Configuring Allowed Coders Groups.
Web: Allowed Coders Mode [IpProfile_SBCAllowedCodersMode]	<p>Determines the mode of the Allowed Coders feature for this IP Profile.</p> <ul style="list-style-type: none"> [0] Restriction = In the incoming SDP offer, the device uses only coders that are also listed in the Allowed Coders Group; the rest are removed from the SDP offer (i.e., only coders common between SDP offered coders and Allowed Coders Group are used). If an Extension Coders Group is also selected (using the IP Profile's SBCExtensionCodersGroupID parameter), these coders are added to the SDP offer. [1] Preference = The device re-arranges the priority (order) of the coders in the incoming SDP offer according to their order of appearance in the Allowed Coders Group list. This option also retains all the coders received in the SDP offer. [2] Restriction and Preference = Performs both Restriction and Preference. <p>Notes:</p> <ul style="list-style-type: none"> If the AllowedCodersGroup parameter is set to None, this parameter is not applicable. To select the Allowed Coders Group ID, use the AllowedCodersGroup parameter. To select the Extension Coders Group ID, use the CodersGroups parameter. For more information on the Allowed Coders feature, see Restricting Coders.
Web: SBC Preferences Mode [SBCPreferencesMode]	<p>Determines the order of the Extension coders (coders added if there are no common coders between SDP offered coders and Allowed coders) and Allowed coders (defined in the Allowed Coders Group table) in the outgoing SIP message (in the SDP).</p> <ul style="list-style-type: none"> [0] Doesn't Include Extensions = (Default) Extension coders are added at the end of the coder list. [1] Include Extensions = Extension coders and Allowed coders are arranged according to their order of appearance in the Allowed Coders Group table. <p>Note: If the SBCExtensionCodersGroupID parameter of the IP Profile table is set to None, this parameter is not applicable.</p>
Web: Diversion Mode	Determines the device's handling of the SIP Diversion header.

Parameter	Description
[IpProfile_SBCDiversioMode]	<p>For more information on interworking of the History-Info and Diversion headers, see Interworking SIP Diversion and History-Info Headers.</p> <ul style="list-style-type: none"> ▪ [0] Don't Care = (Default) Diversion header is not handled. ▪ [1] Add = History-Info header converted to a Diversion header. ▪ [2] Remove = Removes the Diversion header and the conversion to the History-Info header depends on the settings of the SBCHistoryInfoMode parameter.
Web: History Info Mode [IpProfile_SBCHistoryInfoMode]	<p>Determines the device's handling of the History-Info header. For more information on interworking of the History-Info and Diversion headers, see Interworking SIP Diversion and History-Info Headers.</p> <ul style="list-style-type: none"> ▪ [0] Don't Care = (Default) History-Info header is not handled. ▪ [1] Add = Diversion header converted to a History-Info header. ▪ [2] Remove = History-Info header removed from the SIP dialog and the conversion to the Diversion header depends on the settings of the SBCDiversioMode parameter.
Web: Media Security Behavior [IpProfile_SBCMediaSecurityBehaviour]	<p>Determines the transcoding method between SRTP and RTP and enforce an SBC leg to use SRTP or RTP.</p> <ul style="list-style-type: none"> ▪ [0] As is = (Default) No special handling for RTP\SRTP is done. ▪ [1] SRTP = SBC legs negotiate only SRTP media lines, and RTP media lines are removed from the incoming SDP offer\answer. ▪ [2] RTP = SBC legs negotiate only RTP media lines, and SRTP media lines are removed from the incoming offer\answer. ▪ [3] Both = Each offer\answer is extended (if not already) to two media lines - one RTP and the other SRTP. <p>If two SBC legs (after offer\answer negotiation) use different security types (i.e., one RTP and the other SRTP), the device performs RTP-SRTP transcoding. To transcode between RTP and SRTP, the following prerequisites must be met:</p> <ul style="list-style-type: none"> ▪ At least one supported SDP "crypto" attribute and parameters ▪ EnableMediaSecurity must be set to 1 <p>If one of the above transcoding prerequisites is not met, then:</p> <ul style="list-style-type: none"> ▪ any value other than "As is" is discarded. ▪ if the incoming offer is SRTP, force transcoding, coder transcoding, and DTMF extensions are not applied.
Web: RFC 2833 Behavior [IpProfile_SBCRFC2833Behavior]	<p>Determines the RFC 2833 SDP offer\answer negotiation.</p> <ul style="list-style-type: none"> ▪ [0] As is = (Default) The device does not intervene in the RFC 2833 negotiation. ▪ [1] Extend = Each outgoing offer\answer includes RFC 2833 in the offered SDP (the device adds RFC 2833 only if the incoming offer does not include RFC 2833). ▪ [2] Disallow = The device removes RFC 2833 from the incoming offer.
Web: Alternative DTMF Method	The device's first priority for DTMF method at each leg is RFC

Parameter	Description
[IpProfile_SBCAlternativeDTMF Method]	<p>2833. Therefore, if a specific leg negotiates RFC 2833 successfully, the chosen DTMF method for this leg is RFC 2833. For legs where RFC 2833 is not negotiated successfully, the device uses this parameter to determine the chosen DTMF method for the leg.</p> <ul style="list-style-type: none"> ▪ [0] Don't care = (Default) The device does not attempt to interwork any special DTMF method. ▪ [1] Transparent = In Band ▪ [2] INFO - Cisco ▪ [3] INFO - Nortel ▪ [4] INFO - Lucent = INFO, Korea
Web: P-Asserted-Identity [IpProfile_SBCAssertIdentity]	For a description, see the global parameter SBCAssertIdentity.
Web: SBC Fax Coders Group ID [IpProfile_SBCFaxCodersGroup ID]	Selects the supported fax coders (Coders Group ID) for fax negotiation. Coders Groups are configured in the Coders Group Settings table.
Web: SBC Fax Behavior [IpProfile_SBCFaxBehavior]	<p>Defines the negotiation method for fax offer.</p> <ul style="list-style-type: none"> ▪ [0] = (Default) Pass fax transparently, without interference. ▪ [1] = Handle fax according to fax settings in the IP Profile for all offer-answer transactions (including the initial INVITE). ▪ [2] = Handle fax according to fax settings in the IP Profile for all re-INVITE offer-answer transactions (except for initial INVITE).
Web: SBC Fax Offer Mode [IpProfile_SBCFaxOfferMode]	<p>Defines the coders included in the outgoing SDP offer (sent to the called "fax").</p> <ul style="list-style-type: none"> ▪ [0] All = (Default) Use only (and all) the coders of the selected Coders Group ID configured using the SBCFaxCodersGroupID parameter. ▪ [1] Single = Use only one coder. If a coder in the incoming offer (from the calling "fax") matches a coder in the SBCFaxCodersGroupID, then the device uses this coder. If no match exists, then the device uses the first coder listed in the Coders Group ID (SBCFaxCodersGroupID).
Web: SBC Fax Answer Mode [IpProfile_SBCFaxAnswerMode]	<p>Defines the coders included in the outgoing SDP answer (sent to the calling "fax").</p> <ul style="list-style-type: none"> ▪ [0] = Use matched coders between the incoming offer coders (from the calling "fax") and the coders of the selected Coders Group ID (configured using the SBCFaxCodersGroupID parameter). ▪ [1] = (Default) Use only one coder. If the incoming answer (from the called "fax") includes a coder that matches a coder match between the incoming offer coders (from the calling "fax") and the coders of the selected Coders Group ID (SBCFaxCodersGroupID), then the device uses this coder. If no match exists, the device uses the first listed coder of the matched coders between the incoming offer coders (from the calling "fax") and the coders of the selected Coders Group ID.
Web: SBC Session Expires Mode [IpProfile_SBCSessionExpiresM]	<p>Determines the required session expires mode of the IP entity.</p> <ul style="list-style-type: none"> ▪ [0] Transparent = (Default) The device does not interfere

Parameter	Description
ode]	<p>with the session expires negotiation.</p> <ul style="list-style-type: none"> ▪ [1] Observer = If the SIP Session-Expires header is present, the device does not interfere, but maintains an independent timer for each leg to monitor the session. If the session is not refreshed on time, the device disconnects the call. ▪ [2] Not Supported = The device does not allow a session timer with this IP entity. ▪ [3] Supported = The device enables the session timer with this IP entity. If the incoming SIP message does not include any session timers, the device adds the session timer information to the sent message. You can configure the value of the Session-Expires and Min-SE headers, using the SBCSessionExpires and SBCMinSE parameters, respectively.
Web: SBC Remote Early Media RTP [IpProfile_SBCRemoteEarlyMediaRTP]	<p>Defines whether the destination UA sends RTP immediately after it sends 18x response.</p> <ul style="list-style-type: none"> ▪ [0] Immediate = (Default) Remote client sends RTP immediately after it sends 18x response with early media. Device forwards 18x and RTP as is. ▪ [1] Delayed = After sending 18x response, the remote client waits before sending RTP (e.g., Microsoft Lync environment). For the device's handling of this remote UA support, see Interworking SIP Early Media.
Web: SBC Remote Can Play Ringback [IpProfile_SBCRemoteCanPlayRingback]	<p>Defines whether the destination UA can play a local ringback tone.</p> <ul style="list-style-type: none"> ▪ [0] No = UA does not support local ringback tone. The device sends 18x with delayed SDP to the UA. ▪ [1] Yes = (Default) UA supports local ringback tone. For the device's handling of this remote UA support, see Interworking SIP Early Media.
Web: SBC Remote Supports RFC 3960 [IpProfile_SBCRemoteSupportsRFC3960]	<p>Defines whether the destination UA is capable of receiving 18x messages with delayed RTP.</p> <ul style="list-style-type: none"> ▪ [0] Not Supported = (Default) UA does not support receipt of 18x messages with delayed RTP. For the device's handling of this remote UA support, see Interworking SIP Early Media. ▪ [1] Supported = UA is capable of receiving 18x messages with delayed RTP.
Web: SBC Multiple 18x Support [IpProfile_SBCRemoteMultiple18xSupport]	<p>Determines whether multiple 18x responses (including 180 Ringing, 181 Call is Being Forwarded, 182 Call Queued, and 183 Session Progress) are forwarded to the caller.</p> <ul style="list-style-type: none"> ▪ [0] Not Supported = Only the first 18x response is forwarded to the caller. ▪ [1] Supported = (Default) Multiple 18x responses are forwarded to the caller.
Web: SBC Early Media Response Type [IpProfile_SBCRemoteEarlyMediaResponseType]	<p>Determines the SIP provisional response type - 180 or 183 - for forwarding early media to the caller.</p> <ul style="list-style-type: none"> ▪ [0] Transparent = (Default) All early media response types are supported; the device forwards all responses as is (unchanged). ▪ [1] 180 = Early media is sent as 180 response only. ▪ [2] 183 = Early media is sent as 183 response only.

Parameter	Description
Web: SBC Remote Update Support [IpProfile_SBCRemoteUpdatesSupport]	Determines whether endpoints support the UPDATE method. <ul style="list-style-type: none"> ▪ [0] Not Supported = UPDATE method is not supported. ▪ [1] Supported Only After Connect = UPDATE method is supported only after the call is connected. ▪ [2] Supported = (Default) UPDATE method is supported during call setup and after call establishment.
Web: SBC Remote Re-Invite Support [IpProfile_SBCRemoteReinviteSupport]	Determines whether the destination UA of the re-INVITE request supports re-INVITE messages and if so, whether it supports re-INVITE with or without SDP. <ul style="list-style-type: none"> ▪ [0] Not Supported = re-INVITE is not supported and the device does not forward re-INVITE requests. The device sends a SIP response to the re-INVITE request, which can either be a success or a failure, depending on whether the device can bridge the media between the endpoints. ▪ [1] Supported with SDP = re-INVITE is supported, but only with SDP. If the incoming re-INVITE arrives without SDP, the device creates an SDP and adds it to the outgoing re-INVITE. ▪ [2] Supported = (Default) re-INVITE is supported with or without SDP.
Web: SBC Remote Refer Behavior [IpProfile_SBCRemoteReferBehavior]	For a description, see the global parameter SBCReferBehavior.
Web: SBC Remote Early Media Support [IpProfile_SBCRemoteEarlyMediaSupport]	Determines whether a remote side can accept early media or not. <ul style="list-style-type: none"> ▪ [0] Not Supported = Early media is not supported. ▪ [1] Supported = (Default) Early media is supported.
Web: SBC Remote 3xx Behavior [IpProfile_SBCRemote3xxBehavior]	For a description, see the global parameter SBC3xxBehavior.
Web: SBC Remote Delayed Offer Support [IpProfile_SBCRemoteDelayedOfferSupport]	Determines whether the remote endpoint supports delayed offer (i.e., initial INVITEs without an SDP offer). <ul style="list-style-type: none"> ▪ [0] Not Supported = Initial INVITE requests without SDP are not supported. ▪ [1] Supported = (Default) Initial INVITE requests without SDP are supported. <p>Note: For this parameter to function properly, a valid Extension Coders Group ID needs to be configured for IP Profiles that do not support delayed offer.</p>
Web: SBC PRACK Mode [IpProfile_SbcPrackMode]	Determines the PRACK mode required at the remote side: <ul style="list-style-type: none"> ▪ [1] Optional = PRACK is optional for these UAs. If required, the device performs the PRACK process on behalf of the destination UA. ▪ [2] Mandatory = PRACK is required for these UAs. Calls from UAs that do not support PRACK are rejected. Calls destined to these UAs are also required to support PRACK. ▪ [3] Transparent (default) = The device does not intervene with the PRACK process and forwards the request as is.

Parameter	Description
Web: SBC Enforce MKI Size [IpProfile_SBCEnforceMKISize]	<p>Enables MKI length negotiation for SRTP-to-SRTP flows between SIP networks (i.e., IP Groups). This feature includes the capability of modifying the MKI length on the inbound or outbound SBC call leg, using IP Profiles.</p> <ul style="list-style-type: none"> ▪ [0] Disable = (Default) Device forwards the MKI size as is. ▪ [1] Enable = Device changes the MKI length according to the settings of the IP Profile parameter, MKISize.
Web: SBC RFC2833 DTMF Payload Type Value [IpProfile_SBC2833DTMFPayloadType]	<p>Defines the RFC 2833 DTMF Payload Type for a specific SBC leg. This enables the interworking of the DTMF payload type for RFC 2833 between different SBC call legs. For example, if two entities require different DTMF payload types, the SDP offer received by the device from one entity is forwarded to the destination entity with its payload type replaced with the configured payload type, and vice versa.</p> <p>The value range is 96 to 127. The default is 0 (i.e., the device forwards the received payload type as is).</p>
Web: SBC User Registration Time [IpProfile_SBCUserRegistrationTime]	<p>For a description, see the global parameter SBCUserRegistrationTime.</p>
Web: SBC Remote Hold Format [IPProfile_SBCRemoteHoldFormat]	<p>Defines the format of the SDP in the re-INVITE for call hold that the device sends to the held party.</p> <ul style="list-style-type: none"> ▪ [0] transparent = Device forwards SDP as is. ▪ [1] send-only = Device sends SDP with 'a=sendonly'. ▪ [2] send only 0.0.0.0 = Device sends SDP with 'a=sendonly' and 'c=0.0.0.0'. ▪ [3] inactive = Device sends SDP with 'a=inactive'. ▪ [4] inactive 0.0.0.0 = Device sends SDP with 'a=inactive' and 'c=0.0.0.0'. ▪ [5] not supported = Used when remote side cannot identify a call-hold message. The device terminates the received call-hold message (re-INVITE / UPDATE) and sends a 200 OK to the initiator of the call hold. The device plays a held tone to the held party if the 'SBC Play Held Tone' parameter is set to Yes.
Web: SBC Play Held Tone [IpProfile_SBCPlayHeldTone]	<p>Enables the device to play a held tone to the held party. This is useful if the held party does not support playing a local held tone, or for IP entities initiating call hold that do not support the generation of held tones.</p> <ul style="list-style-type: none"> ▪ [0] No (default) ▪ [1] Yes <p>Note: If this parameter is set to Yes, the device plays the tone only if the 'SBC Remote Hold Format' parameter is set to transparent, send-only, send only 0.0.0.0, or not supported.</p>
Web: SBC Reliable Held Tone Source [IPProfile_ReliableHoldToneSource]	<p>Enables the device to consider the received call-hold request (re-INVITE/UPDATE) with SDP containing 'a=sendonly', as genuine.</p> <ul style="list-style-type: none"> ▪ [0] No (default) = Even if the received SDP contains 'a=sendonly', the device plays a held tone to the held party. This is useful in cases where the initiator of the call hold does not support the generation of held tones. ▪ [1] Yes = If the received SDP contains 'a=sendonly', the

Parameter	Description
	<p>device does not play a held tone to the held party (and assumes that the initiator of the call hold plays the held tone).</p> <p>Note: The device plays a held tone only if the 'SBC Play Held Tone' parameter is set to Yes.</p>

Reader's Notes

Part V

Session Border Controller Application

19 SBC Overview

This section provides a detailed description of the device's SBC application.

**Notes:**

- For guidelines on how to design your E-SBC deployment, refer to the *SBC Design Guide* document.
- For SBC functionality, the Software License Key installed on the device must include the SBC feature.

The SBC application supports the following main features:

- NAT traversal: The device supports NAT traversal, allowing, for example, communication with ITSPs with globally unique IP addresses, for LAN-to-WAN VoIP signaling (and bearer), using two independent legs. This also enables communication for "far-end" users located behind a NAT on the WAN. The device supports this by:
 - Continually registering far-end users in its dynamic database.
 - Maintaining remote NAT binding state by frequent registrations, thereby, off-loading far-end registrations from the LAN IP PBX.
 - Using Symmetric RTP (RFC 4961) to overcome bearer NAT traversal.
- VoIP firewall and security for signaling and media:
 - SIP signaling:
 - ◆ Deep and stateful inspection of all SIP signaling packets.
 - ◆ SIP dialog initiations may be rejected based on values of incoming SIP INVITE message and other Layer-3 characteristics.
 - ◆ Packets not belonging to an authorized SIP dialog are discarded.
 - RTP:
 - ◆ Opening pinholes (ports) in the device's firewall based on Offer-Answer SDP negotiations.
 - ◆ Deep packet inspection of all RTP packets.
 - ◆ Late rouge detection - if a SIP session was gracefully terminated and someone tries to "ride on it" with rouge traffic from the already terminated RTP and SIP context, the VoIP Firewall prevents this from occurring.
 - ◆ Disconnects call (after user-defined time) if RTP connection is broken.
 - ◆ Black/White lists for both Layer-3 firewall and SIP classification.
- Topology hiding: The device intrinsically supports topology hiding, limiting the amount of topology information displayed to external parties. For example, IP addresses of ITSPs' equipment (e.g. proxies, gateways, and application servers) can be hidden from outside parties. The device's topology hiding is provided by implementing back-to-back user agent (B2BUA) leg routing:
 - Strips all incoming SIP Via header fields and creates a new Via value for the outgoing message.
 - Each leg has its own Route/Record Route set.
 - Modifies SIP To, From, and Request-URI host names (must be configured using the Message Manipulations table).
 - Generates a new SIP Call-ID header value (different between legs).
 - Changes the SIP Contact header to the device's own address.
 - Layer-3 topology hiding by modifying source IP address in the SIP IP header.
- SIP normalization: The device supports SIP normalization, whereby the SBC

application can overcome interoperability problems between SIP user agents. This is achieved by the following:

- Manipulation of SIP URI user and host parts.
- Connection to ITSP SIP trunks on behalf of an IP-PBX - the device can register and utilize user and password to authenticate for the IP-PBX.
- Survivability:
 - Routing calls to alternative routes such as the PSTN.
 - Routing calls between user agents in the local network using a dynamic database (built according to registrations of SIP user agents).
- Routing:
 - IP-to-IP routing translations of SIP, UDP, TCP, TLS (when extensive transcoding is not required).
 - Load balancing and redundancy of SIP servers.
 - Routing according to Request-URI\Specific IP address\Proxy\FQDN.
 - Alternative routing.
 - Routing between different Layer-3 networks (e.g., LAN and WAN).
- Load balancing\redundancy of SIP servers.
- ITSP accounts.
- SIP URI user and host name manipulations.
- Coder transcoding.

19.1 SIP Network Definitions

The device's SBC application can implement multiple SIP signaling and RTP (media) interfaces.

19.2 SIP Dialog Initiation Process

The device's SIP dialog initiation process concerns all incoming SIP dialog initiation requests. This includes SIP methods such as INVITE, SUBSCRIBE, OPTIONS, REFER, INFO, UNSOLICITED NOTIFY, MESSAGE, and REGISTER.

The SIP dialog initiation process consists of the following stages:

1. **Determining source and destination URL:** The SIP protocol has more than one URL in a dialog-establishing request that may represent the source and destination URLs. When handling an incoming request, the device uses specific SIP headers for obtaining the source and destination URLs. Once these URLs are determined, their user and host parts are used as input for the classification process, message manipulation, and call routing.
 - **All SIP requests (e.g., INVITE) except REGISTER dialogs:**
 - ◆ Source URL: The source URL is obtained from the SIP header according to the following logic:
 - ✓ The source URL is obtained from the From header.
 - ✓ If the From header contains the value 'Anonymous', the source URL is obtained from the P-Preferred-Identity header.
 - ✓ If the P-Preferred-Identity header does not exist, the source URL is obtained from the P-Asserted-Identity header.
 - ◆ Destination URL: The destination URL is obtained from the Request-URI.
 - **REGISTER dialogs:**
 - ◆ Source URL: The source URL is obtained from the To header.

- ◆ Destination URL: The destination URL is obtained from the Request-URI.

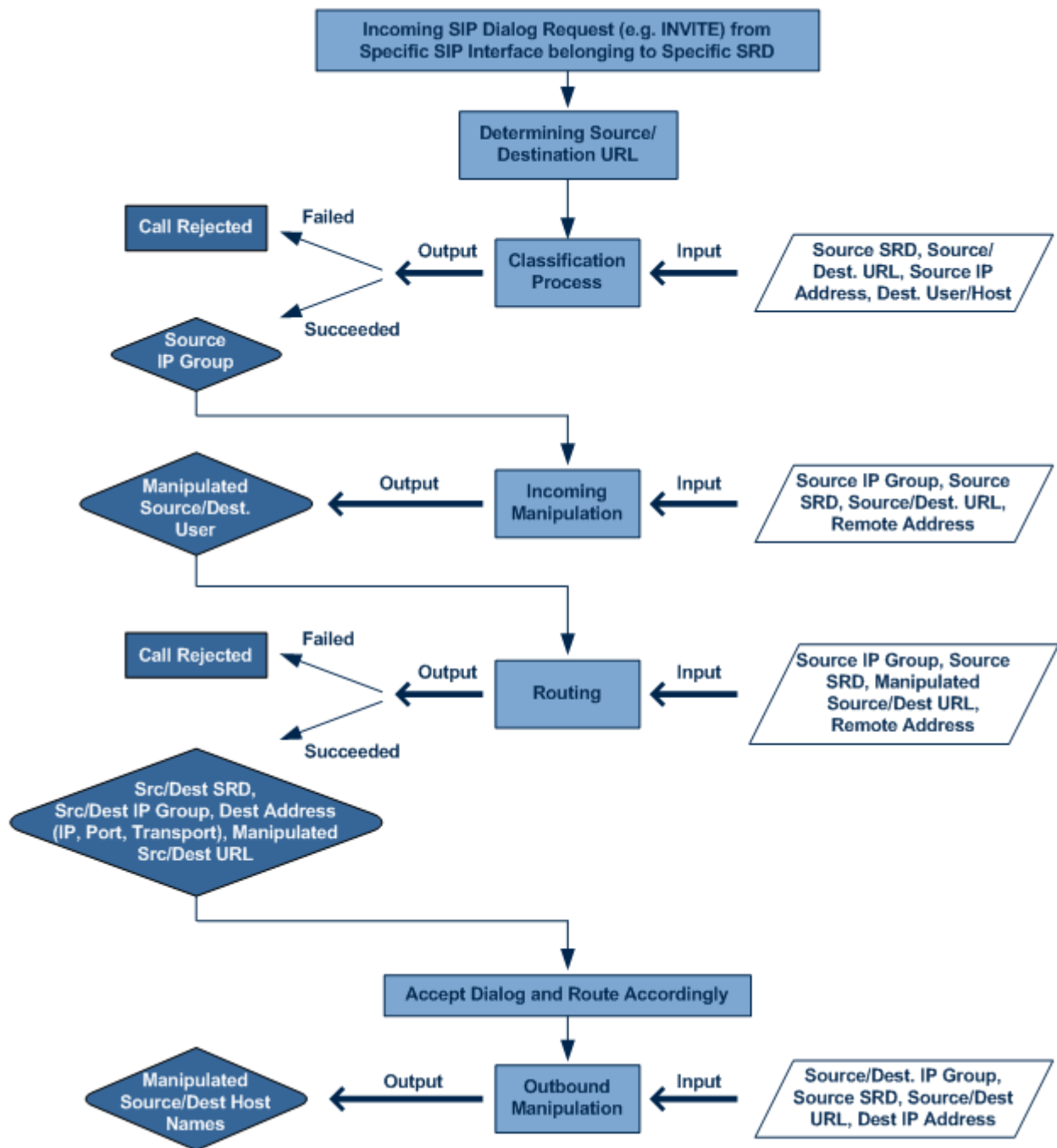


Note: You can determine the SIP header from where the device obtains the source URL in the incoming SIP request. This is done in the IP Group table using the 'Source URI Input' parameter.

- 2. Classifying incoming SIP dialog-initiating requests to a source IP Group:** The classification identifies the incoming SIP dialog request as belonging to a specific IP Group (from where the SIP dialog request originated). For more information, see 'Configuring Classification Rules' on page 272.
- 3. SBC IP-to-IP routing:** The device routes the call to a destination that can be configured to one of the following:
 -
 - IP Group - the destination is the address configured for the Proxy Set associated with the IP Group (allows redundancy/load balancing).
 - Specified destination address (can be based on IP address, host name, port, transport type, and/or SRD). Routing to a host name can be resolved using NAPTR/SRV/A-Record.
 - Request-URI of incoming SIP dialog initiating requests.
 - ENUM query.
 - Hunt Group - used for call survivability.
 - IP address (in dotted-decimal notation or FQDN - NAPTR/SRV/A-Record resolutions) according to a specified Dial Plan index listed in the loaded Dial Plan file.
 - LDAP server or LDAP query result.For more information, see 'Configuring SBC IP-to-IP Routing' on page 279.
- 4. Manipulating SIP URI user part (source and destination) of inbound and/or outbound SIP dialog requests:** You can configure rules for manipulating the SIP URI user part (source and destination) on the inbound and/or outbound leg. For more information, see 'SBC Manipulations' on page 287.
- 5. SIP message manipulations:** You can configure SIP message manipulation rules that can add, remove, and/or modify SIP headers and parameters. For more information, see 'Configuring SIP Message Manipulation' on page 208.

The flowchart below illustrates the SBC process:

Figure 19-1: Routing Process



19.3 User Registration and Internal Database

To allow registrations to traverse the SBC, the device must be configured with at least one User-type IP Group. These IP Groups represent a group of user agents that share the following characteristics:

- Perform registrations and share the same serving proxy/registrar
- Possess identical SIP and media behavior
- Reside on the same Layer-3 network and are associated with the same SRD

Typically, the device is configured as the user agent's outbound proxy and the device is configured (using the IP-to-IP Routing table) to route requests received from this IP Group to the serving proxy and vice versa. Survivability can be achieved using the alternative routing feature.

19.3.1 Initial Registration Request Processing

Registration requests have different processing policies than other SIP methods:

1. Determining source and destination URL's:
 - The source URL is obtained from the To header
 - The destination URL is obtained from the Request URI
2. Classification: The REGISTER classification process is the same as the general classification process (described in previous sections). The source IP Group must be of type User. If classification fails or the source IP Group is not of type User, the registration is rejected.
3. Routing: The REGISTER routing is performed using the IP-to-IP Routing table:
 - The destination type can be an IP Group, specific IP address, Request-URI, or ENUM query (can also use DNS queries).
 - If the destination is a User-type IP Group, then the registration is not be forwarded. Instead, the device accepts (replies with 200 OK response) or rejects (Reply with 4xx) the request according to the user group policy.
4. Internal registration database: If the source IP Group is of type User and registration succeeds (replied with 200 OK by the IP-PBX), then the device adds a record to its database that identified the specific contact of this specific user (AOR). This record is used later to route requests to this specific user (either in normal or in survivability modes).
5. Alternative Routing: Alternative routing can be configured in the IP-to-IP Routing table for REGISTER requests.
6. Inbound Manipulation: The SBC record in the device's database includes the Contact header. Every REGISTER request is added to the database before manipulation, allowing correct user identification in the SBC Classification process for the next received request.
7. Session Admission Control: Applies various limitations on incoming and outgoing REGISTER requests. For example, limiting REGISTER requests from a certain IP Group/SRD. Note that this limitation is only for concurrent register dialogs and not concurrent registrations in the internal database.
8. The device can retain the original value of the SIP Expires header received from the user or proxy, in the outgoing REGISTER message. This feature also applies when the device is in "survivability" state (i.e., REGISTER requests cannot be forwarded to the proxy and is terminated by the device). This is configured by the SBCUserRegistrationTime, SBCProxyRegistrationTime, and SBCSurvivabilityRegistrationTime parameters.
9. By default, the Contact of the outgoing REGISTER is populated with a unique Contact generated by the device and associated with this specific registration. Alternatively, the original user can be retained in the Contact and used in the outgoing REGISTER request (using the SBCKeepContactUserinRegister parameter).

19.3.2 Internal Database

The device manages a dynamic database that is updated according to registration requests that traverse the SBC. Each database entry represents a binding between an AOR and one or more contact. Database bindings are added upon successful registration

responses. For specific registrations, the AOR is obtained from the SIP To header and the contact is taken from the SIP Contact header.

Database bindings are removed in the following cases:

- Successful de-registration responses (REGISTER with Expires header that equals zero)
- Registration failure responses
- Timeout of the Expires header value (in scenarios where the user agent did not send a refresh registration request)

The database has the following limitations:

- Maximum of five contacts per AOR
- The same contact cannot belong to more than one AOR
- Contacts with identical URIs and different ports and transport types are not supported (same key is created)
- Multiple contacts in a single REGISTER is not supported
- One database is shared between all User-type IP Groups

19.3.3 Routing using Internal Database

Typically, routing using the database is applicable to all method types other than registrations. To route to a registered user (using the internal dynamic database), the following steps must be taken:

1. An IP-to-IP Routing rule with the desired input parameters (matching characteristics) and the destination type as IP Group (operation rule).
2. The destination IP Group must be of type User.
3. To find a match for these specific rules, the device attempts to locate a match between the incoming Request-URI and (according to the description order):
 - a. Unique contact - the Contact generated by the SBC and sent in the initial registration request to the serving proxy
 - b. Registered AOR - the AOR of the incoming REGISTER request
 - c. Registered contact - the Contact of the incoming REGISTER request

If registrations are destined to the database (using the above rules), the device does not attempt to find a database match, but instead replies with 200 OK (used for Survivability). Once a match is found, the request is routed either to the contact received in the initial registration or (if the device identifies that the user agent is behind a NAT) to the source IP address of the initial registration.

19.3.4 Registration Refreshes

Registration refreshes are incoming REGISTER requests that are associated with a specific registered user. The association is performed by searching the internal registration database. These refreshes are routed to the serving proxy only if the serving proxy Expires time is about to expire; otherwise, the device responds with a 200 OK without routing the REGISTER. Each such refreshes also refresh the internal timer time set on the device for this specific registration.

19.3.5 Notification of Expired User Registration to SIP Proxy / Registrar

The device automatically notifies SIP Proxy / Registrar servers of users registered in the device's database whose registration timeout has expired. When a user's registration timer expires, the device removes the user record from its Registration database and sends an unregister notification (REGISTER message with the Expires header set to 0) to the Proxy/Registrar.

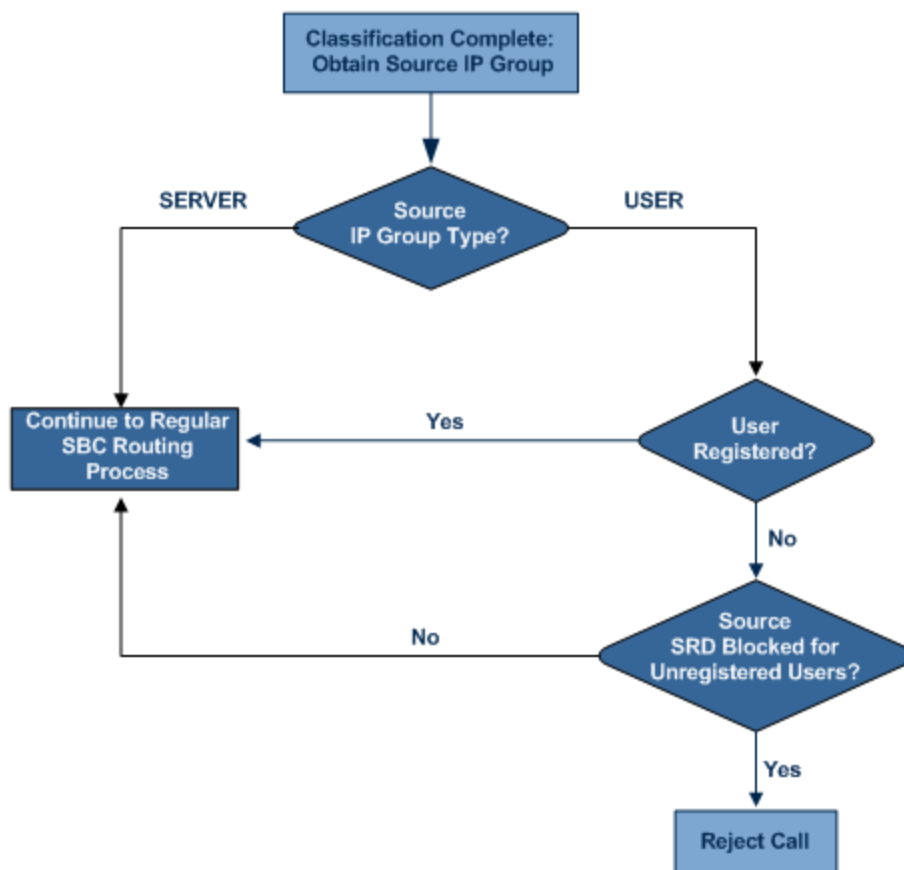
This feature is enabled only if a REGISTER message is sent to an IP Group destination type, configured in the IP-to-IP Routing table.

19.3.6 Registration Restriction Control

The device provides flexibility in controlling user registration:

- **Limiting Number of Registrations per Source SRD and/or IP Group:** You can limit the number of users that can register with the device. This limitation can be applied per source IP Group and/or SRD. By default, no limitation exists for registered users. This is configured using the parameters SRD or IPGroup.
- **Blocking Incoming Calls from Unregistered Users:** You can block incoming calls (INVITE requests) from unregistered users (pertaining to User-type IP Groups). By default, calls from unregistered users are not blocked. This is configured using the parameter SRD. The flowchart below depicts the process for blocking unregistered users. When the call is rejected, the device sends a SIP 500 "Server Internal Error" response to the remote end.

Figure 19-2: Blocking Incoming Calls from Unregistered Users



19.4 SBC Media Handling

Media behavior includes anything related to the establishment, management and termination of media sessions within the SIP protocol. Media sessions are created using the SIP "offer"/"answer" mechanism. If successful, the result is a bi-directional media (RTP) flow (e.g. audio, fax, modem, DTMF). Each offer/answer may create more than one media session of different types (e.g. audio and fax). In a SIP dialog, multiple offer/answer transactions may occur, each may change the media sessions characteristics (e.g. IP

address, port, coders, media types, and RTP mode). The media capabilities exchanged in an offer/answer transaction include the following:

- Media types (Audio, Secure Audio, Video, Fax, Text...)
- IP addresses and ports of the media flow
- Media flow mode (send receive, receive only, send only, inactive)
- Media coders (coders and their characteristics used in each media flow)
- Other (standard or proprietary) media and session characteristics

Even though the device usually does not change the negotiated media capabilities (mainly performed by the remote user agents), it does examine the media exchange to control negotiated media types (if necessary) and to know how to open the RTP media channels (IP addresses, coder type, payload type etc.). The device forwards multiple video streams and text, as is.

The device interworks (normalization) the media (RTP-to-RTP, SRTP-to-RTP, and SRTP-to-SRTP) between its SBC legs. It "re-builds" specific fields in the RTP header when forwarding media packets. The main fields include the sequence number, SSRC, and timestamp.

The device is aware and sometimes active in the offer/answer process due to the following:

- NAT traversal: the device changes the SDP address to be its own address, thereby, resolving NAT problems.
- Firewall and security:
 - RTP pin holes - only RTP packets related to a successful offer/answer negotiation traverse the device: When the device initializes, there are no RTP pin holes opened, this means that each RTP/RTCP packets destined to the device are discarded. Once an offer/answer transaction ends successfully, an RTP pin hole is opened and RTP/RTCP flows between the two remote user agents. Once a pin hole is opened, the payload type and RTP header version is validated for each packet. RTP pin holes close if one of the associated SIP dialogs is closed (may also be due to broken connection).
 - Late rogue detection - once a dialog is disconnected, the related pin holes also disconnect.
 - Deep Packet inspection of the RTP that flows through the opened pin holes.
- Adding of media functionality to SIP user agents:
 - Transcoding (for a description on the transcoding modes, see 'Transcoding Modes' on page 247)
 - Broken connection

According to the above functionalities, the call can be configured to operate in one of the following modes:

- **Media Anchoring without Transcoding (Transparent):** RTP traverses the device with minimal RTP packet changes (no DSP resources needed). This is typically used to solve NAT, firewall, and security issues. In this mode, all the "audio" coders in the received offer are included in the SBC outgoing offer. The Coder Table configuration has no effect on the coders in the outgoing offer. For more information, see 'Media Anchoring without Transcoding (Transparent)' on page 243.
- **Media Anchoring with Transcoding:** RTP traverses the device and each leg uses a different coder or coder parameters (DSP resources are required). For more information, see 'Media Anchoring with Transcoding' on page 243.
- **No Media Anchoring:** The RTP packet flow does not traverse the device. Instead, the two SIP UA's establish a direct RTP/SRTP flow between one another (see 'No Media Anchoring' on page 245).

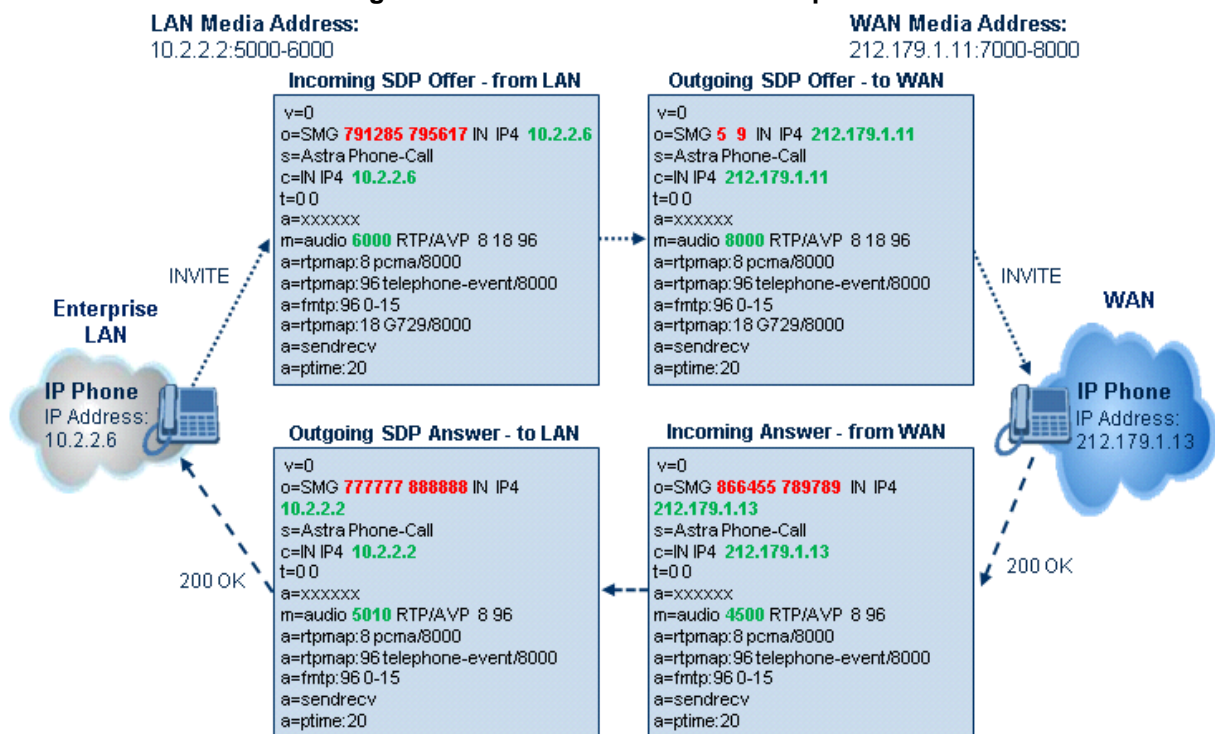
19.4.1 Media Anchoring without Transcoding (Transparent)

To direct the RTP to flow through the device (for NAT traversal, firewall and security), all IP address fields in the SDP are modified:

- Origin: IP address, session and version id
- Session connection attribute ('c=' field)
- Media connection attribute ('c=' field)
- Media port number
- RTCP media attribute IP address and port

Each SBC leg allocates and uses the device's local ports (e.g., for RTP/RTCP/fax). The local ports are allocated from a Media Realm associated with each leg. The legs are associated with a Media Realm as follows: If the leg's IP Group is configured with a Media Realm, then this is the associated Media Realm; otherwise, the leg's SRD Media Realm is the associated one. The figure below illustrates an example of SDP handling for a call between a LAN IP Phone 10.2.2.6 and a remote IP Phone 212.179.1.13 on the WAN.

Figure 19-3: SDP Offer/Answer Example



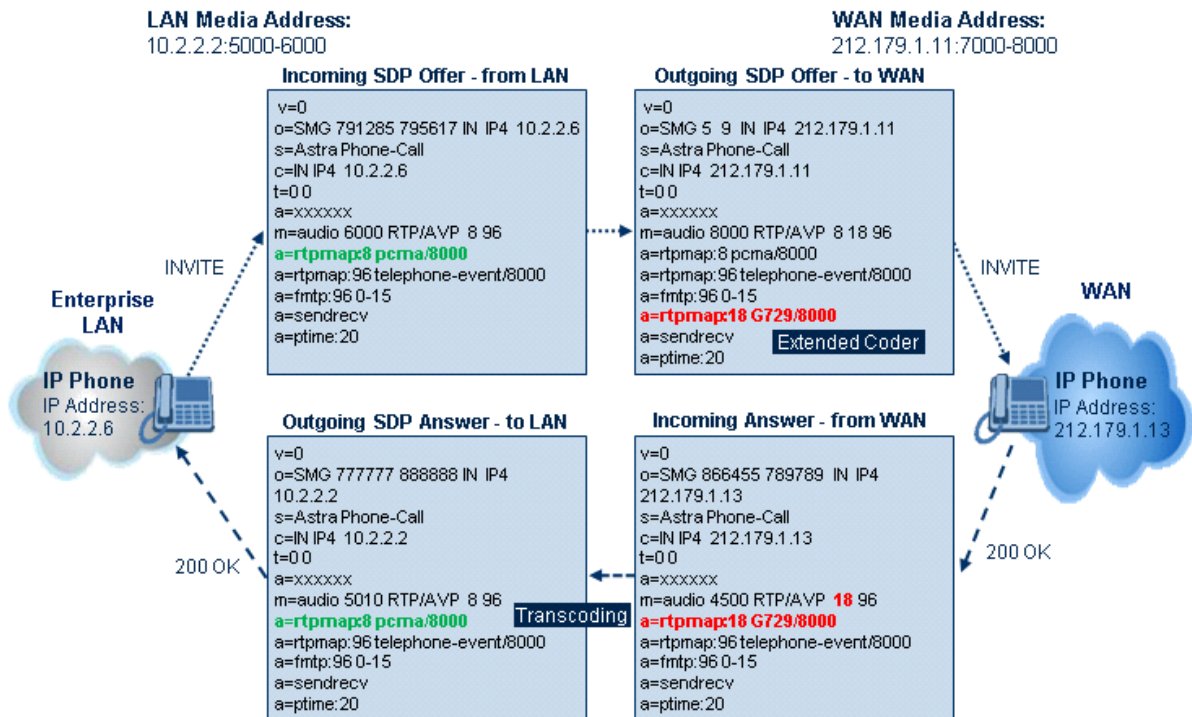
19.4.2 Media Anchoring with Transcoding

The device performs transcoding when there are no common coders between the two user agents (i.e., the SDP answer from one user agent doesn't include any coder included in the offer previously sent by the other user agent). For transcoding, the device can be configured to add media capabilities to user agents pertaining to a specific IP Group, and then perform transcoding in cases where the selected coder in the answer SDP is not one that appears in the original offer. The capabilities that can be added are one or more of the device's supported coders and are configured by using the parameter SBCExtensionCodersGroupID (points to a coders list) in the IP Profile table (which is assigned to the IP Group). Therefore, to allow user agents of different IP Groups to communicate with each other (regardless of their capabilities), an extended coders table with at least one coder that is supported by each IP Groups' user agents needs to be

assigned to each IP Group. Therefore, each offer destined to specific IP Groups include this coder.

In the scenario depicted in the figure below, the IP phone on the LAN side initiates a call to the IP phone on the WAN. The initial SDP offer (from the LAN leg) includes codec G.711 as its supported codec. Since this is sent to a Destination IP Group that is configured with an extended coder list, on the WAN leg the device adds another supported codec G.729 to the SDP, which is now offered to the WAN IP phone. The WAN IP phone chooses the extended codec (G.729) in its SDP answer to the device's WAN leg. Since this codec was not included in the original incoming offer, the device performs transcoding (between G.729 and G.711) between its two legs, allowing the streaming of media to occur.

Figure 19-4: Transcoding using Extended Coders (Example)



For an SDP offer to provide an extended coder list to a remote user agent, the following prerequisites must be fulfilled:

- An extended coders list has been configured for the user agent's IP Group (i.e., Destination IP Group)
- The incoming offer contains at least one supported coder (otherwise, transcoding can't be performed)
- Both legs have available DSP's
- T.38 doesn't appear in the offer

If the above prerequisites are not met, the SDP offer is sent without the extended coders list. The coders from the extended list are added after the ones from the original offer (decreases transcoding probability). Coders common between the extended coders list and those in the original SDP offer are not added. Transcoding may be performed even in scenarios when the same coder has been chosen - this occurs if the coders use different coder parameters (e.g. rate and packetization time).

The device also supports early media, whereby the first offer\answer transaction is finalized and the media flow starts before the SIP call is connected (before the INVITE 200 OK response). The offer and answer options can be included in the following SIP messages:

- Offer in first INVITE, answer on 180, and no or same answer in the 200 OK
- Offer in first INVITE, answer on 180, and a different answer in the 200 OK (not standard)

- INVITE without SDP, offer in 180, and answer in PRACK
- PRACK and UPDATE transactions can also be used for initiating subsequent offer\answer transactions before the INVITE 200 OK response.
- In a SIP dialog life time, media characteristics after originally determined by the first offer\answer transaction can be changed by using subsequent offer\answer transactions. These transactions may be carried either in UPDATE or ReINVITE SIP transactions. The media handling is similar to the original offer\answer handling. If the offer is rejected by the remote party, then no media changes occur (e.g. INVITE without SDP, then 200 OK and ACK, offer\answer within an offer\answer, and Hold ReINVITE with IP address of 0.0.0.0 - IP address is unchanged).

19.4.3 No Media Anchoring

The No Media Anchoring or Anti-Tromboning feature enables the use of SBC signaling capabilities without handling the RTP/SRTP (media) flow between remote SIP user agents (UA). The RTP packet flow does not traverse the device and instead, the two SIP UAs establish a direct RTP/SRTP flow (i.e., direct call) between one another. Signaling continues to traverse the device with minimal intermediation and involvement to enable certain SBC abilities such as routing.

By default, media packets traverse the device. This is done in order to:

- Solve NAT problems
- Enforce media security policy
- Perform media transcoding between the two legs
- Media monitoring

However, since media packets traverse the SBC, media quality may degrade, for example, due to packet delay.

In some setups, specific calls do not require media anchoring, for example, when there is no need for NAT, security, or transcoding. This is typical for calls between users in the LAN:

- Internal LAN calls: When the SBC routes a call between two UAs within the same LAN, the SBC can forward the SDP directly between caller and callee, and direct the RTP to flow between the UAs without traversing the SBC.
- Internal LAN calls via WAN: In this setup, the SBC dynamically identifies that the call is between UAs located in the same network (i.e., LAN) and thereby, directs the RTP to flow between these UAs without traversing the SBC

In contrast to the regular SBC implementation, the No Media Anchoring feature:

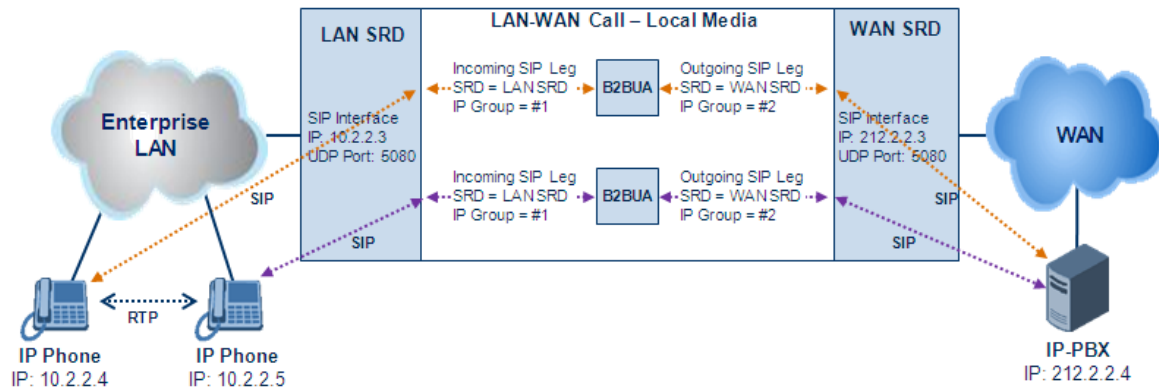
- Does not perform any manipulation on SDP data (offer\answer transaction) such as ports, IP address, coders
- Opening voice channels and allocation of IP media ports are not required

The No Media Anchoring feature is typically implemented in the following scenarios:

- SBC device is located within the LAN.

- Calls between two SIP UA's in the same LAN and signals are sent to a SIP proxy server (or hosted IP PBX) located in the WAN.

Figure 19-5: SBC SIP Signaling without RTP Media Flow



The benefits of implementing the No Media Anchoring include the following:

- Saves network bandwidth
- Reduces CPU usage (no RTP/SRTP handling)
- Avoids interference in SDP negotiation and header manipulation on RTP/SRTP

The No Media Anchoring process is as follows:

1. Identifies a No Media Anchoring call - according to configuration and the call's properties (such as source, destination, IP Group, and SRD).
2. Handles the identified No Media Anchoring call.

The No Media Anchoring feature is enabled using the `SBCDirectMedia` parameter. You can also enable No Media Anchoring per SRD (using the `IntraSRDMediaAnchoring` parameter), whereby calls between two UA's that pertain to the same SRD (source and destination) are handled as No Media Anchoring (direct media) calls.

**Notes:**

- No Media Anchoring can be used when the SBC does not do NAT traversal (for media) where all the users are in the same domain.
- No Media Anchoring calls cannot operate simultaneously with the following SBC features:
 - Force transcoding
 - Extension Coders
 - Extension of RFC 2833/Out-of-band DTMF/In-band DTMF
 - Extension of SRTP/RTPAll restriction features (Allowed Coders, restrict SRTP/SRT, restrict RFC 2833) can operate simultaneously. Once No Media Anchoring is enabled, the features listed above are disabled.
- The Coder Restriction feature operates simultaneously with No Media Anchoring calls. Restricted coders are removed from the SDP offer message.
- When two UA's pertain to the same SRD, the parameter IntraSRDMediaAnchoring is set to 1, and one of the UA's is defined as a foreign user (example, "follow me service") located in the WAN, while the other UA is located in the LAN: calls between these two UA's can't be established until IntraSRDMediaAnchoring is set to 0, as the device doesn't interfere in the SIP signaling. In other words, parameters such as IP addresses are not manipulated for calls between LAN and WAN (although required).
- When the parameter SBCDirectMedia is disabled, No Media Anchoring calls between two UA's belonging to separate SRD's cannot be configured. No Media Anchoring calls between two UA's belonging to the same SRD is configurable only (in this case).

19.4.4 Transcoding Modes

The device supports the configuration of the voice transcoding mode (media negotiation) between the two SBC legs. The device can be configured to perform transcoding only when necessary. Typically, the SBC passes RTP packets transparently (RTP-to-RTP) between the two user agents. If the device is configured to always perform transcoding, then transcoding is performed on the outgoing SBC leg and the device's SBC application interworks the media by implementing PSTN transcoding (since both legs have different media capabilities).

In the SBC application, forced transcoding of voice in an SBC session allows the device to receive capabilities that are not negotiated between the SBC legs. For example, if on the SBC session you want to force Gain Control to use voice transcoding even though both sides of the session have negotiated without SBC intervention (for example, coder extension).

The transcoding mode can be configured using the parameters TranscodingMode and IPProfile.



Note: To implement transcoding, you must configure the number of required DSP channels for transcoding (for example, MediaChannels = 120). Each transcoding session uses two DSP resources.

19.4.5 Restricting Coders

The SBC Allowed Coders (coders restriction) feature determines the coders that can be used for a specific SBC leg. This provides greater control over bandwidth by enforcing the use of specific coders (*allowed coders groups*) while preventing the use of other coders. This is done by defining a group of allowed coders for the SBC leg, as described below:

1. Configure a Coders Group for allowed coders, using the AllowedCodersGroup parameter.
2. Select this Coders Group using the SBCAllowedCodersGroupID parameter of the IP Profile table.
3. Enable this feature by setting the SBCAllowedCodersMode parameter of the IP Profile table to **Restriction**.

Coders that are not listed (including unknown coders) in the Allowed Coders Group are removed from the SDP offer. Therefore, only coders common between the SDP offer and Allowed Coders Group are used. If the SDP offer does not list any of the Allowed Coders, the call is rejected, unless transcoding is configured (using Extension coders – see 'Coder Transcoding' on page 248).



Notes:

- For a list of supported coders, see 'Configuring Coders' on page 217.
- Allowed Coders Groups are applicable only to audio media.

The Allowed Coders process is as follows:

- a. The device receives an incoming SIP message with SDP (offer) and checks the offered coders.
- b. The source (first) leg may have Allowed Coders (i.e. list of coders that can be used - enforced).
- c. The device checks for common coders between the SDP offered coders and the Allowed Coders Group list.

For example, assume the following:

- The SDP coder offer includes the following coders: G.729, G.711, and G.723.
- The source (first) leg includes the following Allowed Coders: G.711 and G.729.

The device selects the common coders, i.e., G.711 and G.729 (with changed preferred coder priority - highest for G.711). In other words, it removes the coders that are not in the Allowed Coders list and the order of priority is first according to the Allowed Coders list.

19.4.6 Coder Transcoding

The device can add coders, referred to as *Extension coders* to the SDP offer in the outgoing leg.

For example, assume the following:

- The SDP coder offer includes the following coders: G.729, G.711, and G.723.
 - The incoming leg includes the following Allowed coders: G.711 and G.729.
1. The device selects the common coders, i.e., G.711 and G.729 (with changed preferred coder priority - highest for G.711). In other words, it removes the coders that are not in the Allowed coders list and the order of priority is first according to the Allowed coders list.
 2. Assuming that the outgoing leg also includes Allowed coders and/or Extension coders: The device performs the Allowed coders procedure (common coders) between the updated coder list and the outgoing leg's Allowed coders. The Extension coders procedure is performed before Allowed coders.

3. Adding to the example, assume the following:

- For the outgoing leg, the device selects the common coders G.711 and G.729 (explained in the example above).
- Outgoing leg includes the Extended coder G.726.
- Outgoing leg includes the following Allowed coders: G.723, G.726, and G.729.

As a result, the device selects the common coders, i.e., G.729 and G.726 (coder priority did not change Extension coder order).

If the Allowed coders policy on the SDP incoming leg returns an empty coders list, the device rejects the call (SIP 488 or ACK and BYE). If both Extension coders and Allowed coders policies on SDP (in this order) returns an empty coders list, the outgoing leg rejects the call (SIP 488, or ACK and BYE).

Below is an example, assuming that Allowed Coders list (ordered) includes G711A-law (PCMA), G729, and G711U-law (PCMU), and Extension Coder is G729.

1. SDP offer - original offer:

```
m=audio 6050 RTP/AVP 0 8 4 96
a=rtpmap:0 PCMU/8000
a=rtpmap:8 PCMA/8000
a=rtpmap:4 G723/8000
a=fmtp:4 annexa=no
a=rtpmap:96 telephone-event/8000
a=fmtp:96 0-15
a=ptime:20
a=sendrecv
```

2. SDP offer - after manipulation:

```
m=audio 6010 RTP/AVP 8 0 96 18
a=rtpmap:0 PCMU/8000
a=rtpmap:8 PCMA/8000
a=rtpmap:96 telephone-event/8000
a=fmtp:96 0-15
a=ptime:20
a=sendrecv
a=rtpmap:18 G729/8000
a=fmtp:18 annexb=no
```

In the SDP, the "m=audio 6010 RTP/AVP 8 0 96 18" line shows that the coder priority has changed - G.711A-law ("8") and then G.711U-law ("0") - and that the Extension coder G.729 ("18") has been added. The G.723 coder ("4") in the original offer was removed as it was not defined in the Allowed Coders list (i.e., a restricted coder).

➤ **To configure Extension coders:**

1. In the Coders Group table (see 'Configuring Coder Groups' on page 220), configure a Coders Group for extension coders.
2. In the IP Profile table, select this Coders Group in the 'Extension Coders Group ID' parameter.
3. In the IP Profile table, enable this feature by setting the 'Allowed Coders Mode' parameter to **Restriction** or **Restriction and Preference**.

19.4.7 Prioritizing Coder List in SDP Offer

In addition to restricting the use of coders with Allowed coders, the device can prioritize the coders listed in the SDP offer. This feature is referred to as *Coder Preference*. This is done on both SBC legs:

- **Incoming SDP offer:** The device arranges the coder list according to the order in the Allowed Coders Group table. The coders listed higher up in the table take preference over ones listed lower down in the table. This feature is enabled by setting the 'Allowed Coders Mode' parameter in the IP Profile table to **Preference** or **Restriction and Preference**. If set to **Preference**, in addition to the Allowed coders that are listed first in the SDP offer, the original coders received in the SDP are retained and listed after the Allowed coders. Thus, this mode does not necessarily restrict coder use to Allowed coders, but uses (prefers) the Allowed coders whenever possible.
- **Outgoing SDP offer:** If only Allowed coders are used, the coders are arranged in the SDP offer as described above. However, if Extension coders are also used, then the coder list is arranged according to the 'SBC Preferences Mode' parameter in the IP Profile table. This parameter can be configured to add the Extension coders after the Allowed coders (i.e., at the end of the list - default) according to their order in the Coders Group table, or arrange Allowed and Extension coders according to their position in the Coders Group table.

19.4.8 SRTP-RTP and SRTP-SRTP Transcoding

The device supports transcoding between SRTP and RTP. The device can also enforce specific SBC legs to use SRTP and/or RTP. The device's handling of SRTP/RTP is configured using the IP Profile parameter, SBCMediaSecurityBehaviour, which provides the following options:

- SBC passes the media as is, regardless of whether it's RTP or SRTP (default).
- SBC legs negotiate only SRTP media lines (m=); RTP media lines are removed from the incoming SDP offer\answer.
- SBC legs negotiate only RTP media lines; SRTP media lines are removed from the incoming offer\answer.
- Each SDP offer\answer is extended (if not already) to two media lines for RTP and SRTP.

If after SDP offer\answer negotiation, one SBC leg uses RTP while the other uses SRTP, then the device performs RTP-SRTP transcoding. To translate between RTP and SRTP, the following prerequisites must be met:

- At least one supported SDP "crypto" attribute.
- The EnableMediaSecurity parameter must be set to 1.

Channel resources are not required for transcoding between RTP and SRTP.

Transcoding where both legs are configured for SRTP is typically required to trans-encrypt and trans-decrypt. This is relevant when the MKI and Symmetric MKI parameters are enabled. In other words, both sides need to both encrypt and decrypt the outgoing and incoming SRTP packets, respectively. Channel resources are not required for transcoding between SRTP and SRTP.

19.4.9 Multiple RTP Media Streams per Call Session

The device's SBC application supports multiple RTP media streams per SBC call session. Up to five different media types can be included in a session:

- Audio (m=audio)
- Video (m=video)
- Text (m=text)
- Fax (m=image)

Therefore, the device can provide transcoding of various attributes in the SDP offer/answer (e.g., codec, port, and packetization time) per media type. If the device is unable to perform transcoding (for example, does not support the codec), it relays the SBC dialog transparently.

19.4.10 Interworking DTMF Methods

The device supports interworking between various DTMF methods such as RFC 2833, In-Band DTMF's, and SIP INFO (Cisco\Nortel\Korea). By default, the device allows the remote user agents to negotiate (in case of RFC 2833) and passes DTMF without intervention. However, if two user agents (UA) support different DTMF methods, the device can interwork these different DTMF methods at each leg.

This DTMF interworking feature is enabled using IP Profiles (*ini* file parameter IPProfile):

- SBCRFC2833Behavior - affects the RFC 2833 SDP offer\answer negotiation:
 - [0] (default): the device does not intervene in the RFC 2833 negotiation.
 - [1]: each outgoing offer\answer includes RFC 2833 in the offered SDP (the device adds RFC 2833 only if the incoming offer does not include RFC 2833).
 - [2]: the device removes RFC 2833 from the incoming offer.
- SBCAlternativeDTMFMethod – the device's first priority for DTMF method at each leg is RFC 2833. Therefore, if a specific leg negotiates RFC 2833 successfully, then the chosen DTMF method for this leg is RFC 2833. For legs where RFC 2833 is not negotiated successfully, the device uses this parameter to determine the DTMF method for the leg.
 - [0] (default): the device does not attempt to interwork any special DTMF method
 - [1]: In Band
 - [2]: INFO, Cisco
 - [3]: INFO, Nortel
 - [4]: INFO, Korea

The chosen DTMF method determines (for each leg) which DTMF method is used for sending DTMF's. If the device interworks between different DTMF methods and one of the methods is In-band\RFC 2833, detection and generation of DTMF methods requires DSP allocation.

19.5 Fax Negotiation and Transcoding

The device can allow fax transmissions to traverse transparently without transcoding or it can handle the fax as follows:

- Allow interoperability between different fax machines, supporting fax transcoding if required.
- Restrict usage of specific fax coders to save bandwidth, enhance performance, or comply with supported coders. These coders include G.711 (A-Law or Mu-Law), VBD (G.711 A-Law or G.711 Mu-Law), and T38.

Fax configuration is done in the IP Profile and Coder Group Settings tables. The IP Profile table determines the supported fax coders and the negotiation method used between the incoming and outgoing fax legs, using the following fax-related parameters:

- SBCFaxBehavior: defines the offer negotiation method - pass fax transparently, negotiate fax according to fax settings in IP Profile, or enforce remote UA to first establish a voice channel before fax negotiation.
- SBCFaxCodersGroupID: defines the supported fax coders (from the Coders Group Settings table).
- SBCFaxOfferMode: determines the fax coders sent in the outgoing SDP offer.
- SBCFaxAnswerMode: determines the fax coders sent in the outgoing SDP answer.



Note: The voice-related coder configuration (Allowed and Extended coders) is independent of the fax-related coder configuration, with the exception of the G.711 coder. If the G.711 coder is restricted by the Allowed Coders Group table, it is not used for fax processing even if it is listed in the Coders Group Settings table for faxes. However, support for G.711 coders for voice is not dependent upon which fax coders are listed in the Coders Group Settings table.

19.6 Limiting SBC Call Duration

You can define a maximum allowed duration (in minutes) for SBC calls. If an established call reaches this user-defined limit, the device terminates the call. This feature ensures calls are properly terminated, allowing available resources for new calls. This feature is configured using the MaxCallDuration parameter.

19.7 SIP Authentication Server for SBC Users

The device can function as an authentication server for SIP SBC message requests, based on HTTP authentication DIGEST with MD5. Alternatively, such requests can be authenticated by an external, third-party server.

When functioning as an authentication server (set by the IP Group table parameter, AuthenticationMode), the device authenticates users belonging to a User-type IP Group. When the device receives an INVITE or REGISTER request from a client (e.g., SIP phone) for SIP message authorization, the device processes the authorization as follows:

1. The device verifies the type of incoming SIP method (e.g., INVITE) that must be challenged for authorization. This is configured using the IP Group table parameter, MethodList.
2. If the message is received without an Authorization header, the device "challenges" the client by sending a 401 or 407 SIP response. The client then resends the request with an Authorization header (containing the user name and password).
3. The device validates the SIP message according to the settings of the parameters, AuthNonceDuration, AuthChallengeMethod and AuthQOP.
 - If validation fails, the message is rejected and the device sends a 403 "Forbidden" response.
 - If validation succeeds, the device verifies identification of the SBC user. This is done by checking that the user name and password received from the user is the same username and password that appears in the device's database. The SBC users in the database are obtained from the User Information file. If the SIP SBC user is not successfully authenticated after three attempts, the device sends a 403 "Forbidden" response.
4. If the user is successfully identified, the SIP message request is processed.

19.8 Interworking SIP Signaling

The device supports interworking of SIP signaling messages to ensure interoperability between communicating SIP UAs or entities. This is critical in network environments where the UAs on opposing SBC legs have different SIP signaling support. For example, some UAs may support different versions of a SIP method while others may not even support a specific SIP method. The configuration method for assigning specific SIP message handling modes to UAs, includes configuring an IP Profile with the required interworking mode, and then assigning the IP Profile to the relevant IP Group.

This section describes some of the device's support for handling SIP methods to ensure interoperability.

19.8.1 Interworking SIP 3xx Redirect Responses

The device supports interworking of SIP 3xx redirect responses. By default, the device's handling of SIP 3xx responses is to send the Contact header unchanged. However, some SIP UAs may support different versions of the SIP 3xx standard while others may not even support SIP 3xx.

The handling of SIP 3xx can be configured for all calls, using the global parameter `SBC3xxBehavior`. For configuring different SIP 3xx handling options for different UAs (i.e., per IP Group), use the IP Profile table parameter, 'SBC Remote 3xx Behavior'.

19.8.1.1 Resultant INVITE Traversing Device

The device can handle SIP 3xx responses so that the new INVITE message sent as a result of the 3xx traverses the device. The reasons for enforcing resultant INVITEs to traverse the device may vary:

- The user that receives the 3xx is unable to route to the 3xx contact (i.e., the user is on the LAN and the new contact is on the WAN). In such a scenario, the device enables the user to reach the WAN contact and overcome NAT problems.
- Enforce certain SBC policies (e.g., call admission control, header manipulation, and transcoding) on the resultant INVITE.

The device enforces this by modifying each Contact in the 3xx response as follows:

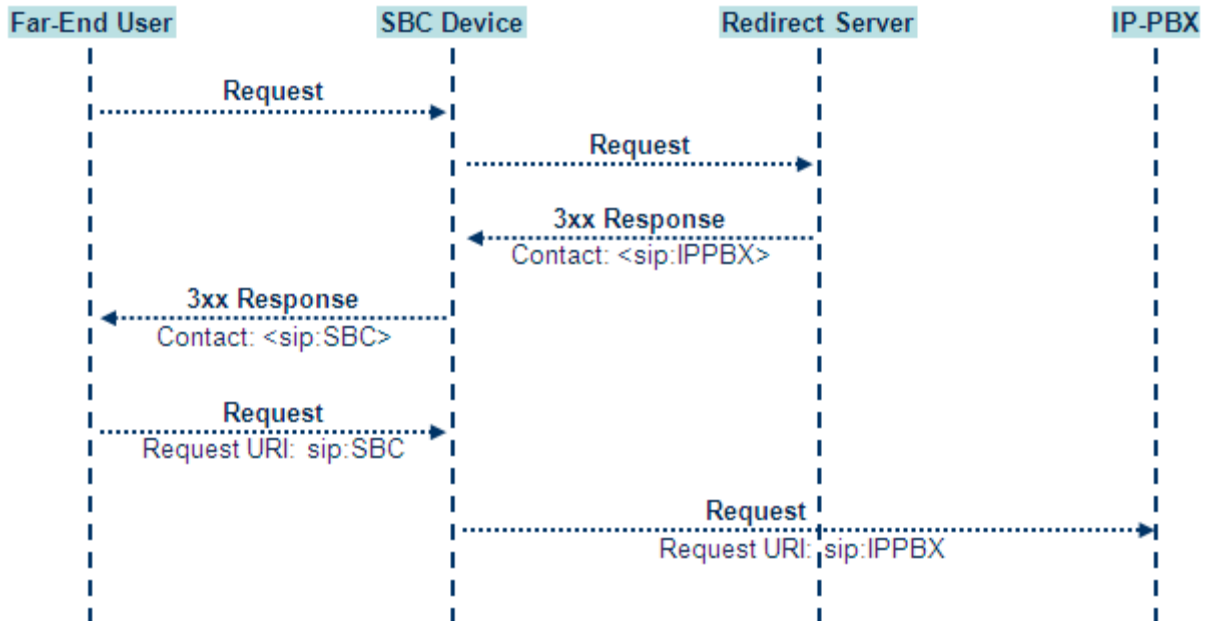
- Changes the host part to the device's IP address – this change causes the remote user agent to send the INVITE to the device.
- Adds a special prefix ("T~&R_") to the Contact user part – to identify the new INVITE as a 3xx resultant INVITE.

The SBC handling for the 3xx resultant INVITE is as follows:

1. The incoming INVITE is identified as a 3xx resultant INVITE according to the special prefix.
2. The device automatically replaces the SBC host part (in the Request-URI) with the host from the 3xx Contact.
3. The prefix ("T~&R_") remains in the user part for the classification, manipulation, and routing mechanisms.
4. The classification, manipulation, and routing processes are done exactly like any other INVITE handling. The special prefix can be used for specific routing rules for 3xx resultant INVITEs.

- The prefix is removed before the resultant INVITE is sent to the destination.

Figure 19-6: SIP 3xx Response Handling



The process of this feature is described using an example:

- The device receives the Redirect server's SIP 3xx response (e.g., Contact: <sip:User@IPPBX:5060;transport=tcp;param=a;q=0.5>).
- The device replaces the Contact header value with the special prefix and database key value as user part, and with the device's URL as host part (e.g., Contact: <sip:Prefix_Key_User@SBC:5070;transport=udp;q=0.5>).
- The device sends this manipulated SIP 3xx response to the Far-End User (FEU).
- The FEU sends a new request with the Request-URI set to the value of the received 3xx response's Contact header (e.g., RequestURI: sip:Prefix_Key_User@SBC:5070;transport=udp).
- Upon receipt of the new request from the FEU, the device replaces the Request-URI with the new destination address (e.g., RequestURI: sip:Prefix_User@IPPBX:5070;transport=tcp;param=a).
- The device removes the user prefix from the Request-URI, and then sends this Request-URI to the new destination (e.g., RequestURI: sip:User@IPPBX:5070;transport=tcp;param=a).

19.8.1.2 Local Handling of SIP 3xx

The device can handle SIP 3xx responses on behalf of the dialog-initiating UA and retry the request (e.g., INVITE) using one or more alternative URIs included in the 3xx response. The new request includes SIP headers from the initial request such as Diversion, History-Info, P-Asserted-Id, and Priority. The source and destination URIs can be manipulated using the regular manipulation mechanism.

The device sends the new request to the alternative destination according to the IP-to-IP Routing table rules. (where the 'Call Trigger' field is set to **3xx**). It is also possible to specify the IP Group that sent the 3xx request as matching criteria for the re-routing rule in this table ('ReRoute IP Group ID' field).

19.8.2 Interworking SIP Diversion and History-Info Headers

This device can be configured to interwork between the SIP Diversion and History-Info headers. This is important, for example, to networks that support the Diversion header but not the History-Info header, or vice versa. Therefore, mapping between these headers is crucial for preserving the information in the SIP dialog regarding how and why (e.g., call redirection) the call arrived at a certain SIP UA.

This feature is configured in the IP Profile table (IPProfile parameter) using the following new parameters:

- SBCDiversionMode - defines the device's handling of the Diversion header
- SBCHistoryInfoMode - defines the device's handling of the History-Info header

The handling of the SIP Diversion and History-Info headers is described in the table below:

Table 19-1: Handling of SIP Diversion and History-Info Headers

Parameter Value	SIP Header Present in Received SIP Message		
	Diversion	History-Info	Diversion and History-Info
HistoryInfoMode = Add DiversionMode = Remove	Diversion converted to History-Info. Diversion removed.	Not present	Diversion removed.
HistoryInfoMode = Remove DiversionMode = Add	Not present.	History-Info converted to Diversion. History-Info removed.	History-Info added to Diversion. History-Info removed.
HistoryInfoMode = Disable DiversionMode = Add	Diversion converted to History-Info.	Not present.	Diversion added to History-Info.
HistoryInfoMode = Disable DiversionMode = Add	Not present.	History-Info converted to Diversion.	History-Info added to Diversion.
HistoryInfoMode = Add DiversionMode = Add	Diversion converted to History-Info.	History-Info converted to Diversion.	Headers are synced and sent.
HistoryInfoMode = Remove DiversionMode = Remove	Diversion removed.	History-Info removed.	Both removed.

19.8.3 Interworking SIP REFER Messages

The device supports interworking of SIP REFER messages. SIP UAs may support different versions of the REFER standard while others may not even support REFER.

This feature supports the following:

- Attended, unattended, and semi-attended call transfers
- Sending INVITE, REFER-notifications, BYE, PRACK and Session Timer on behalf of peer PBXs

- Advanced routing rules for the new, initiated INVITE
- Forwarding early media after REFER while attempting to avoid transcoding (by sending session update)
- Interoperate with environments where different SIP UAs lack basic SIP functionality such as re-INVITE, UPDATE, PRACK, Delayed Offer, re-INVITE without SDP
- Session updates after connect to avoid transcoding

The handling of REFER can be configured for all calls, using the global parameter SBCReferBehavior. For configuring different REFER handling options for different UAs (i.e., IP Groups), use the IP Profile table parameter, 'SBC Remote Refer Behavior'.

- **Local handling of REFER:** This option is used for UAs that do not support REFER. Upon receipt of a REFER request, instead of forwarding it to the IP Group, the device handles it locally. It generates a new INVITE to the alternative destination according to the rules in the IP-to-IP Routing table (where the 'Call Trigger' field is set to **REFER**). It is also possible to specify the IP Group that sent the REFER request, as matching criteria for the re-routing rule in this table ('ReRoute IP Group ID' field).
- **Transparent handling:** The device forwards the REFER with the Refer-To header unchanged.
- **Re-routing through SBC:** The device changes the Refer-To header so that the re-routed INVITE is sent through the SBC application.
- **IP Group Name =** The device sets the host part in the REFER message to the name configured for the IP Group in the IP Group table.

19.8.4 Interworking SIP PRACK Messages

The device supports interworking of SIP Provisional Response ACKnowledgement (PRACK) messages (18x). While some UAs may not support PRACK (RFC 3262), others may require it. The device can be configured to resolve this interoperable issue and enable sessions between such endpoints. SIP PRACK handling is configured using the IP Profile parameter, 'SBC Prack Mode':

- **Optional:** PRACK is optional for these UAs. If required, the device performs the PRACK process on behalf of the destination UA.
- **Mandatory:** PRACK is required for these UAs. Calls from UAs that do not support PRACK are rejected. Calls destined to these UAs are also required to support PRACK.
- **Transparent (default):** The device does not intervene with the PRACK process and forwards the request as is.

19.8.5 Interworking SIP Session Timer

The device supports interworking of the SIP signaling keep-alive mechanism. The SIP standard provides a signaling keep-alive mechanism using re-INVITE and UPDATE messages. In certain setups, keep-alive may be required by some SIP UAs while for others it may not be supported. The device can resolve this mismatch by performing the keep-alive process on behalf of SIP UAs that do not support it.

For configuring the handling of session expires, use the IP Profile parameter, 'SBC Session Expires Mode'.

19.8.6 Interworking SIP Early Media

The device supports various interworking modes for SIP early media between SIP UAs (i.e., IP Groups):

- **Early Media Enabling:** The device supports the interworking of early media between SIP UAs that support early media and those that do not support receipt of early media.

Early media can arrive in provisional responses to an INVITE request. The device forwards the request of early media for IP Groups that support this capability; otherwise, the device terminates it. Provisional responses whose SDP are suppressed are changed to a SIP 180 response. This feature is also supported for delayed offers. This is configured using the IP Profile parameter, 'SBC Remote Early Media Support'. The device refers to this parameter also for features that require early media such as playing ringback tone.

- **Early Media Response Type:** The device supports the interworking of different SIP provisional response types between UAs for forwarding the early media to the caller. This can support all early media response types (default), SIP 180 only, or SIP 183 only, and is configured by the IP Profile parameter, 'SBC Remote Early Media Response Type'.
- **Multiple 18x:** The device supports the interworking of different support for multiple 18x responses (including 180 Ringing, 181 Call is Being Forwarded, 182 Call Queued, and 183 Session Progress) that are forwarded to the caller. The UA can be configured as supporting only receipt of the first 18x response (i.e., the device forwards only this response to the caller), or receipt of multiple 18x responses (default). This is configured by the IP Profile parameter, 'SBC Remote Multiple 18x Support'.
- **Early Media RTP:** The device supports the interworking with remote clients that send 18x responses with early media and whose subsequent RTP is delayed, and with remote clients that do not support this and require RTP to immediately follow the 18x response. Some clients do not support 18x with early media, while others require 18x with early media (i.e., they cannot play ringback tone locally). These various interworking capabilities are configured by the IP Profile parameters, 'SBC Remote Early Media RTP', 'SBC Remote Supports RFC 3960', and 'SBC Remote Can Play Ringback'. See the flowcharts below for the device's handling of such scenarios:

Figure 19-7: SBC Early Media RTP 18x without SDP

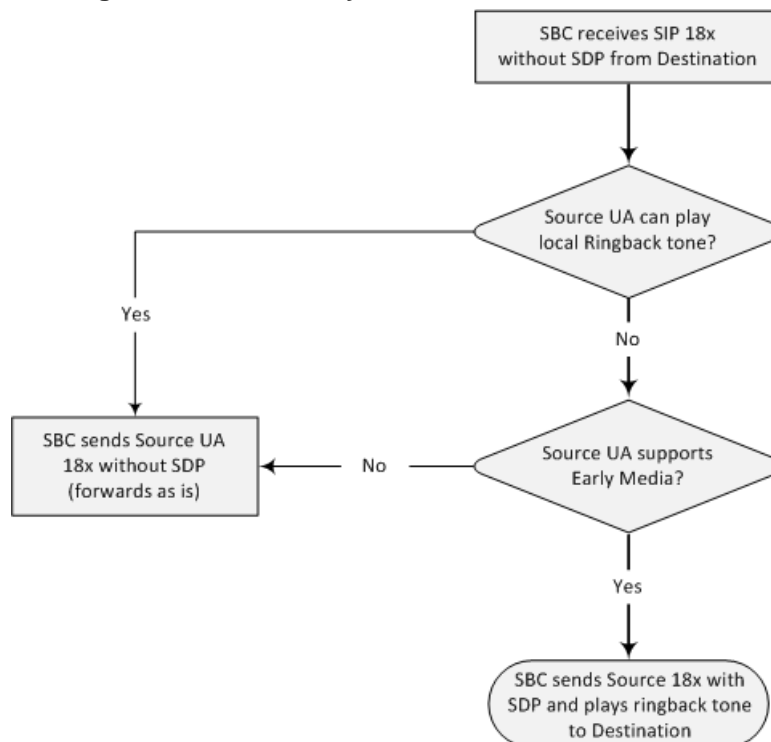
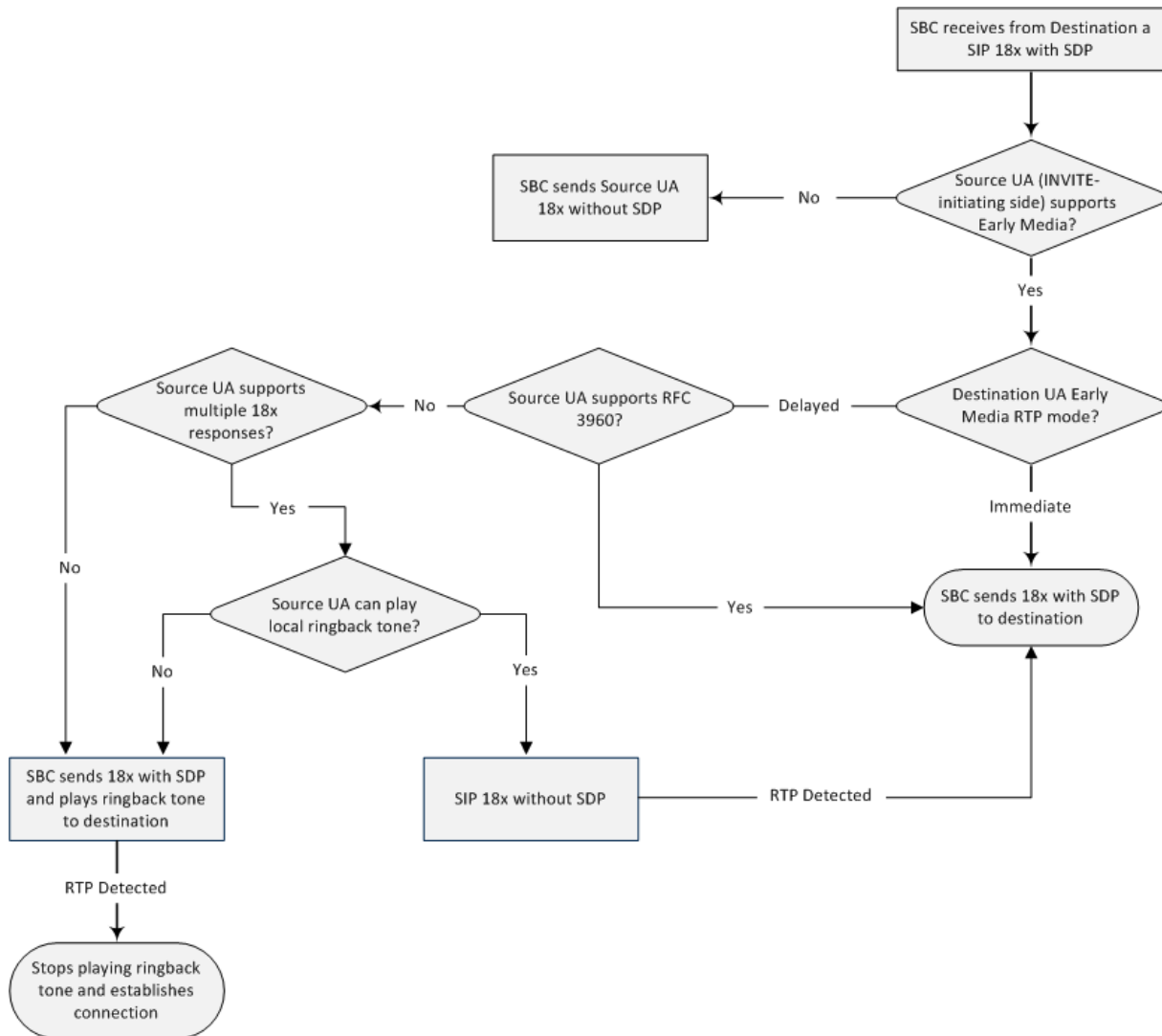


Figure 19-8: Early Media RTP - SIP 18x with SDP


19.8.7 Interworking SIP re-INVITE Messages

The device supports interworking of SIP re-INVITE messages. This enables communication between endpoints that generate re-INVITE requests and those that do not support the receipt of re-INVITEs. The device does not forward re-INVITE requests to IP Groups that do not support it. Instead, it sends a SIP response to the re-INVITE request, which can either be a success or a failure, depending on whether the device can bridge the media between the endpoints. The device can also handle re-INVITEs with or without an SDP body, enabling communication between endpoints that do not support re-INVITE requests without SDP, and those that require SDP. The device generates an SDP offer and adds it to the incoming re-INVITE request if it does not contain an SDP and only then forwards it to the destination endpoint. This interworking support is configured by the IP Profile parameter, 'SBC Remote Reinvite Support'.

19.8.8 Interworking SIP UPDATE Messages

The device supports interworking of the SIP UPDATED message. This enables communication between UAs that generate UPDATE requests and those that do not support the receipt of UPDATE requests. The device does not forward UPDATE requests to IP Groups that do not support it. Instead, it sends a SIP response to the UPDATE

request which can either be a success or a failure, depending on whether the device can bridge the media between the endpoints. The handling of UPDATE messages is configured by the IP Profile parameter 'SBC Remote Update Support'.

19.8.9 Interworking SIP re-INVITE to UPDATE

The device enables communication between endpoints (IP Groups) that do not support re-INVITE requests but support the UPDATE method, and vice versa. The device translates the re-INVITE request to the UPDATE request, and vice versa. Note that if a re-INVITE request arrives without SDP, the device generates the SDP and adds it to the outgoing UPDATE request. To enable this feature, each IP Group needs to be configured with its unique capabilities by associating it with a relevant IP Profile. For example, an IP Group that supports UPDATE requests but not re-INVITES would be configured as follows:

- SBCRemoteUpdateSupport = 2 (Supported)
- SBCRemoteReinviteSupport = 0 (Not Supported)

If a re-INVITE request needs to be forwarded to this IP Group, it is translated to an UPDATE request.

19.8.10 Interworking Delayed Offer

The device enables sessions between endpoints (IP Groups) that send INVITEs without SDP (i.e., delayed media) and those that do not support the receipt of INVITEs without SDP. The device creates an SDP and adds it to INVITEs that arrive without SDP. This intervention in the SDP offer/answer process may require transcoding. Delayed offer is also supported when early media is present.

The interworking of delayed offer is configured using the IP Profile parameter 'SBC Remote Delayed Offer Support'.



Note: For this feature to function properly, a valid Extension Coders Group ID needs to be configured for IP Profiles that do not support delayed offer.

19.8.11 Interworking Call Hold

The device supports the interworking of call hold / retrieve requests between IP entities supporting different call hold capabilities:

- Interworking SDP call hold formats. This is configured by the IP Profile parameter, 'SBC Remote Hold Format'.
- Interworking the play of the held tone for IP entities that cannot play held tones locally. This is configured by the IP Profile parameter, 'SBC Play Held Tone'.
- Interworking generation of held tone where the device generates the tone to the held party instead of the call hold initiator. This is configured by the IP Profile parameter, 'SBC Reliable Held Tone Source'.

For configuring IP Profiles, see Configuring IP Profiles [221](#).

19.9 Call Survivability

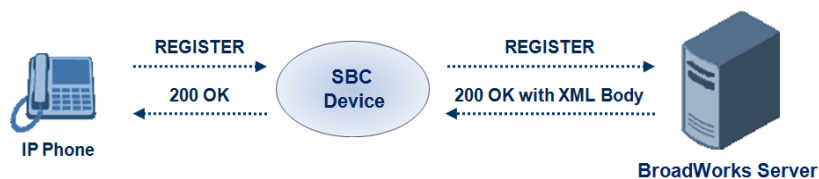
This section describes various call survivability features supported by the SBC device.

19.9.1 Auto-Provisioning of Subscriber-Specific Information for BroadWorks Server for Survivability

This feature enables SBC user registration for interoperability with BroadSoft BroadWorks server to provide call survivability in case of connectivity failure with the BroadWorks server, for example, due to a WAN failure. This feature enables local users to dial a local extension (or any other configured alias) that identifies another local user, in survivability mode. This feature is enabled using the SBCExtensionsProvisioningMode parameter.

In normal operation, when subscribers (such as IP phones) register to the BroadWorks server through the device, the device includes the SIP Allow-Events header in the sent REGISTER message. In response, the BroadWorks server sends the device a SIP 200 OK containing an XML body with subscriber information such as extension number, phone number, and URIs (aliases). The device forwards the 200 OK to the subscriber (without the XML body).

Figure 19-9: Interoperability with BroadWorks Registration Process



The device saves the users in its registration database with their phone numbers and extensions, enabling future routing to these destinations during survivability mode. When in survivability mode, the device routes the call to the Contact associated with the dialed phone number or extension number in the registration database.

Below is an example of an XML body received from the BroadWorks server:

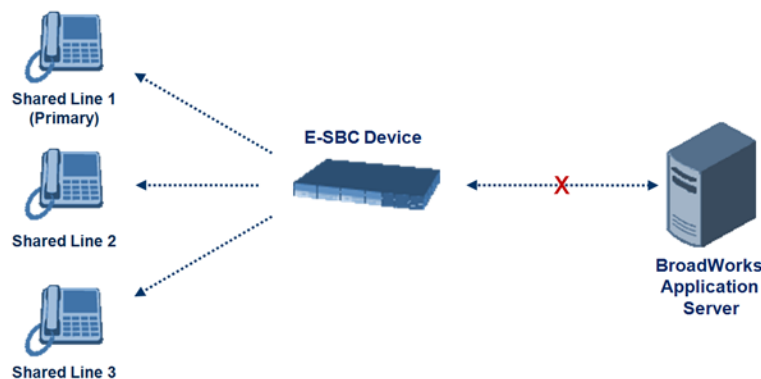
```
<?xml version="1.0" encoding="utf-8"?>
<BroadsoftDocument version="1.0" content="subscriberData">
  <phoneNumbers>
    <phoneNumber>2403645317</phoneNumber>
    <phoneNumber>4482541321</phoneNumber>
  </phoneNumbers>
  <aliases>
    <alias>sip:bob@broadsoft.com</alias>
    <alias>sip:rhughes@broadsoft.com</alias>
  </aliases>
  <extensions>
    <extension>5317</extension>
    <extension>1321</extension>
  </extensions>
</BroadSoftDocument>
```

19.9.2 BroadSoft's Shared Phone Line Call Appearance for SBC Survivability

The device can provide redundancy for BroadSoft's Shared Call Appearance feature. When the BroadSoft application server switch (AS) fails or does not respond, or when the network connection between the device and the BroadSoft AS is down, the device manages the Shared Call Appearance feature for the SIP clients.

This feature is supported by configuring a primary extension and associating it with secondary extensions (i.e., *shared lines*) so that incoming calls to the primary extension also ring at the secondary extensions. The call is established with the first extension to answer the call, and consequently, the ringing at the other extensions stop. For example, assume primary extension number 600 is shared with secondary extensions 601 and 602. In the case of an incoming call to 600, all three phone extensions ring simultaneously (using the device's call forking feature as described in 'SIP Forking Initiated by SIP Proxy Server' on page 265). Note that incoming calls specific to extensions 601 or 602 ring only at these specific extensions.

Figure 19-10: Call Survivability for BroadSoft's Shared Line Appearance



To configure this capability, you need to configure a shared-line, inbound manipulation rule for registration requests to change the destination number of the secondary extension numbers (e.g. 601 and 602) to the primary extension (e.g., 600). Call forking must also be enabled. The procedure below describes the main configuration required.



Notes:

- You can configure whether REGISTER messages from secondary lines are terminated on the device or forwarded transparently (as is), using the `SBCSharedLineRegMode` parameter.
- The device enables outgoing calls from all equipment that share the same line simultaneously (usually only one simultaneous call is allowed per a specific shared line).
- The LED indicator of a shared line may display the wrong current state.

➤ To configure the Shared Line feature:

1. In the IP Group table (see 'Configuring IP Groups' on page 191), add a Server-type IP Group for the BroadWorks server.
2. In the IP Group table, add a User-type IP Group for the IP phone users and set the 'Enable SBC Client Forking' parameter to **Yes** so that the device forks incoming calls to all contacts under the same AOR that are registered in the device's registration database.

3. In the IP-to-IP Routing table (see 'Configuring SBC IP-to-IP Routing' on page 279), add a rule for routing calls between the above configured IP Groups.
4. In the IP to IP Inbound Manipulation table (see 'Configuring IP-to-IP Inbound Manipulations' on page 289), add a manipulation rule for the secondary extensions (e.g., 601 and 602) so that they also register in the device's database under the primary extension contact (e.g., 600):
 - Set the 'Manipulation Purpose' field to **Shared Line**.
 - Set the 'Source IP Group' field to the IP Group ID that you created for the users (e.g., 2).
 - Set the 'Source Username Prefix' field to represent the secondary extensions (e.g., 601 and 602).
 - Set the 'Manipulated URI' field to **Source** to manipulate the source URI.
 - Set the 'Remove From Right' field to "1" to remove the last digit of the extensions (e.g., 601 is changed to 60).
 - Set the 'Suffix to Add' field to "0" to add 0 to the end of the manipulated number (e.g., 60 is changed to 600).

19.9.3 Call Survivability for Call Centers

The device supports call survivability for call centers. When a communication failure (e.g., in the network) occurs with the remote voice application server responsible for handling the call center application (such as IVR), the device routes the incoming calls received from the customer (i.e., from the TDM gateway) to the call center agents.

In normal operation, the device registers the agents in its users registration database. Calls received from the TDM gateway are forwarded by the device to the application server, which processes the calls and sends them to specific call center agents, through the device. Upon a failure with the application server, the device routes the calls from the TDM Gateway to the agents. The device routes the call to the first available user it finds. If the call is not answered by the user, the device routes it to the next available user. The SBC can handle a sequence of up to five users, after which the session is timed out and the call is dropped.

Figure 19-11: Normal Operation in Call Center Application

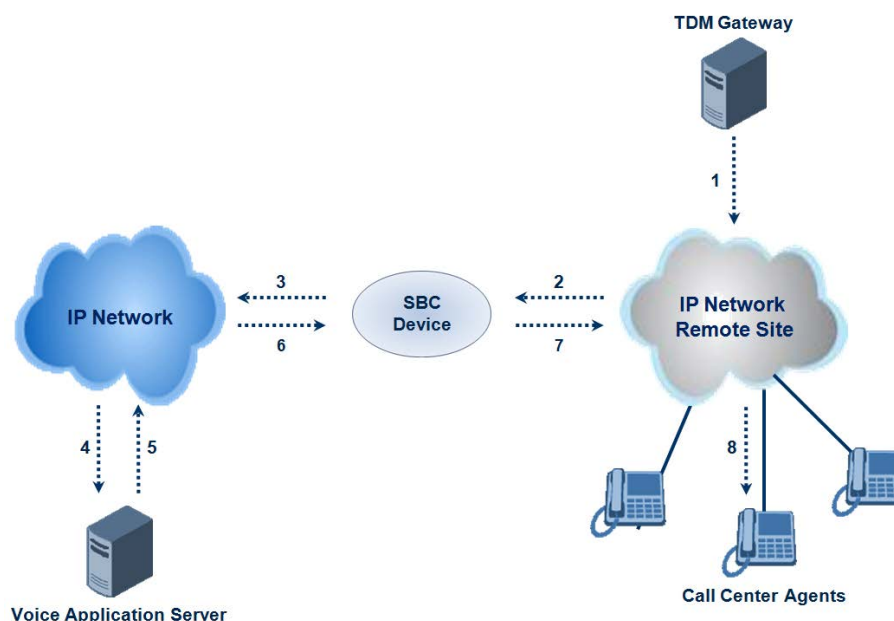
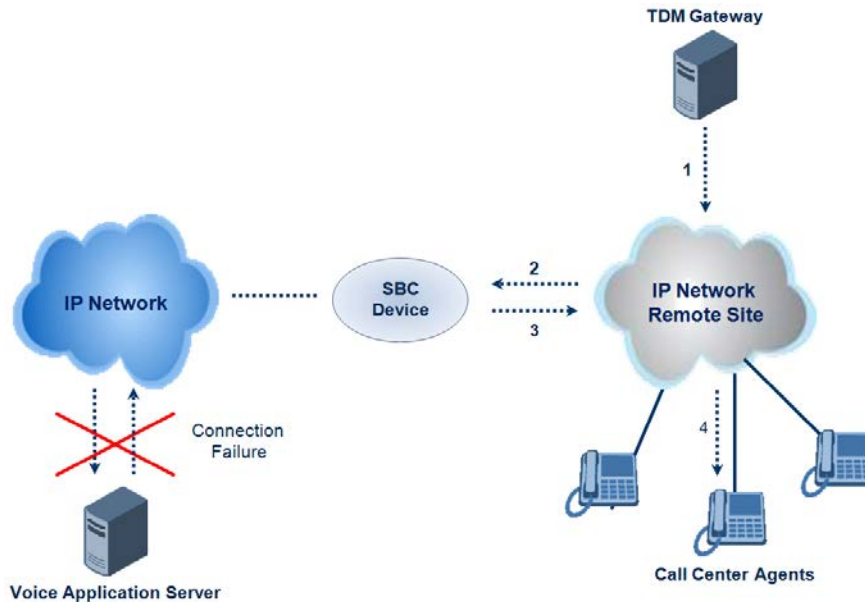


Figure 19-12: Call Survivability for Call Center

➤ **To configure call survivability for a call center application:**

1. In the IP Group table (see 'Configuring IP Groups' on page 191), add IP Groups for the following entities:
 - TDM Gateway (Server-type IP Group). This entity forwards the customer calls through the device to the Application server.
 - Application server (Server-type IP Group). This entity processes the call and sends the call through the device to the specific call center agent located on a different network (remote).
 - Call center agents (User-type IP Group). You can configure multiple IP Groups to represent different groups of call center agents, for example, agents and managers.
2. In the Classification table (see 'Configuring Classification Rules' on page 272), add rules to classify incoming calls that are received from the entities listed in Step 1, to IP Groups.
3. In the SBC IP-to-IP Routing table (see 'Configuring SBC IP-to-IP Routing' on page 279), add the following IP-to-IP routing rules:
 - For normal operation:
 - ◆ Routing from TDM Gateway to Application server.
 - ◆ Routing from Application server to call center agents.
 - For call survivability mode: Routing from TDM Gateway to call center agents. This configuration is unique due to the following settings:
 - ◆ The 'Source IP Group ID' field is set to the IP Group of the TDM Gateway.
 - ◆ The 'Destination Type' field is set to **Hunt Group**, which is specifically used for call center survivability.
 - ◆ The 'Destination IP Group ID' field is set to the IP Group of the call center agents.

The figure below displays a routing rule example, assuming IP Group "1" represents the TDM Gateway and IP Group "3" represents the call center agents:

Figure 19-13: Routing Rule Example for Call Center Survivability

Add Record ✕	
Index	3
Source IPGroup ID	1
Source Username Prefix	*
Source Host	*
Destination Username Prefix	*
Destination Host	*
Request Type	All ▼
Message Condition	None ▼
Destination Type	Hunt Group ▼
Destination IPGroup ID	3
Destination SRD ID	None ▼
Destination Address	
Destination Port	0
Destination Transport Type	▼
Alternative Route Options	Route Row ▼
Cost Group	None ▼
<input type="button" value="Submit"/> <input type="button" value="✕ Cancel"/>	

19.9.4 Survivability Mode Display on Aastra IP Phones

If the SBC device is deployed in an Enterprise network with Aastra IP phones and connectivity with the WAN fails, the device provides call survivability by enabling communication between IP phone users within the LAN enterprise. In such a scenario, the device can be configured to notify the IP phones that it is currently operating in Survivability mode. When this occurs, the Aastra IP phones display the message, "StandAlone Mode" on their LCD screens. This feature is enabled by setting the SBCEnableAASTRASurvivabilityNotice parameter to 1.

When this feature is enabled and the SBC device is in Survivability mode, it responds to SIP REGISTER messages from the IP phones with a SIP 200 OK containing the following XML body:

```
Content-Type: application/xml
<?xml version="1.0" encoding="utf-8"?>
<LMIDocument version="1.0">
  <LocalModeStatus>
    <LocalModeActive>true</LocalModeActive>
    <LocalModeDisplay>StandAlone Mode</LocalModeDisplay>
  </LocalModeStatus>
</LMIDocument>
```


19.10 Call Forking

This section describes various Call Forking features supported by the device.

19.10.1 Initiating SIP Call Forking

The SBC device supports call forking of an incoming call to multiple SBC users (destinations). Call forking is supported by the device's capability of registering multiple SIP client user phone contacts (mobile and fixed-line extensions) under the same Address of Record (AOR) in its registration database. This feature can be implemented in the following example scenarios:

- An enterprise Help Desk, where incoming customer calls are simultaneously sent to multiple customer service agent extensions.
- An employee's phone devices, where the incoming call is simultaneously sent to multiple devices (e.g., to the employee's office phone and mobile SIP phone).
- An enterprise reception desk, where an incoming call is simultaneously sent to multiple receptionists.

The device supports various modes of call forking. For example, in Parallel call forking mode, the device sends the INVITE message simultaneously to all the users registered under the same AOR, resulting in the ringing of all extensions; the first extension to pick up the call receives the call, and all other extensions stop ringing. The Call Forking feature is configured by creating a User-type IP Group and configuring the IP Group table's parameter, 'SBC Client Forking Mode' (see Configuring IP Groups on page 191).

The device can also fork INVITE messages received for a Request-URI of a specific contact (user) registered in the database to all other users located under the same AOR as the specific contact. This is configured using the SBCSendInviteToAllContacts parameter.

19.10.2 SIP Forking Initiated by SIP Proxy Server

The device can handle SIP forking responses received from a proxy server in response to an INVITE forwarded by the device from a UA. In other words, received responses with a different SIP To header 'tag' parameter for the request forwarded by the device. This occurs in scenarios, for example, where a proxy server forks the INVITE request to several UAs, and therefore, the SBC device may receive several replies for a single request. Forked SIP responses may result in a single SDP offer with two or more SDP answers during call setup. The SBC handles this scenario by "hiding" the forked responses from the INVITE-initiating UA. This is achieved by marking the UA that responded first to the INVITE as the active UA, and only requests/responses from that UA are subsequently forwarded. All other requests/responses from other UAs are handled by the SBC (SDP offers from these users are answered with an 'inactive' media).

The SBC supports two forking modes, configured by the SBCForkingHandlingMode parameter:

- Latch On First - only the first received 18x response is forwarded to the INVITE initiating UA, and disregards any subsequently received 18x forking responses (with or without SDP).
- Sequential - all 18x responses are forwarded to the INVITE initiating UA, one at a time in a sequential manner. If 18x arrives with an offer only, only the first offer is forwarded to the INVITE initiating UA.

The SBC also supports media synchronization for call forking. If the active UA is the first one to send the final response (e.g., 200 OK), the call is established and all other final responses are acknowledged and a BYE is sent if needed. If another UA sends the first final response, then it is possible that the SDP answer that was forwarded to the INVITE-initiating UA is not relevant, and media synchronization is needed between the two UAs.

Media synchronization is done by sending a re-INVITE request immediately after the call is established. The re-INVITE is sent without an offer to the INVITE-initiating UA. This causes the UA to send an offer which is forwarded to the UA that confirmed the call. The media synchronization process is enabled by the EnableSBCMediaSync parameter.

19.11 Alternative Routing on Detection of Failed SIP Response

The device can detect failure of a sent SIP response (e.g., TCP timeout, and UDP ICMP). In such a scenario, the device re-sends the response to an alternative destination. This support is in addition to alternative routing if the device detects failed SIP requests.

For example, assume the device sends a SIP 200 OK in response to a received INVITE request. If the device does not receive a SIP ACK in response to this, it sends a new 200 OK to the next alternative destination. This new destination can be the next given IP address resolved from a DNS from the Contact or Record-Route header in the request related to the response.

20 SBC Configuration

This section describes the configuration of the SBC application.



Note: For the SBC application, the following requirements must be met:

- The SBC application must be enabled (see 'Enabling Applications' on page 183).
- The 'SBC' Software License Key must be installed on the device (see 'Software License Key' on page 370).

20.1 Configuring General Settings

The General Settings page allows you to configure general SBC parameters. For a description of these parameters, see 'SBC Parameters' on page 512.

➤ **To configure general parameters:**

1. Open the General Settings page (**Configuration** tab > **VoIP** menu > **SBC** > **General Settings**).

Figure 20-1: General Settings Page

Transcoding Mode	Only If Required
SBC No Answer Timeout	600
SBC GRUU Mode	AsProxy
Minimum Session-Expires [sec]	90
BroadWorks Survivability Feature	Disable
Bye Authentication	Disable
SBC User Registration Time	0
SBC Proxy Registration Time	0
SBC Survivability Registration Time	0
SBC Forking Handling Mode	Latch On First
Allow Unclassified Calls	Reject
SBC Session-Expires [sec]	180
SBC Direct Media	Disable
Server Authentication	
Lifetime of the nonce in seconds	300
Authentication Challenge Method	0
Authentication Quality of Protection	2

2. Configure the parameters as required.
3. Click **Submit** to apply your changes.
4. To save the changes to flash memory, see 'Saving Configuration' on page 360.

20.2 Configuring Admission Control

The Admission Control page allows you to define up to 100 rules for limiting the number of concurrent calls (SIP dialogs). These call limits can be applied per SRD, IP Group, SIP request type (e.g., INVITEs), SIP dialog direction (e.g., inbound), and/or per user (identified by its registered contact). This feature can be useful for implementing Service Level Agreements (SLA) policies.

The SIP dialog limits can be defined per SIP request type and direction. These relate to requests that initiate SIP dialogs and not the subsequent requests that can be of different type and direction. The SIP dialog-initiating request types can include SIP INVITEs, REGISTER, and/or SUBSCRIBE, or it can be configured to include the total number of all dialogs.

This feature also provides support for SIP-dialog rate control, using the "token bucket" mechanism. The token bucket is a control mechanism that dictates the rate of SIP-dialog setups based on the presence of tokens in the bucket – a logical container that holds aggregate SIP dialogs to be accepted or transmitted. Tokens in the bucket are removed ("cached in") for the ability to setup a dialog. Therefore, a flow can setup dialogs up to its peak burst rate if there are adequate tokens in the bucket and if the burst threshold is configured appropriately:

- Every SIP dialog setup request must attempt to take a token from the bucket.
- If there are no tokens, the request is dropped.
- New tokens are added to the bucket at a user-defined rate (token rate).
- If the bucket contains the maximum number of tokens, tokens to be added at that moment are dropped.

Requests that reach the user-defined call limit (maximum concurrent calls and/or call rate) are sent to an alternative route, if configured in the IP-to-IP Routing table for the SRD or IP Group. If no alternative routing rule is located, the device rejects the SIP request with a SIP 486 "Busy Here" response.



Notes:

- The enforcement of a configured limitation for the incoming leg is performed immediately after the Classification process. If the call/request is rejected at this stage, no routing is performed. The enforcement for the outgoing leg is performed within each alternative route iteration. This is accessed from two places: one during initial classification/routing, and another during alternative routing process.
- The Admission Control table can also be configured using the table *ini* file parameter, SBCAdmissionControl.

➤ **To configure Admission Control rules:**

1. Open the Admission Control page (**Configuration** tab > **VoIP** menu > **SBC** > **Admission Control**).
2. Click the **Add** button; the following dialog box appears:

Figure 20-2: Admission Control Page - Add Record Dialog Box

Add Record	
Index	0
Limit Type	IP Group
IP Group ID	-1
SRD ID	-1
Request Type	All
Request Direction	Both
Limit	-1
Limit Per User	-1
Rate	0
Max Burst	0
<input type="button" value="Submit"/> <input type="button" value="Cancel"/>	

3. Configure the rule as required. For a description of the parameters, see the table below.
4. Click **Submit** to apply your changes.

Table 20-1: Admission Control Parameters

Parameter	Description
Limit Type CLI: limit-type [SBCAdmissionControl_LimitType]	Defines the entity to which the rule applies. <ul style="list-style-type: none"> ▪ [0] IP Group (default) ▪ [1] SRD
IP Group ID [SBCAdmissionControl_IPGroupID]	Defines the IP Group to which you want to apply the rule. To apply the rule to all IP Groups, set this parameter to -1 (default). Note: This parameter is applicable only if 'Limit Type' is set to IP Group .
SRD ID [SBCAdmissionControl_SRIDID]	Defines the SRD to which you want to apply the rule. To apply the rule to all SRDs, set this parameter to -1 (default). Note: This parameter is applicable only if 'Limit Type' is set to SRD .

Parameter	Description
Request Type [SBCAdmissionControl_RequestType]	<p>Defines the SIP dialog-initiating request type to which you want to apply the rule (not the subsequent requests that can be of different type and direction).</p> <ul style="list-style-type: none"> [0] All = (Default) Includes the total number of all dialogs. [1] INVITE [2] SUBSCRIBE [3] Other
Request Direction [SBCAdmissionControl_RequestDirection]	<p>Defines the direction of the SIP request to which the rule applies.</p> <ul style="list-style-type: none"> [0] Both = (Default) Rule applies to inbound and outbound SIP dialogs. [1] Inbound = Rule applies only to inbound SIP dialogs. [2] Outbound = Rule applies only to outbound SIP dialogs.
Limit [SBCAdmissionControl_Limit]	<p>Defines the maximum number of concurrent SIP dialogs per IP Group or SRD. You can also use the following special values:</p> <ul style="list-style-type: none"> [0] 0 = Block all these dialogs. [-1] -1 = (Default) No limit.
Limit Per User [SBCAdmissionControl_LimitPerUser]	<p>Defines the maximum number of concurrent SIP dialogs per user belonging to the specified IP Group or SRD. You can also use the following special values:</p> <ul style="list-style-type: none"> [0] 0 = Block all these dialogs. [-1] -1 = (Default) No limit.
Rate [SBCAdmissionControl_Rate]	<p>Defines the rate at which tokens are added to the token bucket per second (i.e., token rate). One token is added to the bucket every 1000 divided by the value of this parameter (in milliseconds).</p> <p>The default is 0 (i.e., unlimited rate).</p> <p>Note: The token bucket feature is per IP Group, SRD, SIP request type, and SIP request direction.</p>
Max Burst [SBCAdmissionControl_MaxBurst]	<p>Defines the maximum number of tokens (SIP dialogs) that the bucket can hold. The device only accepts a SIP dialog if a token exists in the bucket. Once the SIP dialog is accepted, a token is removed from the bucket. If a SIP dialog is received by the device and the token bucket is empty, then the device rejects the SIP dialog. Alternatively, if the bucket is full, for example, 100 tokens, and 101 SIP dialogs arrive (before another token is added to the bucket, i.e., faster than that defined in the Rate field), then the device accepts the first 100 SIP dialogs and rejects the last one.</p> <p>Dropped requests are replied with the SIP 486 “Busy Here” response. Dropped requests are not counted in the bucket.</p> <p>The default is 0 (i.e., unlimited SIP dialogs).</p> <p>Note: The token bucket feature is per IP Group, SRD, SIP request type, and SIP request direction.</p>

20.3 Configuring Allowed Coder Groups

The Allowed Coders Group page allows you to define up to five Allowed Coder Groups, each with up to 10 coders. Allowed Coder Groups determine the coders that can be used for a specific SBC leg. Therefore, the device can enforce the use of specific coders while preventing the use of other coders. Coders excluded from the Allowed Coders Group are removed from the SDP offer. Only common coders between SDP offered coders and coders configured in the Allowed Coder Groups are used. For more information, see 'Restricting Coders' on page 248.

The Allowed Coders Group table is also used to configure Extension coders. These are coders that are added to the SDP offer. For more information on Extension coders, see Coder Transcoding on page 248.

The order of appearance of coders in the Allowed Coder Group determines the coder priority (preference), whereby the first coder is given the highest priority. For more information, see 'Prioritizing Coder List in SDP Offer' on page 249.



Notes:

- Each coder can appear only once per Allowed Coder Group.
- Allowed Coder Groups are applicable only to audio media.
- Allowed Coder Groups can be assigned to IP Profiles (see 'Configuring IP Profiles' on page 221).
- The Allowed Coder Groups table can also be configured using the table ini file parameter, AllowedCodersGroup.

➤ To configure Allowed Coder Groups:

1. Open the Allowed Coders Group page (**Configuration** tab > **VoIP** menu > **SBC** > **Allowed Coders Group**).

Figure 20-3: Allowed Coders Group Page

Coder Name
G.711A-law
G.729

2. From the 'Allowed Coders Group ID' drop-down list, select an ID for the Allowed Coder Group.

3. In the Coder Name table, select coders for the Allowed Coder Group.
4. Click **Submit** to apply your changes.
5. To save the changes to flash memory, see 'Saving Configuration' on page 360.

20.4 Routing SBC

This section describes the configuration of the routing entities for the SBC application. These include the following:

- Classification rules - see 'Configuring the Classification Rules' on page 272
- Condition rules - see 'Configuring Condition Rules' on page 278
- IP-to-IP routing rules - see 'Configuring SBC IP-to-IP Routing' on page 279
- Alternative routing reasons - see 'Configuring Alternative Routing Reasons' on page 286

20.4.1 Configuring Classification Rules

The Classification table enables you to configure up to 100 Classification rules. Classification rules are used to classify incoming SIP dialog-initiating requests (e.g., INVITE messages) to source IP Groups from where the SIP dialog request originated. The identified IP Group is later used in the manipulation and routing processes.

Classification rules also enhance security by allowing you to create a SIP access list, whereby classified calls can be denied (i.e., blacklist) or allowed (i.e., whitelist).

The Classification table is used to classify incoming SIP dialog requests only if the other classification stages fail, as described below:

1. **Classification Stage 1 - Registered Users Database:** The device searches its registration database to check if the incoming SIP dialog arrived from a registered user:

- Compares the SIP Contact header of the received SIP dialog to the Contact of the registered user.
- Compares the URL in the SIP P-Asserted-Identity/From header to the registered address-of-record (AOR).

If this stage fails, the device proceeds to classification based on Proxy Set.

2. **Classification Stage 2 - Proxy Set:** If the database search fails, the device performs classification based on Proxy Set if the 'Classify By Proxy Set' parameter is enabled for the IP Group (see 'Configuring IP Groups' on page 191). If enabled, the device checks whether the INVITE's IP address (if host names, then according to the dynamically resolved IP address list) is defined for a Proxy Set ID (in the Proxy Set table). If a Proxy Set ID has such an IP address, the device classifies the INVITE to the IP Group that is associated with this Proxy Set. (The Proxy Set ID is assigned to the IP Group using the IP Group table's 'Proxy Set ID' parameter.)



Note: For security purposes, it is highly recommended to disable the Classify by Proxy Set feature so that the device can use the Classification table instead, for "strict" classification of incoming calls to IP Groups. In addition, in cases where multiple IP Groups are associated with the same Proxy Set ID, do **not** use the Classify by Proxy Set feature.

If this stage fails (or Classify by Proxy Set is disabled), the device proceeds to classification based on the Classification table.

3. **Classification Stage 3 - Classification Table:** If classification based on Proxy Set fails (or disabled), the device uses the Classification table to classify the SIP dialog to

an IP Group. If it locates a classification rule whose characteristics (such as source IP address) match the incoming SIP dialog, the SIP dialog is assigned to the associated IP Group. In addition, if the classification rule is defined as a whitelist, the SIP dialog is allowed and proceeds with the manipulation, routing and other SBC processes. If the classification rule is defined as a blacklist, the SIP dialog is denied.

If the classification process fails, the device rejects or allows the call, depending on the setting of the 'Unclassified Calls' parameter (on the General Settings page - **Configuration** tab > **VoIP** menu > **SBC** > **General Settings**). If this parameter is set to **Allow**, the incoming SIP dialog is assigned to an IP Group as follows:

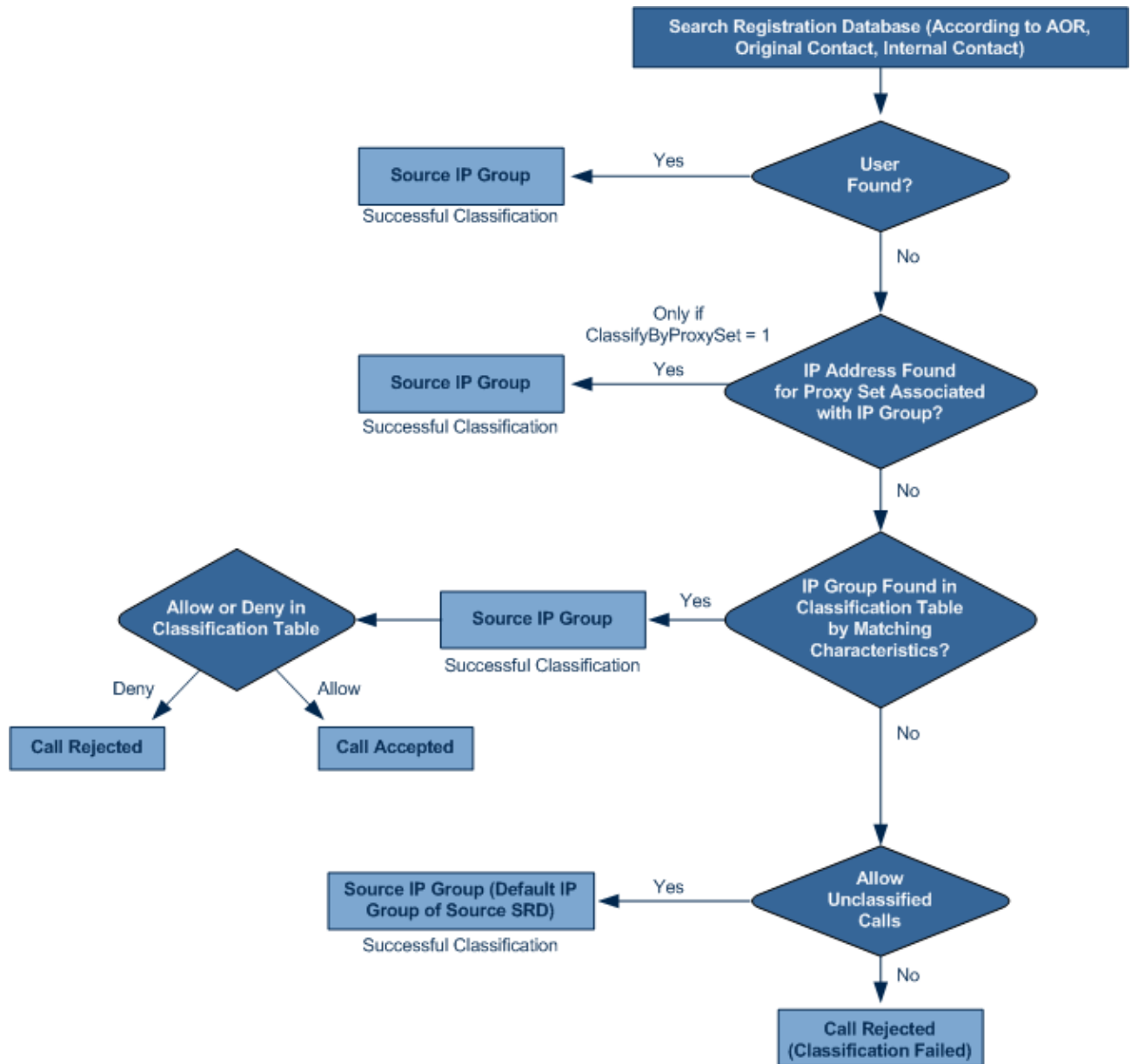
1. The device checks on which SIP listening port (e.g., 5061) the incoming SIP dialog request arrived and the SIP Interface which is configured with this port (in the SIP Interface table).
2. The device checks the SRD that is associated with this SIP Interface (in the SIP Interface table) and then classifies the SIP dialog with the first IP Group that is associated with this SRD. For example, if IP Groups 3 and 4 use the same SRD, the device classifies the call to IP Group 3.



Note: If classification for a SIP request fails and the device is configured to reject unclassified calls, the device can send a specific SIP response code per SIP interface, configured by the 'Classification Failure Response Type' parameter in the SIP Interface table (see 'Configuring SIP Interface Table' on page 188).

The flowchart below illustrates the classification process:

Figure 20-4: Classification Process (Identifying IP Group or Rejecting Call)



Notes:

- Incoming REGISTER messages are saved in the device's registration database and sent to a destination only if they are associated with a source User-type IP Group.
- The Classification table can also be configured using the table ini file parameter, Classification.



The Classification table provides two configuration areas:

- Matching characteristics of incoming IP call, for example, source IP address.
- Operation - classifies call to an IP Group.

If the incoming call matches the characteristics of a rule, then the call is classified to the IP Group configured for that rule.

➤ **To configure classification rules:**

1. Open the Classification Table page (**Configuration** tab > **VoIP** menu > **SBC** > **Routing SBC** > **Classification Table**).
2. Click the **Add** button; the following appears:

Figure 20-5: Classification Table Page

Add Record	
Index	0
Source SRD ID	None
Source IP Address	
Source Port	0
Source Transport Type	ANY
Source Username Prefix	*
Source Host	*
Destination Username Prefix	*
Destination Host	*
Message Condition	None
Source IP Group ID	None
Action Type	Allow
<input type="button" value="Submit"/> <input type="button" value="Cancel"/>	

3. Configure the rule as required. For a description of the parameters, see the table below.
4. Click **Submit** to apply your changes.

Table 20-2: Classification Table Parameters

Parameter	Description
Index	Defines the index number of the table row entry.
Matching Characteristics	
Source SRD ID [Classification_SrcSRDID]	<p>Defines the SRD ID of the incoming SIP dialog. The default is -1 (i.e., no SRD is assigned).</p> <p>Notes:</p> <ul style="list-style-type: none"> ▪ The SRDs are configured in the SRD table (see 'Configuring SRD Table' on page 185). ▪ The SRDs are also associated with a port number as defined by the SIP Interface used by the SRD (see 'Configuring SIP Interface Table' on page 188).
Source IP Address [Classification_SrcAddress]	<p>Defines the source IP address (in dotted-decimal notation) of the incoming SIP dialog.</p> <p>Notes:</p> <ul style="list-style-type: none"> ▪ If this parameter is not configured or is configured as an asterisk (*), then any source IP address is accepted.

Parameter	Description
	<ul style="list-style-type: none"> The IP address can include the "x" wildcard to represent single digits. For example: 10.8.8.xx represents all the addresses between 10.8.8.10 to 10.8.8.99. The IP address can include the asterisk (*) wildcard to represent any number between 0 and 255. For example, 10.8.8.* represents all addresses between 10.8.8.0 and 10.8.8.255.
Source Port [Classification_SrcPort]	Defines the source port number of the incoming SIP dialog.
Source Transport Type [Classification_SrcTransportType]	Defines the source transport type (UDP, TCP, or TLS) of the incoming SIP dialog.
Source Username Prefix [Classification_SrcUsernamePrefix]	<p>Defines the prefix of the source URI user part of the incoming SIP dialog. The URI is typically located in the SIP From header. However, you can configure the SIP header from where the device obtains the source URI. This is done in the IP Group table, using the 'Source URI Input' parameter. For more information on how the device obtains this URI, see 'SIP Dialog Initiation Process' on page 236.</p> <p>Notes:</p> <ul style="list-style-type: none"> For REGISTER requests, the source URL is obtained from the To header. The prefix can be a single digit or a range of digits. For available notations, see 'Dialing Plan Notation for Routing and Manipulation' on page 435.
Source Host [Classification_SrcHost]	<p>Defines the prefix of the source URI host name. The URI is typically located in the SIP From header. However, you can configure the SIP header from where the device obtains the source URI. This is done in the IP Group table, using the 'Source URI Input' parameter. For more information on how the device obtains this URI, see 'SIP Dialog Initiation Process' on page 236.</p> <p>If this routing rule is not required, leave the field empty. The asterisk (*) symbol can be used to depict any source host prefix.</p> <p>Note: For REGISTER requests, the source URL is obtained from the To header.</p>
Destination Username Prefix [Classification_DestUsernamePrefix]	<p>Defines the prefix of the destination Request-URI user part of the incoming SIP dialog.</p> <p>Note: The prefix can be a single digit or a range of digits. For available notations, see 'Dialing Plan Notation for Routing and Manipulation' on page 435.</p>
Destination Host [Classification_DestHost]	Defines the prefix of the destination Request-URI host name of the incoming SIP dialog request. If this routing rule is not required, leave the field empty. The asterisk (*) symbol can be used to depict any destination host prefix.
Message Condition [Classification_MessageCondition]	<p>Assigns a Condition rule which can also be used to classify the incoming SIP dialog.</p> <p>Note: Condition rules are configured in the Condition Table (see 'Configuring Condition Rules' on page 278).</p>
Operation Rule	

Parameter	Description
Source IP Group ID [Classification_SrcIPGroupID]	<p>Defines an IP Group to which the incoming SIP dialog request is assigned if this SIP dialog matches the matching rule.</p> <p>The default is -1 (i.e., no IP Group is assigned).</p> <p>Notes:</p> <ul style="list-style-type: none"> The IP Group must be associated with the selected SRD. The IP Group is used for SBC routing and manipulations. To define IP Groups, see 'Configuring IP Groups' on page 191.
Action Type [Classification_ActionType]	<p>Defines a whitelist or blacklist for incoming SIP dialog requests that match the characteristics of the classification rule.</p> <ul style="list-style-type: none"> [0] Deny = Blocks incoming SIP dialogs that match the characteristics of the Classification rule (blacklist). [1] Allow = Allows incoming SIP dialogs that match the characteristics of the Classification rule (whitelist) and assigns it to the associated IP Group. (default)

20.4.1.1 Classification Based on URI of Selected Header Example

The following example describes how to configure classification of incoming calls to IP Groups, based on source URI in a specific SIP header.

This example assumes the following incoming INVITE message:

```
INVITE sip:8000@10.33.4.226 SIP/2.0
Via: SIP/2.0/UDP 10.33.4.226;branch=z9hG4bKVEBTDHSAHSUYRTEXEDEGJY
From: <sip:100@10.33.4.226>;tag=YSQQKXXREVDPYPTNFMWG
To: <sip:8000@10.33.4.226>
Call-ID: FKPNOYRNKROIMEGBSSKS@10.33.4.226
CSeq: 1 INVITE
Contact: <sip:100@10.33.4.226>
Route: <sip:2000@10.10.10.10>,<sip:300@10.10.10.30>
Supported: em,100rel,timer,replaces
P-Called-Party-ID: <sip:1111@10.33.38.1>
User-Agent: Sip Message Generator V1.0.0.5
Content-Length: 0
```

1. In the Classification table, add the following classification rules:

Index	Source Username Prefix	Destination Username Prefix	Destination Host	Source IP Group ID
0	333	-	-	1
1	1111	2000	10.10.10.10	2

2. In the IP Group table, add the following IP Groups:

Index	Source URI Input	Destination URI Input
1	-	-
2	P-Called-Party-ID	Route

In this example, a match exists only for Classification Rule #1. This is because the source (1111) and destination (2000) username prefixes match those in the INVITE's P-Called-Party-ID header (i.e., "<sip:1111@10.33.38.1>") and Route header (i.e., "<sip:2000@10.10.10.10>"), respectively. These SIP headers were determined in IP Group ID 2.

20.4.2 Configuring Condition Rules

Condition rules define special conditions for the incoming SIP messages. Condition rules are configured using the same syntax as that used for message conditions in the Message Manipulations table (see Configuring SIP Message Manipulation on page 208).

Condition rules are used if assigned to any of the following:

- Classification rules in the Classification table (see Configuring Classification Rules on page 272). This enables you to use SIP message conditions as additional matching criteria for classifying incoming SIP dialogs to IP Groups, thereby increasing the strictness of the classification process.
- IP-to-IP routing rules in the IP-to-IP Routing table (see Configuring SBC IP-to-IP Routing on page 279): This enables you to use SIP message conditions as additional matching criteria for selecting the routing rule.

You can define simple Condition rules, for example, "header.to.host contains company" or complex rules using the "AND" or "OR" Boolean operands. You can also use regular expressions (regex), for example:

- "body.sdp regex pcmu" can be used to enable routing based on the offered codec (G.711 Mu) in the incoming SDP message.
- "body.sdp regex (AVP[0-9]||s)*s8[s]||n)" can be used to enable routing based on payload type 8 in the incoming SDP message.



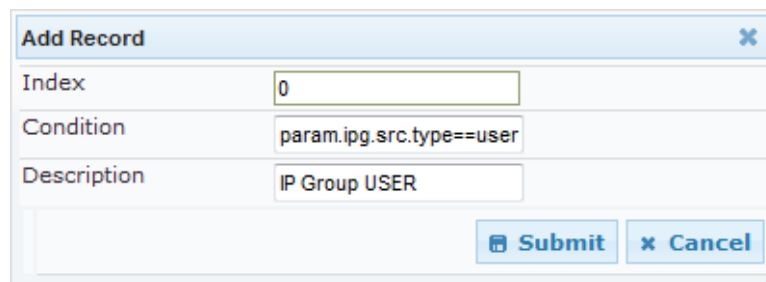
Notes:

- For a detailed description of the syntax for configuring SIP message manipulation rules, refer to *SIP Message Manipulations Quick Reference Guide*.
- The Condition table can also be configured using the table ini file parameter, ConditionTable.

➤ To configure Condition rules:

1. Open the Condition Table page (**Configuration** tab > **VoIP** menu > **SBC** > **Routing SBC** > **Condition Table**).
2. Click the **Add** button; the following dialog box appears:

Figure 20-6: Condition Table Page - Add Record Dialog Box



Add Record	
Index	0
Condition	param.ipg.src.type==user
Description	IP Group USER
<input type="button" value="Submit"/> <input type="button" value="Cancel"/>	

3. Configure the rule as required. For a description of the parameters, see the table below.
4. Click **Submit**.

The figure below shows an example of the Condition table configured with the following rules:

- **Index 1:** Incoming SIP dialog that is classified as belonging to a User-type IP Group.
- **Index 2:** Incoming SIP dialog with a SIP Via header.
- **Index 3:** Incoming SIP dialog with 101 as the user part in the SIP From header.

Figure 20-7: Condition Table Page

Index	Condition	Description
0	param.ipg.src.type==user	IP Group USER
1	header.via.exists	Includes SIP Via header
2	header.from.url.user=='101'	101 user part of From header

Table 20-3: Condition Table Parameters Description

Parameter	Description
Condition [ConditionTable_Condition]	Defines the Condition rule of the SIP message. The valid value is a string. Note: User and host parts must be enclosed in single quotes.
Description [ConditionTable_Description]	Defines a brief description of the Condition rule.

20.4.3 Configuring SBC IP-to-IP Routing

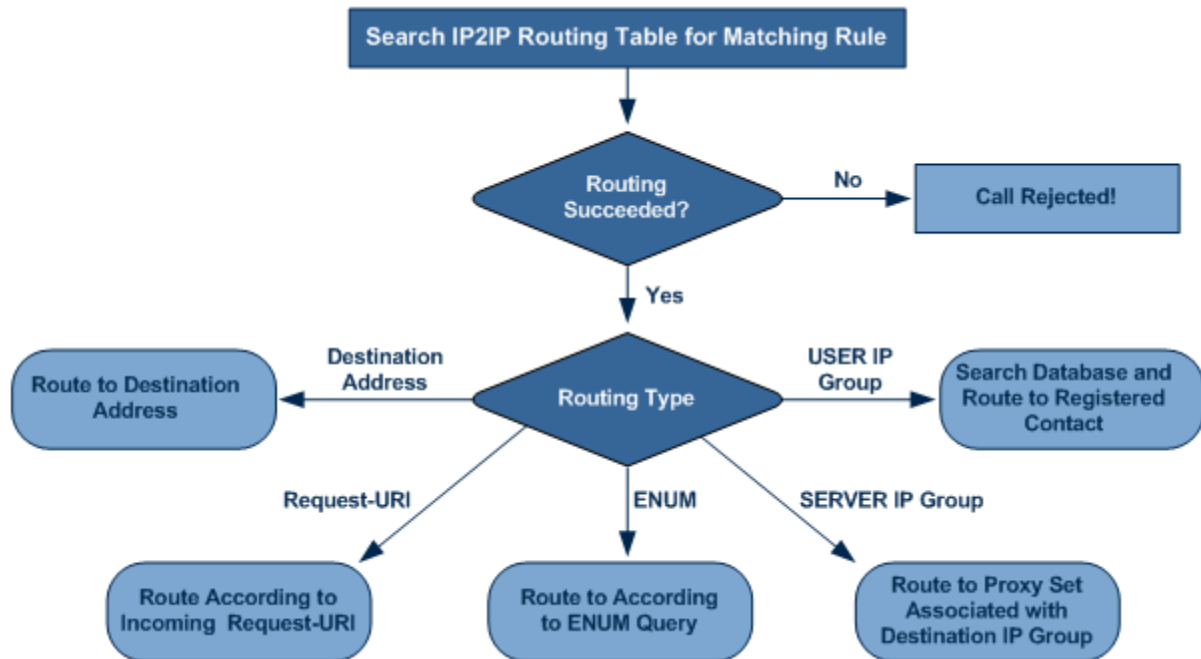
The IP-to-IP Routing table enables you to configure up to 1,000 SBC IP-to-IP routing rules. This table provides enhanced IP-to-IP call routing capabilities for routing received SIP dialog messages (e.g., INVITE) to a destination IP address. The SIP message is routed according to a routing rule whose configured input characteristics (e.g., Source IP Group) match the incoming SIP message. If the characteristics of an incoming call does not match the first rule, the call characteristics is then compared to those of the second rule, and so on until a matching rule is located. If no rule is matched, the call is rejected.

The call can be routed to one of the following IP destinations:

- IP Group - the destination is the address configured for the Proxy Set associated with the IP Group (allows redundancy/load balancing).
- Specified destination address (can be based on IP address, host name, port, transport type, and/or SRD). Routing to a host name can be resolved using NAPTR/SRV/A-Record.
- Request-URI of incoming SIP dialog initiating requests.
- ENUM query.
- Hunt Group - used for call survivability (see 'Call Survivability for Call Centers' on page 262).
- IP address (in dotted-decimal notation or FQDN - NAPTR/SRV/A-Record resolutions) according to a specified Dial Plan index listed in the loaded Dial Plan file.

- LDAP server or LDAP query result. For more information on LDAP-based routing, see 'Routing Based on LDAP Active Directory Queries' on page 167.

Figure 20-8: IP-to-IP Routing Types



For all destination types listed above except destination IP Group, the IP Group can optionally be itself, configured to provide the destination SRD and/or IP Profile. If neither destination SRD nor destination IP Group is defined, the destination SRD is the source SRD and the destination IP Group is its default IP Group.

The IP-to-IP Routing table also provides the following features:

- **Alternative routing or load balancing:** In addition to the alternative routing/load balancing provided by the Proxy Set associated with the destination IP Group, the table allows the configuration of alternative routes whereby if a route fails, the next adjacent (below) rule in the table that is configured as 'Alt Route Ignore/Consider Inputs' are used. The alternative routes rules can be set to enforce the input matching criteria or to ignore any matching criteria. Alternative routing occurs upon one of the following conditions:
 - A request sent by the device is responded with one of the following:
 - ◆ SIP response code (i.e., 4xx, 5xx, and 6xx SIP responses) configured in the SBC Alternative Routing Reasons table (see 'Configuring Alternative Routing Reasons' on page 286).
 - ◆ SIP 408 Timeout or no response (after timeout).
 - The DNS resolution includes IP addresses that the device has yet to try (for the current call).

Messages are re-routed with the same SIP Call-ID and CSeq header fields (increased by 1).

- **Re-routing of SIP requests:** This table enables you to configure "re-routing" rules of requests (e.g., INVITEs) that the device sends upon receipt of SIP 3xx responses or REFER messages. These rules are configured for destinations that do not support receipt of 3xx or REFER and where the device handles the requests locally (instead of forwarding the 3xx or REFER to the destination).
- **Least cost routing (LCR):** If the LCR feature is enabled, the device searches the routing table for matching routing rules and then selects the one with the lowest call cost. The call cost of the routing rule is done by assigning it a Cost Group. For configuring Cost Groups, see 'Least Cost Routing' on page 175. If two routing rules

have identical costs, then the rule appearing higher up in the table (i.e., first-matched rule) is used. If a selected route is unavailable, the device uses the next least-cost routing rule. However, even if a matched rule is not assigned a Cost Group, the device can select it as the preferred route over other matched routing rules with Cost Groups, according to the settings of the LCR parameter, LCRDefaultCost (see 'Enabling LCR and Configuring Default LCR' on page 178).



Note: The IP-to-IP Routing table can also be configured using the table ini file parameter, IP2IPRouting (see 'SBC Parameters' on page 512).

➤ **To configure SBC IP-to-IP routing rules:**

1. Open the IP-to-IP Routing Table page (**Configuration** tab > **VoIP** menu > **SBC** submenu > **Routing SBC** submenu > **IP to IP Routing Table**).
2. Click the **Add** button; the following dialog box appears:

Figure 20-9: IP-to-IP Routing Table - Add Record Dialog Box

Add Record	
Index	0
Source IP Group ID	-1
Source Username Prefix	*
Source Host	*
Destination Username Prefix	*
Destination Host	*
Request Type	All
Message Condition	None
ReRoute IP Group ID	-1
Call Trigger	Any
Destination Type	IP Group
Destination IP Group ID	-1
Destination SRD ID	None
Destination Address	
Destination Port	0
Destination Transport Type	
Alternative Route Options	Route Row
Cost Group	None
<input type="button" value="Submit"/> <input type="button" value="Cancel"/>	

3. Configure the rule as required. For a description of the parameters, see the table below.
4. Click **Submit**.

Table 20-4: IP-to-IP Routing Table Parameters Description

Parameter	Description
Matching Characteristics	
Source IP Group ID [IP2IPRouting_SrcIPGroupID]	Selects the IP Group from where the IP-to-IP call originated. Typically, the IP Group of an incoming SIP dialog is determined (or classified) using the 'Classification' table (see Configuring Classification Rules on page 272). If not used (i.e., any IP Group), simply leave the field empty. The default is -1.
Source Username Prefix [IP2IPRouting_SrcUsernamePrefix]	Defines the prefix of the user part of the incoming SIP dialog's source URI (usually the From URI). You can use special notations for denoting the prefix. For example, to denote any prefix, use the asterisk (*) symbol; to denote calls without a user part in the URI, use the \$ sign. For available notations, see 'Dialing Plan Notation for Routing and Manipulation' on page 435. The default is * (i.e., any prefix).
Source Host [IP2IPRouting_SrcHost]	Defines the host part of the incoming SIP dialog's source URI (usually the From URI). If this rule is not required, leave the field empty. To denote any host name, use the asterisk (*) symbol (default).
Destination Username Prefix [IP2IPRouting_DestUsernamePrefix]	Defines the prefix of the incoming SIP dialog's destination URI (usually the Request URI) user part. You can use special notations for denoting the prefix. For example, to denote any prefix, use the asterisk (*) symbol; to denote calls without a user part in the URI, use the \$ sign. For available notations, see 'Dialing Plan Notation for Routing and Manipulation' on page 435. The default is * (i.e., any prefix).
Destination Host [IP2IPRouting_DestHost]	Defines the host part of the incoming SIP dialog's destination URI (usually the Request-URI). If this rule is not required, leave the field empty. The asterisk (*) symbol (default) can be used to denote any destination host.
Request Type [IP2IPRouting_RequestType]	Defines the SIP dialog request type of the incoming SIP dialog. <ul style="list-style-type: none"> ▪ [0] All (default) ▪ [1] INVITE ▪ [2] REGISTER ▪ [3] SUBSCRIBE ▪ [4] INVITE and REGISTER ▪ [5] INVITE and SUBSCRIBE ▪ [6] OPTIONS
Message Condition [IP2IPRouting_MessageCondition]	Selects a Message Condition rule. To configure Message Condition rules, see 'Configuring Condition Rules' on page 278.
ReRoute IP Group ID [IP2IPRouting_ReRouteIPGroupID]	Defines the IP Group that initiated (sent) the SIP redirect response (e.g., 3xx) or REFER message. This field is typically used for re-routing requests (e.g., INVITEs) when interworking is required for SIP 3xx redirect responses or REFER messages (for more information, see 'Interworking SIP 3xx Redirect Responses' on page 253 and 'Interworking

Parameter	Description
	<p>SIP REFER Messages' on page 255, respectively). This parameter functions together with the 'Call Trigger' field (see below).</p> <p>The default is -1 (i.e., not configured).</p>
Call Trigger [IP2IPRouting_Trigger]	<p>Defines the reason (i.e, trigger) for re-routing the SIP request:</p> <ul style="list-style-type: none"> ▪ [0] Any = (Default) This routing rule is used for all scenarios (re-routes and non-re-routes). ▪ [1] 3xx = Re-routes the request if it was triggered as a result of a SIP 3xx response. ▪ [2] REFER = Re-routes the INVITE if it was triggered as a result of a REFER request. ▪ [3] 3xx or REFER = Applies to options [1] and [2]. ▪ [4] Initial only = This routing rule is used for regular requests that the device forwards to the destination. This rule is not used for re-routing of requests triggered by the receipt of REFER or 3xx.
Operation Routing Rule	
Destination Type [IP2IPRouting_DestType]	<p>Determines the destination type to which the outgoing SIP dialog is sent.</p> <ul style="list-style-type: none"> ▪ [0] IP Group = (Default) The SIP dialog is sent to the IP Group's Proxy Set (SERVER-type IP Group) or registered contact from the database (if USER-type IP Group). ▪ [1] Dest Address = The SIP dialog is sent to the address configured in the following fields: 'Destination SRD ID', 'Destination Address', 'Destination Port', and 'Destination Transport Type'. ▪ [2] Request URI = The SIP dialog is sent to the address indicated in the incoming Request-URI. If the fields 'Destination Port' and 'Destination Transport Type' are configured, the incoming Request-URI parameters are overridden and these fields take precedence. ▪ [3] ENUM = An ENUM query is sent to include the destination address. If the fields 'Destination Port' and 'Destination Transport Type' are configured, the incoming Request-URI parameters are overridden and these fields take precedence. ▪ [4] Hunt Group = Used for call center survivability. For more information, see 'Call Survivability for Call Centers' on page 262. ▪ [5] Dial Plan = The IP destination is determined by a Dial Plan index of the loaded Dial Plan file. The syntax of the Dial Plan index in the Dial Plan file is as follows: <destination / called prefix number>,0,<IP destination> Note that the second parameter "0" is ignored. An example of a configured Dial Plan (# 6) in the Dial Plan file is shown below: <pre>[PLAN6] 200,0,10.33.8.52 ; called prefix 200 is routed to destination 10.33.8.52 201,0,10.33.8.52</pre>

Parameter	Description
	<p>300,0,itsp.com ; called prefix 300 is routed to destination itsp.com</p> <p>Once the Dial Plan is defined, you need to assign it (0 to 7) to the routing rule as the destination in the 'Destination Address' parameter, where "0" denotes [PLAN1], "1" denotes [PLAN2], and so on.</p> <ul style="list-style-type: none"> [7] LDAP = LDAP-based routing.
Destination IP Group ID [IP2IPRouting_DestIPGroupID]	<p>Defines the IP Group ID to where you want to route the call. The SIP dialog messages are sent to the IP address defined for the Proxy Set associated with this IP Group. If you select an IP Group, it is unnecessary to configure a destination IP address (in the 'Destination Address' field). However, if both parameters are configured, then the IP Group takes precedence.</p> <p>If the destination IP Group is of USER type, the device searches for a match between the Request-URI (of the received SIP dialog) to an AOR registration record in the device's database. The SIP dialog is then sent to the IP address of the registered contact.</p> <p>The default is -1.</p> <p>Notes:</p> <ul style="list-style-type: none"> This parameter is only relevant if the parameter 'Destination Type' is set to 'IP Group'. However, regardless of the settings of the parameter 'Destination Type', the IP Group is still used - only for determining the IP Profile or outgoing SRD. If neither IP Group nor SRD are defined in this table, the destination SRD is determined according to the source SRD associated with the Source IP Group (configured in the IP Group table, see 'Configuring IP Groups' on page 191). If this table does not define an IP Group but only an SRD, then the first IP Group associated with this SRD (in the IP Group table) is used. If the selected destination IP Group ID is type SERVER, the request is routed according to the IP Group addresses. If the selected destination IP Group ID is type USER, the request is routed according to the IP Group specific database (i.e., only to registered users of the selected database). If the selected destination IP Group ID is ANY USER ([-2]), the request is routed according to the general database (i.e., any matching registered user).
Destination SRD ID [IP2IPRouting_DestSRDID]	<p>Defines the SRD ID. The default is None.</p> <p>Note: The destination IP Group must belong to the destination SRD, if both are configured in this table.</p>
Destination Address [IP2IPRouting_DestAddress]	<p>Defines the destination to where the call is sent. This can be an IP address or a domain name (e.g., domain.com). If ENUM-based routing is used (i.e., the 'Destination Type' parameter is set to ENUM) this parameter defines the IP address or domain name (FQDN) of the ENUM service, for example, e164.arpa, e164.customer.net, or NREnum.net. The device sends the ENUM query containing the</p>

Parameter	Description
	<p>destination phone number to an external DNS server, configured in the Multiple Interface table. The ENUM reply includes a SIP URI (user@host) which is used as the destination Request-URI in this routing table.</p> <p>Notes:</p> <ul style="list-style-type: none"> This parameter is applicable only if the 'Destination Type' parameter is set to Dest Address [1] or ENUM [3]. When using domain names, enter a DNS server IP address or alternatively, define these names in the Internal DNS table (see Configuring the Internal SRV Table on page 116). To terminate SIP OPTIONS messages at the device (i.e., to handle them locally), set this parameter to "internal".
Destination Port [IP2IPRouting_DestPort]	Defines the destination port to where the call is sent.
Destination Transport Type [IP2IPRouting_DestTransportType]	<p>Defines the transport layer type for sending the call:</p> <ul style="list-style-type: none"> [-1] Not Configured (default) [0] UDP [1] TCP [2] TLS <p>Note: When this parameter is set to -1, the transport type is determined by the parameter SIPTransportType.</p>
Alternative Route Options [IP2IPRouting_AltRouteOptions]	<p>Determines whether this routing rule is the main routing rule or an alternative routing rule (to the rule defined directly above it in the table).</p> <ul style="list-style-type: none"> [0] Route Row (default) = Main routing rule - the device first attempts to route the call to this route if the incoming SIP dialog's input characteristics matches this rule. [1] Alt Route Ignore Inputs = If the call cannot be routed to the main route (Route Row), the call is routed to this alternative route regardless of the incoming SIP dialog's input characteristics. [2] Alt Route Consider Inputs = If the call cannot be routed to the main route (Route Row), the call is routed to this alternative route only if the incoming SIP dialog matches this routing rule's input characteristics. <p>Notes:</p> <ul style="list-style-type: none"> The alternative routing entry ([1] or [2]) must be defined in the next consecutive table entry index to the Route Row entry (i.e., directly below it). For example, if Index 4 is configured as a Route Row, Index 5 must be configured as the alternative route. For IP-to-IP alternative routing, configure alternative routing reasons upon receipt of 4xx, 5xx, and 6xx SIP responses (see Configuring Alternative Routing Reasons on page 286). However, if no response, ICMP, or a SIP 408 response is received, the device attempts to use the alternative route even if no entries are configured in the 'SBC Alternative Routing Reasons' table. Multiple alternative route entries can be configured (e.g.,

Parameter	Description
	Index 1 is the main route - Route Row - and indices 2 through 4 are configured as alternative routes).
Cost Group [IP2IPRouting_CostGroup]	Assigns a Cost Group to the routing rule for determining the cost of the call. To configure Cost Groups, see 'Configuring Cost Groups' on page 179. By default, no Cost Group is assigned to the rule.

20.4.4 Configuring Alternative Routing Reasons

The SBC Alternative Routing Reasons page allows you to define up to five different call release (termination) reasons for call releases. If a call is released as a result of one of these reasons provided in SIP 4xx, 5xx, and 6xx response codes, the device attempts to locate an alternative route for the call. The call release reason type can be configured, for example, when there is no response to an INVITE message (after INVITE re-transmissions), where the device issues an internal 408 'No Response' implicit release reason.

Release reasons can also be configured to indicate that a route for an SRD or IP Group has reached its call admission control limit (i.e., maximum concurrent calls and/or call rate), as set in the Admission Control table (see 'Configuring Admission Control' on page 268). In such a scenario, an alternative route configured in the IP-to-IP Routing table can be used.

Alternative routing rules are configured in the IP-to-IP Routing table where the 'Alternative Route Options' parameter is set to **Alt Route Ignore Inputs** or **Alt Route Consider Inputs**. For more information, see 'Configuring SBC IP-to-IP Routing' on page 279.

Notes:

- Alternative routing occurs even if this table is not configured upon scenarios where no response, ICMP, or a SIP 408 response is received.
- SIP requests pertaining to an SRD or IP Group that reach the call limit (maximum concurrent calls and/or call rate) as defined in the Call Admission table are sent to an alternative route if configured in the IP-to-IP Routing table for the SRD or IP Group. If no alternative routing rule is located, the device automatically rejects the SIP request with a SIP 486 "Busy Here" response.
- The SBC Alternative Routing Reasons table can also be configured using the table ini file parameter, SBCAlternativeRoutingReasons.



➤ To configure SIP reason codes for alternative IP routing:

1. Open the SBC Alternative Routing Reasons page (**Configuration** tab > **VoIP** menu > **SBC** > **Routing SBC** > **Alternative Routing Reasons**).

Figure 20-10: Alternative Routing Reasons Page

SBC Alternative Routing Reasons	
Reason 1	401
Reason 2	
Reason 3	
Reason 4	
Reason 5	

2. Configure different call failure reasons that invoke alternative routing.
3. Click **Submit** to apply your changes.

20.5 SBC Manipulations

This section describes the configuration of the manipulation rules for the SBC application.

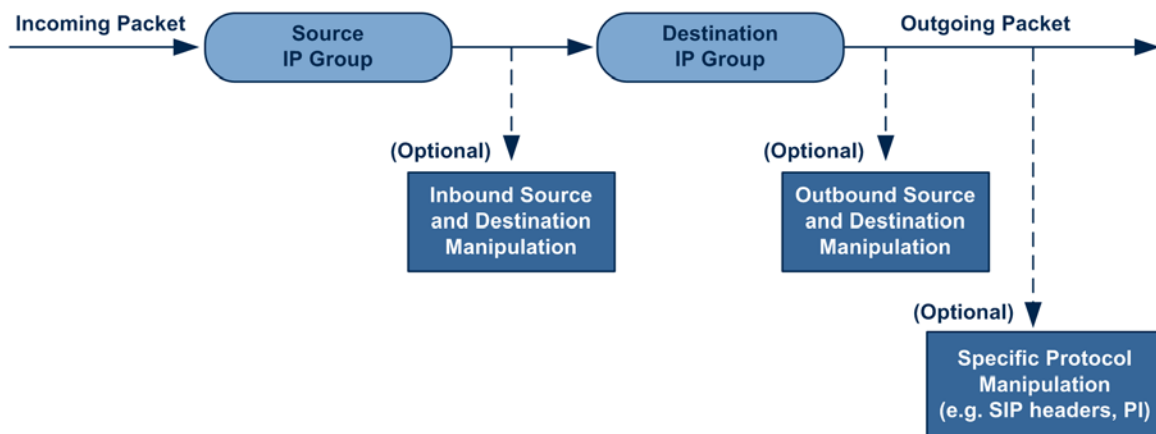


Note: For additional manipulation features, see the following:

- 'Configuring SIP Message Policy Rules'.
- 'Configuring SIP Message Manipulation' on page 208.

The device supports SIP URI user part (source and destination) manipulations for inbound and outbound routing. These manipulations can be applied to a source IP group, source and destination host and user prefixes, and/or user-defined SIP request (e.g., INVITE, OPTIONS, SUBSCRIBE, and/or REGISTER). Since outbound manipulations are performed after routing, the outbound manipulation rule matching can also be done by destination IP Group. Manipulated destination user and host are performed on the following SIP headers: Request-URI, To, and Remote-Party-ID (if exists). Manipulated source user and host are performed on the following SIP headers: From, P-Asserted (if exists), P-Preferred (if exists), and Remote-Party-ID (if exists).

Figure 20-11: SIP URI Manipulation in IP-to-IP Routing



You can also restrict source user identity in outgoing SIP dialogs in the Outbound Manipulation table (using the column PrivacyRestrictionMode). The device identifies an incoming user as restricted if one of the following exists:

- From header user is 'anonymous'.
- P-Asserted-Identity and Privacy headers contain the value 'id'.

All restriction logic is done after the user number has been manipulated.

Host name (source and destination) manipulations are simply host name substitutions with the names defined for the source and destination IP Groups respectively (if any, in the IP Group table).

Below is an example of a call flow and consequent SIP URI manipulations:

■ **Incoming INVITE from LAN:**

```
INVITE sip:1000@10.2.2.3;user=phone;x=y;z=a SIP/2.0
Via: SIP/2.0/UDP 10.2.2.6;branch=z9hGLLLLLan
From: <sip:7000@10.2.2.6;user=phone;x=y;z=a>;tag=01LAN;parameter1
```



```
=abe
To: <sip:1000@10.2.2.3;user=phone>
Call-ID: USELLLAN@10.2.2.3
CSeq: 1 INVITE
Contact: <sip:7000@10.2.2.3>
Supported: em,100rel,timer,replaces
Allow: REGISTER,OPTIONS,INVITE,ACK,CANCEL,BYE,NOTIFY,PRACK
User-Agent: Sip Message Generator V1.0.0.5
Content-Type: application/sdp
Content-Length: 155
v=0
o=SMG 791285 795617 IN IP4 10.2.2.6
s=Phone-Call
c=IN IP4 10.2.2.6
t=0 0
m=audio 6000 RTP/AVP 8
a=rtpmap:8 pcma/8000
a=sendrecv
a=ptime:20
```

■ **Outgoing INVITE to WAN:**

```
INVITE sip: 9721000@ITSP;user=phone;x=y;z=a SIP/2.0
Via: SIP/2.0/UDP 212.179.1.12;branch=z9hGwWan
From:
<sip:97000@IP_PBX;user=phone;x=y;z=a>;tag=OWan;parameter1=abe
To: <sip: 9721000@ ITSP;user=phone>
Call-ID: USEVWWAN@212.179.1.12
CSeq: 38 INVITE
Contact: <sip:7000@212.179.1.12>
Supported: em,100rel,timer,replaces
Allow:
REGISTER,OPTIONS,INVITE,ACK,CANCEL,BYE,NOTIFY,PRACK,REFER
User-Agent: Sip Message Generator V1.0.0.5
Content-Type: application/sdp
Content-Length: 155
v=0
o=SMG 5 9 IN IP4 212.179.1.11
s=Phone-Call
c=IN IP4 212.179.1.11
t=0 0
m=audio 8000 RTP/AVP 8
a=rtpmap:8 pcma/8000
a=sendrecv
a=ptime:20
```

The SIP message manipulations in the example above (contributing to typical topology hiding) are as follows:

■ Inbound source SIP URI user name from "7000" to "97000":

```
From:<sip:7000@10.2.2.6;user=phone;x=y;z=a>;tag=01LAN;parameter1=abe
```

to

```
From:
<sip:97000@IP_PBX;user=phone;x=y;z=a>;tag=OWan;parameter1=abe
```

■ Source IP Group name (i.e., SIP URI host name) from "10.2.2.6" to "IP_PBX":

```
From:<sip:7000@10.2.2.6;user=phone;x=y;z=a>;tag=01LAN;parameter1=abe
```

to

```
From:
<sip:97000@IP_PBX;user=phone;x=y;z=a>;tag=OWan;parameter1=abe
```


- Inbound destination SIP URI user name from "1000" to 9721000:

```
INVITE sip:1000@10.2.2.3;user=phone;x=y;z=a SIP/2.0
To: <sip:1000@10.2.2.3;user=phone>
to
```

```
INVITE sip:9721000@ITSP;user=phone;x=y;z=a SIP/2.0
To: <sip:9721000@ITSP;user=phone>
```

- Destination IP Group name (SIP URI host name) from "10.2.2.3" to "ITSP":

```
INVITE sip:1000@10.2.2.3;user=phone;x=y;z=a SIP/2.0
To: <sip:1000@10.2.2.3;user=phone>
```

```
to
INVITE sip:9721000@ITSP;user=phone;x=y;z=a SIP/2.0
To: <sip:9721000@ITSP;user=phone>
```

20.5.1 Configuring IP-to-IP Inbound Manipulations

The IP to IP Inbound Manipulation table allows you to configure up to 100 manipulation rules for manipulating the source or destination SIP URI user part of inbound SIP dialog requests. You can apply these manipulations to different SIP dialog message types (e.g., INVITE or REGISTER) and SIP headers as follows:

- Manipulated destination URI user part are done on the following SIP headers: Request-URI, To, and Remote-Party-ID (if exists)
- Manipulated source URI user part are done on the following SIP headers: From, P-Asserted-Identity (if exists), P-Preferred-Identity (if exists), and Remote-Party-ID (if exists)

The IP to IP Inbound Manipulation table provides two configuration areas:

- Matching characteristics (*Rule*) of incoming call, for example, source host name.
- Manipulation operation (*Action*), for example, remove user-defined number of characters from the left of the SIP URI user part.

If the incoming call matches the characteristics of a rule, then its manipulation action is applied.



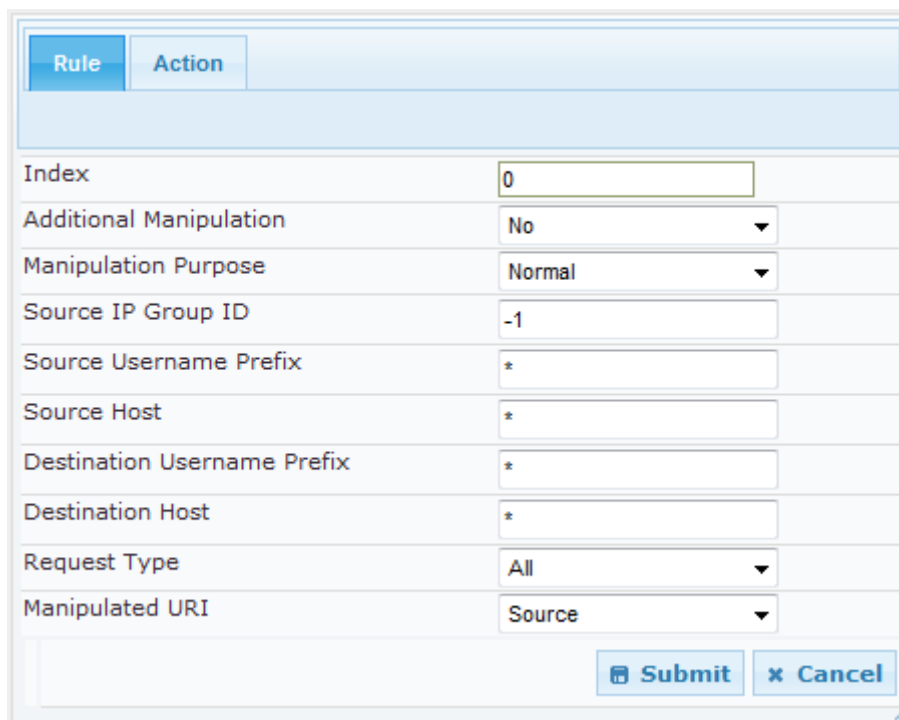
Notes:

- The IP Group table can be used to configure a host name that overwrites the received host name. This manipulation can be done for source or destination IP Groups (see 'Configuring IP Groups' on page 191).
- The IP to IP Inbound Manipulation table can also be configured using the table ini file parameter, IPInboundManipulation.

➤ To configure IP-to-IP inbound manipulation rules:

1. Open the IP to IP Inbound Manipulation page (**Configuration** tab > **VoIP** menu > **SBC** > **Manipulations SBC** > **IP to IP Inbound**).
2. Click the **Add** button; the following dialog box appears:

Figure 20-12: IP to IP Inbound Manipulation Page - Add Dialog Box



3. Configure a rule as required. For a description of the parameters, see the table below.
4. Click **Submit**.

Table 20-5: IP to IP Inbound Manipulation Parameters Description

Parameter	Description
Matching Characteristics (Rule)	
Additional Manipulation [IPInboundManipulation_IsAdditionalManipulation]	<p>Determines whether additional SIP URI user part manipulation is done for the table entry rule listed directly above it.</p> <ul style="list-style-type: none"> ▪ [0] No = (Default) Regular manipulation rule (not done in addition to the rule above it). ▪ [1] Yes = If the above row entry rule matched the call, consider this row entry as a match as well and perform the manipulation specified by this rule. <p>Note: Additional manipulation can only be done on a different SIP URI, source or destination, to the rule configured in the row above as configured by the 'Manipulated URI' parameter (see below).</p>
Manipulation Purpose [IPInboundManipulation_ManipulationPurpose]	<p>Defines the purpose of the manipulation:</p> <ul style="list-style-type: none"> ▪ [0] Normal = (Default) Inbound manipulations affect the routing input and source and/or destination number. ▪ [1] Routing input only = Inbound manipulations affect the routing input only, retaining the original source and destination number.

Parameter	Description
	<ul style="list-style-type: none"> [2] Shared Line = Used for the Shared-Line Appearance feature. This manipulation is for registration requests to change the destination number of the secondary extension numbers to the primary extension. For more information, see 'BroadSoft's Shared Phone Line Call Appearance for SBC Survivability' on page 261.
Source IP Group ID [IPInboundManipulation_SrcIpGroup]	<p>Defines the IP Group from where the incoming INVITE is received.</p> <p>For any IP Group, enter the value "-1".</p>
Source Username Prefix [IPInboundManipulation_SrcUsernamePrefix]	<p>Defines the prefix of the source SIP URI user name (usually in the From header).</p> <p>For any prefix, enter the asterisk "*" symbol (default).</p> <p>Note: The prefix can be a single digit or a range of digits. For available notations, see 'Dialing Plan Notation for Routing and Manipulation' on page 435.</p>
Source Host [IPInboundManipulation_SrcHost]	<p>Defines the source SIP URI host name - full name (usually in the From header). For any host name, enter the asterisk "*" symbol (default).</p>
Destination Username Prefix [IPInboundManipulation_DestUsernamePrefix]	<p>Defines the prefix of the destination SIP URI user name (usually in the Request-URI).</p> <p>For any prefix, enter the asterisk "*" symbol (default).</p> <p>Note: The prefix can be a single digit or a range of digits. For available notations, see 'Dialing Plan Notation for Routing and Manipulation' on page 435.</p>
Destination Host [IPInboundManipulation_DestHost]	<p>Defines the destination SIP URI host name - full name (usually in the Request URI).</p> <p>For any host name, enter the asterisk "*" symbol (default).</p>
Request Type [IPInboundManipulation_RequestType]	<p>Defines the SIP request type to which the manipulation rule is applied.</p> <ul style="list-style-type: none"> [0] All = (Default) All SIP messages. [1] INVITE = All SIP messages except REGISTER and SUBSCRIBE. [2] REGISTER = Only REGISTER messages. [3] SUBSCRIBE = Only SUBSCRIBE messages. [4] INVITE and REGISTER = All SIP messages except SUBSCRIBE. [5] INVITE and SUBSCRIBE = All SIP messages except REGISTER.
Manipulated URI [IPInboundManipulation_ManipulatedURI]	<p>Determines whether the source or destination SIP URI user part is manipulated.</p> <ul style="list-style-type: none"> [0] Source = (Default) Manipulation is done on the source SIP URI user part. [1] Destination = Manipulation is done on the destination SIP URI user part.
Operation Rule (Action)	
Remove From Left [IPInboundManipulation_RemoveFromLeft]	<p>Defines the number of digits to remove from the left of the user name prefix. For example, if you enter 3 and the user name is "john", the new user name is "n".</p>

Parameter	Description
Remove From Right [IPInboundManipulation_RemoveFromRight]	Defines the number of digits to remove from the right of the user name prefix. For example, if you enter 3 and the user name is "john", the new user name is "j". Note: If both 'Remove From Right' and 'Leave From Right' parameters are configured, the 'Remove From Right' setting is applied first.
Leave From Right [IPInboundManipulation_LeaveFromRight]	Defines the number of characters that you want retained from the right of the user name. Note: If both 'Remove From Right' and 'Leave From Right' parameters are configured, the 'Remove From Right' setting is applied first.
Prefix to Add [IPInboundManipulation_Prefix2Add]	Defines the number or string that you want added to the front of the user name. For example, if you enter 'user' and the user name is "john", the new user name is "userjohn".
Suffix to Add [IPInboundManipulation_Suffix2Add]	Defines the number or string that you want added to the end of the user name. For example, if you enter '01' and the user name is "john", the new user name is "john01".

20.5.2 Configuring IP-to-IP Outbound Manipulations

The IP to IP Outbound Manipulation page allows you to configure up to 100 manipulation rules for manipulating SIP URI user part (source and destination) of outbound SIP dialog requests. Manipulation rules in the table are located according to the source IP Group, and source and destination host and user prefixes and can be applied to a user-defined SIP request type (e.g., INVITE, OPTIONS, SUBSCRIBE, and /or REGISTER). However, since outbound manipulations are done only after routing, the outbound manipulation rule matching can also be done by destination IP Group.

- Manipulated destination URI user part are performed on the following SIP headers: Request URI, To, and Remote-Party-ID (if exists).
- Manipulated source URI user part are performed on the following SIP headers: From, P-Asserted (if exists), P-Preferred (if exists), and Remote-Party-ID (if exists).

The IP to IP Outbound Manipulation table provides two configuration areas:

- Matching characteristics (*Rule*) of incoming call, for example, source host name.
- Manipulation operation (*Action*), for example, remove user-defined number of characters from the left of the SIP URI user part.

If the incoming call matches the characteristics of a rule, then its manipulation action is applied.

**Notes:**

- Manipulated destination SIP URI user names are done on the following SIP headers: Request URI, To, and Remote-Party-ID (if exists).
- Manipulated source SIP URI user names are done on the following SIP headers: From, P-Asserted (if exists), P-Preferred (if exists), and Remote-Party-ID (if exists)
- SIP URI host name (source and destination) manipulations can also be configured in the IP Group table. These manipulations are simply host name substitutions with the names defined for the source and destination IP Groups respectively.
- The IP to IP Outbound Manipulation table can also be configured using the table ini file parameter, IPOutboundManipulation.

➤ **To configure IP-to-IP outbound manipulation rules:**

1. Open the IP to IP Outbound Manipulation page (**Configuration** tab > **VoIP** menu > **SBC > Manipulations SBC > IP to IP Outbound**).
2. Click the **Add** button; the following dialog box appears:

Figure 20-13: IP to IP Outbound Manipulation Page - Add Dialog Box

Rule	Action
Index	<input type="text" value="0"/>
Additional Manipulation	<input type="text" value="No"/>
Source IP Group ID	<input type="text" value="-1"/>
Destination IP Group ID	<input type="text" value="-1"/>
Source Username Prefix	<input type="text" value="*"/>
Source Host	<input type="text" value="*"/>
Destination Username Prefix	<input type="text" value="*"/>
Destination Host	<input type="text" value="*"/>
Request Type	<input type="text" value="All"/>
ReRoute IP Group ID	<input type="text" value="-1"/>
Call Trigger	<input type="text" value="Any"/>
Manipulated URI	<input type="text" value="Source"/>
<input type="button" value="Submit"/> <input type="button" value="Cancel"/>	

3. Configure a rule as required. For a description of the parameters, see the table below.
4. Click **Submit**.

Table 20-6: IP to IP Outbound Manipulation Table Parameters Description

Parameter	Description
Matching Characteristics (Rule)	
Additional Manipulation [IPOutboundManipulation_IsAdditionalManipulation]	<p>Determines whether additional SIP URI user part manipulation is done for the table entry rule listed directly above it.</p> <ul style="list-style-type: none"> [0] No = (Default) Regular manipulation rule - not done in addition to the rule above it. [1] Yes = If the previous table row entry rule matched the call, consider this row entry as a match as well and perform the manipulation specified by this rule. <p>Note: Additional manipulation can only be performed on a different SIP URI (either source or destination) to the rule configured in the row above (configured by the 'Manipulated URI' parameter).</p>
Source IP Group ID [IPOutboundManipulation_SrcIPGroupID]	<p>Defines the IP Group from where the INVITE is received.</p> <p>For any Source IP Group, enter the value -1.</p>
Destination IP Group ID [IPOutboundManipulation_DestIPGroupID]	<p>Defines the IP Group to where the INVITE is to be sent.</p> <p>For any Destination IP Group, enter the value -1.</p>
Source Username Prefix [IPOutboundManipulation_SrcUsernamePrefix]	<p>Defines the prefix of the source SIP URI user name (usually in the From header).</p> <p>For any prefix, enter the asterisk "*" symbol (default).</p> <p>Note: The prefix can be a single digit or a range of digits. For available notations, see 'Dialing Plan Notation for Routing and Manipulation' on page 435.</p>
Source Host [IPOutboundManipulation_SrcHost]	<p>Defines the source SIP URI host name - full name (usually in the From header).</p> <p>For any host name, enter the asterisk "*" symbol (default).</p>
Destination Username Prefix [IPOutboundManipulation_DestUsernamePrefix]	<p>Defines the prefix of the destination SIP URI user name (usually in the Request-URI).</p> <p>For any prefix, enter the asterisk "*" symbol (default).</p> <p>Note: The prefix can be a single digit or a range of digits. For available notations, see 'Dialing Plan Notation for Routing and Manipulation' on page 435.</p>
Destination Host [IPOutboundManipulation_DestHost]	<p>Defines the destination SIP URI host name - full name (usually in the Request URI).</p> <p>For any host name, enter the asterisk "*" symbol (default).</p>
Request Type [IPOutboundManipulation_RequestType]	<p>Defines the SIP request type to which the manipulation rule is applied.</p> <ul style="list-style-type: none"> [0] All = (Default) all SIP messages. [1] INVITE = All SIP messages except REGISTER and SUBSCRIBE. [2] REGISTER = Only SIP REGISTER messages. [3] SUBSCRIBE = Only SIP SUBSCRIBE messages. [4] INVITE and REGISTER = All SIP messages except SUBSCRIBE. [5] INVITE and SUBSCRIBE = All SIP messages except REGISTER.

Parameter	Description
ReRoute IP Group ID [IPOutboundManipulation_ReRouteIPGroupID]	<p>Defines the IP Group that initiated (sent) the SIP redirect response (e.g., 3xx) or REFER message. This parameter is typically used for re-routing requests (e.g., INVITEs) when interworking is required for SIP 3xx redirect responses or REFER messages.</p> <p>The default is -1 (i.e., not configured).</p> <p>Notes:</p> <ul style="list-style-type: none"> This parameter functions together with the 'Call Trigger' parameter (see below). For more information on interworking of SIP 3xx redirect responses or REFER messages, see 'Interworking SIP 3xx Redirect Responses' on page 253 and 'Interworking SIP REFER Messages' on page 255, respectively.
Call Trigger [IPOutboundManipulation_Trigger]	<p>Defines the reason (i.e, trigger) for the re-routing of the SIP request:</p> <ul style="list-style-type: none"> [0] Any = (Default) Re-routed for all scenarios (re-routes and non-re-routes). [1] 3xx = Re-routed if it triggered as a result of a SIP 3xx response. [2] REFER = Re-routed if it triggered as a result of a REFER request. [3] 3xx or REFER = Applies to options [1] and [2]. [4] Initial only = Regular requests that the device forwards to a destination. In other words, re-routing of requests triggered by the receipt of REFER or 3xx does not apply.
Manipulated URI [IPOutboundManipulation_IsAdditionalManipulation]	<p>Determines whether the source or destination SIP URI user part is manipulated.</p> <ul style="list-style-type: none"> [0] Source = (Default) Manipulation is done on the source SIP URI user part. [1] Destination = Manipulation is done on the destination SIP URI user part.
Operation Manipulation Rule (Action)	
Remove From Left [IPOutboundManipulation_RemoveFromLeft]	Defines the number of digits to remove from the left of the user name prefix. For example, if you enter 3 and the user name is "john", the new user name is "n".
Remove From Right [IPOutboundManipulation_RemoveFromRight]	Defines the number of digits to remove from the right of the user name prefix. For example, if you enter 3 and the user name is "john", the new user name is "j".
Leave From Right [IPOutboundManipulation_LeaveFromRight]	Defines the number of characters that you want retained from the right of the user name.
Prefix to Add [IPOutboundManipulation_Prefix2Add]	Defines the number or string that you want added to the front of the user name. For example, if you enter 'user' and the user name is "john", the new user name is "userjohn".
Suffix to Add [IPOutboundManipulation_Suffix2Add]	Defines the number or string that you want added to the end of the user name. For example, if you enter '01' and the user name is "john", the new user name is "john01".
Privacy Restriction Mode [IPOutboundManipulation_Privacy]	Determines user privacy handling (i.e., restricting source user

Parameter	Description
RestrictionMode]	<p>identity in outgoing SIP dialogs).</p> <ul style="list-style-type: none"> ▪ [0] Transparent = (Default) No intervention in SIP privacy. ▪ [1] Don't change privacy = The user identity remains the same as in the incoming SIP dialog. If a restricted number exists, the restricted presentation is normalized as follows: <ul style="list-style-type: none"> ✓ From URL header: anonymous@anonymous.invalid. ✓ If a P-Asserted-Identity header exists (either in the incoming SIP dialog or added by the device), a Privacy header is added with the value "id". ▪ [2] Restrict = The user identity is restricted (the restricted presentation is as mentioned above). ▪ [3] Remove Restriction = The device attempts to reveal the user identity by setting user values in the From header and removing the privacy "id" value if the Privacy header exists. <p>If the From header user is anonymous, the value is taken from the P-Preferred-Identity, P-Asserted-Identity, or Remote-Party-ID header (if exists).</p> <p>The device identifies an incoming user as restricted if one of the following exists:</p> <ul style="list-style-type: none"> ▪ From header user is anonymous. ▪ P-Asserted-Identity and Privacy headers contain the value "id". <p>Note: All restriction logic is performed after the user number has been manipulated.</p>

Part VI

Cloud Resilience Package Application

21 CRP Overview

The device's Cloud Resilience Package (CRP) application enhances cloud-based or hosted communications environments by ensuring survivability, high voice quality and security at enterprise branch offices and cloud service customer premises. CRP is designed to be deployed at customer sites and branches of:

- Cloud-based and hosted communications
- Cloud-based or hosted contact-center services
- Distributed PBX or unified communications deployments

The CRP application is based on the functionality of the SBC application, providing branch offices with call routing and survivability support similar to the SAS application. CRP is implemented in a network topology where the device is located at the branch office, routing calls between the branch users, and/or between the branch users and other users located elsewhere (at headquarters or other branch offices), through a hosted server (IP PBX) located at the Enterprise headquarters. The device maintains call continuity even if a failure occurs in communication with the hosted IP PBX. It does this by using its Call Survivability feature, enabling the branch users to call one another or make external calls through the device's PSTN gateway interface (if configured).

For cloud providers, CRP ensures uninterrupted communications in the event of lost connection with the cloud providers' control systems. For distributed enterprises and contact centers, CRP is an essential solution for enterprises deploying geographically distributed communications solutions or distributed call centers with many branch offices. CRP ensures the delivery of internal and external calls even when the connection with the centralized control servers is lost.

Table 7: Key Features

Survivability	Quality of Experience/Service	Security
<ul style="list-style-type: none"> ■ PSTN fallback* ■ WAN redundancy ■ Local mode ■ High availability* ■ Emergency calling (E911) 	<ul style="list-style-type: none"> ■ QoE monitoring ■ Call Admission Control ■ SLA fulfillment ■ SIP mediation ■ Media transcoding ■ Test call agent 	<ul style="list-style-type: none"> ■ Layer 3 to 7 protection ■ Media encryption ■ Call control encryption ■ NAT traversal ■ Topology hiding

One of the main advantages of CRP is that it enables quick-and-easy configuration. This is accomplished by its pre-configured routing entities, whereby only minimal configuration is required such as for IP addresses to get the device up and running and deployed in the network.

Reader's Notes

22 CRP Configuration

This section describes configuration specific to the CRP application. As CRP has similar functionality to the SBC application, for configuration common to the SBC, which is not covered in this section, see the following SBC sections:

- Configuring General Settings (see Configuring General Settings on page 267)
- Configuring Admission Control (see Configuring Admission Control on page 268)
- Configuring Allowed Coder Groups (see Configuring Allowed Coder Groups on page 271)
- Configuring Classification Rules (see Configuring Classification Rules on page 272)
- Configuring Condition Rules (see Configuring Condition Rules on page 278)
- Configuring SBC IP-to-IP Routing (see Configuring SBC IP-to-IP Routing on page 279)
- Configuring Alternative Routing Reasons (see Configuring Alternative Routing Reasons on page 286)
- Configuring IP-to-IP Inbound Manipulations (see Configuring IP-to-IP Inbound Manipulations on page 289)
- Configuring IP-to-IP Outbound Manipulations (see Configuring IP-to-IP Outbound Manipulations on page 292)



Note: The main difference in the common configuration between the CRP and SBC applications is the navigation menu paths to opening these Web configuration pages. Wherever "SBC" appears in the menu path, for the CRP application it appears as "CRP".

22.1 Enabling the CRP Application

The procedure below describes how to enable the CRP application.



Note: For CRP support, the device must be installed with the relevant Software License Key. This License Key must also be defined with the required number of call sessions.

➤ **To enable the CRP application:**

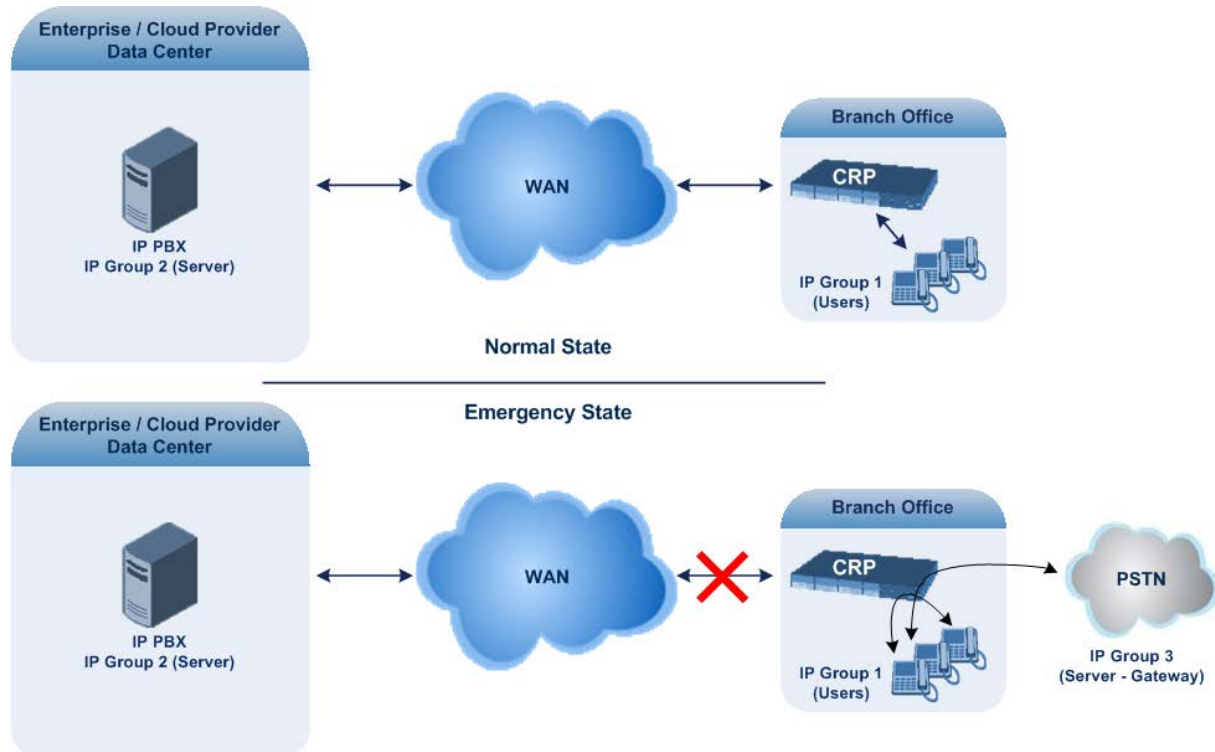
1. Open the Applications Enabling page (**Configuration** tab > **VoIP** menu > **Applications Enabling** > **Applications Enabling**).
2. From the 'Enable CRP Application' drop-down list, select **Enable**.
3. Save (burn) the changes to the device's flash memory with a device reset (see Saving Configuration on 360).

22.2 Configuring Call Survivability Mode

The CRP can be configured to operate in one of the following call survivability modes:

- **Normal (Default):** The CRP interworks between the branch users and the IP PBX located at headquarters. The CRP forwards all requests (such as for registration) from the branch users to the IP PBX, and routes the calls based on the IP-to-IP routing rules. If communication with the IP PBX fails (i.e., Emergency mode), it routes the calls between the branch users themselves. If this fails, it routes the calls to the PSTN (if employed).

Figure 14: CRP in Normal & Auto Answer to Registrations Modes



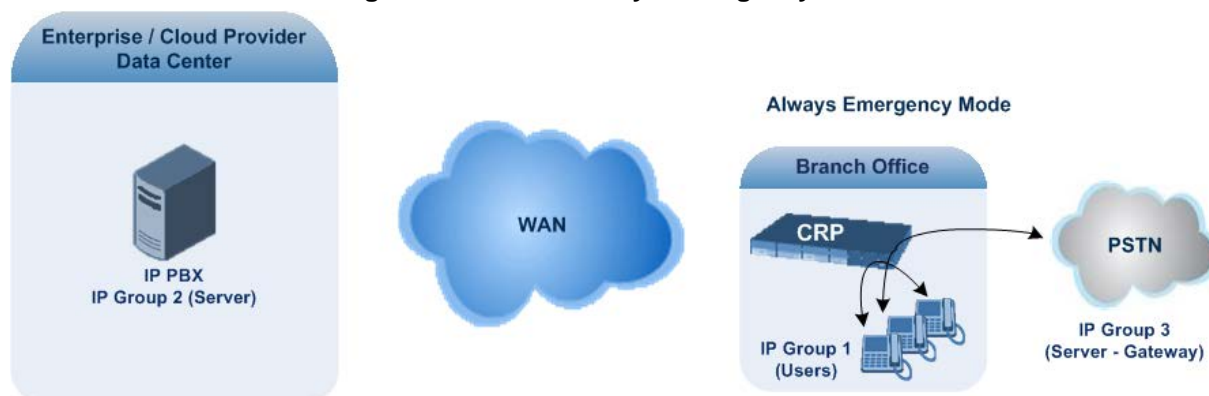
- **Auto Answer to Registrations:** This mode is the same as the Normal mode, except that the CRP registers the branch users in its registration database instead of forwarding them to the IP PBX.



Note: SIP REGISTER and OPTIONS requests are terminated at the CRP.

- **Always Emergency:** The CRP routes the calls between the branch users themselves as if connectivity failure has occurred with the IP PBX. The CRP also registers the branch users in its registration database.

Figure 15: CRP in Always Emergency Mode



➤ **To configure the Call Survivability mode:**

1. Open the General Settings page (**Configuration** tab > **VoIP** menu > **CRP** > **General Settings**).
2. From the 'CRP Survivability Mode' drop-down list, select the required mode.
3. Click **Submit**.

22.3 Pre-Configured IP Groups

The IP Group table for CRP is pre-configured with the following IP Groups:

Table 8: Pre-configured IP Groups in the IP Group Table

Index	Type	Description
1	User	Users
2	Server	Proxy
3	Server	Gateway

These IP Groups represent the following IP entities:

- **"Users" IP Group:** LAN users (e.g., IP phones) at the branch office
- **"Server" IP Group:** server (e.g., hosted IP PBX at headquarters)
- **"Gateway" IP Group:** interface with the PSTN

These IP Groups are used in the IP-to-IP Routing rules to indicate the source and destination of the call (see "Pre-Configured IP-to-IP Routing Rules" on page 304).



Notes:

- These IP Groups cannot be deleted and additional IP Groups cannot be configured. The IP Groups can be edited, except for the fields listed above, which are read-only.
- For accessing the IP Group table and for a description of its parameters, see Configuring IP Groups on page 191.

22.4 Pre-Configured IP-to-IP Routing Rules

The IP-to-IP Routing table is pre-configured with IP-to-IP routing rules. These rules depend on the configured Call Survivability mode, as described in "Configuring Call Survivability Mode" on page 302.



Notes:

- The IP-to-IP Routing table is read-only.
- For accessing the IP-to-IP Routing table and for a description of its parameters, see Configuring SBC IP-to-IP Routing on page 279.

Table 9: Pre-Configured Rules in IP-to-IP Routing Table

Mode	Index	Source IP Group ID	Request Type	Destination Type	Destination IP Group ID	Destination Address	Alternative Route Options
Normal	1	*	OPTIONS	Dest Address	-	Internal	Route Row
	3	1	All	IP Group	2	-	Route Row
	4	1	All	IP Group	1	-	Alternative
	5	1	All	IP Group	3	-	Alternative
	6	2	All	IP Group	1	-	Route Row
	7	3	All	IP Group	2	-	Route Row
	8	3	All	IP Group	1	-	Alternative
Always Emergency	1	*	OPTIONS	Dest Address	-	Internal	Route Row
	2	*	REGISTER	Dest Address	-	Internal	Route Row
	4	1	All	IP Group	1	-	Route Row
	5	1	All	IP Group	3	-	Alternative
	8	3	All	IP Group	1	-	Route Row
Auto Answer to Registrations	1	*	OPTIONS	Dest Address	-	Internal	Route Row
	2	*	REGISTER	Dest Address	-	Internal	Route Row
	3	1	All	IP Group	2	-	Route Row
	4	1	All	IP Group	1	-	Alternative
	5	1	All	IP Group	3	-	Alternative
	6	2	All	IP Group	1	-	Route Row
	7	3	All	IP Group	2	-	Route Row
	8	3	All	IP Group	1	-	Alternative

Part VII

Stand-Alone Survivability Application

23 SAS Overview

The device's Stand-Alone Survivability (SAS) feature ensures telephony communication continuity (survivability) for enterprises using hosted IP services (such as IP Centrex) or IP-PBX in cases of failure of these entities. In case of failure of the IP Centrex, IP-PBX servers (or even WAN connection and access Internet modem), the enterprise typically loses its internal telephony service at any branch, between its offices, and with the external environment. Typically, these failures also lead to the inability to make emergency calls (e.g., 911 in North America). Despite these possible points of failure, the device's SAS feature ensures that the enterprise's telephony services (e.g., SIP IP phones or soft phones) are maintained, by routing calls to the PSTN (i.e., providing PSTN fallback).



Notes:

- The SAS application is available only if the device is installed with the SAS Software License Key.
- Throughout this section, the term *user agent* (UA) refers to the enterprise's LAN phone user (i.e., SIP telephony entities such as IP phones).
- Throughout this section, the term *proxy* or *proxy server* refers to the enterprise's centralized IP Centrex or IP-PBX.
- Throughout this section, the term SAS refers to the SAS application running on the device.

23.1 SAS Operating Modes

The device's SAS application can be implemented in one of the following main modes:

- **Outbound Proxy:** In this mode, SAS receives SIP REGISTER requests from the enterprise's UAs and forwards these requests to the external proxy (i.e., outbound proxy). When a connection with the external proxy fails, SAS enters SAS emergency state and serves as a proxy, by handling internal call routing for the enterprise's UAs - routing calls between UAs and if setup, routing calls between UAs and the PSTN. For more information, see 'SAS Outbound Mode' on page 308.
- **Redundant Proxy:** In this mode, the enterprise's UAs register with the external proxy and establish calls directly through the external proxy, without traversing SAS (or the device per se'). Only when connection with the proxy fails, do the UAs register with SAS, serving now as the UAs redundant proxy. SAS then handles the calls between UAs, and between the UAs and the PSTN (if setup). This mode is operational only during SAS in emergency state. This mode can be implemented, for example, for proxies that accept only SIP messages that are sent directly from the UAs. For more information, see 'SAS Redundant Mode' on page 309.



Note: It is recommended to implement the SAS outbound mode.

23.1.1 SAS Outbound Mode

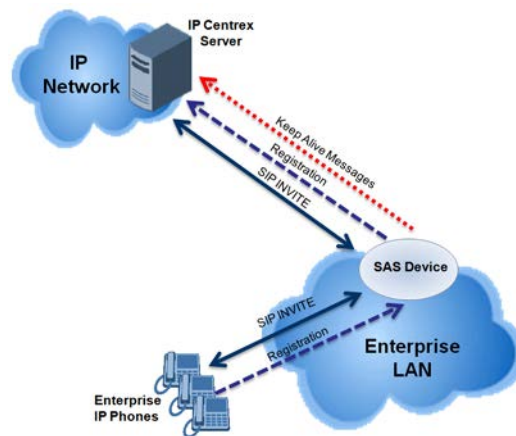
This section describes the SAS outbound mode, which includes the following states:

- Normal state (see 'Normal State' on page 308)
- Emergency state (see 'Emergency State' on page 308)

23.1.1.1 Normal State

In normal state, SAS receives REGISTER requests from the enterprise's UAs and forwards them to the external proxy (i.e., outbound proxy). Once the proxy replies with a SIP 200 OK, the device records the Contact and address of record (AOR) of the UAs in its internal SAS registration database. Therefore, in this mode, SAS maintains a database of all the registered UAs in the network. SAS also continuously maintains a keep-alive mechanism toward the external proxy, using SIP OPTIONS messages. The figure below illustrates the operation of SAS outbound mode in normal state:

Figure 23-1: SAS Outbound Mode in Normal State (Example)



23.1.1.2 Emergency State

When a connection with the external proxy fails (detected by the device's keep-alive messages), the device enters SAS emergency state. The device serves as a proxy for the UAs, by handling internal call routing of the UAs (within the LAN enterprise).

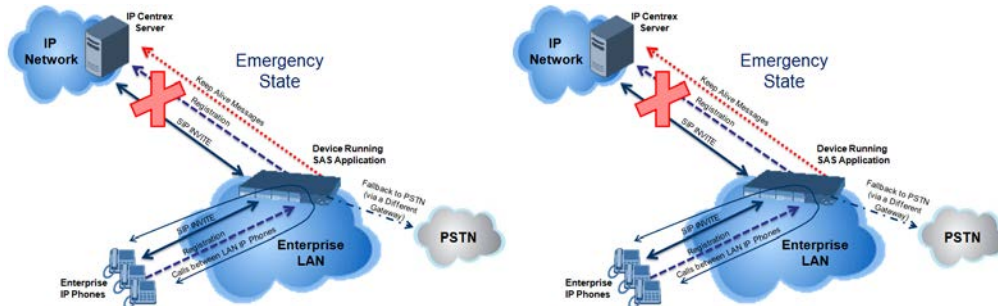


Note: SAS can also enter Emergency state if no response is received from the proxy for sent OPTIONS, INVITE, or REGISTER messages. To configure this, set the SASEnteringEmergencyMode parameter to 1.

When the device receives calls, it searches its SAS registration database to locate the destination address (according to AOR or Contact). If the destination address is not found, SAS forwards the call to the default gateway. Typically, the default gateway is defined as the device itself (on which SAS is running), and if the device has PSTN interfaces, the enterprise preserves its capability for outgoing calls (from UAs to the PSTN network).

The routing logic of SAS in emergency state is described in detail in 'SAS Routing in Emergency State' on page 313.

The figure below illustrates the operation of SAS outbound mode in emergency state:



When emergency state is active, SAS continuously attempts to communicate with the external proxy, using keep-alive SIP OPTIONS. Once connection to the proxy returns, the device exits SAS emergency state and returns to SAS normal state, as explained in 'Exiting Emergency and Returning to Normal State' on page 310.

23.1.2 SAS Redundant Mode

In SAS redundant mode, the enterprise's UAs register with the external proxy and establish calls directly through it, without traversing SAS (or the device per se'). Only when connection with the proxy fails, do the UAs register with SAS, serving now as the UAs redundant proxy. SAS then handles the calls between UAs, and between the UAs and the PSTN (if setup).

This mode is operational only during SAS in emergency state.

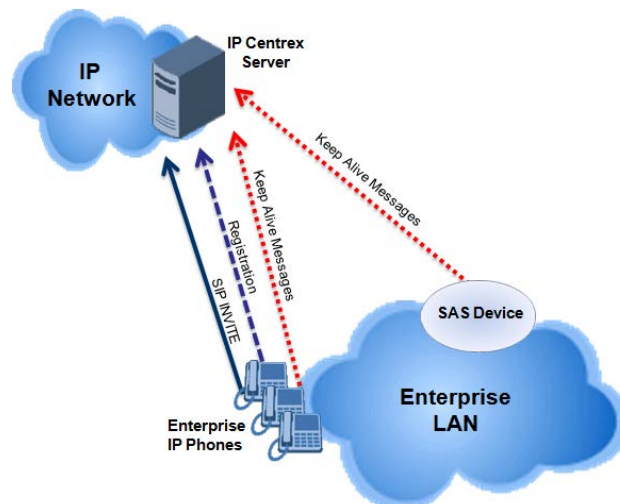


Note: In this SAS deployment, the UAs (e.g., IP phones) must support configuration for primary and secondary proxy servers (i.e., proxy redundancy), as well as homing. Homing allows the UAs to switch back to the primary server from the secondary proxy once the connection to the primary server returns (UAs check this using keep-alive messages to the primary server). If homing is not supported by the UAs, you can configure SAS to ignore messages received from UAs in normal state (the 'SAS Survivability Mode' parameter must be set to 'Always Emergency' / 2) and thereby, "force" the UAs to switch back to their primary proxy.

23.1.2.1 Normal State

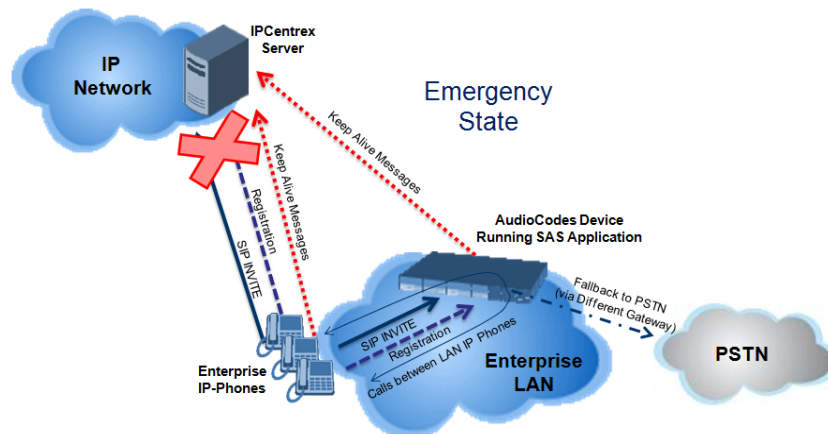
In normal state, the UAs register and operate directly with the external proxy.

Figure 23-2: SAS Redundant Mode in Normal State (Example)



23.1.2.2 Emergency State

If the UAs detect that their primary (external) proxy does not respond, they immediately register to SAS and start routing calls to it.



23.1.2.3 Exiting Emergency and Returning to Normal State

Once the connection with the primary proxy is re-established, the following occurs:

- **UAs:** Switch back to operate with the primary proxy.
- **SAS:** Ignores REGISTER requests from the UAs, forcing the UAs to switch back to the primary proxy.

Note: This is applicable only if the 'SAS Survivability Mode' parameter is set to 'Always Emergency' (2).

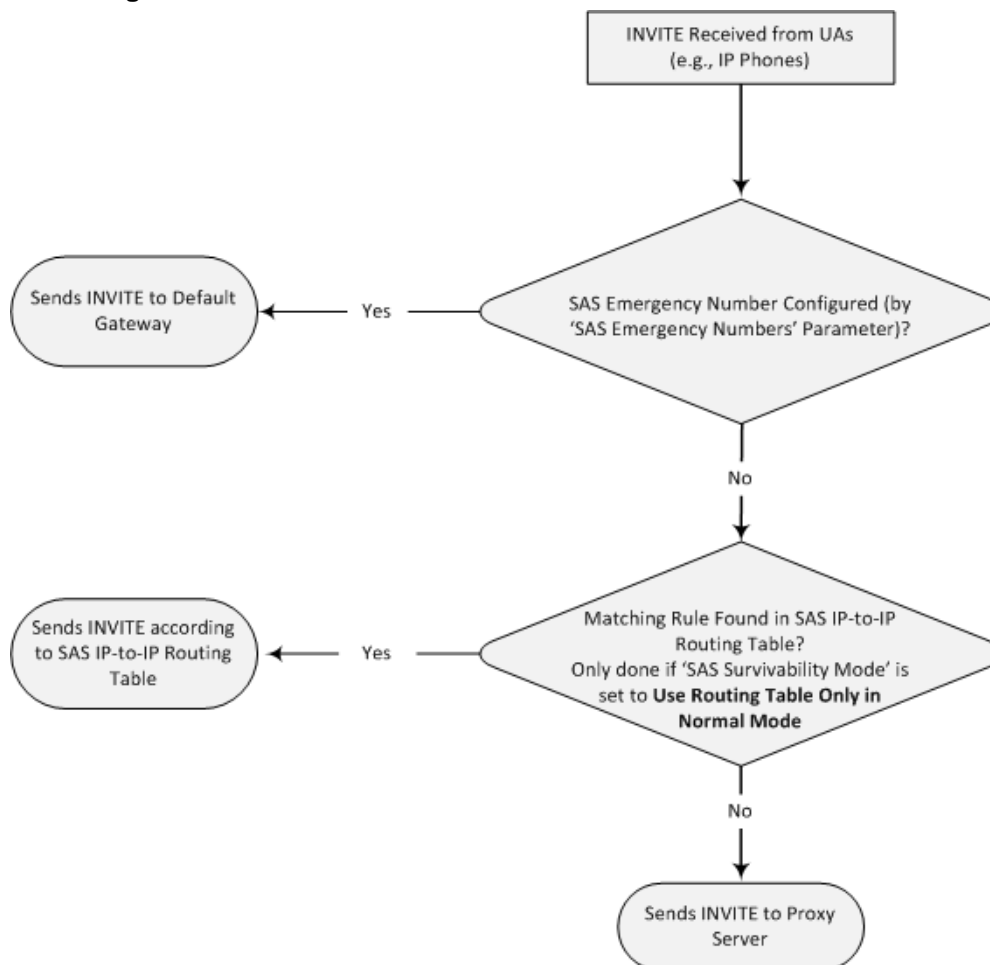
23.2 SAS Routing

This section provides flowcharts describing the routing logic for SAS in normal and emergency states.

23.2.1 SAS Routing in Normal State

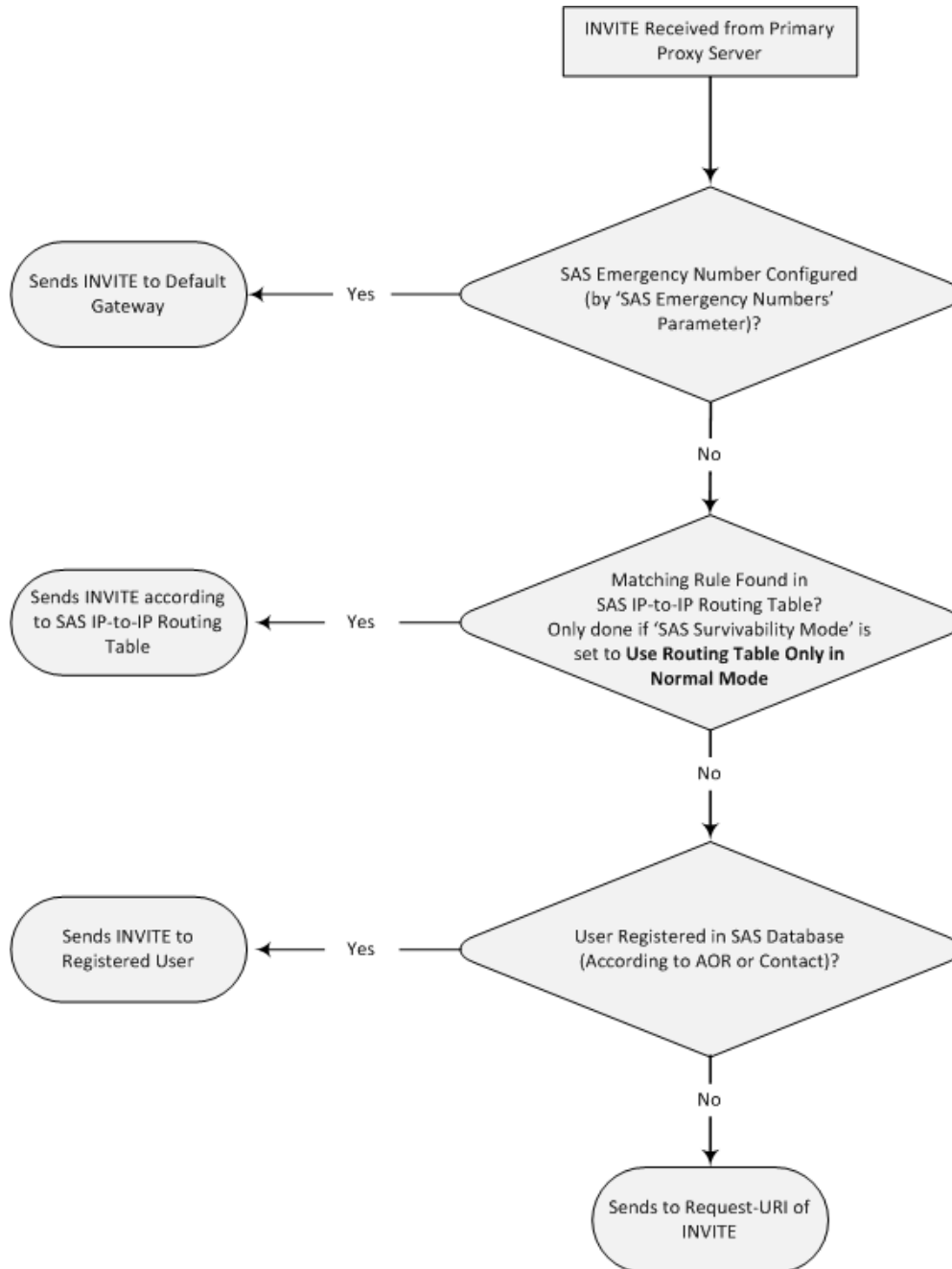
The flowchart below displays the routing logic for SAS in normal state for INVITE messages received from UAs:

Figure 23-3: Flowchart of INVITE from UA's in SAS Normal State



The flowchart below displays the routing logic for SAS in normal state for INVITE messages received from the external proxy:

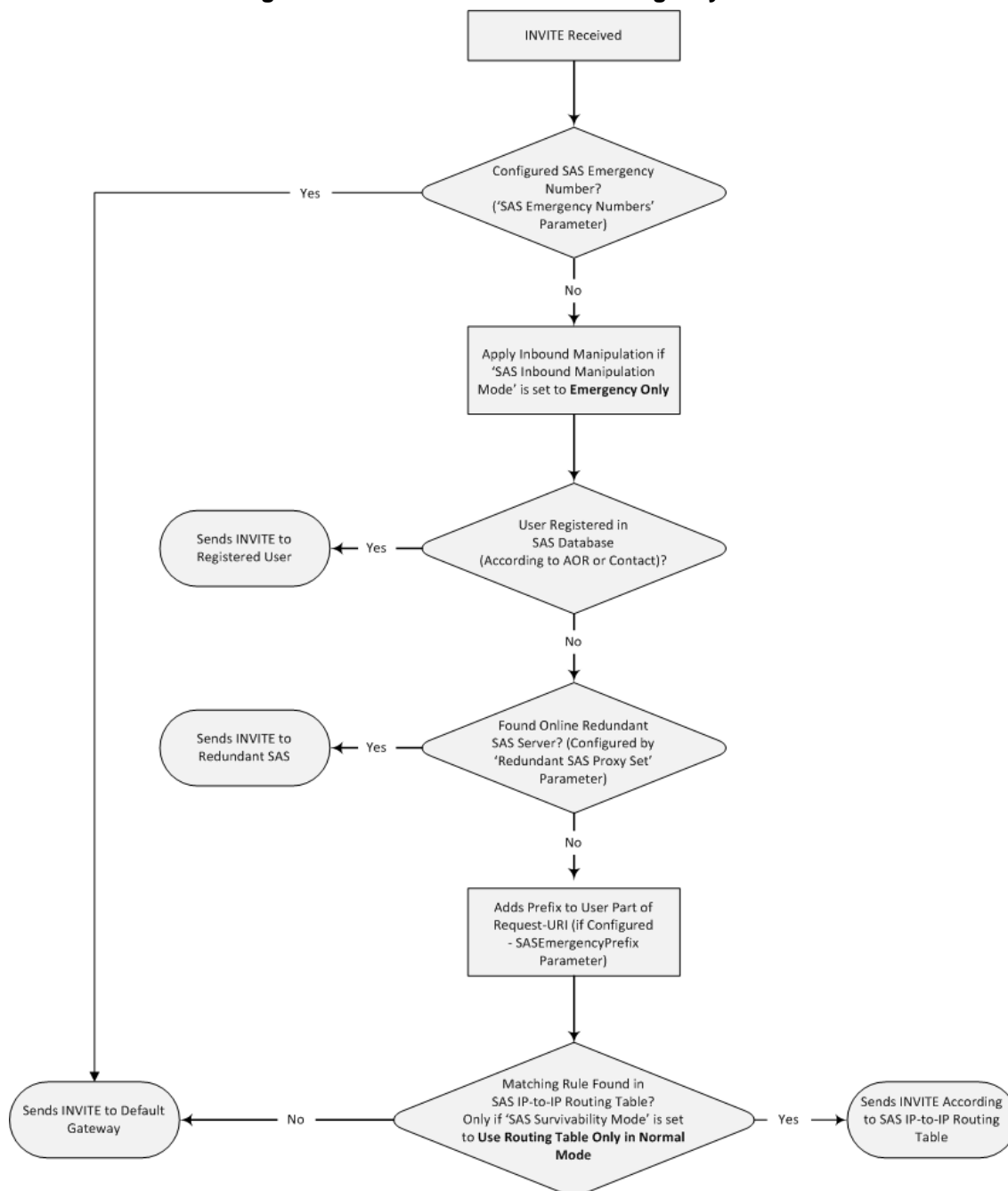
Figure 23-4: Flowchart of INVITE from Primary Proxy in SAS Normal State



23.2.2 SAS Routing in Emergency State

The flowchart below shows the routing logic for SAS in emergency state:

Figure 23-5: Flowchart for SAS Emergency State



Reader's Notes

24 SAS Configuration

SAS supports various configuration possibilities, depending on how the device is deployed in the network and the network architecture requirements. This section provides step-by-step procedures on configuring the SAS application, using the device's Web interface.

The SAS configuration includes the following:

- General SAS configuration that is common to all SAS deployment types (see 'General SAS Configuration' on page 315)
- SAS outbound mode (see 'Configuring SAS Outbound Mode' on page 318)
- SAS redundant mode (see 'Configuring SAS Redundant Mode' on page 318)
- Gateway and SAS applications deployed together (see 'Configuring Gateway Application with SAS' on page 319)
- Optional, advanced SAS features (see 'Advanced SAS Configuration' on page 322)

24.1 General SAS Configuration

This section describes the general configuration required for the SAS application. This configuration is applicable to all SAS modes.

24.1.1 Enabling the SAS Application

Before you can configure SAS, you need to enable the SAS application on the device. Once enabled, the **SAS** menu and related pages appear in the device's Web interface.



Note: The SAS application is available only if the device is installed with the SAS Software License Key. If your device is not installed with the SAS feature, contact your AudioCodes representative.

➤ **To enable the SAS application:**

1. Open the Applications Enabling page (**Configuration** tab > **VoIP** menu > **Applications Enabling** > **Applications Enabling**).
2. From the 'SAS Application' drop-down list, select **Enable**.

Figure 24-1: Enabling SAS Application



3. Click **Submit**.
4. Save the changes to the flash memory with a device reset.

24.1.2 Configuring Common SAS Parameters

The procedure below describes how to configure SAS settings that are common to all SAS modes. This includes various SAS parameters as well as configuring the Proxy Set for the SAS proxy (if required). The SAS Proxy Set ID defines the address of the UAs' external proxy.

➤ **To configure common SAS settings:**

1. Open the SAS Configuration page (**Configuration** tab > **VoIP** menu > **SAS** > **Stand Alone Survivability**).
2. Define the port used for sending and receiving SAS messages. This can be any of the following port types:
 - UDP port - defined in the 'SAS Local SIP UDP Port' field
 - TCP port - defined in the 'SAS Local SIP TCP Port' field
 - TLS port - defined in the 'SAS Local SIP TLS Port' field

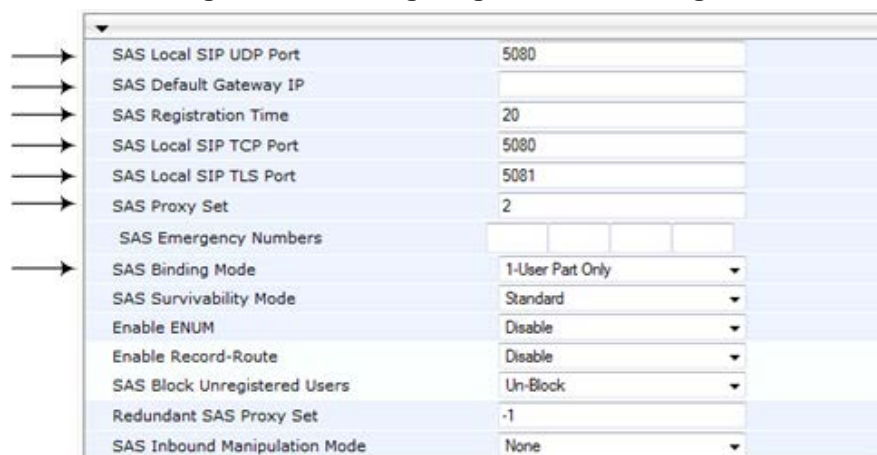


Note: This SAS port must be different than the device's local gateway port (i.e., that defined for the 'SIP UDP/TCP/TLS Local Port' parameter in the SIP General Parameters page - **Configuration** tab > **VoIP** menu > **SIP Definitions** > **General Parameters**).

3. In the 'SAS Default Gateway IP' field, define the IP address and port (in the format *x.x.x.x:port*) of the device (i.e., Gateway application). Note that the port of the device is defined by the parameter 'SIP UDP Local Port' (refer to the note in Step 2 above).
4. In the 'SAS Registration Time' field, define the value for the SIP Expires header, which is sent in the 200 OK response to an incoming REGISTER message when SAS is in emergency state.
5. From the 'SAS Binding Mode' drop-down list, select the database binding mode:
 - **0-URI:** If the incoming AOR in the REGISTER request uses a 'tel:' URI or 'user=phone', the binding is done according to the Request-URI user part only. Otherwise, the binding is done according to the entire Request-URI (i.e., user and host parts - user@host).
 - **1-User Part Only:** Binding is done according to the user part only.

You must select **1-User Part Only** in cases where the UA sends REGISTER messages as SIP URI, but the INVITE messages sent to this UA include a Tel URI. For example, when the AOR of an incoming REGISTER is sip:3200@domain.com, SAS adds the entire SIP URI (e.g., sip:3200@domain.com) to its database (when the parameter is set to '0-URI'). However, if a subsequent Request-URI of an INVITE message for this UA arrives with sip:3200@10.1.2.3 user=phone, SAS searches its database for "3200", which it does not find. Alternatively, when this parameter is set to '1-User Part Only', then upon receiving a REGISTER message with sip:3200@domain.com, SAS adds only the user part (i.e., "3200") to its database. Therefore, if a Request-URI of an INVITE message for this UA arrives with sip:3200@10.1.2.3 user=phone, SAS can successfully locate the UA in its database.

Figure 24-2: Configuring Common Settings



SAS Local SIP UDP Port	5080
SAS Default Gateway IP	
SAS Registration Time	20
SAS Local SIP TCP Port	5080
SAS Local SIP TLS Port	5081
SAS Proxy Set	2
SAS Emergency Numbers	
SAS Binding Mode	1-User Part Only
SAS Survivability Mode	Standard
Enable ENUM	Disable
Enable Record-Route	Disable
SAS Block Unregistered Users	Un-Block
Redundant SAS Proxy Set	-1
SAS Inbound Manipulation Mode	None

6. In the 'SAS Proxy Set' field, enter the Proxy Set used for SAS. The SAS Proxy Set must be defined only for the following SAS modes:

- **Outbound mode:** In SAS normal state, SAS forwards REGISTER and INVITE messages received from the UAs to the proxy servers defined in this Proxy Set.
- **Redundant mode and only if UAs don't support homing:** SAS sends keep-alive messages to this proxy and if it detects that the proxy connection has resumed, it ignores the REGISTER messages received from the UAs, forcing them to send their messages directly to the proxy.

If you define a SAS Proxy Set ID, you must configure the Proxy Set as described in Step 8 below.

7. Click **Submit** to apply your settings.
8. If you defined a SAS Proxy Set ID in Step 6 above, then you must configure the SAS Proxy Set ID:
- Open the Proxy Sets Table page (**Configuration** tab > **VoIP** menu > **Control Networks** > **Proxy Set Table**).
 - From the 'Proxy Set ID' drop-down list, select the required Proxy Set ID.



Notes:

- The selected Proxy Set ID number must be the same as that specified in the 'SAS Proxy Set' field in the 'SAS Configuration page (see Step 6).
- Do not use Proxy Set ID 0.

- In the 'Proxy Address' field, enter the IP address of the external proxy server.
- From the 'Enable Proxy Keep Alive' drop-down list, select **Using Options**. This instructs the device to send SIP OPTIONS messages to the proxy for the keep-alive mechanism.

Figure 24-3: Defining SAS Proxy Server

	Proxy Address	Transport Type
1	10.15.4.52	TLS
2		
3		
4		
5		

Enable Proxy Keep Alive	Using Options
Proxy Keep Alive Time	60
Proxy Load Balancing Method	Disable
Is Proxy Hot Swap	No
Proxy Redundancy Mode	Not Configured
SRD Index	0
Classification Input	IP only

- Click **Submit** to apply your settings.

24.2 Configuring SAS Outbound Mode

This section describes how to configure the SAS outbound mode. These settings are in addition to the ones described in 'Configuring Common SAS Parameters' on page 315.



Note: The VoIP CPEs (such as IP phones or residential gateways) need to be defined so that their proxy and registrar destination addresses and ports are the same as that configured for the device's SAS IP address and SAS local SIP port. In some cases, on the UAs, it is also required to define SAS as their outbound proxy, meaning that messages sent by the UAs include the host part of the external proxy, but are sent (on Layer 3/4) to the IP address / UDP port of SAS.

➤ **To configure SAS outbound mode:**

1. Open the SAS Configuration page (**Configuration** tab > **VoIP** menu > **SAS** > **Stand Alone Survivability**).
2. From the 'SAS Survivability Mode' drop-down list, select **Standard**.
3. Click **Submit**.

24.3 Configuring SAS Redundant Mode

This section describes how to configure the SAS redundant mode. These settings are in addition to the ones described in 'Configuring Common SAS Parameters' on page 315.



Note: The VoIP CPEs (such as IP phones or residential gateways) need to be defined so that their primary proxy is the external proxy, and their redundant proxy destination addresses and port is the same as that configured for the device's SAS IP address and SAS SIP port.

➤ **To configure SAS redundant mode:**

1. Open the SAS Configuration page (**Configuration** tab > **VoIP** menu > **SAS** > **Stand Alone Survivability**).
2. From the 'SAS Survivability Mode' drop-down list, select one of the following, depending on whether the UAs support homing (i.e., they always attempt to operate with the primary proxy, and if using the redundant proxy, they switch back to the primary proxy whenever it's available):
 - **UAs support homing:** Select **Always Emergency**. This is because SAS does not need to communicate with the primary proxy of the UAs; SAS serves only as the redundant proxy of the UAs. When the UAs detect that their primary proxy is available, they automatically resume communication with it instead of with SAS.
 - **UAs do not support homing:** Select **Ignore REGISTER**. SAS uses the keep-alive mechanism to detect availability of the primary proxy (defined by the SAS Proxy Set). If the connection with the primary proxy resumes, SAS ignores the messages received from the UAs, forcing them to send their messages directly to the primary proxy.
3. Click **Submit**.

24.4 Configuring Gateway Application with SAS

If you want to run both the Gateway and SAS applications on the device, the configuration described in this section is required. The configuration steps depend on whether the Gateway application is operating with SAS in outbound mode or SAS in redundant mode.



Note: The Gateway application must use the same SAS operation mode as the SIP UAs. For example, if the UAs use the SAS application as a redundant proxy (i.e., SAS redundancy mode), then the Gateway application must do the same.

24.4.1 Gateway with SAS Outbound Mode

The procedure below describes how to configure the Gateway application with SAS outbound mode.

➤ **To configure Gateway application with SAS outbound mode:**

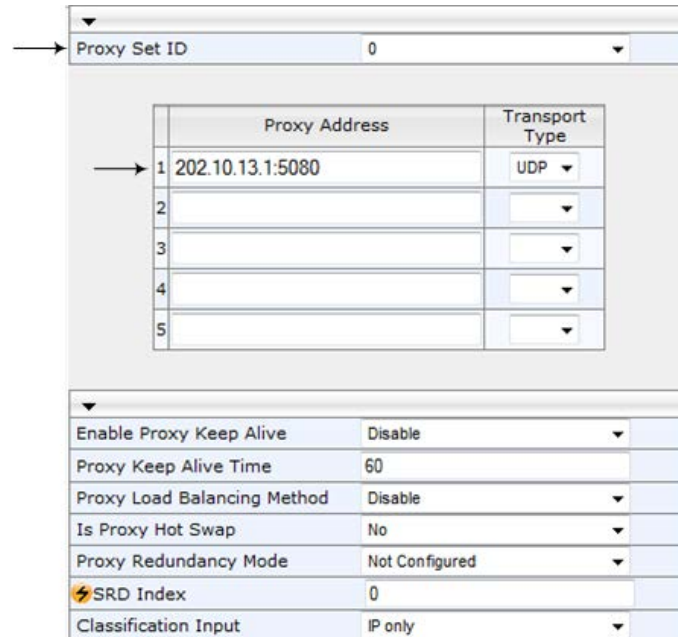
1. Define the proxy server address for the Gateway application:
 - a. Open the Proxy & Registration page (**Configuration** tab > **VoIP** menu > **SIP Definitions** submenu > **Proxy & Registration**).
 - b. From the 'Use Default Proxy' drop-down list, select **Yes**.

Figure 24-4: Enabling Proxy Server for Gateway Application

- c. Click **Submit**.
- d. Open the Proxy Sets Table page (**Configuration** tab > **VoIP** menu > **Control Network** submenu > **Proxy Sets** Table).
- e. From the 'Proxy Set ID' drop-down list, select **0**.

- f. In the first 'Proxy Address' field, enter the IP address and port of the device (in the format `x.x.x.x:port`). This is the port as defined in the 'SAS Local UDP/TCP/TLS Port' field (see 'Configuring Common SAS Parameters' on page 315).

Figure 24-5: Defining Proxy Server for Gateway Application

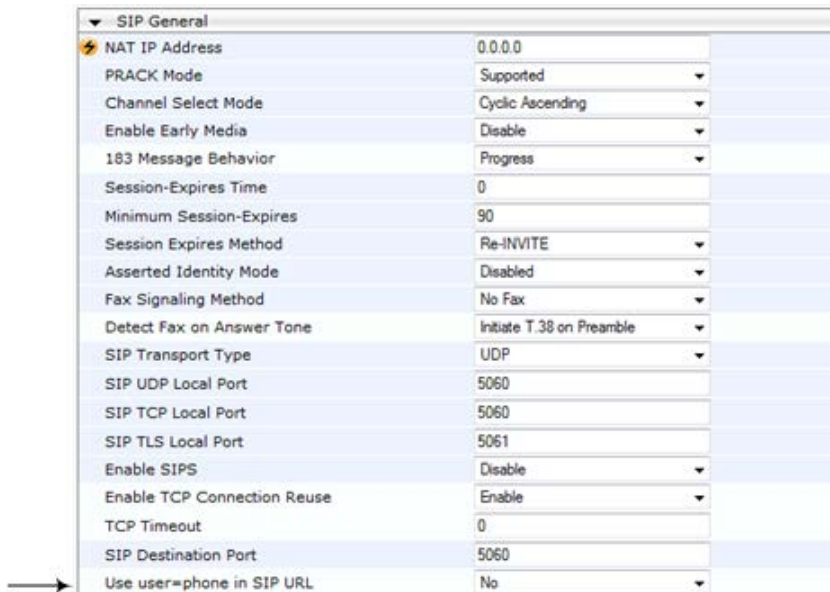


	Proxy Address	Transport Type
1	202.10.13.1:5080	UDP
2		
3		
4		
5		

Enable Proxy Keep Alive	Disable
Proxy Keep Alive Time	60
Proxy Load Balancing Method	Disable
Is Proxy Hot Swap	No
Proxy Redundancy Mode	Not Configured
SRD Index	0
Classification Input	IP only

- g. Click **Submit**.
2. Disable use of `user=phone` in SIP URL:
 - a. Open the SIP General Parameters page (**Configuration** tab > **VoIP** menu > **SIP Definitions** submenu > **General Parameters**).
 - b. From the 'Use `user=phone` in SIP URL' drop-down list, select **No**. This instructs the Gateway application not to use `user=phone` in the SIP URL and therefore, REGISTER and INVITE messages use SIP URI. (By default, REGISTER messages are sent with `sip uri` and INVITE messages with `tel uri`.)

Figure 24-6: Disabling `user=phone` in SIP URL



SIP General	
NAT IP Address	0.0.0.0
PRACK Mode	Supported
Channel Select Mode	Cyclic Ascending
Enable Early Media	Disable
183 Message Behavior	Progress
Session-Expires Time	0
Minimum Session-Expires	90
Session Expires Method	Re-INVITE
Asserted Identity Mode	Disabled
Fax Signaling Method	No Fax
Detect Fax on Answer Tone	Initiate T.38 on Preamble
SIP Transport Type	UDP
SIP UDP Local Port	5060
SIP TCP Local Port	5060
SIP TLS Local Port	5061
Enable SIPs	Disable
Enable TCP Connection Reuse	Enable
TCP Timeout	0
SIP Destination Port	5060
Use <code>user=phone</code> in SIP URL	No

- c. Click **Submit**.

24.4.2 Gateway with SAS Redundant Mode

The procedure below describes how to configure the Gateway application with SAS redundant mode.

➤ **To configure Gateway application with SAS redundant mode:**

1. Define the proxy servers for the Gateway application:
 - a. Open the Proxy & Registration page (**Configuration** tab > **VoIP** menu > **SIP Definitions** submenu > **Proxy & Registration**).
 - b. From the 'Use Default Proxy' drop-down list, select **Yes**.

Figure 24-7: Enabling Proxy Server for Gateway Application

Use Default Proxy: Yes

Proxy Set Table

Proxy Name

- c. Click **Submit**.
- d. Open the Proxy Sets Table page (**Configuration** tab > **VoIP** menu > **Control Network** submenu > **Proxy Sets Table**).
- e. From the 'Proxy Set ID' drop-down list, select **0**.
- f. In the first 'Proxy Address' field, enter the IP address of the external proxy server.
- g. In the second 'Proxy Address' field, enter the IP address and port of the device (in the format x.x.x.x:port). This is the same port as defined in the 'SAS Local UDP/TCP/TLS Port' field (see 'Configuring Common SAS Parameters' on page 315).
- h. From the 'Proxy Redundancy Mode' drop-down list, select **Homing**.

Figure 24-8: Defining Proxy Servers for Gateway Application

Proxy Set ID: 0

	Proxy Address	Transport Type
1	202.10.13.1:5080	UDP
2	10.13.4.1	UDP
3		
4		
5		

Enable Proxy Keep Alive: Using Options

Proxy Keep Alive Time: 60

Proxy Load Balancing Method: Disable

Is Proxy Hot Swap: No

Proxy Redundancy Mode: Homing

SRD Index: 0

Classification Input: IP only

- i. Click **Submit**.
2. Disable the use of *user=phone* in the SIP URL:
 - a. Open the SIP General Parameters page (**Configuration** tab > **VoIP** menu > **SIP Definitions** submenu > **General Parameters**).

- b. From the 'Use user=phone in SIP URL' drop-down list, select **No**. This instructs the Gateway application not to use *user=phone* in SIP URL and therefore, REGISTER and INVITE messages use SIP URI. (By default, REGISTER messages are sent with *sip uri* and INVITE messages with *tel uri*.)
- c. Click **Submit**.

24.5 Advanced SAS Configuration

This section describes the configuration of advanced SAS features that can optionally be implemented in your SAS deployment.

24.5.1 Manipulating URI user part of Incoming REGISTER

There are scenarios in which the UAs register to the proxy server with their full phone number (for example, "976653434"), but can receive two types of INVITE messages (calls):

- INVITES whose destination is the UAs' full number (when the call arrives from outside the enterprise)
- INVITES whose destination is the last four digits of the UAs' phone number ("3434" in our example) when it is an internal call within the enterprise

Therefore, it is important that the device registers the UAs in the SAS registered database with their extension numbers (for example, "3434") in addition to their full numbers. To do this, you can define a manipulation rule to manipulate the SIP Request-URI user part of the AOR (in the To header) in incoming REGISTER requests. Once manipulated, it is saved in this manipulated format in the SAS registered users database in addition to the original (un-manipulated) AOR.

For example: Assume the following incoming REGISTER message is received and that you want to register in the SAS database the UA's full number as well as the last four digits from the right of the SIP URI user part:

```
REGISTER sip:10.33.38.2 SIP/2.0
Via: SIP/2.0/UDP 10.33.4.226:5050;branch=z9hG4bKac10827
Max-Forwards: 70
From: <sip: 976653434@10.33.4.226>;tag=1c30219
To: <sip: 976653434@10.33.4.226>
Call-ID: 16844@10.33.4.226
CSeq: 1 REGISTER
Contact: <sip: 976653434@10.10.10.10:5050>;expires=180
Allow:
REGISTER, OPTIONS, INVITE, ACK, CANCEL, BYE, NOTIFY, PRACK, REFER, INFO, SUB
SCRIBE, UPDATE
Expires: 180
User-Agent: Audiocodes-Sip-Gateway-/v.
Content-Length: 0
```

After manipulation, SAS registers the user in its database as follows:

- **AOR:** 976653434@10.33.4.226
- **Associated AOR:** 3434@10.33.4.226 (after manipulation, in which only the four digits from the right of the URI user part are retained)
- **Contact:** 976653434@10.10.10.10

The procedure below describes how to configure the above manipulation example.

➤ To manipulate incoming Request-URI user part of REGISTER message:

1. Open the SAS Configuration page (**Configuration** tab > **VoIP** menu > **SAS** > **Stand Alone Survivability**).
2. Under the **SAS Registration Manipulation** group, in the 'Leave From Right' field, enter the number of digits (e.g., "4") to leave from the right side of the user part. This

field defines the number of digits to retain from the right side of the user part; all other digits in the user part are removed.

Figure 24-9: Manipulating User Part in Incoming REGISTER

SAS Local SIP UDP Port	5080
SAS Default Gateway IP	10.0.0.2:5080
SAS Registration Time	20
SAS Local SIP TCP Port	5080
SAS Local SIP TLS Port	5081
SAS Proxy Set	0
SAS Emergency Numbers	
SAS Binding Mode	0-URI
SAS Survivability Mode	Always Emergency
Enable ENUM	Disable
Enable Record-Route	Disable
SAS Block Unregistered Users	Un-Block
Redundant SAS Proxy Set	-1
SAS Inbound Manipulation Mode	None

SAS Registration Manipulation	
Remove From Right	Leave From Right
0	4

SAS Routing	
SAS Routing Table	

3. Click **Submit**.



Notes:

- The device first does manipulation according to the Remove From Right parameter and only then according to the Leave From Right parameter.
- Only one manipulation rule can be configured.
- You can also configure SAS registration manipulation using the table ini file parameter, SASRegistrationManipulation.


24.5.2 Manipulating Destination Number of Incoming INVITE

You can define a manipulation rule to manipulate the destination number in the Request-URI of incoming INVITE messages when SAS is in emergency state. This is required, for example, if the call is destined to a registered user but the destination number in the received INVITE is not the number assigned to the registered user in the SAS registration database. To overcome this and successfully route the call, you can define manipulation rules to change the INVITE's destination number so that it matches that of the registered user in the database. This is done using the IP to IP Inbound Manipulation table.

For example, in SAS emergency state, assume an incoming INVITE has a destination number "7001234" which is destined to a user registered in the SAS database as "552155551234". In this scenario, the received destination number needs to be manipulated to the number "552155551234". The outgoing INVITE sent by the device then also contains this number in the Request-URI user part.

In normal state, the numbers are not manipulated. In this state, SAS searches the number 552155551234 in its database and if found, it sends the INVITE containing this number to the UA.

➤ **To manipulate the destination number in SAS emergency state:**

1. Open the SAS Configuration page (**Configuration** tab > **VoIP** menu > **SAS** > **Stand Alone Survivability**).
2. From the 'SAS Inbound Manipulation Mode' (*SASInboundManipulationMode*) drop-down list, select **Emergency Only**.
3. Click **Submit**; the **SAS Inbound Manipulation Mode Table**  button appears on the page.
4. Click this button to open the IP to IP Inbound Manipulation page.
5. Add your SAS manipulation rule as required. See the table below for descriptions of the parameters.
6. Click **Submit** to save your changes.



Notes:

- The following fields in the IP to IP Inbound Manipulation table are not applicable to SAS and must be left at their default values:
 - 'Additional Manipulation' - default is **0**
 - 'Manipulation Purpose' - default is **Normal**
 - 'Source IP Group' - default is **-1**
- The IP to IP Inbound Manipulation table can also be configured using the table ini file parameter, *IPInboundManipulation*.

Table 24-1: SAS IP to IP Inbound Manipulation Parameters

Parameter	Description
Matching Characteristics (Rule)	
Additional Manipulation [IPInboundManipulation_IsAdditionalManipulation]	<p>Determines whether additional SIP URI user part manipulation is done for the table entry rule listed directly above it.</p> <ul style="list-style-type: none"> ▪ [0] No = (Default) Regular manipulation rule (not done in addition to the rule above it). ▪ [1] Yes = If the above row entry rule matched the call, consider this row entry as a match as well and perform the manipulation specified by this rule. <p>Note: Additional manipulation can only be done on a different SIP URI, source or destination, to the rule configured in the row above as configured by the 'Manipulated URI' parameter (see below).</p>
Manipulation Purpose [IPInboundManipulation_ManipulationPurpose]	<p>Defines the purpose of the manipulation:</p> <ul style="list-style-type: none"> ▪ [0] Normal = (Default) Inbound manipulations affect the routing input and source and/or destination number. ▪ [1] Routing input only = Inbound manipulations affect the routing input only, retaining the original source and destination number. ▪ [2] Shared Line = Used for the Shared-Line Appearance feature. This manipulation is for registration requests to change the destination number of the secondary extension numbers to the primary extension. For more information, see 'BroadSoft's Shared Phone Line Call Appearance for SBC

Parameter	Description
	Survivability' on page 261.
Source IP Group ID [IPInboundManipulation_SrcIpGroup]	Defines the IP Group from where the incoming INVITE is received. For any IP Group, enter the value "-1".
Source Username Prefix [IPInboundManipulation_SrcUsernamePrefix]	Defines the prefix of the source SIP URI user name (usually in the From header). For any prefix, enter the asterisk "*" symbol (default). Note: The prefix can be a single digit or a range of digits. For available notations, see 'Dialing Plan Notation for Routing and Manipulation' on page 435.
Source Host [IPInboundManipulation_SrcHost]	Defines the source SIP URI host name - full name (usually in the From header). For any host name, enter the asterisk "*" symbol (default).
Destination Username Prefix [IPInboundManipulation_DestUsernamePrefix]	Defines the prefix of the destination SIP URI user name (usually in the Request-URI). For any prefix, enter the asterisk "*" symbol (default). Note: The prefix can be a single digit or a range of digits. For available notations, see 'Dialing Plan Notation for Routing and Manipulation' on page 435.
Destination Host [IPInboundManipulation_DestHost]	Defines the destination SIP URI host name - full name (usually in the Request URI). For any host name, enter the asterisk "*" symbol (default).
Request Type [IPInboundManipulation_RequestType]	Defines the SIP request type to which the manipulation rule is applied. <ul style="list-style-type: none"> [0] All = (Default) All SIP messages. [1] INVITE = All SIP messages except REGISTER and SUBSCRIBE. [2] REGISTER = Only REGISTER messages. [3] SUBSCRIBE = Only SUBSCRIBE messages. [4] INVITE and REGISTER = All SIP messages except SUBSCRIBE. [5] INVITE and SUBSCRIBE = All SIP messages except REGISTER.
Manipulated URI [IPInboundManipulation_ManipulatedURI]	Determines whether the source or destination SIP URI user part is manipulated. <ul style="list-style-type: none"> [0] Source = (Default) Manipulation is done on the source SIP URI user part. [1] Destination = Manipulation is done on the destination SIP URI user part.
Operation Rule (Action)	
Remove From Left [IPInboundManipulation_RemoveFromLeft]	Defines the number of digits to remove from the left of the user name prefix. For example, if you enter 3 and the user name is "john", the new user name is "n".

Parameter	Description
Remove From Right [IPInboundManipulation_RemoveFromRight]	Defines the number of digits to remove from the right of the user name prefix. For example, if you enter 3 and the user name is "john", the new user name is "j". Note: If both 'Remove From Right' and 'Leave From Right' parameters are configured, the 'Remove From Right' setting is applied first.
Leave From Right [IPInboundManipulation_LeaveFromRight]	Defines the number of characters that you want retained from the right of the user name. Note: If both 'Remove From Right' and 'Leave From Right' parameters are configured, the 'Remove From Right' setting is applied first.
Prefix to Add [IPInboundManipulation_Prefix2Add]	Defines the number or string that you want added to the front of the user name. For example, if you enter 'user' and the user name is "john", the new user name is "userjohn".
Suffix to Add [IPInboundManipulation_Suffix2Add]	Defines the number or string that you want added to the end of the user name. For example, if you enter '01' and the user name is "john", the new user name is "john01".

24.5.3 SAS Routing Based on IP-to-IP Routing Table

SAS routing that is based on SAS Routing table rules is applicable for the following SAS states:

- Normal, if the 'SAS Survivability Mode' parameter is set to **Use Routing Table only in Normal mode**.
- Emergency,, if the 'SAS Survivability Mode' parameter is **not** set to **Use Routing Table only in Normal mode**.

The SAS routing rule destination can be an IP Group, IP address, Request-URI, or ENUM query.

The IP-to-IP Routing Table page allows you to configure up to 120 SAS routing rules (for Normal and Emergency modes). The device routes the SAS call (received SIP INVITE message) once a rule in this table is matched. If the characteristics of an incoming call do not match the first rule, the call characteristics is then compared to the settings of the second rule, and so on until a matching rule is located. If no rule is matched, the call is rejected.

When SAS receives a SIP INVITE request from a proxy server, the following routing logic is performed:

- a. Sends the request according to rules configured in the IP-to-IP Routing table.
- b. If no matching routing rule exists, the device sends the request according to its SAS registration database.
- c. If no routing rule is located in the database, the device sends the request according to the Request-URI header.



Note: The IP-to-IP Routing table can also be configured using the table *ini* file parameter, IP2IPRouting (see 'Configuration Parameters Reference' on page 437).

➤ **To configure the IP-to-IP Routing table for SAS:**


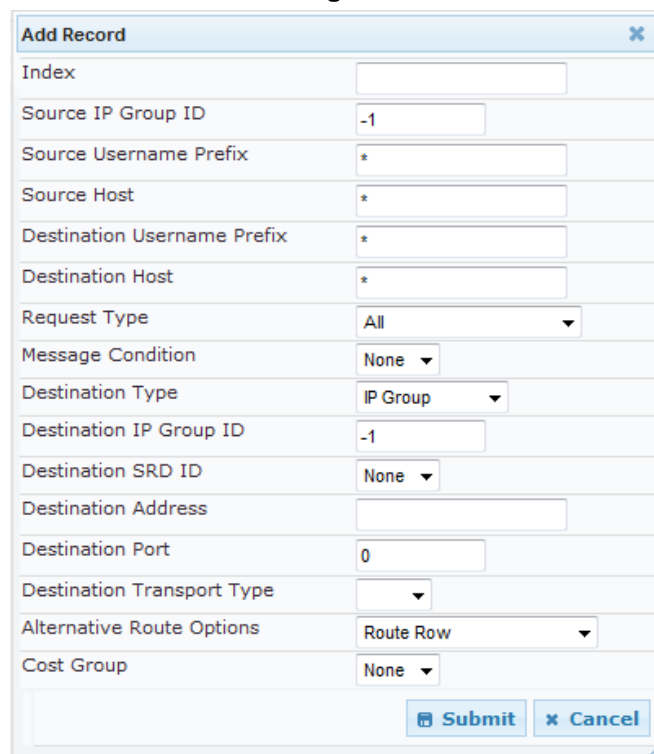
1. In the SAS Configuration page, click the **SAS Routing Table**  button; the IP-to-IP Routing Table page appears.
2. Click **Add**; the Add Record dialog box appears:

Figure 24-10: Add Record Dialog Box of SAS IP2IP Routing Page



Index	<input type="text"/>
Source IP Group ID	<input type="text" value="-1"/>
Source Username Prefix	<input type="text" value="*"/>
Source Host	<input type="text" value="*"/>
Destination Username Prefix	<input type="text" value="*"/>
Destination Host	<input type="text" value="*"/>
Request Type	<input type="text" value="All"/>
Message Condition	<input type="text" value="None"/>
Destination Type	<input type="text" value="IP Group"/>
Destination IP Group ID	<input type="text" value="-1"/>
Destination SRD ID	<input type="text" value="None"/>
Destination Address	<input type="text"/>
Destination Port	<input type="text" value="0"/>
Destination Transport Type	<input type="text"/>
Alternative Route Options	<input type="text" value="Route Row"/>
Cost Group	<input type="text" value="None"/>

3. Configure the rule according to the table below.
4. Click **Submit** to apply your changes.
5. To save the changes to flash memory, see 'Saving Configuration' on page 360.



Note: The following parameters are not applicable to SAS and must be ignored:

- 'Source IP Group ID'
- 'Destination IP Group ID'
- 'Destination SRD ID'
- 'Alternative Route Options'

Table 24-2: SAS IP-to-IP Routing Table Parameters

Parameter	Description
Matching Characteristics	
Source IP Group ID [IP2IPRouting_SrcIPGroupID]	Selects the IP Group from where the IP-to-IP call originated. Typically, the IP Group of an incoming SIP dialog is determined (or classified) using the 'Classification' table (see Configuring Classification Rules on page 272). If not used (i.e., any IP Group), simply leave the field empty.

Parameter	Description
	The default is -1.
Source Username Prefix [IP2IPRouting_SrcUsernamePrefix]	Defines the prefix of the user part of the incoming SIP dialog's source URI (usually the From URI). You can use special notations for denoting the prefix. For example, to denote any prefix, use the asterisk (*) symbol; to denote calls without a user part in the URI, use the \$ sign. For available notations, see 'Dialing Plan Notation for Routing and Manipulation' on page 435. The default is * (i.e., any prefix).
Source Host [IP2IPRouting_SrcHost]	Defines the host part of the incoming SIP dialog's source URI (usually the From URI). If this rule is not required, leave the field empty. To denote any host name, use the asterisk (*) symbol (default).
Destination Username Prefix [IP2IPRouting_DestUsernamePrefix]	Defines the prefix of the incoming SIP dialog's destination URI (usually the Request URI) user part. You can use special notations for denoting the prefix. For example, to denote any prefix, use the asterisk (*) symbol; to denote calls without a user part in the URI, use the \$ sign. For available notations, see 'Dialing Plan Notation for Routing and Manipulation' on page 435. The default is * (i.e., any prefix).
Destination Host [IP2IPRouting_DestHost]	Defines the host part of the incoming SIP dialog's destination URI (usually the Request-URI). If this rule is not required, leave the field empty. The asterisk (*) symbol (default) can be used to denote any destination host.
Request Type [IP2IPRouting_RequestType]	Defines the SIP dialog request type of the incoming SIP dialog. <ul style="list-style-type: none"> [0] All (default) [1] INVITE [2] REGISTER [3] SUBSCRIBE [4] INVITE and REGISTER [5] INVITE and SUBSCRIBE [6] OPTIONS
Message Condition [IP2IPRouting_MessageCondition]	Selects a Message Condition rule. To configure Message Condition rules, see 'Configuring Condition Rules' on page 278.
ReRoute IP Group ID [IP2IPRouting_ReRouteIPGroupID]	Defines the IP Group that initiated (sent) the SIP redirect response (e.g., 3xx) or REFER message. This field is typically used for re-routing requests (e.g., INVITEs) when interworking is required for SIP 3xx redirect responses or REFER messages (for more information, see 'Interworking SIP 3xx Redirect Responses' on page 253 and 'Interworking SIP REFER Messages' on page 255, respectively). This parameter functions together with the 'Call Trigger' field (see below). The default is -1 (i.e., not configured).
Call Trigger [IP2IPRouting_Trigger]	Defines the reason (i.e, trigger) for re-routing the SIP request: <ul style="list-style-type: none"> [0] Any = (Default) This routing rule is used for all scenarios (re-routes and non-re-routes). [1] 3xx = Re-routes the request if it was triggered as a result of a SIP 3xx response. [2] REFER = Re-routes the INVITE if it was triggered as a

Parameter	Description
	<p>result of a REFER request.</p> <ul style="list-style-type: none"> ▪ [3] 3xx or REFER = Applies to options [1] and [2]. ▪ [4] Initial only = This routing rule is used for regular requests that the device forwards to the destination. This rule is not used for re-routing of requests triggered by the receipt of REFER or 3xx.
Operation Routing Rule	
Destination Type [IP2IPRouting_DestType]	<p>Determines the destination type to which the outgoing SIP dialog is sent.</p> <ul style="list-style-type: none"> ▪ [0] IP Group = (Default) The SIP dialog is sent to the IP Group's Proxy Set (SERVER-type IP Group) or registered contact from the database (if USER-type IP Group). ▪ [1] Dest Address = The SIP dialog is sent to the address configured in the following fields: 'Destination SRD ID', 'Destination Address', 'Destination Port', and 'Destination Transport Type'. ▪ [2] Request URI = The SIP dialog is sent to the address indicated in the incoming Request-URI. If the fields 'Destination Port' and 'Destination Transport Type' are configured, the incoming Request-URI parameters are overridden and these fields take precedence. ▪ [3] ENUM = An ENUM query is sent to include the destination address. If the fields 'Destination Port' and 'Destination Transport Type' are configured, the incoming Request-URI parameters are overridden and these fields take precedence. ▪ [4] Hunt Group = Used for call center survivability. For more information, see 'Call Survivability for Call Centers' on page 262. ▪ [5] Dial Plan = The IP destination is determined by a Dial Plan index of the loaded Dial Plan file. The syntax of the Dial Plan index in the Dial Plan file is as follows: <destination / called prefix number>,0,<IP destination> Note that the second parameter "0" is ignored. An example of a configured Dial Plan (# 6) in the Dial Plan file is shown below: <pre>[PLAN6] 200,0,10.33.8.52 ; called prefix 200 is routed to destination 10.33.8.52 201,0,10.33.8.52 300,0,itsp.com ; called prefix 300 is routed to destination itsp.com</pre> Once the Dial Plan is defined, you need to assign it (0 to 7) to the routing rule as the destination in the 'Destination Address' parameter, where "0" denotes [PLAN1], "1" denotes [PLAN2], and so on. ▪ [7] LDAP = LDAP-based routing.
Destination IP Group ID [IP2IPRouting_DestIPGroupID]	<p>Defines the IP Group ID to where you want to route the call. The SIP dialog messages are sent to the IP address defined for the Proxy Set associated with this IP Group. If you select an IP Group, it is unnecessary to configure a destination IP address (in the 'Destination Address' field). However, if both parameters are configured, then the IP Group takes precedence.</p>

Parameter	Description
	<p>If the destination IP Group is of USER type, the device searches for a match between the Request-URI (of the received SIP dialog) to an AOR registration record in the device's database. The SIP dialog is then sent to the IP address of the registered contact. The default is -1.</p> <p>Notes:</p> <ul style="list-style-type: none"> This parameter is only relevant if the parameter 'Destination Type' is set to 'IP Group'. However, regardless of the settings of the parameter 'Destination Type', the IP Group is still used - only for determining the IP Profile or outgoing SRD. If neither IP Group nor SRD are defined in this table, the destination SRD is determined according to the source SRD associated with the Source IP Group (configured in the IP Group table, see 'Configuring IP Groups' on page 191). If this table does not define an IP Group but only an SRD, then the first IP Group associated with this SRD (in the IP Group table) is used. If the selected destination IP Group ID is type SERVER, the request is routed according to the IP Group addresses. If the selected destination IP Group ID is type USER, the request is routed according to the IP Group specific database (i.e., only to registered users of the selected database). If the selected destination IP Group ID is ANY USER ([-2]), the request is routed according to the general database (i.e., any matching registered user).
Destination SRD ID [IP2IPRouting_DestSRDID]	<p>Defines the SRD ID. The default is None.</p> <p>Note: The destination IP Group must belong to the destination SRD, if both are configured in this table.</p>
Destination Address [IP2IPRouting_DestAddress]	<p>Defines the destination IP address (or domain name, e.g., domain.com) to where the call is sent.</p> <p>Notes:</p> <ul style="list-style-type: none"> This parameter is applicable only if the parameter 'Destination Type' is set to 'Dest Address' [1]. When using domain names, enter a DNS server IP address or alternatively, define these names in the 'Internal DNS Table' (see 'Configuring the Internal SRV Table' on page 116).
Destination Port [IP2IPRouting_DestPort]	<p>Defines the destination port to where the call is sent.</p>
Destination Transport Type [IP2IPRouting_DestTransportType]	<p>Defines the transport layer type for sending the call:</p> <ul style="list-style-type: none"> [-1] Not Configured (default) [0] UDP [1] TCP [2] TLS <p>Note: When this parameter is set to -1, the transport type is determined by the parameter SIPTransportType.</p>

Parameter	Description
Alternative Route Options [IP2IPRouting_AltRouteOptions]	<p>Determines whether this routing rule is the main routing rule or an alternative routing rule (to the rule defined directly above it in the table).</p> <ul style="list-style-type: none"> ▪ [0] Route Row (default) = Main routing rule - the device first attempts to route the call to this route if the incoming SIP dialog's input characteristics matches this rule. ▪ [1] Alt Route Ignore Inputs = If the call cannot be routed to the main route (Route Row), the call is routed to this alternative route regardless of the incoming SIP dialog's input characteristics. ▪ [2] Alt Route Consider Inputs = If the call cannot be routed to the main route (Route Row), the call is routed to this alternative route only if the incoming SIP dialog matches this routing rule's input characteristics. <p>Notes:</p> <ul style="list-style-type: none"> ▪ The alternative routing entry ([1] or [2]) must be defined in the next consecutive table entry index to the Route Row entry (i.e., directly below it). For example, if Index 4 is configured as a Route Row, Index 5 must be configured as the alternative route. ▪ For IP-to-IP alternative routing, configure alternative routing reasons upon receipt of 4xx, 5xx, and 6xx SIP responses (see Configuring Alternative Routing Reasons on page 286). However, if no response, ICMP, or a SIP 408 response is received, the device attempts to use the alternative route even if no entries are configured in the 'SBC Alternative Routing Reasons' table. ▪ Multiple alternative route entries can be configured (e.g., Index 1 is the main route - Route Row - and indices 2 through 4 are configured as alternative routes).
Cost Group [IP2IPRouting_CostGroup]	<p>Assigns a Cost Group to the routing rule for determining the cost of the call. To configure Cost Groups, see 'Configuring Cost Groups' on page 179.</p> <p>By default, no Cost Group is assigned to the rule.</p>

24.5.4 Blocking Calls from Unregistered SAS Users

To prevent malicious calls, for example, service theft, it is recommended to configure the feature for blocking SIP INVITE messages received from SAS users that are not registered in the SAS database. This applies to SAS in normal and emergency states.

➤ To block calls from unregistered SAS users:

1. Open the SAS Configuration page (**Configuration** tab > **VoIP** menu > **SAS Stand Alone Survivability**).
2. From the 'SAS Block Unregistered Users' drop-down list, select **Block**.
3. Click **Submit** to apply your changes.

24.5.5 Configuring SAS Emergency Calls

You can configure SAS to route emergency calls (such as 911 in North America) directly to the PSTN through its . Thus, even during a communication failure with the external proxy, enterprise UAs can still make emergency calls.

You can define up to four emergency numbers, where each number can include up to four digits. When SAS receives a SIP INVITE (from a UA) that includes one of the user-defined emergency numbers in the SIP user part, it forwards the INVITE directly to the default gateway (see 'SAS Routing in Emergency State' on page 313). The default gateway is defined in the 'SAS Default Gateway IP' field, and this is the device itself. The device then sends the call directly to the PSTN.

This feature is applicable to SAS in normal and emergency states.

➤ To configure SAS emergency numbers:

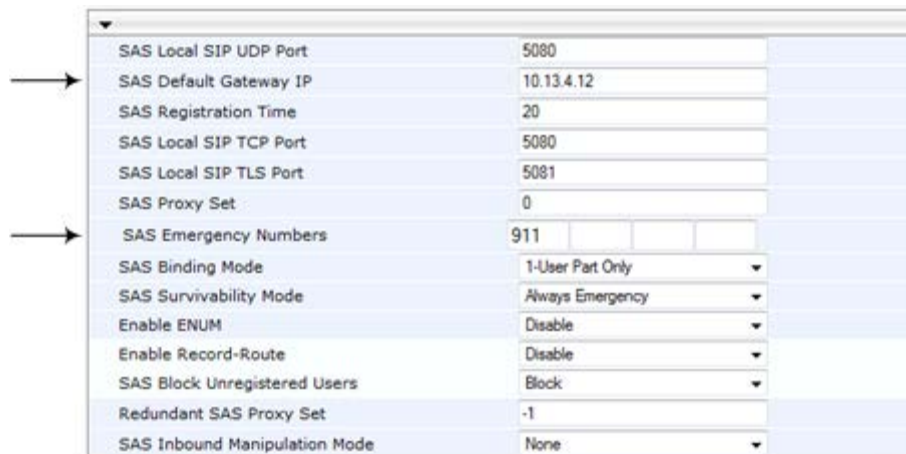
1. Open the SAS Configuration page (**Configuration** tab > **VoIP** menu > **SAS** > **Stand Alone Survivability**).
2. In the 'SAS Default Gateway IP' field, define the IP address and port (in the format *x.x.x.x:port*) of the device (Gateway application).



Note: The port of the device is defined in the 'SIP UDP/TCP/TLS Local Port' field in the SIP General Parameters page (**Configuration** tab > **VoIP** menu > **SIP Definitions** > **General Parameters**).

3. In the 'SAS Emergency Numbers' field, enter an emergency number in each field box.

Figure 24-11: Configuring SAS Emergency Numbers



SAS Local SIP UDP Port	5080
SAS Default Gateway IP	10.13.4.12
SAS Registration Time	20
SAS Local SIP TCP Port	5080
SAS Local SIP TLS Port	5081
SAS Proxy Set	0
SAS Emergency Numbers	911
SAS Binding Mode	1-User Part Only
SAS Survivability Mode	Always Emergency
Enable ENUM	Disable
Enable Record-Route	Disable
SAS Block Unregistered Users	Block
Redundant SAS Proxy Set	-1
SAS Inbound Manipulation Mode	None

4. Click **Submit** to apply your changes.

24.5.6 Adding SIP Record-Route Header to SIP INVITE

You can configure SAS to add the SIP Record-Route header to SIP requests (e.g. INVITE) received from enterprise UAs. SAS then sends the request with this header to the proxy. The Record-Route header includes the IP address of the SAS application. This ensures that future requests in the SIP dialog session from the proxy to the UAs are routed through the SAS application. If not configured, future request within the dialog from the proxy are sent directly to the UAs (and do not traverse SAS). When this feature is enabled, the SIP Record-Route header includes the URI "lr" parameter, indicating loose routing, as shown in the following example:

```
Record-Route: <sip:server10.biloxi.com;lr>
```



Note: This feature is applicable only to the SAS Outbound mode.

➤ **To enable the Record-Route header:**

1. Open the SAS Configuration page (**Configuration** tab > **VoIP** menu > **SAS** > **Stand Alone Survivability**).
2. From the 'Enable Record-Route' drop-down list, select **Enable**.
3. Click **Submit** to apply your changes.

24.5.7 Re-using TCP Connections

You can enable the SAS application to re-use the same TCP connection for sessions (multiple SIP requests / responses) with the same SIP UA. The benefits of this feature include less CPU and memory usage because fewer TCP connections are open and reduced network congestion. For example, assume User A sends a REGISTER message to SAS with transport=TCP, and User B sends an INVITE message to A using SAS. In this scenario, the SAS application forwards the INVITE request using the same TCP connection that User A initially opened with the REGISTER message.

➤ **To re-use TCP connection sessions in SAS**

1. Open the SAS Configuration page (**Configuration** tab > **VoIP** menu > **SAS** > **Stand Alone Survivability**).
2. From the 'SAS Connection Reuse' drop-down list, select **Enable**.
3. Click **Submit** to apply your changes.

24.5.8 Replacing Contact Header for SIP Messages

You can configure SAS to change the SIP Contact header so that it points to the SAS host. This ensures that in the message, the top-most SIP Via header and the Contact header point to the same host.



Notes:

- This feature is applicable only to the SAS Outbound mode.
- The device may become overloaded if this feature is enabled, as all incoming SIP dialog requests traverse the SAS application.

Currently, this feature can be configured only by the *ini* file parameter, `SASEnableContactReplace`:

- **[0]** (Default): Disable - when relaying requests, SAS adds a new Via header (with the IP address of the SAS application) as the top-most Via header and retains the original Contact header. Thus, the top-most Via header and the Contact header point to different hosts.
- **[1]**: Enable - SAS changes the Contact header so that it points to the SAS host and therefore, the top-most Via header and the Contact header point to the same host.

24.6 Viewing Registered SAS Users

You can view all the users that are registered in the SAS registration database. This is displayed in the 'SAS/SBC Registered Users' page, as described in 'Viewing Registered Users' on page [397](#).



Note: You can increase the maximum number of registered SAS users, by implementing the SAS Cascading feature, as described in 'SAS Cascading' on page [335](#).

25 SAS Cascading

The SAS Cascading feature allows you to increase the number of SAS users above the maximum supported by the SAS gateway. This is achieved by deploying multiple SAS gateways in the network. For example, if the SAS gateway supports up to 600 users, but your enterprise has 1,500 users, you can deploy three SAS gateways to accommodate all users: the first SAS gateway can service 600 registered users, the second SAS gateway the next 600 registered users, and the third SAS gateway the rest (i.e., 300 registered users).

In SAS Cascading, the SAS gateway first attempts to locate the called user in its SAS registration database. Only if the user is not located, does the SAS gateway send it on to the next SAS gateway according to the SAS Cascading configuration.

There are two methods for configuring SAS Cascading. This depends on whether the users can be identified according to their phone extension numbers:

- **SAS Routing Table:** If users can be identified with unique phone extension numbers, then the SAS Routing table is used to configure SAS Cascading. This SAS Cascading method routes calls directly to the SAS Gateway (defined by IP address) to which the called SAS user is registered.

The following is an example of a SAS Cascading deployment of users with unique phone extension numbers:

- users registered to the first SAS gateway start with extension number "40"
- users registered to the second SAS gateway start with extension number "20"
- users registered to the third SAS gateway start with extension number "30"

The SAS Routing table rules for SAS Cascading are created using the destination (called) extension number prefix (e.g., "30") and the destination IP address of the SAS gateway to which the called user is registered. Such SAS routing rules must be configured at each SAS gateway to allow routing between the SAS users. The routing logic for SAS Cascading is similar to SAS routing in Emergency state (see the flowchart in 'SAS Routing in Emergency State' on page 313). For a description on the SAS Routing table, see 'SAS Routing Based on IP-to-IP Routing Table' on page 326.

The figure below illustrates an example of a SAS Cascading call flow configured using the SAS Routing table. In this example, a call is routed from SAS Gateway (A) user to a user on SAS Gateway (B).

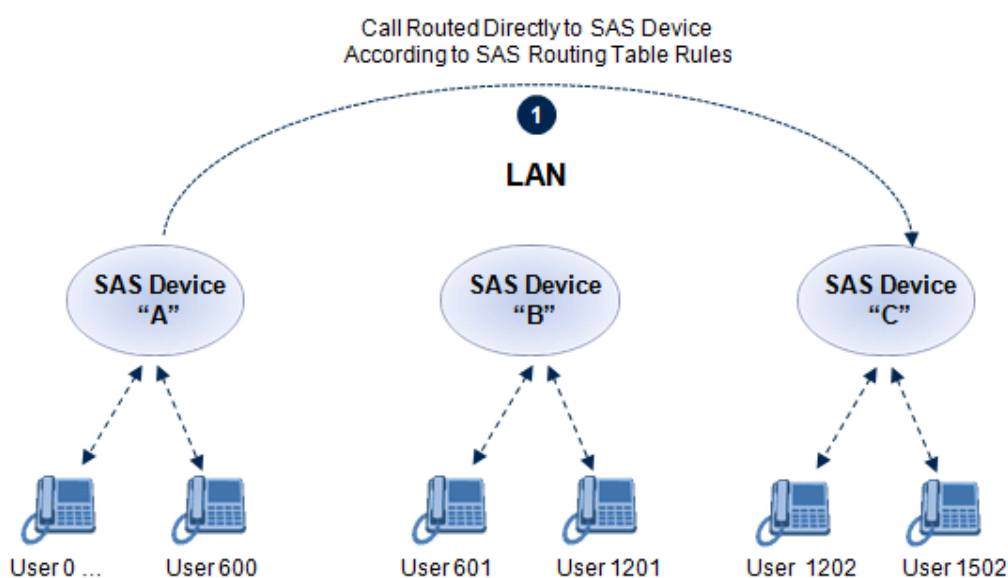


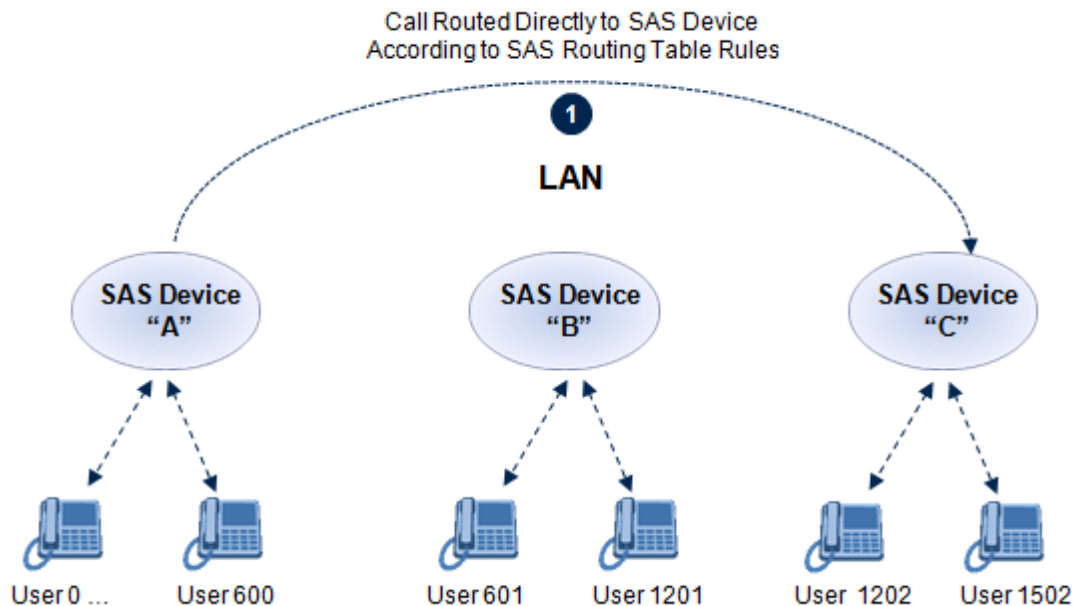
Figure 25-1: SAS Cascading Using SAS Routing Table - Example

- SAS Redundancy mode:** If users cannot be distinguished (i.e., associated to a specific SAS gateway), then the SAS Redundancy feature is used to configure SAS Cascading. This mode routes the call in a loop fashion, from one SAS gateway to the next, until the user is located. Each SAS gateway serves as the redundant SAS gateway (“redundant SAS proxy server”) for the previous SAS gateway (in a one-way direction). For example, if a user calls a user that is not registered on the same SAS gateway, the call is routed to the second SAS gateway, and if not located, it is sent to the third SAS gateway. If the called user is not located on the third (or last) SAS gateway, it is then routed back to the initial SAS gateway, which then routes the call to the default gateway (i.e., to the PSTN).

Each SAS gateway adds its IP address to the SIP via header in the INVITE message before sending it to the next (“redundant”) SAS gateway. If the SAS gateway receives an INVITE and its IP address appears in the SIP via header, it sends it to the default gateway (and not to the next SAS gateway), as defined by the SASDefaultGatewayIP parameter. Therefore, this mode of operation prevents looping between SAS gateways when a user is not located on any of the SAS gateways.

The figure below illustrates an example of a SAS Cascading call flow when configured using the SAS Redundancy feature. In this example, a call is initiated from a SAS Gateway (A) user to a user that is not located on any SAS gateway. The call is subsequently routed to the PSTN.

Figure 25-2: SAS Cascading Using SAS Redundancy Mode - Example



Part VIII

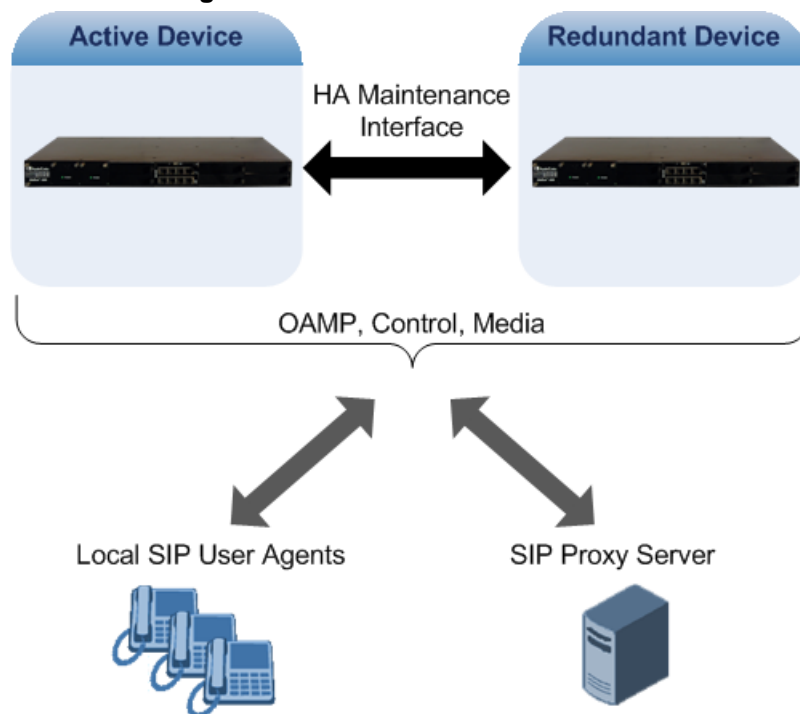
High Availability System

26 HA Overview

The device's High Availability (HA) feature provides 1+1 system redundancy using two Mediant 4000 devices. If failure occurs in the active device, a switchover occurs to the redundant device which takes over the call handling process. Thus the continuity of call services is ensured. All active calls (signaling and media) are maintained upon switchover.

The figure below illustrates the Active-Redundant HA devices under normal operation. Communication between the two devices is through a Maintenance interface, having a unique IP address for each device. The devices have identical software and configuration including network interfaces (i.e., OAMP, Control, and Media), and have identical local-port cabling of these interfaces.

Figure 26-1: Two Devices in HA State



26.1 Connectivity and Synchronization between Devices

In HA mode, the Ethernet connectivity between the two devices is through a special LAN interface on each device, referred to as the *Maintenance* interface. Each device has its own Maintenance interface with a unique address, and each device knows the Maintenance address of the other. The Maintenance interface can use a dedicated Ethernet port group or share the same Ethernet port group with the other network interface types (i.e., OAMP, Media, and Control).

When only one of the devices is operational it is in HA stand-alone state. This means that the device has no connectivity to the second device. When the second device is powered up, it recognizes the active device through the Maintenance network and acquires the HA redundant state. It then begins synchronizing for HA with the active device through the Maintenance network. During synchronization, the active device sends the redundant device its current configuration settings, including auxiliary files. The active device also sends its software file (.cmp) if the redundant device is running a different software version. Once loaded to the redundant device, the redundant device reboots to apply the new configuration and/or software.

Thus, under normal operation, one of the devices is in active state while the other is in redundant state, where both devices share the same configuration and software. Any

subsequent configuration update or software upgrade on the active device is also done on the redundant device.

In the active device, all logical interfaces (i.e., Media, Control, OAMP, and Maintenance) are active. In the redundant device, only the Maintenance interface is active, which is used for connectivity to the active device. Therefore, management is done only through the active device. Upon a failure in the active device, the redundant device becomes active and activates all its logical interfaces exactly as was used on the active device.

26.2 Device Switchover upon Failure

When a failure occurs in the active device, a switchover occurs to the redundant device making it the new active device. Whether a switchover is later done back to the repaired failed device, depends on whether you have enabled the Revertive mode:

- **Revertive mode enabled:** The Revertive mode specifies one of the device's as the "preferred" device between the two devices. This is done by assigning a priority level to each device (1 to 10, where 1 is the lowest). Whenever the device with higher priority recovers from a failure, it first becomes the redundant device but then initiates a switchover to become the active device once again; otherwise, after recovery, it becomes the redundant device and remains as redundant. If you change the priority level of the redundant device to one that is higher than the active device and then reset the redundant device, a switchover occurs to the redundant device making it the active device and the "preferred" device. If both devices are configured with the same priority level, then Revertive mode is irrelevant.
- **Revertive mode disabled:** A switchover is done only upon failure of the currently active device.

Failure detection by the devices is done by the constant keep-alive messages they send between themselves to verify connectivity. Upon detection of a failure in one of the devices, the following occurs:

- **Failure in active device:** The redundant device initiates a switchover. The failed device resets and the previously redundant device becomes the active device in stand-alone mode. If at a later stage this newly active device detects that the failed device has been repaired, the system returns to HA mode. If Revertive mode is enabled and the originally active device was configured with a higher priority, a switchover occurs to this device; otherwise, if it was configured with a lower priority (or Revertive mode was disabled), the repaired device is initialized as the redundant device.
- **Failure in redundant device:** The active device moves itself into stand-alone mode until the redundant device is returned to operation. If the failure in the redundant device is repaired after reset, it's initialized as the redundant device once again and the system returns to HA mode.

Connectivity failure triggering a switchover can include, for example, one of the following:

- **Loss of physical (link) connectivity:** If one or more physical network groups (i.e., Ethernet port pair) used for one or more network interfaces of the active device disconnects (i.e., no link) and these physical network groups are connected OK on the redundant device, then a switchover occurs to the redundant device.
- **Loss of network (logical) connectivity:** No network connectivity, verified by keep-alive packets between the devices. This applies only to the Maintenance interface.

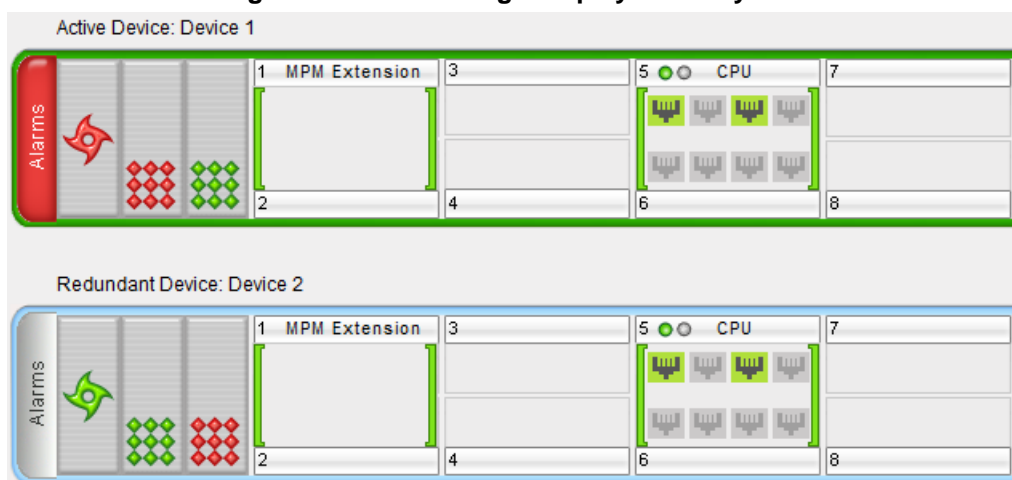


Note: Switchover triggered by loss of physical connectivity in one or more Ethernet port-group is not done if the active device has been set to a Revertive priority level of 10. In such a scenario, the device remains active despite the loss of connectivity in one or more of its Ethernet port groups.

26.3 HA Status

The device's Web Home page displays the status of the HA system. The Home page provides a graphical display of both active and redundant devices:

Figure 26-2: Home Page Display of HA System



- Active device:
 - Color border: The active device is surrounded by a green border.
 - Title: The default title of the device is Active Device: "Device 1".
- Redundant device:
 - Color border: The redundant device is surrounded by a blue border.
 - Title: The default title of the device is Redundant Device: "Device 2".

The title of each device can be configured as described below:

➤ **To define a name for the device:**

1. Open the HA Settings page (**Configuration** tab > **System** menu > **HA Settings**).
2. In the 'HA Device Name' field, enter a name for the device, and then click **Submit**.

The Home page also displays the HA operational status of the device to which you are currently logged in. This is displayed in the 'High Availability' field under the General Information pane:

- "Not Operational": HA is not configured or the device is not installed with the HA Software License Key
- "Synchronizing": Redundant device is synchronizing with Active device
- "Operational": The device is in HA mode
- "Stand Alone": HA is configured but the Redundant device is missing and HA is currently unavailable
- "Not Available": HA is not configured correctly (error)

Reader's Notes

27 HA Configuration

This section describes the configuration of the HA system.

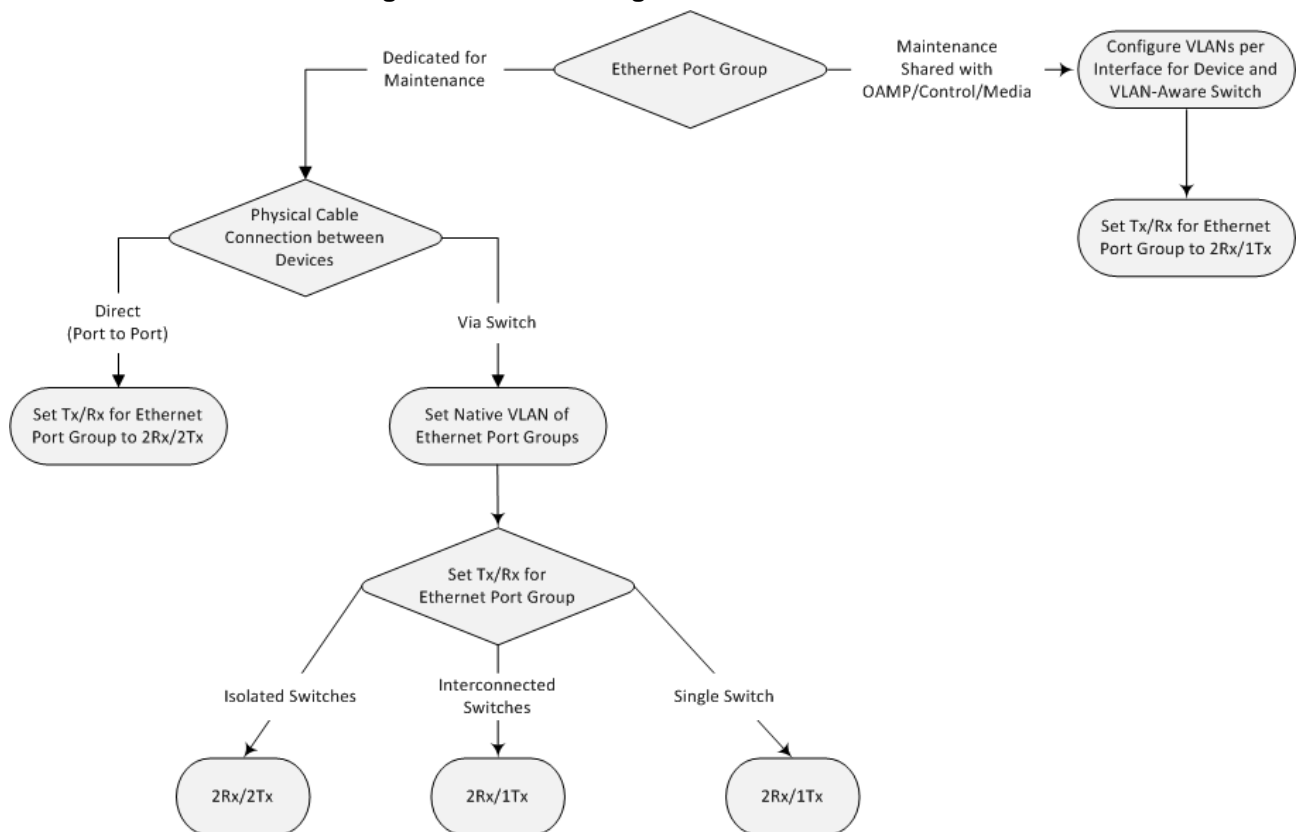
27.1 Initial HA Configuration

By default, HA is disabled on the device. When a device is loaded with valid HA configuration and it is the first device to be loaded, it becomes the active device. The second device that is loaded with HA configuration becomes the redundant (standby) device.

27.1.1 Network Topology Types

The initial configuration of HA depends on how you want to deploy your HA system in the network. The flowchart below shows the different configuration setups according to network topology:

Figure 27-1: HA Configuration Flowchart

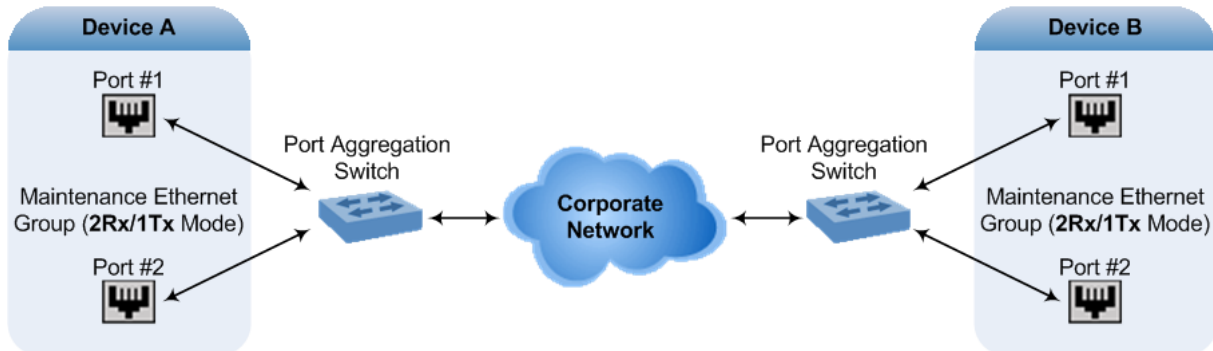


27.1.1.1 Tx/Rx Port Settings

The required transmit (TX) / receive (Rx) mode that you need to configure for the port pair in the Ethernet Port Group used by the Maintenance interface depends on whether this Ethernet Port Group is shared or not with the other network interfaces (i.e., OAMP, and/or Control, and/or Media):

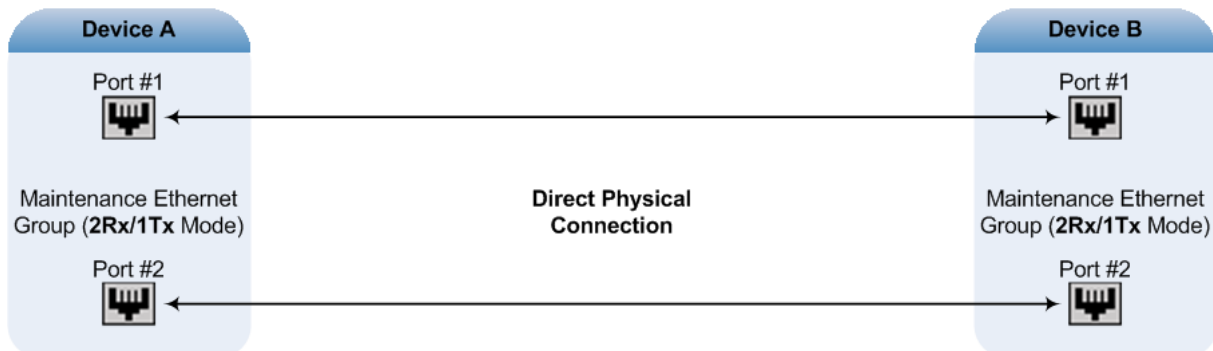
- For Geographical HA (both units are located far from each other), **2Rx/1Tx** port mode connected to a port aggregation switch is the recommended option:

Figure 27-2: Rx/Tx Mode for Geographical HA



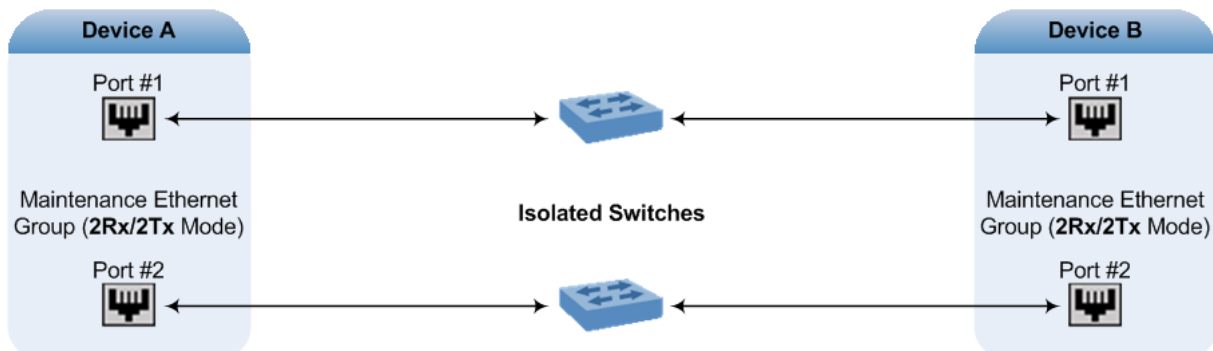
- If the Maintenance ports of both devices are connected directly to each other without intermediation of switches, configure the mode to **2RX/1TX**:

Figure 27-3: Rx/Tx Mode for Direct Connection



- If the two devices are connected through two (or more) isolated LAN switches (i.e., packets from one switch cannot traverse the second switch), configure the mode to **2RX/2TX**:

Figure 27-4: Redundancy Mode for Two Isolated Switches



Notes:

- When two LAN switches are used, the LAN switches must be in the same subnet (i.e., broadcast domain).
- To configure Tx/Rx modes of the Ethernet ports, see 'Configuring Tx/Rx for Ethernet Port-Pair Groups' on page 97.

27.1.2 Configuring the Two HA Devices

This section describes how to initially configure the two devices comprising the HA system. This configuration is done in the following chronological order:

1. Configuring the first device for HA - see 'Step 1: Configure the First Device' on page 345
2. Configuring the second device for HA - see 'Step 2: Configure the Second Device' on page 347
3. Activating HA on the devices - see 'Step 3: Initialize HA on the Devices' on page 348



Notes:

- The physical connections of the first and second devices to the network (i.e., Maintenance interface and OAMP, Control and Media interfaces) **must be identical**. This also means that the two devices must also use the same Ethernet Port Groups and the port numbers belonging to these Ethernet Port Groups. For example, if the first device uses Ethernet Port Group 1 (with ports 1 and 2), the second device must also use Ethernet Port Group 1 (with ports 1 and 2).
- Before configuring HA, determine the required network topology, as described in 'Network Topology Types' on page 343.
- For HA support, ensure that both devices are installed with a Software License Key that includes the HA feature.
- It is recommended to avoid using Spanning Tree Protocol (STP) on the Maintenance interface; the Ethernet connectivity of the Maintenance interface between the two devices should be constantly reliable without any disturbances.

27.1.2.1 Step 1: Configure the First Device

The first stage is to configure the first device for HA, as described in the procedure below:



Note: During this stage, ensure that the second device is powered off or disconnected from the network.

➤ To configure the first device for HA:

1. Configure the network interfaces, including the default OAMP interface:
 - a. Connect your PC to the device using a local, direct physical cable connection and then access the Web interface using the default OAMP network address. For more information on initial access, see 'Assigning the OAMP IP Address' on page 19.
 - b. Open the Multiple Interface table (**Configuration** tab > **VoIP** menu > **Network** > **IP Settings**).
 - c. Change the default OAMP network settings to suite your networking scheme.
 - d. Configure the Control and Media network interfaces, as required.
 - e. Add the HA Maintenance interface (i.e., the **MAINTENANCE** Application Type).

Notes:

- The used Ethernet Port Group, selected in the 'Underlying Interface' field, depends on the following:
 - ✓ **Dedicated Ethernet Port Group for Maintenance interface:** Use a different Ethernet Port Group for the Maintenance interface than for the other network interfaces.
 - ✓ **Shared Ethernet Port Group for Maintenance and OAMP/Control/Media interfaces:** Use the same Ethernet Port Group for all these interfaces.
- If you are using an Ethernet Port Group that is shared between the Maintenance interface and the other network interfaces, assign them different VLANs. Ensure that you also configure these VLANs on the third-party, VLAN-aware switch to which the Ethernet Port Group is connected.



The Multiple Interface table below shows an example where the Maintenance interface is assigned to Ethernet Port Group 2 ("GROUP_2") and the other interfaces to Ethernet Port Group 1 ("GROUP_1"):

Figure 27-5: Multiple Interface Table with Maintenance Interface

Index	Application Type	Interface Mode	IP Address	Prefix Length	Gateway	VLAN ID	Interface Name	Primary DNS Server IP Address	Secondary DNS Server IP Address	Underlying
0	OAMP + Media + Control	IPv4 Manual	10.8.244.80	16	10.8.0.1	1	Voice	0.0.0.0	0.0.0.0	GROUP_1
1	MAINTENANCE	IPv4 Manual	10.3.0.51	16	10.3.0.1	2	Unknown	0.0.0.0	0.0.0.0	GROUP_2

2. If the Maintenance interface uses a dedicated Ethernet Port Group and the connection is through a switch, the packets of both interfaces should generally be untagged. In such a scenario, set the Native VLAN ID of each Ethernet Port Group so that it is the same as the VLAN ID set for each interface assigned to that Ethernet Port Group. The Native VLAN ID is configured in the Physical Ports Settings page (see 'Configuring Physical Ethernet Ports' on page 95). The figure below shows an example whereby the Native VLAN IDs of the Ethernet Port Groups are set to the same VLAN IDs of the interfaces using these Ethernet Port Groups:

Figure 27-6: Native VLAN for Ethernet Port Groups of Maintenance and Other Interfaces

OAMP, Control, Media



Index	Port	Mode	Native Vlan	Speed&Duplex	Description	Group Member	Group Status
1	GE_1	Enable	1	Auto Negotiation	User Port #0	GROUP_1	Active
2	GE_2	Enable	1	Auto Negotiation	User Port #1	GROUP_1	Redundant
3	GE_3	Enable	2	Auto Negotiation	User Port #2	GROUP_2	Active
4	GE_4	Enable	2	Auto Negotiation	User Port #3	GROUP_2	Redundant

Maintenance

3. Set the Ethernet port Tx / Rx mode of the Ethernet Port Group used by the Maintenance interface. This is configured in the Ethernet Group Settings page (see 'Configuring Tx/Rx for Ethernet Port-Pair Groups' on page 97). The port mode depends on the type of Maintenance connection between the devices, as described in 'Tx/Rx Port Settings' on page 343.

4. Configure the HA parameters in the HA Settings page (**Configuration** tab > **System** menu > **HA Settings**):

Figure 27-7: HA Settings Page

HA Settings		
HA Remote Address	10.31.4.61	
HA Revertive	Disable	
HA Priority	5	
Redundant HA Priority	5	

- a. In the 'HA Remote Address' field, enter the Maintenance IP address of the **second** device.
 - b. (Optional) Enable the Revertive mode by setting the 'HA Revertive' parameter to **Enable** and then setting the priority level of this device in the 'HA Priority' field.
5. Burn the configuration to flash **without** a reset.
6. Power down the device.
7. Continue to 'Step 2: Configure the Second Device' on page 347 for configuring the second device.

27.1.2.2 Step 2: Configure the Second Device

Once you have configured the first device for HA, you can configure the second device for HA. As the configuration of the second device is similar to the first device, the procedure below briefly describes each procedural step. For detailed configuration such as the path to the Web configuration pages, refer to the section on configuring the first device ('Step 1: Configure the First Device' on page 345).



Note: During this stage, ensure that the first device is powered off or disconnected from the network.

➤ **To configure the second device for HA:**

1. Connect to the device in the same way as you did with the first device.
2. Configure the **same** OAMP, Media, and Control interfaces as you configured for the first device.
3. Configure a Maintenance interface for this device. The IP address must be different to that configured for the Maintenance interface of the first device. However, the Maintenance interfaces of the devices must be in the same subnet.
4. Configure the **same** Native VLAN IDs of the Ethernet Port Groups and VLAN IDs of the network interfaces as you configured for the first device.
5. Configure the **same** Ethernet port Tx / Rx mode of the Ethernet Port Group used by the Maintenance interface as you configured for the first device.
6. Configure the HA parameters in the HA Settings page (**Configuration** tab > **System** menu > **HA Settings**):
 - a. In the 'HA Remote Address' field, enter the Maintenance IP address of the **first** device.
 - b. (Optional) Enable the Revertive mode by setting the 'HA Revertive' parameter to **Enable** and then setting the priority level of this second device in the 'Redundant HA Priority' field.
7. Burn the configuration to flash **without** a reset.

8. Power down the device.
9. Continue to 'Step 3: Initialize HA on the Devices' on page 348 for completing the HA configuration.

27.1.2.3 Step 3: Initialize HA on the Devices

Once you have configured both devices for HA as described in the previous sections, follow the procedure below to complete and initialize HA so that the devices become operational in HA. This last stage applies to both devices.

➤ **To initialize the devices for HA:**

1. Cable the devices to the network.



Note: You must connect both ports (two) in the Ethernet Port Group of the Maintenance interface to the network (i.e., two network cables are used). This provides 1+1 Maintenance port redundancy.

2. Power up the devices; the redundant device synchronizes with the active device and updates its configuration according to the active device. The synchronization status is indicated as follows:

- Active device: The Web interface's Home page displays the HA status as "Synchronizing".
- Redundant device: The ↔ LED is lit yellow on the E-SBC module.

When synchronization completes successfully, the redundant device resets to apply the received configuration and software.

When both devices become operational in HA, the HA status is indicated as follows:

- Both devices: The Web interface's Home page displays the HA status as "Operational".
 - Active device: The ↔ LED is lit green.
 - Redundant device: The ↔ LED flashes yellow.
3. Access the active device with its OAMP IP address and configure the device as required. For information on configuration done after HA is operational, see 'Configuration while HA State is Operational' on page 348.

27.2 Configuration while HA is Operational

When the devices are operating in HA state, subsequent configuration is as follows:

- All configuration, including HA is done on the active device **only**.
- Non-HA configuration on the active device is automatically updated on the redundant device (through the Maintenance interface).
- HA-related configuration on the active device is automatically updated on the redundant device:
 - Maintenance interface:
 - ◆ Modified Maintenance interface address of the active device: this address is set as the new 'HA Remote Address' value on the redundant device.
 - ◆ Modified 'HA Remote Address' value on the active device: this address is set as the new Maintenance interface address on the redundant device. This requires a device reset.

- ◆ Modifications on all other Maintenance interface parameters (e.g., Default Gateway and VLAN ID): updated to the Maintenance interface on the redundant device.
- 'HA Revertive' mode (this requires a device reset).
- 'HA Priority' parameter is set for the active device.
- Modified 'Redundant HA Priority' value is set for the redundant device. This requires a device reset.



Note: If the HA system is already in Revertive mode and you want to change the priority of the device, to ensure that system service is maintained and traffic is not disrupted, it is recommended to set the higher priority to the redundant device and then reset it. After it synchronizes with the active device, it initiates a switchover and becomes the new active device (the former active device resets and becomes the new redundant device).

27.3 Monitoring IP Entity and HA Switchover upon Ping Failure

The device can monitor a specified network entity, using pings. If the device does not receive a ping response from the entity, a switchover to the redundant device occurs. The switchover happens only if a ping was initially successful and then a subsequent ping failed. This feature can be used, for example, to check connectivity with a nearby router (first hop) that the device uses to reach other destinations.

The network entity is defined by IP address. The IP interface from where the ping is sent can be selected from one of the device's configured network interfaces (in the Multiple Interface table).



Notes:

- The ping feature is not functional under the following conditions:
 - ✓ HA is disabled (i.e., active device is in standalone mode).
 - ✓ HA Priority is used (to prevent endless loops of switchovers).
 - ✓ Number of Ethernet Groups in the redundant device that are in "up" state are less than on the active device (to prevent endless loops of switchovers).
- For a detailed description of the HA ping parameters, see 'HA Ping Parameters' on page [456](#).

➤ **To configure monitoring of IP entity using pings:**

1. Open the HA Settings page (**Configuration** tab > **System** menu > **HA Settings**).

Figure 8: HA Settings Page - Monitor Destination Settings

▼ Monitor Destination Settings	
Enable HA Network reachability	Disable ▼
HA Network reachability destination address	0.0.0.0
HA Network reachability source If name	
HA Network reachability ping timeout	1
HA Network reachability ping retries	2

▼ Monitor Destination Status	
HA Network watchdog status: Disabled by configuration and HA state.	

2. Under the Monitor Destination Settings group, do the following:
 - Set the 'Enable HA Network Reachability' field to **Enable**.
 - In the 'HA Network Reachability Destination Address' field, set the address of the IP entity that you want to monitor.
 - In the 'HA Network Reachability source If Name' field, set the device's IP network interface from where you want to ping the destination entity.
 - In the 'HA Network Reachability Ping Timeout' field, set the timeout for which the ping request waits for a response.
 - In the 'HA Network Reachability Ping Retries' field, set the number of ping requests that the device sends after no ping response is received from the destination, before the destination is declared unavailable.
3. Click **Submit**.

If this feature is operational, the status of the connectivity to the pinged destination is displayed in the read-only 'Monitor Destination Status' field:

- "Enabled": Ping is sent as configured.
- "Disabled by configuration and HA state": HA and ping are not configured.
- "Disabled by HA state": same as above.
- "Disabled by configuration": same as above.
- "Disabled by invalid configuration": invalid configuration, for example, invalid interface name or destination address (destination address must be different than a local address and from the redundant device's Maintenance address).
- "Disabled by HA priority in use": when HA priority is used, ping mechanism is disabled.
- "Disabled by Eth groups error": when the number of Ethernet Groups in the redundant device becomes less than in the active device, the ping mechanism is disabled.
- "Failed to be activated": Internal error (failed activating the ping mechanism).

27.4 Configuring Firewall Allowed Rules

If you add firewall rules in the Firewall Settings page (see 'Configuring Firewall Settings' on page 125) that block specified traffic, you also need to add rules that ensure traffic related to the HA feature is allowed. These allowed HA rules include the following:

- Keep-alive packets between the HA devices (e.g., rules #1 and #2 in the figure below).
- HA control and data packets between the HA devices (e.g., rules #3 and #4 in the figure below).
- HA control and data packets between the HA devices after switchover (e.g., rules #5 and #6 in the figure below). These rules are the same as rules #3 and #4 respectively, but are required as the TCP source and destination port IDs are not symmetric.
- HTTP protocol for file transferring (e.g., Rule #7 in the figure below).
- HTTP protocol for file transferring after switchover (e.g., Rule #8 - same as Rule #7 - in the figure below).

The figure below displays an example of the required firewall rules. In this example, 10.31.4.61 is the Maintenance interface of the redundant device and 10.31.4.62 is the Maintenance interface of the active device. "HA_IF" is the name of the Maintenance interface.

Figure 27-9: Allowed Firewall Rules for HA

Edit Rule	Rule Status	Source IP	Source Port	Prefix Length	Local Port Range	Protocol	Use Specific Interface	Interface Name	Packet Size	Byte rate	Burst Bytes	Action Upon Match	Match Count
0	<input type="radio"/> Active	0.0.0.0	0	0	80-80	tcp	Enable	O+M+C	0	0	0	ALLOW	248
1	<input type="radio"/> Active	10.31.4.61	669	32	669-669	udp	Enable	HA_IF	0	0	0	ALLOW	921
2	<input type="radio"/> Active	10.31.4.62	669	32	669-669	udp	Enable	HA_IF	0	0	0	ALLOW	0
3	<input type="radio"/> Active	10.31.4.61	0	32	2442-2442	TCP	Enable	HA_IF	0	0	0	ALLOW	57
4	<input type="radio"/> Active	10.31.4.62	2442	32	0-65535	TCP	Enable	HA_IF	0	0	0	ALLOW	0
5	<input type="radio"/> Active	10.31.4.61	2442	32	0-65535	TCP	Enable	HA_IF	0	0	0	ALLOW	0
6	<input type="radio"/> Active	10.31.4.62	0	32	2442-2442	TCP	Enable	HA_IF	0	0	0	ALLOW	0
7	<input type="radio"/> Active	10.31.4.61	80	32	0-65535	TCP	Enable	HA_IF	0	0	0	ALLOW	0
8	<input type="radio"/> Active	10.31.4.62	80	32	0-65535	TCP	Enable	HA_IF	0	0	0	ALLOW	0
9	<input checked="" type="radio"/> Not Active	0.0.0.0	0	0	0-65535	Any	Disable	None	0	0	0	Block	0

Reader's Notes

28 HA Maintenance

This section describes HA maintenance procedures.

28.1 Maintenance of Redundant Device

The only interface that is operational on the redundant device is the Maintenance interface. For maintenance, there are several protocols available for this interface (unlike the active device which uses the logical OAMP / management interface for these protocols):

- **Syslog:** To receive Syslog messages from the redundant device, ensure that there is a valid VLAN and route configured from the maintenance network to where the Syslog server is located on the network.
- **Telnet:** A Telnet server is always available on the redundant device (even if disabled by configuration).

28.2 Replacing a Failed Device

If you need to replace a non-functional device with a new one, the new device must be configured exactly as the second device, as described in 'Configuring the HA Devices' on page 345.

28.3 Forcing a Switchover

If required, you can force a switchover between active and redundant devices. For more information, see 'High Availability Maintenance' on page 361.

28.4 Software Upgrade

The following types of software upgrades are available on the HA system:

- **Software Upgrade with Device Reset:** Both active and redundant devices burn and reboot with the new software version. This method is quick and simple, but it disrupts traffic (i.e., traffic affecting).
- **Hitless Software Upgrade:** This method maintains service (i.e., not traffic affecting) and is as follows:
 - a. The redundant device burns and resets with the new software version.
 - b. A switchover is done between the active and redundant devices, whereby the redundant device becomes the active one.
 - c. The previously active device burns and resets with the new software version.
 - d. The previously active device switches over to become the active device.

For more information on upgrading the software, see 'Software Upgrade Wizard' on page 373.

Reader's Notes

Part IX

Maintenance

29 Basic Maintenance

The Maintenance Actions page allows you to perform the following:

- Reset the device - see 'Resetting the Device' on page 357
- Lock and unlock the device - see 'Locking and Unlocking the Device' on page 359
- Save configuration to the device's flash memory - see 'Saving Configuration' on page 360

➤ **To access the Maintenance Actions page, do one of the following:**

- On the toolbar, click the **Device Actions** button, and then from the drop-down menu, choose **Reset**.
- On the Navigation bar, click the **Maintenance** tab, and then in the Navigation tree, select the **Maintenance** menu and choose **Maintenance Actions**.

Figure 29-1: Maintenance Actions Page

▼ Reset Configuration	
Reset Board	<input type="button" value="Reset"/>
Burn To FLASH	Yes <input type="button" value="v"/>
Graceful Option	No <input type="button" value="v"/>
▼ LOCK / UNLOCK	
Lock	<input type="button" value="LOCK"/>
Graceful Option	No <input type="button" value="v"/>
Current Admin State	UNLOCKED
▼ Save Configuration	
Burn To FLASH	<input type="button" value="BURN"/>

29.1 Resetting the Device

The Maintenance Actions page allows you to remotely reset the device. In addition, before resetting the device, you can choose the following options:

- Save the device's current configuration to the device's flash memory (non-volatile).
- Perform a graceful shutdown, whereby device reset starts only after a user-defined time (i.e., timeout) or after no more active traffic exists (the earliest thereof).



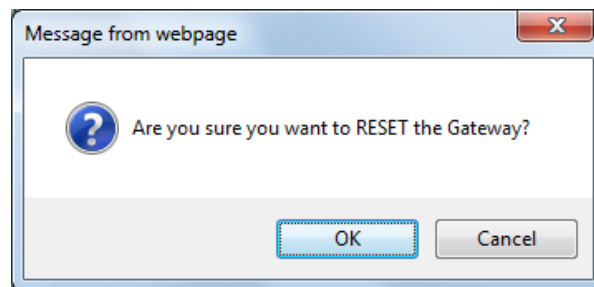
Notes:

- Throughout the Web interface, parameters displayed with a lightning ⚡ symbol are not applied on-the-fly and require that you reset the device for them to take effect.
- When you modify parameters that require a device reset, once you click the **Submit** button in the relevant page, the toolbar displays "Reset" (see 'Toolbar Description' on page 30) to indicate that a device reset is required.
- After you reset the device, the Web GUI is displayed in Basic view (see 'Displaying Navigation Tree in Basic and Full View' on page 31).

➤ **To reset the device:**

1. Open the Maintenance Actions page (see 'Basic Maintenance' on page 357).
2. Under the 'Reset Configuration' group, from the 'Burn To FLASH' drop-down list, select one of the following options:
 - **Yes:** The device's current configuration is saved (*burned*) to the flash memory prior to reset (default).
 - **No:** Resets the device without saving the current configuration to flash (discards all unsaved modifications).
3. Under the 'Reset Configuration' group, from the 'Graceful Option' drop-down list, select one of the following options:
 - **Yes:** Reset starts only after the user-defined time in the 'Shutdown Timeout' field (see Step 4) expires or after no more active traffic exists (the earliest thereof). In addition, no new traffic is accepted.
 - **No:** Reset starts regardless of traffic, and any existing traffic is terminated at once.
4. In the 'Shutdown Timeout' field (relevant only if the 'Graceful Option' in the previous step is set to **Yes**), enter the time after which the device resets. Note that if no traffic exists and the time has not yet expired, the device resets.
5. Click the **Reset** button; a confirmation message box appears, requesting you to confirm.

Figure 29-2: Reset Confirmation Message Box



6. Click **OK** to confirm device reset; if the parameter 'Graceful Option' is set to **Yes** (in Step 3), the reset is delayed and a screen displaying the number of remaining calls and time is displayed. When the device begins to reset, a message appears notifying you of this.

29.2 Remotely Resetting Device using SIP NOTIFY

The device can be remotely reset upon the receipt of a SIP NOTIFY that includes an Event header set to 'check-sync;reboot=true', as shown in the example below:

```
NOTIFY sip:<user>@<dsthost> SIP/2.0
To: sip:<user>@<dsthost>
From: sip:sipsak@<srchost>
CSeq: 10 NOTIFY
Call-ID: 1234@<srchost>
Event: check-sync;reboot=true
```

➤ **To enable remote reset upon receipt of SIP NOTIFY:**

1. Open the Advanced Parameters page (**Configuration** tab > **VoIP** menu > **SIP Definitions** > **Advanced Parameters**).
2. Under the Misc Parameters group, set the 'SIP Remote Rest' parameter to **Enable**.
3. Click **Submit**.



Note: This SIP Event header value is proprietary to AudioCodes.

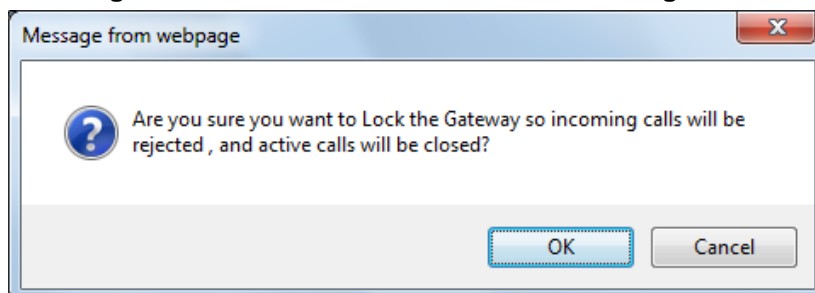
29.3 Locking and Unlocking the Device

The Lock and Unlock option allows you to lock the device so that it doesn't accept any new calls and maintains only the current calls. This is useful when, for example, you are uploading new software files to the device and you don't want any traffic to interfere with the process.

➤ **To lock the device:**

1. Open the Maintenance Actions page (see 'Basic Maintenance' on page 357).
 2. Under the 'LOCK / UNLOCK' group, from the 'Graceful Option' drop-down list, select one of the following options:
 - **Yes:** The device is 'locked' only after the user-defined time in the 'Lock Timeout' field (see Step 3) expires or no more active traffic exists (the earliest thereof). In addition, no new traffic is accepted.
 - **No:** The device is 'locked' regardless of traffic. Any existing traffic is terminated immediately.
- Note:** These options are only available if the current status of the device is in the Unlock state.
3. In the 'Lock Timeout' field (relevant only if the parameter 'Graceful Option' in the previous step is set to **Yes**), enter the time (in seconds) after which the device locks. Note that if no traffic exists and the time has not yet expired, the device locks.
 4. Click the **LOCK** button; a confirmation message box appears requesting you to confirm device Lock.

Figure 29-3: Device Lock Confirmation Message Box



5. Click **OK** to confirm device Lock; if 'Graceful Option' is set to **Yes**, the lock is delayed and a screen displaying the number of remaining calls and time is displayed. Otherwise, the lock process begins immediately. The 'Current Admin State' field displays the current state - "LOCKED" or "UNLOCKED".

➤ **To unlock the device:**

1. Open the Maintenance Actions page (see 'Maintenance Actions' on page 357).
2. Under the 'LOCK / UNLOCK' group, click the **UNLOCK** button. Unlock starts immediately and the device accepts new incoming calls.



Note: The Home page's General Information pane displays whether the device is locked or unlocked (see 'Viewing the Home Page' on page 49).

29.4 Saving Configuration

The Maintenance Actions page allows you to save (*burn*) the current parameter configuration (including loaded auxiliary files) to the device's *non-volatile* memory (i.e., flash). The parameter modifications that you make throughout the Web interface's pages are temporarily saved (to the *volatile* memory - RAM) when you click the **Submit** button on these pages. Parameter settings that are saved only to the device's RAM revert to their previous settings after a hardware/software reset (or power failure). Therefore, to ensure that your configuration changes are retained, you must save them to the device's flash memory using the burn option described below.

➤ **To save the changes to the non-volatile flash memory :**

1. Open the Maintenance Actions page (see 'Basic Maintenance' on page 357).
2. Under the 'Save Configuration' group, click the **BURN** button; a confirmation message appears when the configuration successfully saves.



Notes:

- Saving configuration to the *non-volatile* memory may disrupt current traffic on the device. To avoid this, disable all new traffic before saving, by performing a graceful lock (see 'Locking and Unlocking the Device' on page 359).
- Throughout the Web interface, parameters displayed with the lightning ⚡ symbol are not applied on-the-fly and require that you reset the device for them to take effect (see 'Resetting the Device' on page 357).
- The Home page's General Information pane displays whether the device is currently "burning" the configuration (see 'Viewing the Home Page' on page 49).

30 High Availability Maintenance

The High Availability Maintenance page allows you to perform a switch-over between the Active and Redundant devices. It also allows you to reset the Redundant device.

**Notes:**

- When performing a switchover or a reset on a Redundant device, the HA mode becomes temporarily unavailable.
- This page can also be accessed from the toolbar's **Device Actions** drop-down menu options - **Switch Over** and **Reset Redundant** (see 'Toolbar Description' on page 30).

➤ **To perform a switch-over:**

1. Open the High Availability Maintenance page (**Maintenance** tab > **Maintenance** menu > **High Availability Maintenance**).

Figure 30-1: High Availability Maintenance Page

▼ Switch Over	
Switch Between Active And Redundant Boards	<input type="button" value="Switch Over"/>
▼ Redundant Options	
Reset The Redundant Board	<input type="button" value="Reset"/>

2. Under the 'Switch Over' group, click **Switch Over**; a confirmation box appears requesting you to confirm.
 3. Click **OK**.
- **To reset the Redundant device:**
1. Under the 'Redundant Options' group, click **Reset**; a confirmation box appears requesting you to confirm.
 2. Click **OK**.

Reader's Notes

31 Software Upgrade

The **Software Update** menu allows you to do the following:

- Load Auxiliary Files (see 'Loading Auxiliary Files' on page 363)
- Load Software License Key (see 'Software License Key' on page 370)
- Upgrade device using Software Upgrade Wizard (see 'Software Upgrade Wizard' on page 373)
- Load / save Configuration File (see 'Backing Up and Loading Configuration File' on page 377)

31.1 Loading Auxiliary Files

Various Auxiliary files can be installed on the device. These Auxiliary files provide the device with additional configuration settings. The table below lists the different types of Auxiliary files:

Table 31-1: Auxiliary Files

File	Description
INI	Configures the device. The Web interface enables practically full device provisioning. However, some features may only be configured by ini file or you may wish to configure your device using the ini file. For more information on using the ini file to configure the device, see 'INI File-Based Management' on page 79.
Call Progress Tones	Region-specific, telephone exchange-dependent file that contains the Call Progress Tones (CPT) levels and frequencies for the device. The default CPT file is U.S.A. For more information, see 'Call Progress Tones File' on page 364.
Dial Plan	Provides dialing plans, for example for obtaining the destination IP address for outbound IP routing. For more information, see 'Dial Plan File' on page 367.
User Info	The User Information file maps PBX extensions to IP numbers. This file can be used to represent PBX extensions as IP phones in the global 'IP world'. For more information, see 'User Information File' on page 368.

The Auxiliary files can be loaded to the device using one of the following methods:

- Web interface.
- TFTP: This is done by specifying the name of the Auxiliary file in an *ini* file (see Auxiliary and Configuration Files Parameters) and then loading the *ini* file to the device. The Auxiliary files listed in the *ini* file are then automatically loaded through TFTP during device startup. If the *ini* file does not contain a specific auxiliary file type, the device uses the last auxiliary file of that type that was stored on its non-volatile memory.


Notes:

- You can schedule automatic loading of updated auxiliary files using HTTP/HTTPS. For more information on automatic updates, see 'Automatic Update' on page 379.
- When loading an *ini* file using this Web page, parameters that are excluded from the loaded *ini* file retain their current settings (*incremental*).
- Saving an auxiliary file to flash memory may disrupt traffic on the device. To avoid this, disable all traffic on the device by performing a graceful lock as described in 'Locking and Unlocking the Device' on page 359.
- For deleting auxiliary files, see 'Viewing Device Information' on page 389.

The procedure below describes how to load Auxiliary files using the Web interface.

➤ **To load auxiliary files to the device using the Web interface:**

1. Open the Load Auxiliary Files page (**Maintenance** tab > **Software Update** menu > **Load Auxiliary Files**).

Figure 31-1: Load Auxiliary Files Page




Note: The appearance of certain file load fields depends on the installed Software License Key.

2. Click the **Browse** button corresponding to the file type that you want to load, navigate to the folder in which the file is located, and then click **Open**; the name and path of the file appear in the field next to the **Browse** button.
3. Click the **Load File** button corresponding to the file you want to load.
4. Repeat steps 2 through 3 for each file you want to load.
5. Save the loaded auxiliary files to flash memory, see 'Saving Configuration' on page 360 and reset the device (if you have loaded a Call Progress Tones file), see 'Resetting the Device' on page 357.

31.1.1 Call Progress Tones File

The Call Progress Tones (CPT) auxiliary file includes the definitions of the Call Progress Tones (levels and frequencies) that are detected / generated by the device.

You can use one of the supplied auxiliary files (.dat file format) or create your own file. To create your own file, it's recommended to modify the supplied *usa_tone.ini* file (in any standard text editor) to suit your specific requirements and then convert the modified *ini* file into binary format, using AudioCodes DConvert utility. For a description on converting a CPT *ini* file into a binary *dat* file, refer to the *DConvert Utility User's Guide*.



Note: Only the *dat* file format can be loaded to the device.

You can create up to 32 different Call Progress Tones, each with frequency and format attributes. The frequency attribute can be single or dual-frequency (in the range of 300 to 1980 Hz) or an Amplitude Modulated (AM). Up to 64 different frequencies are supported. Only eight AM tones, in the range of 1 to 128 kHz, can be configured (the detection range is limited to 1 to 50 kHz). Note that when a tone is composed of a single frequency, the second frequency field must be set to zero.

The format attribute can be one of the following:

- **Continuous:** A steady non-interrupted sound (e.g., a dial tone). Only the 'First Signal On time' should be specified. All other on and off periods must be set to zero. In this case, the parameter specifies the detection period. For example, if it equals 300, the tone is detected after 3 seconds (300 x 10 msec). The minimum detection time is 100 msec.
- **Cadence:** A repeating sequence of on and off sounds. Up to four different sets of on/off periods can be specified.
- **Burst:** A single sound followed by silence. Only the 'First Signal On time' and 'First Signal Off time' should be specified. All other on and off periods must be set to zero. The burst tone is detected after the off time is completed.

You can specify several tones of the same type. These additional tones are used only for tone detection. Generation of a specific tone conforms to the first definition of the specific tone. For example, you can define an additional dial tone by appending the second dial tone's definition lines to the first tone definition in the *ini* file. The device reports dial tone detection if either of the two tones is detected.

The Call Progress Tones section of the *ini* file comprises the following segments:

- **[NUMBER OF CALL PROGRESS TONES]:** Contains the following key:
'Number of Call Progress Tones' defining the number of Call Progress Tones that are defined in the file.
- **[CALL PROGRESS TONE #X]:** containing the Xth tone definition, starting from 0 and not exceeding the number of Call Progress Tones less 1 defined in the first section (e.g., if 10 tones, then it is 0 to 9), using the following keys:
 - **Tone Type:** Call Progress Tone types:
 - ◆ **[1]** Dial Tone
 - ◆ **[2]** Ringback Tone
 - ◆ **[3]** Busy Tone
 - ◆ **[4]** Congestion Tone
 - ◆ **[6]** Warning Tone
 - ◆ **[7]** Reorder Tone
 - ◆ **[17]** Call Waiting Ringback Tone - heard by the calling party
 - ◆ **[18]** Comfort Tone
 - ◆ **[23]** Hold Tone
 - ◆ **[46]** Beep Tone
 - **Tone Modulation Type:** Amplitude Modulated (1) or regular (0)
 - **Tone Form:** The tone's format can be one of the following:
 - ◆ Continuous (1)
 - ◆ Cadence (2)
 - ◆ Burst (3)

- **Low Freq [Hz]:** Frequency (in Hz) of the lower tone component in case of dual frequency tone, or the frequency of the tone in case of single tone. This is not relevant to AM tones.
- **High Freq [Hz]:** Frequency (in Hz) of the higher tone component in case of dual frequency tone, or zero (0) in case of single tone (not relevant to AM tones).
- **Low Freq Level [-dBm]:** Generation level 0 dBm to -31 dBm in dBm (not relevant to AM tones).
- **High Freq Level:** Generation level of 0 to -31 dBm. The value should be set to 32 in the case of a single tone (not relevant to AM tones).
- **First Signal On Time [10 msec]:** 'Signal On' period (in 10 msec units) for the first cadence on-off cycle. For continuous tones, this parameter defines the detection period. For burst tones, it defines the tone's duration.
- **First Signal Off Time [10 msec]:** 'Signal Off' period (in 10 msec units) for the first cadence on-off cycle (for cadence tones). For burst tones, this parameter defines the off time required after the burst tone ends and the tone detection is reported. For continuous tones, this parameter is ignored.
- **Second Signal On Time [10 msec]:** 'Signal On' period (in 10 msec units) for the second cadence on-off cycle. Can be omitted if there isn't a second cadence.
- **Second Signal Off Time [10 msec]:** 'Signal Off' period (in 10 msec units) for the second cadence on-off cycle. Can be omitted if there isn't a second cadence.
- **Third Signal On Time [10 msec]:** 'Signal On' period (in 10 msec units) for the third cadence on-off cycle. Can be omitted if there isn't a third cadence.
- **Third Signal Off Time [10 msec]:** 'Signal Off' period (in 10 msec units) for the third cadence on-off cycle. Can be omitted if there isn't a third cadence.
- **Fourth Signal On Time [10 msec]:** 'Signal On' period (in 10 msec units) for the fourth cadence on-off cycle. Can be omitted if there isn't a fourth cadence.
- **Fourth Signal Off Time [10 msec]:** 'Signal Off' period (in 10 msec units) for the fourth cadence on-off cycle. Can be omitted if there isn't a fourth cadence.
- **Carrier Freq [Hz]:** Frequency of the carrier signal for AM tones.
- **Modulation Freq [Hz]:** Frequency of the modulated signal for AM tones (valid range from 1 to 128 Hz).
- **Signal Level [-dBm]:** Level of the tone for AM tones.
- **AM Factor [steps of 0.02]:** Amplitude modulation factor (valid range from 1 to 50). Recommended values from 10 to 25.


Notes:

- When the same frequency is used for a continuous tone and a cadence tone, the 'Signal On Time' parameter of the continuous tone must have a value that is greater than the 'Signal On Time' parameter of the cadence tone. Otherwise, the continuous tone is detected instead of the cadence tone.
- The tones frequency must differ by at least 40 Hz between defined tones.

For example, to configure the dial tone to 440 Hz only, enter the following text:

```
[NUMBER OF CALL PROGRESS TONES]
Number of Call Progress Tones=1
#Dial Tone
[CALL PROGRESS TONE #0]
Tone Type=1
Tone Form =1 (continuous)
Low Freq [Hz]=440
```

```

High Freq [Hz]=0
Low Freq Level [-dBm]=10 (-10 dBm)
High Freq Level [-dBm]=32 (use 32 only if a single tone is
required)
First Signal On Time [10msec]=300; the dial tone is detected after
3 sec
First Signal Off Time [10msec]=0
Second Signal On Time [10msec]=0
Second Signal Off Time [10msec]=0

```

31.1.2 Dial Plan File

The Dial Plan file can be used for various digit mapping features, as described in this section.

31.1.2.1 Creating a Dial Plan File

Creating a Dial Plan file is similar between all Dial Plan features. The main difference is the syntax used in the Dial Plan file and the method for selecting the Dial Plan index to use for the specific feature.

The Dial Plan file is a text-based file that can contain up to eight Dial Plans (Dial Plan indices) and up to 8,000 rules (lines). The general syntax rules for the Dial Plan file are as follows (syntax specific to the feature is described in the respective section):

- Each Dial Plan index must begin with a Dial Plan name enclosed in square brackets "[...]" on a new line.
- Each line under the Dial Plan index defines a rule.
- Empty lines are ignored.
- Lines beginning with a semicolon ";" are ignored. The semicolon can be used for comments.

➤ To create a Dial Plan file:

1. Create a new file using a text-based editor (such as Notepad) and configure your Dial Plans, as required.
2. Save the file with the *ini* file extension name (e.g., mydialplanfile.ini).
3. Convert the *ini* file to a *dat* binary file, using AudioCodes DConvert utility. For more information, refer to *DConvert Utility User's Guide*.
4. Install the converted file on the device, as described in 'Loading Auxiliary Files' on page 363.
5. Select the Dial Plan index that you want to use. This depends on the feature and is described in the respective section.

31.1.2.2 Obtaining IP Destination from Dial Plan File

You can use a Dial Plan index listed in a loaded Dial Plan file for determining the IP destination of SBC calls. This enables the mapping of called numbers to IP addresses (in dotted-decimal notation) or FQDNs (up to 15 characters).

➤ To configure routing to an IP destination based on Dial Plan:

1. Create the Dial Plan file. The syntax of the Dial Plan index for this feature is as follows:

```
<destination / called prefix number>,0,<IP destination>
```

Note that the second parameter "0" is not used and ignored.

An example of a configured Dial Plan (# 6) in the Dial Plan file is shown below:

```
[ PLAN6 ]
200,0,10.33.8.52      ; called prefix 200 is routed to
10.33.8.52
201,0,10.33.8.52
300,0,itsp.com       ; called prefix 300 is routed to itsp.com
```

2. Convert the file to a loadable file and then load it to the device.
3. Assign the Dial Plan index to the required routing rule:
 - a. Open the SBC IP-to-IP Routing table.
 - b. Set the 'Destination Type' field to Dial Plan.
 - c. In the 'Destination Address' field, enter the required Dial Plan index, where "0" denotes [PLAN1] in the Dial Plan file, "1" denotes [PLAN2], and so on.

31.1.3 User Information File

This section describes the various uses of the User Info file.

You can load the User Info file using any of the following methods:

- Web interface (see 'Loading Auxiliary Files' on page 363)
- *ini* file - using the `UserInfoFileName` parameter, e.g., `UserInfoFileName = 'UserInformationFile.txt'` (see 'Auxiliary and Configuration File Name Parameters' on page 529)
- Automatic update mechanism - using the `UserInfoFileURL` parameter, e.g., `UserInfoFileUrl = 'http://192.168.0.250/Audiocodes/ UserInformationFile.txt'` (see 'Automatic Update Mechanism' on page 379)

31.1.3.1 User Information File for SBC User Database

You can create a User Info table of SBC users from a loaded User Info file.

The device can use the SBC User Info for the following:

- Registering each user to an external registrar server.
- Authenticating (for any SIP request and as a client) each user if challenged by an external server.
- Authenticating as a server incoming user requests (for SBC security).

If the device registers on behalf of users and the users do not perform registration, any SIP request destined to the user is routed to the Proxy Set associated with the user's IP Group.

The User Info file is a text-based file that you can create using any text-based program such as Notepad. To add SBC users to this file, use the following syntax:

```
[ SBC ]
FORMAT LocalUser,UserName,Password,IPGroupID
john,john_user,john_pass,2
sue,sue_user,sue_pass,1
```

where:

- *LocalUser* is the user and is used as the Request-URI user part for the AOR in the database
- *UserName* is the user's authentication username
- *Password* is the user's authentication password
- *IPGroupID* is the IP Group ID to which the user belongs and is used as the Request-URI source host part for the AOR in the database

**Notes:**

- To enable the User Info table, see 'Enabling the User Info Table' on page 369.
- To modify the Use Info table, you need to load a new User Info file containing your modifications.

31.1.3.2 Enabling the User Info Table

The procedure below describes how to load a User Info file to the device and enable the use of the User Info table:

➤ **To enable the User Info table:**

1. Open the Advanced Parameters page (**Configuration** tab > **VoIP** menu > **SIP Definitions** > **Advanced Parameters**).
2. Set the 'Enable User-Information Usage' parameter to **Enable**.

31.2 Software License Key

The device is shipped with a pre-installed Software License Key, which determines the device's supported features, capabilities, and available resources. You can upgrade or change your device's supported features by purchasing and installing a new Software License Key to match your requirements.



Notes:

- For the High Availability (HA) system, the Software License Key includes the HA feature and is installed on both devices - active and redundant. If the redundant device's Software License Key is missing or invalid, the system is moved to mismatch configuration mode (alerted by SNMP).
- The availability of certain Web pages depends on the installed Software License Key.

31.2.1 Obtaining the Software License Key File

Before you can install a new Software License Key, you need to obtain a Software License Key file for your device with the required features from your AudioCodes representative. The Software License Key is an encrypted key in string format that is associated with the device's serial number ("S/N") and supplied in a text-based file.

If you need a Software License Key for more than one device, the Software License Key file can include multiple Software License Keys (see figure below). In such cases, each Software License Key in the file is associated with a unique serial number identifying the specific device. When loading such a Software License Key file, the device installs only the Software License Key that is associated with its serial number.

Figure 31-2: Software License Key File with Multiple S/N Lines



```
sample.ini - Notepad
[LicenseKeys]
;Board Type 29
S/N241182 =
okRTr5topwYMBIZd4NN2a3Qhm4NjIidaagUyehso94APbBF85f4by0cmQZf2B8bMcze7JQ9kMSa5h641R1aOkeEb9AddF894Zx
S/N242519 = tmxTr5to0mIMbIZdoPd2a3Qh9zJlIdafilyehsogOQPbBF8pj4by0c9pdl2B8eOoze7JQgywSa5h6o391aOkeTlIAddF8c6Fx
S/N226403 = tmxTr5to0IsMblZdoOB2a3Qh9yJlIdafilyehsogN4PbBF8piZ4by0c9pdl2B8eOoze7JQgwgSa5h6o2x1aOkeTlIAddF8c6Fx
S/N226417 = r6xTr5to25sMblZdfIB2a3Qh5OJlIda92lyehsoix4PbBF8eOZ4by0c52dl2B88yoze7JQlNgSa5h6fpx1aOkeXZlAddF8amFx
;Board Type 24
S/N241182 =
okRTr5topwYMBIZd4NN2a3wk4NjIidaagUyehso94APbBF85f4by0cmQZf2B8bMcze7JQ9kMSa5h641R1aOkeEb9AddF8938s
S/N242519 = tmxTr5to0mIMbIZdoPd2a3wk9zJlIdafilyehsogOQPbBF8pj4by0c9pdl2B8eOoze7JQgywSa5h6o391aOkeTlIAddF8c1ss
S/N226403 = tmxTr5to0IsMblZdoOB2a3wk9yJlIdafilyehsogN4PbBF8piZ4by0c9pdl2B8eOoze7JQgwgSa5h6o2x1aOkeTlIAddF8c1ss
S/N226417 = r6xTr5to25sMblZdfIB2a3wk5OJlIda92lyehsoix4PbBF8eOZ4by0c52dl2B88yoze7JQlNgSa5h6fpx1aOkeXZlAddF8ahss
```

➤ To obtain a Software License Key:

- Make a note of the MAC address and/or serial number of the device:
 - Open the Device Information page (**Status & Diagnostics** tab > **System Status** menu > **Device Information**).
 - The MAC address is displayed in the "MAC Address" field and the serial number in the "Serial Number" field.
- If you need a Software License Key for more than one device, repeat Step 1 for each device.
- Request the required Software License Key from your AudioCodes representative and provide them with the MAC address and/or serial number of the device(s).

4. When you receive the new Software License Key file, check the file as follows:
 - a. Open the file with any text-based program such as Notepad.
 - b. Verify that the first line displays "[LicenseKeys]".
 - c. Verify that the file contains one or more lines in the following format:
 "S/N<serial number> = <Software License Key string>".
 For example: "S/N370604 = jCx6r5tovCIKaBBbhPtT53Yj..."
 - d. Verify that the "S/N" value reflects the serial number of your device. If you have multiple Software License Keys, ensure that each "S/N" value corresponds to a device.



Warning: Do not modify the contents of the Software License Key file.

5. Install the Software License Key on the device as described in 'Installing the Software License Key' on page 371.

31.2.2 Installing the Software License Key

Once you have received your Software License Key file from your AudioCodes representative, you can install it on the device using one of the following management tools:

- Web interface - see 'Installing Software License Key using Web Interface' on page 371
- AudioCodes EMS - refer to the EMS User's Manual or EMS Product Description



Note: When you install a new Software License Key, it is loaded to the device's non-volatile flash memory and overwrites the previously installed Software License Key.

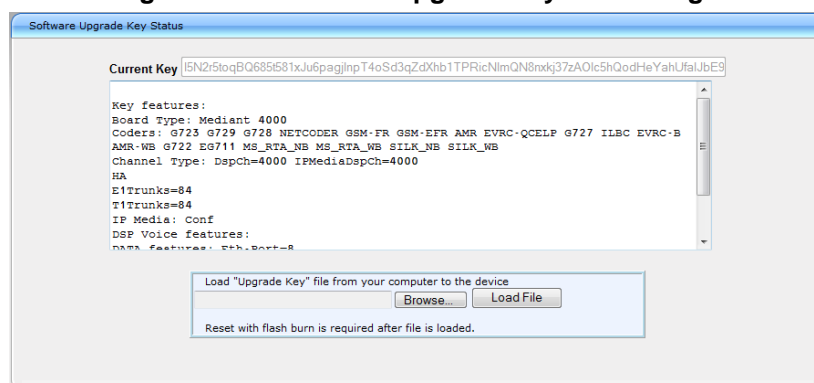
31.2.2.1 Installing Software License Key using Web Interface

The procedure below describes how to install the Software License Key using the Web interface.

➤ To install the Software License Key using the Web interface:

1. Open the Software Upgrade Key Status page (**Maintenance** tab > **Software Update** menu > **Software Upgrade Key**).

Figure 31-3: Software Upgrade Key Status Page



2. As a precaution, backup the Software License Key currently installed on the device. If the new Software License Key does not comply with your requirements, you can re-load this backup to restore the device's original capabilities.
 - a. In the 'Current Key' field, select the entire text string and copy it to any standard text file (e.g., Notepad).
 - b. Save the text file with any file name and file extension (e.g., key.txt) to a folder on your computer.
3. Depending on whether you are loading a Software License Key file with a single Software License Key (i.e., one "S/N") or with multiple Software License Keys (i.e., more than one "S/N"), do one of the following:
 - **Loading a File with a Single Software License Key:**
 - a. Open the Software License Key file using a text-based program such as Notepad.
 - b. Copy-and-paste the string from the file to the 'Add a Software Upgrade Key' field.
 - c. Click the **Add Key** button.
 - **Loading a File with Multiple Software License Keys:**
 - a. In the 'Load Upgrade Key file ...' field, click the **Browse** button and navigate to the folder in which the Software License Key file is located on your computer.
 - b. Click **Load File**; the new key is installed on the device.

If the Software License Key is valid, it is burned to the device's flash memory and displayed in the 'Current Key' field.
4. Verify that the Software License Key was successfully installed, by doing one of the following:
 - In the Software Upgrade Key Status page, check that the listed features and capabilities activated by the installed Software License Key match those that were ordered.
 - Access the Syslog server and ensure that the following message appears in the Syslog server:
"S/N___ Key Was Updated. The Board Needs to be Reloaded with ini file\n"
5. Reset the device; the new capabilities and resources enabled by the Software License Key are active.



Note: If the Syslog server indicates that the Software License Key was unsuccessfully loaded (i.e., the "SN_" line is blank), do the following preliminary troubleshooting procedures:

1. Open the Software License Key file and check that the "S/N" line appears. If it does not appear, contact AudioCodes.
2. Verify that you have loaded the correct file. Open the file and ensure that the first line displays "[LicenseKeys]".
3. Verify that the content of the file has not been altered.

31.3 Software Upgrade Wizard

The Software Upgrade Wizard allows you to upgrade the device's firmware. The firmware file has the .cmp file extension name. The wizard also enables you to load an *ini* file and/or auxiliary files (typically loaded using the Load Auxiliary File page described in 'Loading Auxiliary Files' on page 363). However, it is mandatory when using the wizard to first load a .cmp file to the device. You can then choose to also load an *ini* file and/or auxiliary files, but this cannot be done without first loading a .cmp file. For the *ini* and each auxiliary file type, you can choose to load a new file or not load a file but use the existing file (i.e., maintain existing configuration) running on the device.

The wizard also allows you to perform Hitless Upgrade (non-traffic affecting upgrade), whereby the upgrade process begins only after all current calls have been terminated. For a description of this process, see Software Upgrade on page 354.



Warning: The Software Upgrade Wizard requires the device to be reset at the end of the process, which may disrupt traffic. To avoid this, disable all traffic on the device before initiating the wizard by performing a graceful lock (see 'Basic Maintenance' on page 357).



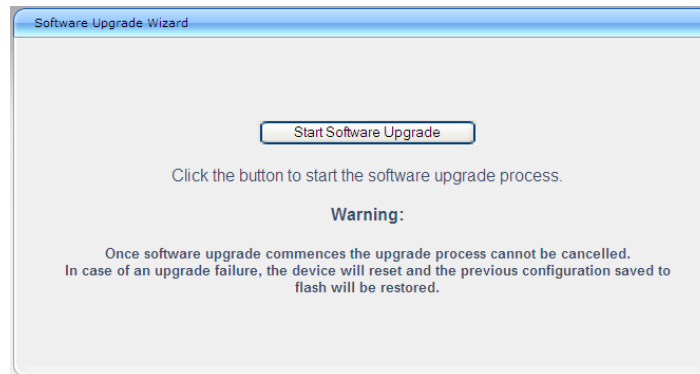
Notes:

- You can get the latest software files from AudioCodes Web site at <http://www.audiocodes.com/downloads>.
- Before upgrading the device, it is recommended that you save a copy of the device's configuration settings (i.e., *ini* file) to your computer. If an upgrade failure occurs, you can then restore your configuration settings by uploading the backup file to the device. For saving and restoring configuration, see 'Backing Up and Loading Configuration File' on page 377.
- If you wish to also load an *ini* or auxiliary file, it is mandatory to first load a .cmp file.
- When you activate the wizard, the rest of the Web interface is unavailable. After the files are successfully loaded, access to the full Web interface is restored.
- If you upgraded your .cmp and the "SW version mismatch" message appears in the Syslog or Web interface, then your Software License Key does not support the new .cmp file version. If this occurs, contact AudioCodes support for assistance.
- If you use the wizard to load an *ini* file, parameters excluded from the *ini* file are assigned default values (according to the .cmp file running on the device) thereby, overriding values previously defined for these parameters.
- You can schedule automatic loading of these files using HTTP/HTTPS (see 'Automatic Update' on page 379).

➤ **To load files using the Software Upgrade Wizard:**


1. Stop all traffic on the device using the Graceful Lock feature (refer to the warning bulletin above).
2. Open the Software Upgrade wizard, by performing one of the following:
 - Select the **Maintenance** tab, click the **Software Update** menu, and then click **Software Upgrade Wizard**.
 - On the toolbar, click **Device Actions**, and then choose **Software Upgrade Wizard**.

Figure 31-4: Start Software Upgrade Wizard Screen



3. Click the **Start Software Upgrade** button; the wizard starts, requesting you to browses to a .cmp file for uploading.




Note: At this stage, you can quit the Software Update Wizard, by clicking **Cancel** , without requiring a device reset. However, once you start uploading a cmp file, the process must be completed with a device reset. If you choose to quit the process in any of the subsequent pages, the device resets.

4. Click the **Browse** button, navigate to the .cmp file, and then click **Load File**; a progress bar appears displaying the status of the loading process. When the .cmp file is successfully loaded to the device, a message appears notifying you of this.
5. If your device is in HA mode, select one of the following options:
 - **Hitless Upgrade: (Default)** To perform a Hitless Upgrade (non-traffic effecting upgrade), whereby the upgrade process begins only after all current calls have been terminated.
 - **System Reset Upgrade:** Both devices immediately reset with the newly loaded .cmp file.








Note: If you select Hitless Upgrade, you can upload only a .cmp file (auxiliary files and ini files cannot be uploaded as well).

6. If you want to load **only** a .cmp file, then click the **Reset**  button to reset the device with the newly loaded .cmp file, utilizing the existing configuration (*ini*) and auxiliary files. To load additional files, skip to the next Step.



Note: Device reset may take a few minutes depending on cmp file version (this may even take up to 10 minutes).

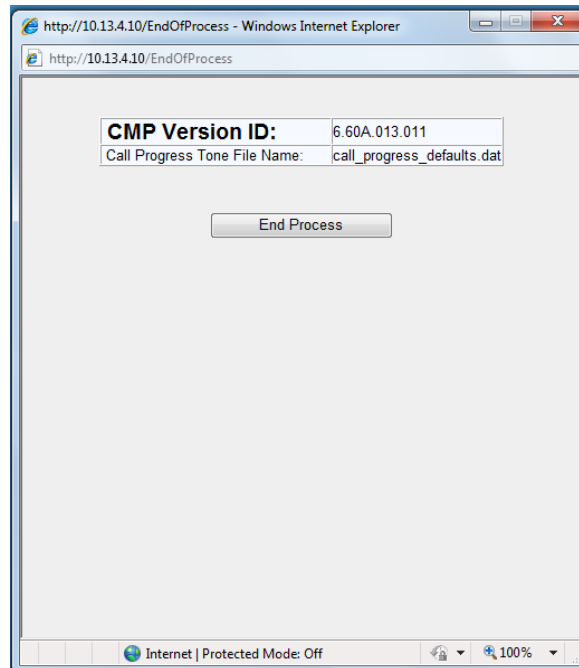
7. Click the **Next**  button; the wizard page for loading an *ini* file appears. You can now perform one of the following:
- Load a new *ini* file: Click **Browse**, navigate to the *ini* file, and then click **Send File**; the *ini* file is loaded to the device and you're notified as to a successful loading.
 - Retain the existing configuration (*ini* file): Do not select an *ini* file, and ensure that the 'Use existing configuration' check box is selected (default).
 - Return the device's configuration settings to factory defaults: Do not select an *ini* file, and clear the 'Use existing configuration' check box.
8. Click the **Next**  button to progress to the relevant wizard pages for loading the desired auxiliary files. To return to the previous wizard page, click the **Back**  button. As you navigate between wizard pages, the relevant file type corresponding to the Wizard page is highlighted in the left pane.
9. When you have completed loading all the desired files, click the **Next**  button until the last wizard page appears ("FINISH" is highlighted in the left pane).
10. Click the **Reset**  button to complete the upgrade process; the device 'burns' the newly loaded files to flash memory and then resets the device.



Note: Device reset may take a few minutes (depending on .cmp file version, this may even take up to 30 minutes).

After the device resets, the End of Process wizard page appears displaying the new .cmp and auxiliary files loaded to the device.

Figure 31-5: Software Upgrade Process Completed Successfully



11. Click **End Process** to close the wizard; the Web Login dialog box appears.
12. Enter your login user name and password, and then click **OK**; a message box appears informing you of the new .cmp file.
13. Click **OK**; the Web interface becomes active, reflecting the upgraded device.

31.4 Backing Up and Loading Configuration File

You can save a copy/backup of the device's current configuration settings as an *ini* file to a folder on your computer, using the Configuration File page. The saved *ini* file includes only parameters that were modified and parameters with other than default values. The Configuration File page also allows you to load an *ini* file to the device. If the device has "lost" its configuration, you can restore the device's configuration by loading the previously saved *ini* file or by simply loading a newly created *ini* file.



Note: When loading an *ini* file using this Web page, parameters not included in the *ini* file are reset to default settings.

➤ **To save the *ini* file:**

1. Open the Configuration File page by doing one of the following:
 - From the Navigation tree, click the **Maintenance** tab, click the **Software Update** menu, and then click **Configuration File**.
 - On the toolbar, click **Device Actions**, and then from the drop-down menu, choose **Load Configuration File** or **Save Configuration File**.

2. To save the *ini* file to a folder on your computer, do the following:
 - a. Click the **Save INI File** button; the File Download dialog box appears.
 - b. Click the **Save** button, navigate to the folder where you want to save the *ini* file, and then click **Save**.
3. To load the *ini* file to the device, do the following:
 - a. Click the **Browse** button, navigate to the folder where the *ini* file is located, select the file, and then click **Open**; the name and path of the file appear in the field beside the **Browse** button.
 - b. Click the **Load INI File** button, and then at the prompt, click **OK**; the device uploads the *ini* file and then resets (from the *cmp* version stored on the flash memory). Once complete, the Web Login screen appears, requesting you to enter your user name and password.

Reader's Notes

32 Automatic Update

Automatic provisioning is useful for large-scale deployment of devices. In some cases, the devices are shipped to the end customer directly from the manufacturer. In other cases, they may pass through a staging warehouse. Configuration may occur at the staging warehouse or at the end-customer premises.

The devices may be pre-configured during the manufacturing process (commonly known as *private labeling*). Typically, a two-stage configuration process is implemented such that initial configuration includes only basic configuration, while the final configuration is done when the device is deployed in a live network.

Automatic provisioning can be used to update the following files:

- Software file (*cmp*)
- Auxiliary files (e.g., Call Progress Tones)
- Configuration file (*ini*)

The Automatic Update mechanism is applied per file, using specific parameters that define the URLs to the servers where the files are located, and the file names (see Automatic Update Parameters on page 530). These files can be stored on any standard Web, FTP, or NFS server and can be loaded periodically to the device using HTTP, HTTPS, FTP, or NFS. This mechanism can be used even for devices that are installed behind NAT and firewalls.

The Automatic Update mechanism can be triggered by the following:

- Upon device startup.
- At a user-defined time of day (e.g., 18:00), configured by the *ini* file parameter `AutoUpdatePredefinedTime`.
- Periodically (e.g., every 60 minutes), configured by the *ini* file parameter `AutoUpdateFrequency`.
- Upon startup but before the device is operational, if the Secure Startup feature is enabled (see 'Loading Files Securely (Disabling TFTP)' on page 383).
- Upon receipt of a special SIP Notify message (see 'Remotely Triggering Auto Update using SIP NOTIFY' on page 384)

When implementing Automatic Updates using HTTP/S, the device determines whether the file on the provisioning server is an updated one as follows:

- **Configuration file:** The device checks the timestamp according to the HTTP server response. Cyclical Redundancy Check (CRC) is only checked if the `AUPDCheckIfIniChanged` parameter is enabled. The device downloads the configuration file only if it was modified since the last successful configuration update.
- **Software file (*cmp*):** The device first downloads the file and then checks if its version number is different from the software version file currently stored on the device's flash memory.
- **Auxiliary files (e.g., CPT):** These files are updated only once. To update the auxiliary file again, you must modify the settings of the related parameter that configures its URL.

32.1 Configuring Automatic Update

The procedure below describes how to configure the Automatic Update feature. It describes a scenario where the devices download a "master" configuration file with common settings from an HTTP server. This "master" file applies common configuration and instructs each device to download a specific configuration file based on the device's MAC address from an HTTP server.



Warning: Do not use the Web interface to configure the device when the Automatic Update feature is implemented. If you do and save (burn) the new settings to the device's flash memory, the IniFileURL parameter is automatically set to 0 and Automatic Updates are consequently disabled. To enable Automatic Updates again, you would need to re-load the ini file (using the Web interface or BootP) with the correct IniFileURL settings. As a safeguard to an unintended burn-to-flash when resetting the device, if the device is configured for Automatic Updates, the 'Burn To FLASH' field under the Reset Configuration group in the Web interface's Maintenance Actions page is automatically set to **No** by default.



Note: For a description of all the Automatic Update *ini* file parameters, see Automatic Update Parameters on page 530.

➤ **To configure the Automatic Update feature (ini file example):**

1. Setup a Web server (e.g., <http://www.corp.com>) and place all the required configuration files on this server.

2. For each device, preconfigure the following parameter (DHCP / DNS are assumed):

```
IniFileURL = 'http://www.corp.com/master_configuration.ini'
```

3. Create a file named *master_configuration.ini* with the following text:

```
# Common configuration for all devices
# -----
CptFileURL = 'http://www.corp.com/call_progress.dat'
# Check for updates every 60 minutes
AutoUpdateFrequency = 60
# Additional configuration per device
# -----
# Each device loads a file named based on its MAC address
# (e.g., config_00908F033512.ini)
IniFileURL = 'http://www.corp.com/config_<MAC>.ini'
# Reset the device after configuration is updated.
# The device resets after all files are processed.
ResetNow = 1
```

You can modify the *master_configuration.ini* file (or any of the *config_<MAC>.ini* files) at any time. The device queries for the latest version every 60 minutes and applies the new settings immediately.

4. For additional security, use HTTPS or FTPS. The device supports HTTPS (RFC 2818) and FTPS using the AUTH TLS method <draft-murray-auth-ftp-ssl-16>.
5. To download configuration files from an NFS server, the NFS file system parameters should be defined in the *ini* file. The following is an example of an *ini* file for downloading files from NFS servers using NFS version 2:

```
# Define NFS servers for Automatic Update
[ NFSServers ]
FORMAT NFSServers_Index = NFSServers_HostOrIP,
NFSServers_RootPath, NFSServers_NfsVersion;
NFSServers 1 = 10.31.2.10, /usr/share, 2 ;
NFSServers 2 = 192.168.100.7, /d/shared, 2 ;
[ \NFSServers ]
CptFileUrl =
'file://10.31.2.10/usr/share/public/usa_tones.dat'
```

```
VpFileUrl =
'file://192.168.100.7/d/shared/gateways/voiceprompt.dat'
```

The following *ini* file example can be used to activate the Automatic Update mechanism.

```
# DNS is required for specifying domain names in URLs
[ InterfaceTable ]
FORMAT InterfaceTable_Index = InterfaceTable_ApplicationTypes,
InterfaceTable_InterfaceMode, InterfaceTable_IPAddress,
InterfaceTable_PrefixLength, InterfaceTable_Gateway,
InterfaceTable_VlanID, InterfaceTable_InterfaceName,
InterfaceTable_PrimaryDNSServerIPAddress,
InterfaceTable_SecondaryDNSServerIPAddress,
InterfaceTable_UnderlyingInterface;
InterfaceTable 0 = 6, 10, 10.13.4.12, 16, 10.13.0.1, 1, Mng,
10.1.1.11, 0.0.0.0, ;
[ \InterfaceTable ]

# Load an extra configuration ini file using HTTP
IniFileURL = 'http://webserver.corp.com/Gateway/inifile.ini'
# Load Call Progress Tones file using HTTPS
CptFileUrl = 'https://10.31.2.17/usa_tones.dat'
# Load Voice Prompts file using FTPS with user 'root' and password
'wheel'
VpFileUrl = 'ftps://root:wheel@ftpserver.corp.com/vp.dat'
# Update every day at 03:00 AM
AutoUpdatePredefinedTime = '03:00'
# Note: The cmp file isn't updated since it's disabled by default
(AutoUpdateCmpFile).
```

Notes:

- The Automatic Update mechanism assumes that the external Web server conforms to the HTTP standard. If the Web server ignores the If-Modified-Since header, or doesn't provide the current date and time during the HTTP 200 OK response, the device may reset itself repeatedly. To overcome this problem, modify the update frequency, using the AutoUpdateFrequency parameter.
- To load a different configuration file (ini file) per device, add the string "<MAC>" to the URL (e.g., IniFileURL = 'http://www.corp.com/config_<MAC>.ini'). This mnemonic is replaced with the device's hardware MAC address, resulting in an ini file name request that contains the device's MAC address (e.g., config_00908F033512.ini).
- To prevent the device from accidentally upgrading its software, by default the Automatic Update feature does not apply a downloaded *cmp* file even if its URL was configured (using the CmpFileURL parameter). To enable this, set the AutoUpdateCmpFile parameter to 1.
- To enable the device to automatically reset after an ini file has been loaded, set the ResetNow parameter to 1. This is important if the downloaded configuration file includes parameters that require a device reset for its settings to be applied.
- By default, parameters that are not included in the downloaded configuration file are set to default. To retain the current settings of these parameters, set the SetDefaultOnINIFileProcess parameter to 0.



32.2 Automatic Configuration Methods

This section describes available methods that can be used for automatic device configuration.

32.2.1 DHCP-based Configuration Server

The DHCP server can be configured to automatically provide each device with a temporary IP address so that individual MAC addresses are not required. Configuration occurs at a staging warehouse for this method.

Below is an example configuration file for Linux DHCP server (dhcpd.conf). The devices are allocated temporary IP addresses in the range 10.31.4.53 to 10.31.4.75. TFTP is assumed to be on the same computer as the DHCP server (alternatively, the "next-server" directive may be used).

```
ddns-update-style ad-hoc;
default-lease-time 60;
max-lease-time 60;
class "gateways" {
    match if(substring(hardware, 1, 3) = 00:90:8f);
}
subnet 10.31.0.0 netmask 255.255.0.0 {
    pool {
        allow members of "audiocodes";
        range 10.31.4.53 10.31.4.75;
        filename "SIP_F6.60A.217.003.cmp -fb;device.ini";
        option routers                10.31.0.1;
        option subnet-mask            255.255.0.0;
    }
}
```

32.2.2 HTTP-based Automatic Updates

An HTTP/S server can be placed in the customer's network where configuration and software updates are available for download. This does not require additional servers at the customer premises and is NAT-safe.

For example, assume the core network HTTPS server is <https://www.corp.com>. A master configuration *ini* file should be placed on the server, e.g., <https://www.corp.com/gateways/master.ini>. This file could point to additional *ini* files, auxiliary files (e.g., call progress tones), and software files (cmp), all on the same HTTP server or different HTTP servers in the network.

The main advantage of this method is that the HTTP configuration can be checked periodically when the device is deployed at the customer site. HTTP(S) is not sensitive to NAT devices, enabling configuration whenever needed without on-site intervention.

For additional security, the URL may contain a different port, and username and password.

The devices should only be pre-configured with the URL of the initial *ini* file, using one of the following methods:

- Methods described in 'DHCP-based Configuration Server' on page 382 or above, via TFTP at a staging warehouse. The configuration URL is configured using the IniFileURL parameter.
- Private labeling.
- Manually on-site, using the RS-232 port or Web interface.

When the device is deployed at the customer site, local DHCP server provides the devices with IP addressing and DNS server information. From the URL provided in the DHCP response, the device can then contact the HTTP server at the core network and

automatically download its configuration. The URL can be a simple file name or contain the device's MAC or IP address, e.g.:

- `http://corp.com/config-<MAC>.ini` - which becomes, for example, `http://corp.com/config-00908f030012.ini`
- `http://corp.com/<IP>/config.ini` - which becomes, for example, `http://corp.com/192.168.0.7/config.ini`

32.2.3 Configuration using FTP or NFS

Some networks block access to HTTP(S). The Automatic Update feature provides limited support for FTP/FTPS connectivity. Periodic polling for updates is not possible since these protocols don't support conditional fetching, i.e., updating files only if it is changed on the server.

The only difference between this method and those described in 'HTTP-based Automatic Updates' on page 382 is that the protocol in the URL is "ftp" (instead of "http").



Notes:

- Unlike FTP, NFS is not NAT-safe.
- NFS v2/v3 is also supported.

32.2.4 Configuration using AudioCodes EMS

AudioCodes EMS server functions as a core-network provisioning server. The device's SNMP Manager should be configured with the IP address of the EMS server, using one of the methods detailed in the previous sections. As soon as a registered device contacts the EMS server through SNMP, the EMS server handles all required configuration automatically, upgrading software as needed. This alternative method doesn't require additional servers at the customer premises, and is NAT-safe.

32.3 Loading Files Securely (Disabling TFTP)

The TFTP protocol is not considered secure and some network operators block it using a firewall. It is possible to disable TFTP completely, using the *ini* file parameter `EnableSecureStartup` (set to 1). This way, secure protocols such as HTTPS may be used to fetch the device configuration.

➤ **To download the ini file to the device using HTTPS instead of TFTP:**

1. Prepare the device's configuration file on an HTTPS server and obtain a URL to the file (e.g., `https://192.168.100.53/gateways.ini`).
2. Enable DHCP, if necessary.
3. Enable SSH and connect to it.
4. In the CLI, use the *ini* file parameters `IniFileURL` (for defining the URL of the configuration file) and `EnableSecureStartup` (for disabling TFTP), and then restart the device with the new configuration:

```
/conf/scp IniFileURL https://192.168.100.53/gateways.ini
/conf/scp EnableSecureStartup 1
/conf/sar bootp
```



Note: Once Secure Startup has been enabled, it can only be disabled by setting `EnableSecureStartup` to 0 using the CLI.

32.4 Remotely Triggering Auto Update using SIP NOTIFY

The device can be remotely triggered to start the Automatic Update process upon receipt of a SIP NOTIFY that includes an Event header set to 'check-sync;reboot=false', as shown in the example below:

```
NOTIFY sip:<user>@<dsthost> SIP/2.0
To: sip:<user>@<dsthost>
From: sip:sipsak@<srchost>
CSeq: 10 NOTIFY
Call-ID: 1234@<srchost>
Event: check-sync;reboot=false
```

For this feature to function, Automatic Update must be enabled on the device. In other words, it must have a loaded ini file with the Automatic Update settings.

➤ **To enable remote trigger of Auto Update upon receipt of SIP NOTIFY:**

1. Open the Advanced Parameters page (**Configuration** tab > **VoIP** menu > **SIP Definitions** > **Advanced Parameters**).
2. Under the **Misc Parameters** group, set the 'SIP Remote Reset' parameter to **Enable**.
3. Click **Submit**.



Note: This SIP Event header value is proprietary to AudioCodes.

33 Restoring Factory Defaults

You can restore the device's configuration to factory defaults using one of the following methods:

- CLI (see 'Restoring Defaults using CLI' on page 385)
- Loading an empty *ini* file (see 'Restoring Defaults using an ini File' on page 386)

33.1 Restoring Defaults using CLI

The device can be restored to factory defaults using CLI, as described in the procedure below.

➤ **To restore factory defaults using CLI:**

1. Access the CLI:
 - a. Connect the RS-232 serial port of the device to the communication port on your computer. For serial cabling, refer to the *Hardware Installation Manual*.
 - b. Establish serial communication with the device using a serial communication program (such as HyperTerminal™) with the following communication port settings:
 - ◆ **Baud Rate:** 115,200 bps
 - ◆ **Data Bits:** 8
 - ◆ **Parity:** None
 - ◆ **Stop Bits:** 1
 - ◆ **Flow Control:** None
2. At the CLI prompt, type the username (default is "Admin" - case sensitive), and then press Enter:
Username: Admin
3. At the prompt, type the password (default is "Admin" - case sensitive), and then press Enter:
Password: Admin
4. At the prompt, type the following, and then press Enter:
enable
5. At the prompt, type the password again, and then press Enter:
Password: Admin
6. At the prompt, type the following to reset the device to default settings, and then press Enter:
write factory

33.2 Restoring Defaults using an ini File

You can restore the device to factory default settings by loading an empty *ini* file to the device. This is done using the Web interface's Configuration File page (see 'Backing Up and Loading Configuration File' on page [377](#)). If the *ini* file does include content (e.g., parameters), ensure that they are on lines beginning with comment signs (i.e., semicolons ";") so that the device ignores them.



Note: The only settings that are not restored to default are the management (OAMP) IP address and the Web interface's login user name and password.

Part X

Status, Performance Monitoring and Reporting

34 System Status

This section describes how to view various system statuses.

34.1 Viewing Device Information

The Device Information page displays various hardware and software information of the device. This page also lists any Auxiliary files that have been installed on the device and allows you to remove them.

➤ **To access the Device Information page:**

- Open the Device Information page (**Status & Diagnostics** tab > **System Status** menu > **Device Information**).

Figure 34-1: Device Information Page

▼ General Settings	
MAC Address:	00908f3ab169
Serial Number:	3846504
Board Type:	Mediant 4000
Device Up Time:	0d:1h:14m:42s:80th
Device Administrative State:	Unlocked
Device Operational State:	Enabled
Flash Size [bytes]:	264241152
RAM Size [bytes]:	4213809152
CPU Speed [MHz]:	800
▼ Versions	
Version ID:	6.40Kor.004.003
DSP Type:	1
DSP Software Version:	64010
DSP Software Name:	5014AE3_H
Flash Version:	560
▲ Loaded Files	

➤ **To delete a loaded file:**

- Click the **Delete** button corresponding to the file that you want to delete. Deleting a file takes effect only after device reset (see 'Resetting the Device' on page [357](#)).

34.2 Viewing Ethernet Port Information

The Ethernet Port Information page displays read-only information on the Ethernet port connections. This includes information such as duplex mode, and speed.



Note: The Ethernet Port Information page can also be accessed from the Home page (see 'Viewing the Home Page' on page 49).

➤ **To view Ethernet port information:**

- Open the Ethernet Port Information page (**Status & Diagnostics** tab > **System Status** menu > **Ethernet Port Information**).

Figure 34-2: Ethernet Port Information Page

	Active	Speed	Duplex Mode	State
Group no.1				
1	<input type="text" value="Yes"/>	<input type="text" value="1000 Mbps"/>	<input type="text" value="Full Duplex"/>	<input type="text" value="Forwarding"/>
2	<input type="text" value="No"/>	<input type="text" value="10 Mbps"/>	<input type="text" value="Half Duplex"/>	<input type="text" value="Disabled"/>
Group no.2				
3	<input type="text" value="Yes"/>	<input type="text" value="1000 Mbps"/>	<input type="text" value="Full Duplex"/>	<input type="text" value="Forwarding"/>
4	<input type="text" value="No"/>	<input type="text" value="10 Mbps"/>	<input type="text" value="Half Duplex"/>	<input type="text" value="Disabled"/>
Group no.3				
5	<input type="text" value="No"/>	<input type="text" value="10 Mbps"/>	<input type="text" value="Half Duplex"/>	<input type="text" value="Disabled"/>
6	<input type="text" value="Yes"/>	<input type="text" value="1000 Mbps"/>	<input type="text" value="Full Duplex"/>	<input type="text" value="Forwarding"/>
Group no.4				
7	<input type="text" value="Yes"/>	<input type="text" value="1000 Mbps"/>	<input type="text" value="Full Duplex"/>	<input type="text" value="Forwarding"/>
8	<input type="text" value="No"/>	<input type="text" value="10 Mbps"/>	<input type="text" value="Half Duplex"/>	<input type="text" value="Disabled"/>

Table 34-1: Ethernet Port Information Parameters

Parameter	Description
State	Displays the state of the port: <ul style="list-style-type: none"> ▪ "Forwarding": Active port (data is being received and sent) ▪ "Disabled": Redundancy port
Group Member	Displays the port-pair group ID to which the port belongs.

35 Carrier-Grade Alarms

This section describes how to view the following types of alarms:

- Active alarms - see 'Viewing Active Alarms' on page 391
- Alarm history - see 'Viewing Alarm History' on page 392

35.1 Viewing Active Alarms

The Active Alarms page displays a list of currently active alarms. You can also access this page from the Home page (see 'Viewing the Home Page' on page 49).

➤ To view the list of active alarms:

- Open the Active Alarms page (**Status & Diagnostics** tab > **System Status** menu > **Carrier-Grade Alarms** > **Active Alarms**).

Figure 35-1: Active Alarms Page

Sequential number	Severity	Source	Description	Date
1	Major	Chassis#0/PowerSupply#1	Power-Supply Alarm. Power-Supply is missing	26.10.2011 , 11:30:58
3	Minor	Board#1/EthernetLink#2	Ethernet link alarm. LAN port number 2 is down.	26.10.2011 , 11:31:3
4	Minor	Board#1/EthernetLink#4	Ethernet link alarm. LAN port number 4 is down.	26.10.2011 , 11:31:3
5	Minor	Board#1/EthernetLink#5	Ethernet link alarm. LAN port number 5 is down.	26.10.2011 , 11:31:3
6	Minor	Board#1/EthernetLink#8	Ethernet link alarm. LAN port number 8 is down.	26.10.2011 , 11:31:3
10	Major	Board#1	Configuration mismatch in the system. SYS_HA: Active and Redundant modules have different feature keys.	26.10.2011 , 11:36:37

For each alarm, the following information is provided:

- **Severity:** severity level of the alarm:
 - Critical (red)
 - Major (orange)
 - Minor (yellow)
- **Source:** unit from which the alarm was raised
- **Description:** brief explanation of the alarm
- **Date:** date and time that the alarm was generated

You can view the next 20 alarms (if exist), by clicking the **Go to page** button.

35.2 Viewing Alarm History

The Alarms History page displays a list of alarms that have been raised and traps that have been cleared.

➤ **To view the list of history alarms:**

- Open the Alarms History page (**Status & Diagnostics** tab > **System Status** menu > **Carrier-Grade Alarms** > **Alarms History**).

Figure 35-2: Alarms History Page

Sequential number	Severity	Source	Description	Date
1	Major	Chassis#0/PowerSupply#1	Power-Supply Alarm. Power-Supply is missing	26.10.2011 , 11:30:58
2	Major	Board#1	No HA! SYS_HA: Waiting for redundant unit.	26.10.2011 , 11:31:2
3	Minor	Board#1/EthernetLink#2	Ethernet link alarm. LAN port number 2 is down.	26.10.2011 , 11:31:3
4	Minor	Board#1/EthernetLink#4	Ethernet link alarm. LAN port number 4 is down.	26.10.2011 , 11:31:3
5	Minor	Board#1/EthernetLink#5	Ethernet link alarm. LAN port number 5 is down.	26.10.2011 , 11:31:3
6	Minor	Board#1/EthernetLink#8	Ethernet link alarm. LAN port number 8 is down.	26.10.2011 , 11:31:3
7	Major	Board#1	Configuration mismatch in the system. SET commands are disabled due to update-process.	26.10.2011 , 11:31:12
8	Cleared	Board#1	Alarm cleared: No HA! SYS_HA: Waiting for redundant unit.	26.10.2011 , 11:36:37
9	Cleared	Board#1	SET commands are available.	26.10.2011 , 11:36:37
10	Major	Board#1	Configuration mismatch in the system. SYS_HA: Active and Redundant modules have different feature keys.	26.10.2011 , 11:36:37

For each alarm, the following information is provided:

- **Severity:** severity level of the alarm:
 - Critical (red)
 - Major (range)
 - Minor (yellow)
 - Cleared (green)
- **Source:** unit from which the alarm was raised
- **Description:** brief explanation of the alarm
- **Date:** date and time that the alarm was generated

You can view the next 20 alarms (if exist), by clicking the **Go to page** button.

➤ **To delete all the alarms in the table:**

1. Click the **Delete History Table** button; a confirmation message box appears.
2. Click **OK** to confirm.

36 Performance Monitoring

This section describes how to view performance monitoring.

36.1 Viewing MOS per Media Realm

The MOS Per Media Realm page displays statistics on Media Realms (configured in 'Configuring Media Realms' on page 157). This page provides two graphs:

- Upper graph: displays the Mean Opinion Score (MOS) quality in RTCP data per selected Media Realm.
- Lower graph: displays the bandwidth of transmitted media (in Kbps) in RTCP data per Media Realm.



➤ To view the MOS per Media Realm graph:

1. Open the MOS Per Media Realm page (**Status & Diagnostics** tab > **Performance Monitoring** menu > **MOS Per Media Realm**).

Figure 36-1: MOS Per Media Realm Graph



2. From the 'Media Realm' drop-down list, select the Media Realm for which you want to view.

Use the **Zoom In**  button to increase the displayed time resolution or the **Zoom Out**  button to decrease it. Instead of using these zoom buttons, you can use the slide ruler. As you increase the resolution, more data is displayed on the graph. The minimum resolution is about 30 seconds; the maximum resolution is about an hour.

To pause the graph, click the **Pause** button; click **Play** to resume.

36.2 Viewing Quality of Experience

The Quality Of Experience page provides statistical information on calls per SRD or IP Group. The statistics can be further filtered to display incoming and/or outgoing call direction, and type of SIP dialog (INVITE, SUBSCRIBE, or all).

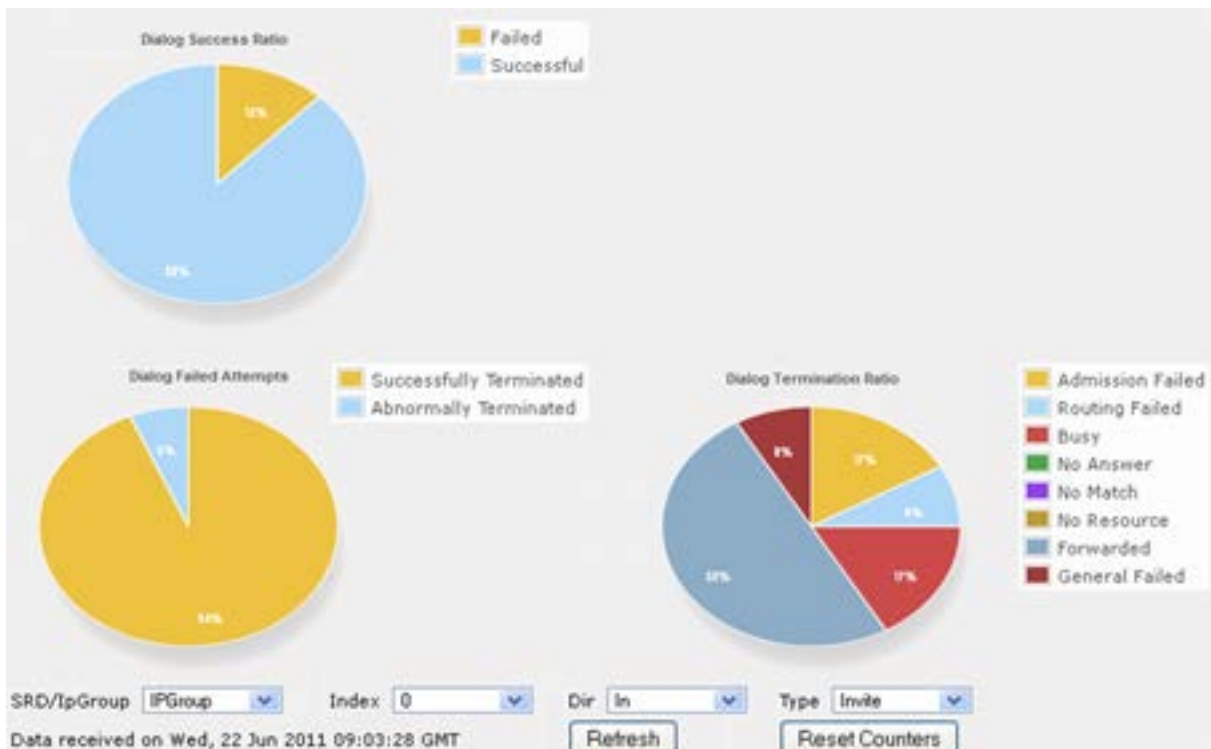
This page provides three pie charts:

- Dialog Success Ratio: displays the SIP call and subscribe (SUBSCRIBE) dialog success-failed ratio.
- Dialog Failed Attempts: displays the failed call attempts. This includes the number of calls and subscribes which were successfully and abnormally terminated.
- Dialog Termination Ratio: displays call termination by reason (e.g., due to no answer).

➤ **To view Quality of Experience:**

1. Open the Quality Of Experience page (**Status & Diagnostics** tab > **Performance Monitoring** menu > **Quality Of Experience**).

Figure 36-2: Quality Of Experience Graph



2. From the 'SRD/IpGroup' drop-down list, select whether you want to view QoE for an SRD or IP Group.
3. From the 'Index' drop-down list, select the SRD or IP Group index.
4. From the 'Dir' drop-down list, select the call direction:
 - **In** - incoming calls
 - **Out** - outgoing calls
 - **Both** - incoming and outgoing calls
5. From the 'Type' drop-down list, select the SIP message type:
 - **Invite** - INVITE
 - **Subscribe** - SUBSCRIBE
 - **Other** - all SIP messages

To refresh the charts, click **Refresh**. To reset the counters, click **Reset Counters**.

36.3 Viewing Average Call Duration

The Average Call Duration page displays information about a specific SRD or IP Group. This page includes two graphs:

- Upper graph: displays the number of calls (INVITEs).
- Lower graph: displays the average call duration.



➤ **To view average call duration:**

1. Open the Average Call Duration page (**Status & Diagnostics** tab > **Performance Monitoring** menu > **Average Call Duration**).

Figure 36-3: Average Call Duration Graph



2. From the 'SRD/IpGroup' drop-down list, select whether you want to view information for an SRD or IP Group.
3. From the 'Index' drop-down list, select the SRD or IP Group index.

Use the **Zoom In**  button to increase the displayed time resolution or the **Zoom Out**  button to decrease it. Instead of using these zoom buttons, you can use the slide ruler. As you increase the resolution, more data is displayed on the graph. The minimum resolution is about 30 seconds; the maximum resolution is about an hour.

To pause the graph, click the **Pause** button; click **Play** to resume.

Reader's Notes

37 VoIP Status

This section describes how to view VoIP status and statistics.

37.1 Viewing Active IP Interfaces

The IP Interface Status page displays the device's active IP interfaces that are listed in the Multiple Interface Table page (see 'Configuring IP Network Interfaces' on page 98).

➤ **To view the active IP network interfaces:**

- Open the IP Interface Status page (**Status & Diagnostics** tab > **VoIP Status** menu > **IP Interface Status**).

Figure 37-1: IP Interface Status Page

Index	Application Type	Address Type	Interface Mode	IP Address	Prefix Length	Gateway	VLAN ID	Interface Name
1	Maintenance	IPv4	IPv4 Manual	10.66.33.44	16	10.66.0.1	25	HA
0	O+M+C	IPv4	IPv4 Manual	10.3.50.40	16	10.3.0.1	1	O+M+C
2	Media	IPv4	IPv4 Manual	10.11.50.40	16	0.0.0.0	11	Media1
3	Media	IPv4	IPv4 Manual	10.13.50.40	16	0.0.0.0	13	Media2
4	Media	IPv4	IPv4 Manual	10.15.50.40	16	0.0.0.0	15	Media3
5	Media	IPv4	IPv4 Manual	10.12.50.40	16	0.0.0.0	112	Media4
6	Media	IPv4	IPv4 Manual	10.14.50.40	16	0.0.0.0	114	Media5

37.2 Viewing Registered Users

The SAS/SBC Registered Users page displays a list of registered SASSBC users recorded in the device's database.

➤ **To view registered SASSBC users:**

- Open the Registration Status page (**Status & Diagnostics** tab > **VoIP Status** menu > **Registered Users**).

Figure 37-2: SAS/SBC Registered Users Page

Address Of Record	Contact
1000@10.8.5.71	<sip:1000@10.8.5.71:5060>;expires=180; Active status: 1
1001@10.8.5.71	<sip:1001@10.8.5.71:5060>;expires=180; Active status: 1
1100@10.8.5.71	<sip:1100@10.8.5.71:5060>;expires=180; Active status: 1
1101@10.8.5.71	<sip:1101@10.8.5.71:5060>;expires=180; Active status: 1
2000@10.8.5.72	<sip:2000@10.8.5.72:5060>;expires=180; Active status: 1

Table 37-1: SAS/SBC Registered Users Parameters

Column Name	Description
Address of Record	An address-of-record (AOR) is a SIP or SIPS URI that points to a domain with a location service that can map the URI to another URI (Contact) where the user might be available.
Contact	SIP URI that can be used to contact that specific instance of the User Agent for subsequent requests.

Reader's Notes

38 Reporting Information to External Party

This section describes features for reporting various information to an external party.

38.1 RTP Control Protocol Extended Reports (RTCP XR)

RTP Control Protocol Extended Reports (RTCP XR) is a VoIP management control that defines a set of metrics containing information for assessing VoIP call quality and for diagnosing problems. RTCP XR (RFC 3611) extends the RTCP reports defined in RFC 3550 by providing additional VoIP metrics. RTCP XR information publishing is implemented in the device according to <draft-johnston-sipping-rtcp-summary-07>. This draft defines how a SIP User Agent (UA) publishes the detailed information to a defined collector. RTCP XR measures VoIP call quality such as packet loss, delay, signal / noise / echo levels, estimated R-factor, and mean opinion score (MOS). RTCP XR measures these parameters using metrics as listed in the table below.



Note: RTCP XR is a customer ordered feature and thus, must be included in the Software License Key installed on the device.

RTCP XR messages containing key call-quality-related metrics are exchanged periodically (user-defined) between the device and the SIP UA. This allows an analyzer to monitor these metrics midstream, or a device to retrieve them using SNMP. The device can send RTCP XR reports to an Event State Compositor (ESC) server using PUBLISH messages. These reports can be sent at the end of each call and according to a user-defined interval between consecutive reports.

Table 38-1: RTCP XR Published VoIP Metrics

Group	Metric Name
General	Start Timestamp
	Stop Timestamp
	Call-ID
	Local Address (IP, Port & SSRC)
	Remote Address (IP, Port & SSRC)
Session Description	Payload Type
	Payload Description
	Sample Rate
	Frame Duration
	Frame Octets
	Frames per Packets
	Packet Loss Concealment
	Silence Suppression State
Jitter Buffer	Jitter Buffer Adaptive
	Jitter Buffer Rate

Group	Metric Name
	Jitter Buffer Nominal
	Jitter Buffer Max
	Jitter Buffer Abs Max
Packet Loss	Network Packet Loss Rate
	Jitter Buffer Discard Rate
Burst Gap Loss	Burst Loss Density
	Burst Duration
	Gap Loss Density
	Gap Duration
	Minimum Gap Threshold
Delay	Round Trip Delay
	End System Delay
	One Way Delay
	Interarrival Jitter
	Min Absolute Jitter
	Signal
	Signal Level
	Noise Level
	Residual Echo Return Noise
Quality Estimates	Listening Quality R
	RLQ Est. Algorithm
	Conversational Quality R
	RCQ Est. Algorithm
	External R In
	Ext. R In Est. Algorithm
	External R Out
	Ext. R Out Est. Algorithm
	MOS-LQ
	MOS-LQ Est. Algorithm
	MOS-CQ
	MOS-CQ Est. Algorithm
	QoE Est. Algorithm

➤ **To configure RTCP XR:**

1. Open the RTP/RTCP Settings page (**Configuration** tab > **VoIP** menu > **Media** > **RTP/RTCP Settings**). The RTCP XR parameters are listed under the 'RTCP XR Settings' group, as shown below:

Figure 38-1: RTCP XR Parameters in RTP/RTCP Settings Page

▼ RTCP XR Settings	
Burst Threshold	-1
Delay Threshold	-1
R-Value Delay Threshold	-1
⚡ Enable RTCP XR	CE_VQMON_DISABLE ▼
Minimum Gap Size	16
RTCP XR Report Mode	Disable ▼
RTCP XR Packet Interval	0
Disable RTCP XR Interval Randomization	Disable ▼
RTCP XR Collection Server	
RTCP XR Collection Server Transport Type	Not Configured ▼

2. Configure the RTCP XR parameters, as required:
 - 'Enable RTCP XR' (*VQMonEnable*) - enables voice quality monitoring and RTCP XR.
 - 'Minimum Gap Size' (*VQMonGMin*) - defines the voice quality monitoring - minimum gap size (number of frames).
 - 'Burst Threshold' (*VQMonBurstTHR*) - defines the voice quality monitoring - excessive burst alert threshold.
 - 'Delay Threshold' (*VQMonDelayTHR*) - defines the voice quality monitoring - excessive delay alert threshold.
 - 'R-Value Delay Threshold' (*VQMonEOCRValTHR*) - defines the voice quality monitoring - end of call low quality alert threshold.
 - 'RTCP XR Report Mode' (*RTCPXRReportMode*) - determines whether RTCP XR reports are sent to the ESC and defines the interval in which they are sent.
 - 'RTCP XR Packet Interval' (*RTCPInterval*) - defines the time interval between adjacent RTCP reports.
 - 'Disable RTCP XR Interval Randomization' (*DisableRTCPRandomize*) - determines whether RTCP report intervals are randomized or whether each report interval accords exactly to the parameter RTCPInterval.
 - 'RTCP XR Collection Server' (*RTCPXREscIP*) - defines the IP address of the Event State Compositor (ESC).
 - 'RTCP XR Collection Server Transport Type' (*RTCPXRESCTransportType*) - determines the transport layer for outgoing SIP dialogs initiated by the device to the RTCP XR Collection Server.
3. Click **Submit**.
4. Reset the device for the settings to take effect.

38.2 Generating Call Detail Records

The Call Detail Record (CDR) contains vital statistic information on calls made from the device. The device can be configured to generate and report CDRs for various stages of the call, including SIP messages and/or media. You can configure when CDRs for a call are generated, for example, only at the end of the call or only at the start and end of the call. Once generated, the device sends the CDRs to a user-defined Syslog server.

The CDR Syslog message complies with RFC 3161 and is identified by Facility 17 (local1) and Severity 6 (Informational).

For CDR in RADIUS format, see 'RADIUS Accounting CDR Attributes' on page 406.

38.2.1 Configuring CDR Reporting

The procedure below describes how to configure CDR reporting.

➤ **To configure CDR reporting:**

1. Enable the Syslog feature for sending log messages generated by the device to a collecting log message server. For more information, see 'Configuring Syslog' on page 415.
2. Open the Advanced Parameters page (**Configuration** tab > **VoIP** menu > **SIP Definitions** > **Advanced Parameters**). The CDR parameters appear under the 'CDR and Debug' group, as shown below:

Figure 38-2: CDR Parameters in Advanced Parameters Page

CDR and Debug	
CDR Server IP Address	10.8.6.55
CDR Report Level	Start & End & Connect Call
Media CDR Report Level	End Media

3. Configure the parameters as required. For a description of the parameters, see 'Syslog, CDR and Debug Parameters' on page 453.
4. Click **Submit**.



Note: If the CDR server IP address is not configured, the CDRs are sent to the Syslog server, configured in 'Configuring Syslog' on page 415.

38.2.2 CDR Field Description

This section describes the CDR fields that are generated by the device.

38.2.2.1 CDR Fields for SBC Signaling

The CDR fields for SBC signaling are listed in the table below. The signaling CDRs are published for each SBC leg.

Table 38-2: CDR Fields for SBC Signaling

CDR Field Name	Description
SBCReportType	Report Type: <ul style="list-style-type: none">▪ CALL_START▪ CALL_CONNECT▪ CALL_END▪ DIALOG_START▪ DIALOG_END
EPTyp	Endpoint type (SBC)
SIPMethod	SIP message type
SIPCallId	Unique ID of call
SessionId	Unique Session ID
Orig	Call originator: <ul style="list-style-type: none">▪ LCL - for local▪ RMT - for remote
SourceIp	Source IP address
SourcePort	Source UDP port
DestIp	Destination IP address
DestPort	Destination UDP port
TransportType	Transport type: <ul style="list-style-type: none">▪ UDP▪ TCP▪ TLS
SrcURI	Source URI
SrcURIBeforeMap	Source URI before manipulation
DstURI	Destination URI
DstURIBeforeMap	Destination URI before manipulation
Durat	Call duration
TrmSd	Termination side (local or remote)
TrmReason	Termination reason

CDR Field Name	Description
TrmReasonCategory	Termination reason category: <ul style="list-style-type: none"> Calls with duration 0 (i.e., not connected): <ul style="list-style-type: none"> ✓ NO_ANSWER - GWAPP_NORMAL_CALL_CLEAR, GWAPP_NO_USER_RESPONDING, GWAPP_NO_ANSWER_FROM_USER_ALERTED ✓ BUSY - GWAPP_USER_BUSY ✓ NO_RESOURCES - GWAPP_RESOURCE_UNAVAILABLE_UNSPECIFIED, RELEASE_BECAUSE_NO_CONFERECE_RESOURCES_LEFT, RESOURCE_BECAUSE_NO_TRANSCODING_RESOURCES_LEFT, RELEASE_BECAUSE_GW_LOCKED ✓ NO_MATCH - RELEASE_BECAUSE_UNMATCHED_CAPABILITIES ✓ FORWARDED - RELEASE_BECAUSE_FORWARD ✓ GENERAL_FAILED - any other reason Calls with duration: <ul style="list-style-type: none"> ✓ NORMAL_CALL_CLEAR - GWAPP_NORMAL_CALL_CLEAR ✓ ABNORMALLY_TERMINATED - Anything else N/A - Reasons not belonging to above categories
SetupTime	Call setup time
ConnectTime	Call connect time
ReleaseTime	Call release time
RedirectReason	Redirect reason
RedirectURINum	Redirection URI
RedirectURINumBeforeManip	Redirect URI number before manipulation
TxSigIPDiffServ	Signaling IP DiffServ
IPGroup	IP Group description
SrdId	SRD name
SIPInterfaceId	SIP Interface ID
ProxySetId	Proxy Set ID
IpProfileId	IP Profile name
MediaRealmId	Media Realm name
DirectMedia	Direct media or traversing SBC: <ul style="list-style-type: none"> yes no
SIPTrmReason	SIP call termination reason (BYE, CANCEL, or SIP error codes, e.g., 404)

CDR Field Name	Description
SipTermDesc	<p>Description of SIP termination reason:</p> <ul style="list-style-type: none"> SIP Reason header, if exists, for example: SIP ;cause=200 ;text="Call completed elsewhere". If no SIP Reason header exists, the description is taken from the reason text, if exists, of the SIP response code, for example: "417 Unknown Resource-Priority". If no reason text exists in the SIP response code, the description is taken from an internal SIP response mapping mechanism. For example, if the device receives a SIP response "422", it sends in the CDR "422 Session Interval Too Small method" as the description.

An example of an SBC signaling CDR sent by the device is shown below:

```
[S=1] |SBCReportType |EPTyp| SIPCallId| SessionId |Orig |SourceIp
|SourcePort |DestIp |DestPort |TransportType |SrcURI
|SrcURIBeforeMap |DstURI |DstURIBeforeMap |Durat |TrmSd |TrmReason
|TrmReasonCategory |SetupTime |ConnectTime |ReleaseTime
|RedirectReason |RedirectURINum |RedirectURINumBeforeMap
|TxSigIPDiffServ |IPGroup (description) |SrdId (name)
|SIPInterfaceId |ProxySetId |IpProfileId (name) |MediaRealmId
(name) |DirectMedia |SIPTrmReason
[S=3] |CALL_END |SBC |170369730753201211288@10.132.10.245 |0 |RMT
|10.132.10.245 |5060 |10.132.10.250 |5070 |UDP |103@audiocodes.com
|103@audiocodes.com |101@10.132.10.250 |101@10.132.10.250 |0 |RMT
|GWAPP_NORMAL_CALL_CLEAR |NO_ANSWER |06:13:54.950 UTC Thu Mar 02
2012 | |06:14:01.175 UTC Thu Mar 02 2012 |-1 | | |40 |2 ( ) |0
(5070SRD) |2 |3 |0 ( ) |0 (lanmedia) |no |CANCEL
```

38.2.2.2 CDR Fields for SBC Media

The CDR fields for SBC media are listed in the table below. The media CDRs are published for each active media stream, thereby allowing multiple media CDRs, where each media CDR has a unique call ID corresponding to the signaling CDR.

Table 38-3: CDR Fields for SBC Media

CDR Field Name	Description
MediaReportType	Report type (media start, update, or end)
SIPCallId	Unique call ID
Cid	Channel CID
MediaType	Media type (audio, video, or text)
Coder	Coder name
PacketInterval	Coder packet interval
LocalRtpIp	Local RTP IP address
LocalRtpPort	Local RTP port
RemoteRtpIp	Remote RTP IP address
RemoteRtpPort	Remote RTP port
InPackets	Number of received packets

CDR Field Name	Description
OutPackets	Number of sent packets
LocalPackLoss	Local packet loss
RemotePackLoss	Remote packet loss
RTPdelay	RTP delay
RTPjitter	RTP jitter
TxRTPssrc	Tx RTP SSRC
RxRTPssrc	Local RTP SSRC
LocalRFactor	Local conversation quality
RemoteRFactor	Remote conversation quality
LocalMosCQ	Local MOS for conversation
RemoteMosCQ	Remote MOS for conversation
TxRTPIPDiffServ	Media IP DiffServ
LatchedRtplp	Remote IP address of the incoming RTP stream that the device "latched" on to as a result of the RTP latching mechanism for NAT traversal.
LatchedRtpPort	Remote RTP port of the incoming RTP stream that the device "latched" on to as a result of the RTP latching mechanism for NAT traversal.

38.2.3 Supported RADIUS Attributes

The table below describes the RADIUS Accounting CDR attributes included in the communication packets transmitted between the device and a RADIUS server.

Table 38-4: Supported RADIUS Accounting CDR Attributes

Attribute Number	Attribute Name	Vendor Specific Attribute (VSA) No.	Purpose	Value Format	Example	AAA
Request Attributes						
1	user-name	-	Account number or calling party number or blank	String up to 15 digits long	5421385747	Start Acc Stop Acc
4	nas-ip-address	-	IP address of the requesting device	Numeric	192.168.14.43	Start Acc Stop Acc
6	service-type	-	Type of service requested	Numeric	1: login	Start Acc Stop Acc
26	h323-incoming-conf-id	1	SIP call identifier	Up to 32 octets	-	Start Acc Stop Acc
26	h323-remote-address	23	IP address of the remote gateway	Numeric	-	Stop Acc

Attribute Number	Attribute Name	Vendor Specific Attribute (VSA) No.	Purpose	Value Format	Example	AAA
26	h323-conf-id	24	H.323/SIP call identifier	Up to 32 octets	-	Start Acc Stop Acc
26	h323-setup-time	25	Setup time in NTP format 1	String	-	Start Acc Stop Acc
26	h323-call-origin	26	The call's originator: Answering (IP) or Originator (PSTN)	String	Answer, Originate etc	Start Acc Stop Acc
26	h323-call-type	27	Protocol type or family used on this leg of the call	String	VoIP	Start Acc Stop Acc
26	h323-connect-time	28	Connect time in NTP format	String	-	Stop Acc
26	h323-disconnect-time	29	Disconnect time in NTP format	String	-	Stop Acc
26	H323-Disconnect-Cause	30	Q.931 disconnect cause code	Numeric	-	Stop Acc
26	h323-gw-id	33	Name of the gateway	String	SIPIDString	Start Acc Stop Acc
26	sip-call-id	34	SIP Call ID	String	abcde@ac.com	Start Acc Stop Acc
26	call-terminator	35	The call's terminator: PSTN-terminated call (Yes); IP-terminated call (No).	String	Yes, No	Stop Acc
30	called-station-id	-	Destination URI	String	8004567145	Start Acc
31	calling-station-id	-	Source URI	String	5135672127	Start Acc Stop Acc
40	acct-status-type	-	Account Request Type (start or stop) Note: 'start' isn't supported on the Calling Card application.	Numeric	1: start, 2: stop	Start Acc Stop Acc
41	acct-delay-time	-	No. of seconds tried in sending a particular record	Numeric	5	Start Acc Stop Acc
44	acct-session-id	-	A unique accounting identifier - match start & stop	String	34832	Start Acc Stop Acc

Attribute Number	Attribute Name	Vendor Specific Attribute (VSA) No.	Purpose	Value Format	Example	AAA
46	acct-session-time	-	For how many seconds the user received the service	Numeric	-	Stop Acc
47	acct-input-packets	-	Number of packets received during the call	Numeric	-	Stop Acc
48	acct-output-packets	-	Number of packets sent during the call	Numeric	-	Stop Acc
61	nas-port-type	-	Physical port type of device on which the call is active	String	0: Asynchronous	Start Acc Stop Acc
Response Attributes						
26	h323-return-code	103	The reason for failing authentication (0 = ok, other number failed)	Numeric	0 Request accepted	Stop Acc
44	acct-session-id	-	A unique accounting identifier – match start & stop	String	-	Stop Acc

Below is an example of RADIUS Accounting, where the non-standard parameters are preceded with brackets:

```
Accounting-Request (361)
user-name = 111
acct-session-id = 1
nas-ip-address = 212.179.22.213
nas-port-type = 0
acct-status-type = 2

acct-session-time = 1
acct-input-packets = 122
acct-output-packets = 220
called-station-id = 201
calling-station-id = 202
// Accounting non-standard parameters:
(4923 33) h323-gw-id =
(4923 23) h323-remote-address = 212.179.22.214
(4923 1) h323-ivr-out = h323-incoming-conf-id:02102944 600a1899
3fd61009 0e2f3cc5
(4923 30) h323-disconnect-cause = 22 (0x16)
(4923 27) h323-call-type = VOIP
(4923 26) h323-call-origin = Originate
(4923 24) h323-conf-id = 02102944 600a1899 3fd61009 0e2f3cc5
```


Part XI

Diagnostics

39 Syslog and Debug Recordings

Syslog is an event notification protocol that enables a device to send event notification messages across IP networks to event message collectors, also known as Syslog servers. The device contains an embedded Syslog client, which sends error reports / events that it generates to a remote Syslog server using the IP / UDP protocol. This information is a collection of error, warning, and system messages that records every internal operation of the device.

For receiving Syslog messages generated by the device, you can use any of the following Syslog servers:

- **Device's embedded Syslog server:** The device provides an embedded Syslog server, which is accessed through the Web interface. This provides limited Syslog server functionality.
- **Wireshark:** Third-party network protocol analyzer (<http://www.wireshark.org>).
- **Third-party, Syslog server:** Any third-party Syslog server program that enables filtering of messages according to parameters such as priority, IP sender address, time, and date.

39.1 Syslog Message Format

The Syslog message is sent from the device to a Syslog server as an ASCII (American Standard Code for Information Interchange) message. Syslog uses UDP as its underlying transport layer mechanism. By default, UDP port 514 is assigned to Syslog, but this can be changed (see 'Configuring Syslog' on page 415).

Below is an example of a Syslog message:

```
13:10:57.811 : 10.13.4.12 : NOTICE : [S=235][SID:1034099026] (
lgr_flow)(63          ) UdpTransportObject#0- Adding socket event
for address 10.33.2.42:5060 [Time: 04-19-2012@18:29:39]
```

Table 39-1: Syslog Message Format Description

Message Item	Description
Message Types	<p>Syslog generates the following types of messages:</p> <ul style="list-style-type: none"> ■ ERROR: Indicates that a problem has been identified that requires immediate handling. ■ WARNING: Indicates an error that might occur if measures are not taken to prevent it. ■ NOTICE: Indicates that an unusual event has occurred. ■ INFO: Indicates an operational message. ■ DEBUG: Messages used for debugging. <p>Notes:</p> <ul style="list-style-type: none"> ■ The INFO and DEBUG messages are required only for advanced debugging. Therefore, by default, they are not sent by the device. ■ When viewing Syslog messages in the Web interface, these message types are color coded.
Message Sequence Number [S=<number>]	<p>Syslog messages are sequentially numbered in the format [S=<number>], for example, "[S=643]".</p> <p>A skip in the number sequence of messages indicates a loss of message packets. For example, in the below Syslog message generation, messages 238 through 300 were not received. In other words, three Syslog messages were lost (the sequential numbers</p>

Message Item	Description
	<p>are indicated below in bold font):</p> <pre> 18:38:14. 52 : 10.33.45.72 : NOTICE: [S=235][SID:1034099026] (lgr_psbrdex)(619) recv <-- DIGIT(0) Ch:0 OnTime:0 InterTime:100 Direction:0 System:1 [File: Line:-1] 18:38:14. 83 : 10.33.45.72 : NOTICE: [S=236][SID:1034099026] (lgr_flow)(620) #0:DIGIT_EV [File: Line:-1] 18:38:14. 83 : 10.33.45.72 : NOTICE: [S=237][SID:1034099026] (lgr_flow)(621) #0:DIGIT_EV [File: Line:-1] 18:38:14.958 : 10.33.45.72 : NOTICE: [S=301][SID:1034099026] (lgr_flow)(625) #0:DIGIT_EV [File: Line:-1] </pre>
Log Number (lgr)(number)	Ignore this number; it has been replaced by the Message Sequence Number (described previously).
Session ID	<p>Automatically assigned (random), unique session identifier (session-id / SID) number per call in the CDR of sent Syslog messages and debug recording packets. This enables you to filter the information (such as SIP, Syslog, and media) according to the SID.</p> <ul style="list-style-type: none"> SBC application: A session is considered as both the outgoing and incoming legs, where both legs share the same SID. <p>The benefit of this unique numbering is that it enables you to filter the information (such as SIP, Syslog, and media) according to a specific SID.</p> <p>Note: Forked legs and alternative legs share the same SID.</p>
Message Body	Describes the message.
Timestamp	When the Network Time Protocol (NTP) is enabled, a timestamp string [hour:minutes:seconds] is added to all Syslog messages.

39.1.1 Event Representation in Syslog Messages

The Syslog message events that the device sends are represented by unique abbreviations. An example of an abbreviated event in a Syslog message indicating packet loss (PL) is shown below:

```
Apr  4 12:00:12 172.30.1.14 PL:5 [Code:3a002] [CID:3294] [Time:
20:17:00]
```

The table below lists these unique event abbreviations:

Table 39-2: Syslog Error Name Descriptions

Error Abbreviation	Error Name Description
AA	Invalid Accumulated Packets Counter
AC	Invalid Channel ID
AL	Invalid Header Length
AO	Invalid Codec Type
AP	Unknown Aggregation Payload Type

Error Abbreviation	Error Name Description
AR	Invalid Routing Flag Received
AT	Simple Aggregation Packets Lost
CC	Command Checksum Error
CE	Invalid Cell Coder Code
CS	Command Sequence Error
ES	8 sec Timeout Before Disconnect
HO	Host Received Overrun
IA	Invalid AMR Payload
IC	Invalid CID Error
IG	Invalid G723 Code
IP	Invalid payload length
IR	Invalid RTCP Packet
IS	Invalid SID Length
LC	Transmitter Received Illegal Command
LF	Lost Fax Frames In High Speed Mode
LM	Lost Modem Frames In High Speed Mode
MI	Misalignment Error
MR	Modem Relay Is Not Supported
OR	DSP JB Overrun
PH	Packet Header Error
PL	RTP Packet Loss
RB	Counts the number of BFI Frames Received From The Host
RD	No Available Release Descriptor
RO	RTP Reorder
RP	Unknown RTP Payload Type
RS	RTP SSRC Error
UF	Unrecognized Fax Relay Command
AA	Invalid Accumulated Packets Counter
AC	Invalid Channel ID
AL	Invalid Header Length
AO	Invalid Codec Type
AP	Unknown Aggregation Payload Type
AR	Invalid Routing Flag Received

39.1.2 Identifying AudioCodes Syslog Messages using Facility Levels

The device's Syslog messages can easily be identified and distinguished from Syslog messages from other equipment, by setting its Facility level. The Facility levels of the device's Syslog messages are numerically coded with decimal values. Facility level may use any of the "local use" facilities (0 through 7), according to RFC 3164. Implementing Facility levels is useful, for example, if you collect the device's as well as other equipments' Syslog messages on the same server. Therefore, in addition to filtering Syslog messages according to IP address, the messages can be filtered according to Facility level.

The Facility level is configured using the SyslogFacility ini file parameter, which provides the following options:

Table 39-3: Syslog Facility Levels

Numerical Value	Facility Level
16 (default)	local use 0 (local0)
17	local use 1 (local1)
18	local use 2 (local2)
19	local use 3 (local3)
20	local use 4 (local4)
21	local use 5 (local5)
22	local use 6 (local6)
23	local use 7 (local7)

Syslog messages begin with a less-than (" $<$ ") character, followed by a number, which is followed by a greater-than (" $>$ ") character. This is optionally followed by a single ASCII space. The number is known as the *Priority* and represents both the Facility level and the Severity level. A Syslog message with Facility level 16 is shown below:

```
Facility: LOCAL0 - reserved for local use (16)
```

39.1.3 SNMP Alarms in Syslog Messages

SNMP alerts are sent to the Syslog server using the following formats:

- **Raised Alarms:** RAISE-ALARM: <Alarm Name>; Textual Description: <Textual Description>; Severity <Alarm Severity>; Source <Alarm Source>; Unique ID: <Alarm Unique ID >.

If additional information exists in the alarm, then these are also added: Additional Info1:/ Additional Info2:/ Additional Info3

The Messages' Severity is as follows:

Table 39-4: Syslog Message Severity

ITU Perceived Severity (SNMP Alarm's Severity)	AudioCodes' Syslog Severity
Critical	RecoverableMsg
Major	RecoverableMsg
Minor	RecoverableMsg
Warning	Notice
Indeterminate	Notice

ITU Perceived Severity (SNMP Alarm's Severity)	AudioCodes' Syslog Severity
Cleared	Notice

- **Cleared Alarms:** CLEAR-ALARM: <Alarm Name>; Textual Description: <Textual Description>; Severity <Alarm Severity>; Source <Alarm Source>; Unique ID: <Alarm Unique ID >; If exists Additional Info1:/ Additional Info2:/ Additional Info3:

39.2 Configuring Syslog Settings

The procedure below describes how to configure Syslog. This includes defining the Syslog server address as well as selecting the activities on the device (for example, a parameter value change) that you want reported to the server.



Notes:

- For configuring CDR reporting, see 'Configuring CDR Reporting' on page 402.
- For viewing Syslog messages in the Web interface, see 'Viewing Syslog Messages' on page 420.
- For a detailed description on the Syslog parameters, see 'Syslog, CDR and Debug Parameters' on page 453.

➤ To configure Syslog :

1. Open the Syslog Settings page (**Configuration** tab > **System** menu > **Syslog Settings**).

Figure 39-1: Syslog Settings Page

▼ Syslog Settings	
Enable Syslog	Enable ▼
Syslog Server IP Address	10.8.2.2
Syslog Server Port	514
Debug Level	0 ▼
▼ Activity Types to Report via 'Activity Log' Messages	
Parameters Value Change	<input type="checkbox"/>
Auxiliary Files Loading	<input type="checkbox"/>
Device Reset	<input type="checkbox"/>
Flash Memory Burning	<input type="checkbox"/>
Device Software Update	<input type="checkbox"/>
Access to Restricted Domains	<input type="checkbox"/>
Non-Authorized Access	<input type="checkbox"/>
Sensitive Parameters Value Change	<input type="checkbox"/>
Login and Logout	<input type="checkbox"/>

2. Enable the Syslog feature by setting the 'Enable Syslog' to **Enable**.
3. Define the Syslog server using the 'Syslog Server IP Address' and 'Syslog Server Port' parameters.

4. Configure the debug level using the 'Debug Level' parameter.
5. Under the 'Activity Types to Report ...' group, select the activities to report.
6. Click **Submit** to apply your changes.

39.3 Configuring Debug Recording

The device enables you to activate debug recording and send debug recording packets to a defined capturing server. When the debug recording is activated, the device duplicates all messages that are sent and/or received by it and then sends them to an external IP address. The debug recording can be done for different types of traffic for example, RTP/RTCP, T.38 and SIP.

Debug recording is used for advanced debugging when you need to analyze internal messages and signals. Debug recording is also useful for recording network traffic in environments where hub or port mirroring is unavailable and for recording internal traffic between two endpoints on the same device.



Note: Debug recording is collected only on the device's OAMP interface.

➤ To configure and activate debug recording:

1. Open the Logging Settings page (**Configuration** tab > **System** menu > **Logging** > **Logging Settings**).

Figure 39-2: Logging Settings Page

▼ Debug Recording	
Debug Recording Destination IP	10.13.4.22
Debug Recording Destination Port	925
Debug Recording Status	Start ▼

2. Configure the debug capturing server using the 'Debug Recording Destination IP' and 'Debug Recording Destination Port' parameters.
3. From the 'Debug Recording Status' drop-down list, select **Start** to start the debug recording or **Stop** to end the recording.
4. Click **Submit** to apply your changes.

39.4 Filtering Syslog Messages and Debug Recordings

The device can filter Syslog messages and debug recording (DR) packets, sent by the device to a Syslog server and packet capturing application (such as Wireshark) respectively. This can be useful to reduce CPU consumption and minimize negative impact on VoIP performance.

You can configure up to 30 filtering rules, each based on a selected filtering criteria (e.g., an IP Group). Each filtering criteria can be configured with a range. For example, you can filter Syslog messages for IP Groups 1 through 4. For each filter criteria, you can enable or disable Syslog messages and debug recording.

Debug recording can also be filtered using various filtering criteria such as SIP signaling or signaling and media.

➤ **To configure logging filtering rules:**

1. Open the Logging Filters Table page (**Configuration** tab > **System** menu > **Logging** > **Logging Filters Table**).
2. Click the **Add** button; the Add Record dialog box appears:

Figure 39-3: Logging Filters Table - Add Record Dialog Box

The dialog box titled 'Add Record' has a close button (X) in the top right corner. It contains the following fields:

- Index:** A text input field containing the value '1'.
- Type:** A dropdown menu with 'Any Filter' selected.
- Value:** An empty text input field.
- Syslog:** A dropdown menu with 'Enable' selected.
- Capture Type:** A dropdown menu with 'Signaling + Media' selected.

At the bottom right, there are two buttons: 'Submit' (with a document icon) and 'Cancel' (with an X icon).

3. Configure the logging filter, as required. See the table below for a description of the parameters.
4. Click **Submit** to save your changes.



Notes:

- To configure the Syslog debug level, use the 'Debug Level' parameter (see 'Configuring Syslog' on page 415).
- The Logging Filters table can also be configured using the table ini file parameter, LoggingFilters.

Table 39-5: Logging Filters Table Parameters Description

Parameter	Description
Filter Type CLI: filter-type [LoggingFilters_Type]	<p>Defines the filter criteria.</p> <ul style="list-style-type: none"> ▪ [1] Any (default) ▪ [8] IP Group = Filters according to a specified IP Group ID listed in the IP Group table. ▪ [9] SRD = Filters according to a specified SRD ID listed in the SRD table. ▪ [10] Classification = Filters according to a specified Classification rule listed in the Classification table (applicable only to the SBC and application). ▪ [11] IP-to-IP Routing = Filters according to a specified SBC IP-to-IP routing rule listed in the IP-to-IP Routing table (applicable only to the SBC application). ▪ [12] User = Filters according to a specified user defined by username or user@host. ▪ [13] IP Trace = Filters according to a specified IP network trace wireshark-like expression. For a detailed description on configuring IP traces, see 'Filtering IP Network Traces' on page 418.

Parameter	Description
Value CLI: value [LoggingFilters_Value]	Defines the value of the selected filtering type in the 'Filter Type' parameter. The value can be the following: <ul style="list-style-type: none"> ▪ A single value ▪ A range, using a hyphen "-" between the two values, e.g., "1-3" ▪ Multiple, non-contiguous values, using commas "," between each value, e.g., "1,3,9" ▪ Any to indicate all ▪ For IP trace expressions, see e 'Filtering IP Network Traces' on page 418
Syslog [LoggingFilters_Syslog]	Enables Syslog messages for the defined logging filter: <ul style="list-style-type: none"> ▪ [0] Disable (default) ▪ [1] Enable
Capture Type [LoggingFilters_CaptureType]	Enables debug recordings for the defined logging filter and defines what to record: <ul style="list-style-type: none"> ▪ [0] None (default) ▪ [1] Signaling = Information related to signaling such as SIP signaling messages, Syslog, and CDR. ▪ [2] Signaling & Media = Signaling and media (RTP/RTCP/T.38). ▪ [3] Signaling & Media & PCM = Signaling, media, and PCM.

39.4.1 Filtering IP Network Traces

You can filter Syslog and debug recording messages for IP network traces, by setting the 'Filter Type' parameter to **IP Trace** in the Logging Filters table. IP traces are used to record any IP stream, according to destination and/or source IP address, or port and Layer-4 protocol (UDP, TCP or any other IP type as defined by <http://www.iana.com>).

When the **IP Trace** option is selected, only the 'Value' parameter is applicable; the 'Syslog' and 'Capture Type' parameters are not relevant. The 'Value' parameter configures the Wireshark-like filtering expressions for your IP trace. The following Wireshark-like expressions are supported:

Table 39-6: Supported Wireshark-like Expressions for 'Value' Parameter

Expression	Description
ip.src, ip.dst	Source and destination IP address
ip.addr	IP address - up to two IP addresses can be entered
ip.proto	IP protocol type (PDU) entered as an enumeration value (e.g., 1 is ICMP, 6 is TCP, 17 is UDP)
udp, tcp, icmp, sip, ldap, http, https	Single expressions for protocol type
udp.port, tcp.port	Transport layer
udp.srcport, tcp.srcport	Transport layer for source port
udp.dstport, tcp.dstport	Transport layer for destination port
and, &&, ==, <, >	Between expressions

Below are examples of configured expressions for the 'Value' parameter:

- `udp && ip.addr==10.8.6.55`
- `ip.src==10.8.6.55 && udp.port>=5000 and udp.port<6000`
- `ip.dst==10.8.0.1/16`
- `ip.addr==10.8.6.40`

For conditions requiring the "or" / "||" expression, add multiple table rows. For example, the Wireshark condition "(ip.src == 1.1.1.1 or ip.src == 2.2.2.2) and ip.dst == 3.3.3.3" can be configured using the following two table row entries:

1. `ip.src == 1.1.1.1 and ip.dst == 3.3.3.3`
2. `ip.src == 2.2.2.2 and ip.dst == 3.3.3.3`



Note: If the 'Value' field is left empty, the device will record all IP traffic types.

39.5 Viewing Syslog Messages

You can use the following tools to view the Syslog messages sent by the device:

- Web interface's Message Log page (see below).
- Any third-party Syslog server (e.g., Wireshark).

The procedure below describes how to view Syslog messages in the Web interface.



Notes:

- It's not recommended to keep a Message Log session open for a prolonged period. This may cause the device to overload. For prolonged (and detailed) debugging, use an external Syslog server.
- You can select the Syslog messages in this page, and copy and paste them into a text editor such as Notepad. This text file (*txt*) can then be sent to AudioCodes Technical Support for diagnosis and troubleshooting.

➤ To activate the Web interface's Message Log:

1. Enable Syslog (see 'Configuring Syslog' on page 415).
2. Open the Message Log page (**Status & Diagnostics** tab > **System Status** menu > **Message Log**); the Message Log page is displayed and the log is activated.

Figure 39-4: Message Log Page

```
Log is Activated

11d:14h:43m:9s ( lgr_psbrdex) (2662 ) recv <-- ON_HOOK Ch:1
11d:14h:43m:9s ( lgr_flow) (2663 ) #1:ON_HOOK_EV
11d:14h:43m:9s ( lgr_flow) (2664 ) | #1:ON_HOOK_EV
11d:14h:43m:9s ( lgr_psbrdif) (2665 ) #1:cpDigitMapHndlr_Stop - Stopped (0)
11d:14h:43m:9s ( lgr_psbrdif) (2666 ) #1:CloseChannel: ChannelNum=1
11d:14h:43m:9s ( lgr_psbrdif) (2667 ) Open channel: IsVoiceOn: 1, IsT38On: 1, IsVbdOn: 0, Is
11d:14h:43m:9s ( lgr_psbrdif) (2668 ) #1:OpenChannel:on Trunk -1 BChannel:1 CID=1 with Voice
11d:14h:43m:9s ( lgr_psbrdif) (2669 ) #1:OpenChannel VoiceVolume= 0, DTHFVolume = -11, Input
11d:14h:43m:9s ( lgr_psbrdif) (2670 ) OpenChannel, CoderType = 15, Interval = 4, M = 1
11d:14h:43m:9s ( lgr_psbrdif) (2671 ) #1:FAXTransportType = 1
11d:14h:43m:9s ( lgr_psbrdif) (2672 ) #1:ConfigFaxModemChannelParams NSEMode=0, CNGDetMode=
11d:14h:43m:9s ( lgr_psbrdif) (2673 ) Detectors: Amd:0, Ans:0 En:0 IBScmd:0xa1
11d:14h:43m:9s ( lgr_psbrdif) (2674 ) #1:PSOSBoardInterface::StopPlayTone- Called
11d:14h:43m:9s ( lgr_psbrdex) (2675 ) recv <-- OFF_HOOK Ch:1
11d:14h:43m:9s ( lgr_flow) (2676 ) #1:OFF_HOOK_EV
11d:14h:43m:9s ( lgr_flow) (2677 ) | #1:OFF_HOOK_EV
11d:14h:43m:9s ( lgr_psbrdif) (2678 ) UpdateChannelParams, Channel 1
11d:14h:43m:9s ( lgr_psbrdif) (2679 ) #1:ConfigFaxModemChannelParams NSEMode=0, CNGDetMode=
11d:14h:43m:9s ( lgr_psbrdif) (2680 ) ActivatedDigitMap for channel : 1, MaxDialStringLength
```

The displayed logged messages are color-coded as follows:

- Yellow - fatal error message
- Blue - recoverable error message (i.e., non-fatal error)
- Black - notice message

➤ To stop and clear the Message Log:

- Close the Message Log page by accessing any another page in the Web interface.

39.6 Collecting Debug Recording Messages

To collect debug recording packets, the open source program Wireshark is used. AudioCodes proprietary plug-in files for Wireshark, which are shipped in your software kit, are also required.



Notes:

- The default debug recording port is 925. You can change the port in Wireshark (**Edit** menu > **Preferences** > **Protocols** > **AC DR**).
- The plug-ins are per major software release and are applicable to Wireshark Ver. 1.62.
- The plug-ins are backward compatible.
- From Wireshark Ver. 99.08, the tpncp.dat file must be located in the folder, ...WireShark\tpncp.

➤ To install Wireshark and the plug-ins for debug recording:

1. Install Wireshark on your computer. The Wireshark program can be downloaded from <http://www.wireshark.org>.
2. Copy the supplied AudioCodes plug-in files to the directory in which you installed Wireshark, as follows:

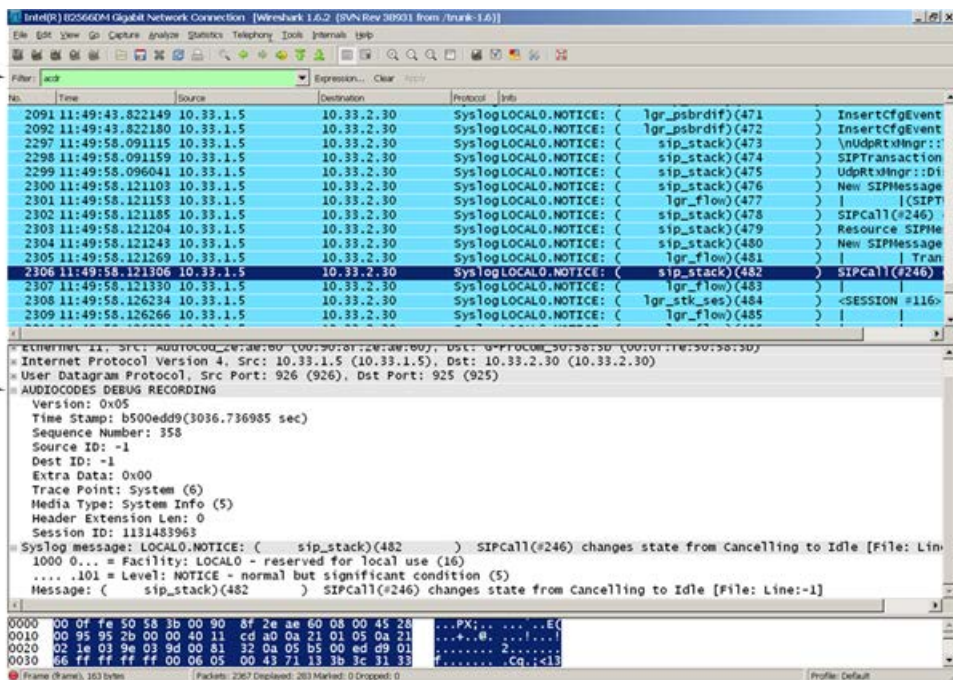
Copy this file	To this folder
...\dtds\cdr.dtd	Wireshark\dtds\
...\plugins\1.6.2*.dll	Wireshark\plugins\1.6.2
...\tpncp\tpncp.dat	Wireshark\tpncp

3. Start Wireshark.
4. In the Filter field, type "acdr" (see the figure below) to view the debug recording messages. Note that the source IP address of the messages is always the OAMP IP address of the device.

The device adds the header "AUDIOCODES DEBUG RECORDING" to each debug recording message, as shown below:

ACDR Filter →

Proprietary Header →



Wireshark capture showing SIP messages. The packet list displays several SyslogLOCAL0.NOTICE messages. The packet details for packet 2306 show the proprietary header with fields like Version, Time Stamp, Sequence Number, Source ID, Dest ID, Extra Data, Trace Point, Media Type, Header Extension Len, and Session ID. The message content is "Syslog message: LOCAL0.NOTICE: (sip_stack)(482) SIPcall(#246) changes state from Cancelling to Idle [File: Lin 1000 0... = Facility: LOCAL0 - reserved for local use (16)101 = Level: NOTICE - normal but significant condition (5) Message: (sip_stack)(482) SIPcall(#246) changes state from Cancelling to Idle [File: Line:-1]".

40 Testing SIP Signaling Calls

A simulated endpoint can be configured on the device to test SIP signaling of calls between it and a remote destination. This feature is useful in that it can remotely verify SIP message flow without involving the remote end side in the debug process. The SIP test call simulates the SIP signaling process - call setup, SIP 1xx responses, and through to completing the SIP transaction with a 200 OK.

The test call sends Syslog messages to a Syslog server, showing the SIP message flow, DTMF signals, termination reasons, as well as voice quality statistics.

40.1 Configuring Test Call Endpoints

The Test Call table enables you to test the SIP signaling (setup and registration) of calls and media (DTMF signals) between a simulated phone on the device and a remote endpoint. These tests involve both incoming and outgoing calls, where the test endpoint can be configured as the caller or called party. Test calls can be dialed automatically at a user-defined interval and/or manually when required. The simulated phone and remote endpoints are defined as SIP URIs (user@host) and the remote destination can be defined as an IP Group, IP address, or according to an Outbound IP Routing rule. You can also enable automatic registration of the endpoint.

When a SIP test call is initiated, the device generates a SIP INVITE towards the remote endpoint (e.g., a SIP proxy server or softswitch). It simulates the SIP call setup process, managing SIP 1xx responses and completing the SIP transaction with a 200 OK.



Notes:

- By default, you can configure up to five test calls. This maximum can be increased by installing the relevant Software License Key. For more information, contact your AudioCodes sales representative.
- The Test Call Endpoint table can also be configured using the table ini file parameter Test_Call (see 'SIP Test Call Parameters' on page 452).

➤ To configure test calls:

1. Open the Test Call Table page (**Configuration** tab > **System** menu > **Test Call** > **Test Call Table**).
2. Click the **Add** button; the following dialog box appears:

Figure 40-1: General Tab of Test Call Table

3. Configure the test endpoint parameters as desired. See the table below for a description of these parameters.
4. Click **Submit** to apply your settings.

Test Call Table Parameters

Parameter	Description
General Tab	
Endpoint URI [Test_Call_EndpointURI]	Defines the endpoint's URI. This can be defined as a user or user@host. The device identifies this endpoint only by the URI's user part. The URI's host part is used in the SIP From header in REGISTER requests. The valid value is a string of up to 150 characters. By default, this parameter is not configured.
Called URI [Test_Call_CalledURI]	Defines the destination (called) URI (user@host). The valid value is a string of up to 150 characters. By default, this parameter is not configured.
Route By [Test_Call_DestType]	Defines the type of routing method. This applies to incoming and outgoing calls. <ul style="list-style-type: none"> ▪ [0] GW Tel2IP = (Default) Calls are matched by (or routed to) an SRD and Application type (defined in the SRD and Application Type parameters below). ▪ [1] IP Group = Calls are matched by (or routed to) an IP Group ID. ▪ [2] Dest Address = Calls are matched by (or routed to) an SRD and application type. Notes: <ul style="list-style-type: none"> ▪ For REGISTER messages, the option [0] cannot be used as the routing method. ▪ For REGISTER messages, if option [1] is used, only Server-type IP Groups can be used.
IP Group ID [Test_Call_IPGroupID]	Defines the IP Group ID to which the test call is sent or from which it is received. Notes: <ul style="list-style-type: none"> ▪ This parameter is applicable only if option [1] is configured for the 'Route By' parameter. ▪ This IP Group is used for incoming and outgoing calls.
Destination Address [Test_Call_DestAddress]	Defines the destination host. This can be defined as an IP address[:port] or DNS name[:port]. Note: This parameter is applicable only if the 'Route By' parameter is set to [2] (Dest Address).
Destination Transport Type [Test_Call_DestTransportType]	Defines the transport type for outgoing calls. <ul style="list-style-type: none"> ▪ [-1] Not configured (default) ▪ [0] UDP ▪ [1] TCP ▪ [2] TLS Note: This parameter is applicable only if the 'Route By' parameter is set to [2] (Dest Address).

Parameter	Description
SRD [Test_Call_SRD]	Defines the SRD for the endpoint. The default is SRD 0. Note: This parameter is applicable only if the 'Route By' parameter is set any option except [1] (IP Group).
Application Type [Test_Call_ApplicationType]	Defines the application type for the endpoint. This, in effect, associates the IP Group and SRD to a specific SIP interface. <ul style="list-style-type: none"> [0] GW & IP2IP (default) [2] SBC
Authentication Tab	
Note: These parameters are applicable only if the test endpoint is set to Caller (see the 'Call Party' parameter).	
Auto Register [Test_Call_AutoRegister]	Enables automatic registration of the endpoint. The endpoint can register to the device itself or to the 'Destination Address' or 'IP Group ID' parameter settings (see above). <ul style="list-style-type: none"> [0] False (default) [1] True
User Name [Test_Call_UserName]	Defines the authentication username. By default, no username is defined.
Password [Test_Call_Password]	Defines the authentication password. By default, no password is defined.
Test Settings Tab	
Call Party [Test_Call_CallParty]	Defines whether the test endpoint is the initiator or receiving side of the test call. <ul style="list-style-type: none"> [0] Caller (default) [1] Called
Maximum Channels for Session [Test_Call_MaxChannels]	Defines the maximum number of concurrent channels for the test session. For example, if you have configured an endpoint "101" and you set this parameter to "3", the device automatically creates three simulated endpoints - "101", "102" and "103" (i.e., consecutive endpoint URIs are assigned). The default is 1.
Call Duration [Test_Call_CallDuration]	Defines the call duration (in seconds). The valid value is -1 to 100000. The default is 20. A value of 0 means infinite. A value of -1 means that the parameter value is automatically calculated according to the values of the 'Calls per Second' and 'Maximum Channels for Session' parameters. Note: This parameter is applicable only if 'Call Party' is set to Caller .
Calls per Second [Test_Call_CallsPerSecond]	Defines the number of calls per second. Note: This parameter is applicable only if 'Call Party' is set to Caller .

Parameter	Description
Test Mode [Test_Call_TestMode]	<p>Defines the test session mode.</p> <ul style="list-style-type: none"> ▪ [0] Once = (Default) The test runs until the lowest value between the following is reached: <ul style="list-style-type: none"> ✓ Maximum channels is reached for the test session, configured by 'Maximum Channels for Session'. ✓ Call duration ('Call Duration') multiplied by calls per second ('Calls per Second'). ✓ Test duration expires, configured by 'Test Duration'. ▪ [1] Continuous = The test runs until the configured test duration is reached. If it reaches the maximum channels configured for the test session (in the 'Maximum Channels for Session'), it waits until the configured call duration of a currently established tested call expires before making the next test call. In this way, the test session stays within the configured maximum channels. <p>Note: This parameter is applicable only if 'Call Party' is set to Caller.</p>
Test Duration [Test_Call_TestDuration]	<p>Defines the test duration (in minutes).</p> <p>The valid value is 0 to 100000. The default is 0 (i.e., unlimited).</p> <p>Note: This parameter is applicable only if 'Call Party' is set to Caller.</p>
Play [Test_Call_Play]	<p>Enables playing a user-defined DTMF signal to the answered side of the call.</p> <ul style="list-style-type: none"> ▪ [0] Disable (default) ▪ [1] DTMF <p>Notes:</p> <ul style="list-style-type: none"> ▪ To configure the played DTMF signal, see 'Configuring DTMF Tones for Test Calls' on page 428. ▪ To configure the DTMF signaling type (e.g., out-of-band or in-band) use the 'DTMF Transport Type' parameter (see 'Configuring DTMF Transport Types' on page 154). ▪ This parameter is applicable only if 'Call Party' is set to Caller.
Schedule Interval [Test_Call_ScheduleInterval]	<p>Defines the interval (in minutes) between automatic outgoing test calls.</p> <p>The valid value range is 0 to 100000. The default is 0 (i.e., scheduling is disabled).</p> <p>Note: This parameter is applicable only if 'Call Party' is set to Caller.</p>

40.1.1 Starting, Stopping and Restarting Test Calls

The procedure below describes how to start, stop, and restart test calls.

➤ To start, stop, and restart a test call:

1. In the Test Call table, select the required test call entry; the **Actions** button appears above the table.
2. From the **Actions** drop-down list, choose the required command:
 - **Dial:** starts the test call (this action is applicable only if the test call party is the caller).
 - **Drop Call:** stops the test call.
 - **Restart:** ends all established calls and then starts the test call session again.

The status of the test call is displayed in the 'Test Status' field of the Test Call table:

- "Idle": test call is not active.
- "Scheduled": test call is planned to run (according to 'Schedule Interval' parameter settings)
- "Running": test call has been started (i.e., the **Dial** command was clicked)
- "Receiving": test call has been automatically activated by calls received for the test call endpoint from the remote endpoint (when all these calls end, the status returns to "Idle")
- "Terminating": test call is in the process of terminating the currently established calls (this occurs if the **Drop Call** command is clicked to stop the test)
- "Done": test call has been successfully completed (or was prematurely stopped by clicking the **Drop Call** command)

A more detailed description of this field is displayed below the table when you click the **Show/Hide** button (see 'Viewing Test Call Statistics' on page 427).

40.1.2 Viewing Test Call Statistics

In addition to viewing a brief status description of the test call in the 'Test Status' field (as described in 'Starting, Stopping and Restarting Test Calls' on page 426), you can also view a more detailed status description which includes test call statistics.

➤ To view statistics of a test call:

1. Open the Test Call Table page (**Configuration** tab > **System** menu > **Test Call** > **Test Call Table**).
2. Select the test call table entry whose call statistics you want to view.
3. Click the **Show/Hide** button; the call statistics are displayed in the **Test Statistics** pane located below the table, as shown in the figure below:

Figure 40-2: Viewing Test Call Statistics

The screenshot shows the 'Test Call Table' interface. At the top, there are buttons for 'Add +', 'Edit', 'Delete -', and 'Action'. Below these is a table with columns: Index, Endpoint URI, Called URI, Route By, IP Group ID, Destination Address, SRD, Application Type, Call Party, and Test Status. One entry is visible with Index 0, Endpoint URI 101, Called URI 102, Route By GW Tel2IP, IP Group ID -1, Destination Address 10.13.4.12, SRD 0, Application Type GW & IP2IP, Call Party Caller, and Test Status Running. To the right of the table is a 'Show/Hide' button. Below the table, there is a section titled 'Test Call Table #0' containing details for the selected entry. At the bottom, there is a 'Test Statistics' section with various metrics.

Index	Endpoint URI	Called URI	Route By	IP Group ID	Destination Address	SRD	Application Type	Call Party	Test Status
0	101	102	GW Tel2IP	-1	10.13.4.12	0	GW & IP2IP	Caller	Running

Test Call Table #0

Endpoint URI: 101
 Route By: GW Tel2IP
 Destination Address: 10.13.4.12
 SRD: 0
 Auto Register: Disable
 Password:
 Maximum Channels for Session: 4
 Calls per Second: 10
 Test Duration (minutes): 4
 Schedule Interval (minutes): 0

Called URI: 102
 IP Group ID: -1
 Destination Transport Type:
 Application Type: GW & IP2IP
 User Name:
 Call Party: Caller
 Call Duration (seconds): 20
 Test Mode: Once
 Play: DTMF

Test Statistics

Elapsed Time [HH:MM:SS]: 00:00:11
 Call Attempts: 4
 Total Failed Attempts: 2
 Test Status: Running
 Detailed Status: Running (Calls: 2, ASR: 50%)

Active Calls: 2
 Total Established Calls: 2
 Remote Disconnections Count: 0
 Average CPS:

The 'Test Statistics' pane displays the following test session information:

- **Elapsed Time:** Duration of the test call since it was started (or restarted).
- **Active Calls:** The number of currently active test calls.
- **Call Attempts:** The number of calls that were attempted.
- **Total Established Calls:** The total number of calls that were successfully established.
- **Total Failed Attempts:** The total number of calls that failed to be established.

- **Remote Disconnections Count:** Number of calls that were disconnected by the remote side.
- **Average CPS:** The average calls per second.
- **Test Status:** Displays the status (brief description) as displayed in the 'Test Status' field (see 'Starting, Stopping and Restarting Test Calls' on page 426).
- **Detailed Status:** Displays a detailed description of the test call status::
 - "Idle": The test call is currently not active.
 - "Scheduled - Established Calls: <established calls>, ASR: <%>": The test call is planned to run (according to 'Schedule Interval' parameter settings) and also shows the following summary of completed test calls:
 - ◆ Total number of test calls that were established.
 - ◆ Number of successfully answered calls out of the total number of calls attempted (ASR).
 - "Running (Calls: <number of active calls>, ASR: <%>)": The test call has been started (i.e., the **Dial** command was clicked) and shows the following:
 - ◆ Number of currently active test calls.
 - ◆ Number of successfully answered calls out of the total number of calls attempted (Answer Seizure Ratio or ASR).
 - "Receiving (<number of active calls>)": The test call has been automatically activated by calls received for this configured test call endpoint from the configured remote endpoint. When all these calls terminate, the status returns to "Idle".
 - "Terminating (<number of active calls>)": The **Drop Call** command has been clicked to stop the test call and the test call is in the process of terminating the currently active test calls.
 - "Done - Established Calls: <established calls>, ASR: <%>": The test call has been successfully completed (or was prematurely stopped by clicking the **Drop Call** command) and shows the following:
 - ◆ Total number of test calls that were established.
 - ◆ Number of successfully answered calls out of the total number of calls attempted (ASR).



Note: On the receiving side, when the first call is accepted in "Idle" state, statistics are reset.

40.2 Configuring DTMF Tones for Test Calls

By default, no DTMF signal is played to an answered test call (incoming or outgoing). However, you can enable this per configured test call in the Test Call table (see 'Configuring Test Call Endpoints' on page 423). If enabled, the default DTMF signal that is played is "3212333". You can change this as described below.



Notes:

- The DTMF signaling type (e.g., out-of-band or in-band) can be configured using the 'DTMF Transport Type' parameter. For more information, see 'Configuring DTMF Transport Types' on page 154).
- To generate DTMF tones, the device's DSP resources are required.

➤ **To configure the played DTMF signal to answered test call:**

1. Open the Test Call Settings page (**Configuration** tab > **System** menu > **Test Call** > **Test Call Settings**).

Figure 40-3: Test Call Settings Page

Test Call DTMF String	3212333
-----------------------	---------

2. In the 'Test Call DTMF String' field, enter the DTMF string (up to 15 digits).
3. Click **Submit**.

40.3 Configuring SBC Test Call with External Proxy

The SBC Test Call feature tests incoming SBC SIP call flow between a simulated test endpoint on the device and a remote SIP endpoint, when registration and routing is done through an external proxy/registrar server such as a hosted IP PBX in the WAN. In other words, the complete SIP flow, including the path to/from the external proxy/registrar can be tested.

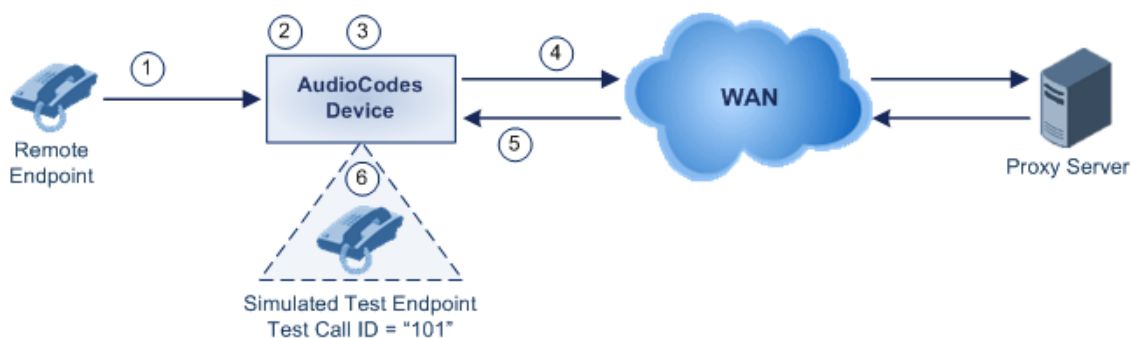
As this test call type involves an SBC call, you need to configure regular SBC rules such as classification and IP-to-IP routing. Therefore, this test call also allows you to verify correct SBC configuration.

For this test call, you also need to configure the following call IDs:

- Test Call ID - prefix number of the simulated endpoint on the device.
- SBC Test ID - prefix number of called number for identifying incoming call as SBC test call. The device removes this prefix, enabling it to route the call according to the IP-to-IP Routing rules to the external proxy/registrar, instead of directly to the simulated endpoint. Only when the device receives the call from the proxy/registrar, does it route the call to the simulated endpoint.

The figure below displays an example of an SBC test call:

Figure 40-4: SBC Test Call Example



1. The call is received from the remote endpoint with the called number prefix "8101".
2. As the 'SBC Test ID' parameter is set to "8", the device identifies this call as a test call and removes the digit "8" from the called number prefix, leaving it as "101".
3. The device performs the regular SBC processing such as classification and manipulation.
4. The device routes the call, according to the configured SBC IP-to-IP routing rules, to the proxy server.

5. The device receives the call from the proxy server.
6. As the 'Test Call ID' parameter is set to "101", the device identifies the incoming call as a test call and sends it directly to the simulated test endpoint "101".

➤ **To configure SBC call testing:**

1. Configure the test call parameters:
 - a. Open the Test Call Settings page (**Configuration** tab > **System** menu > **Test Call** > **Test Call Settings**).

Figure 40-5: Test Call Settings Page

Test Call ID	<input type="text"/>
SBC Test ID	<input type="text"/>

- b. In the 'Test Call ID' field, enter a prefix number for the simulated test endpoint on the device.
 - c. In the 'SBC Test ID' field, enter a called prefix number for identifying the call as an SBC test call.
 - d. Click **Submit** to apply your settings.
2. Configure regular SBC call processing rules for called number prefix "101", such as classification and IP-to-IP routing through a proxy server.



Notes:

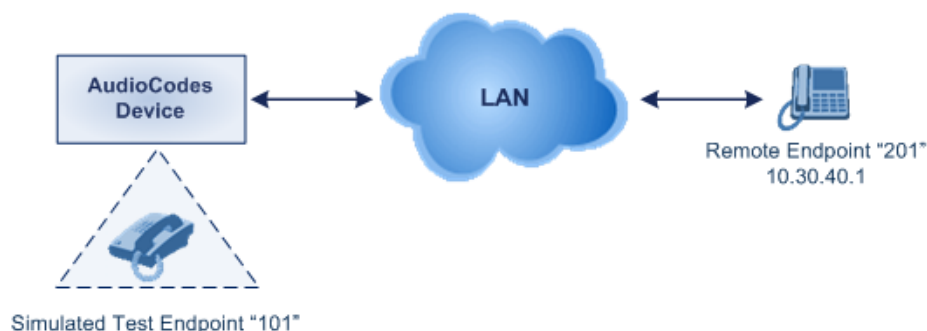
- For a full description of this parameter, see 'SIP Test Call Parameters' on page 452.
- This test call is initiated only upon receipt of incoming calls and with the configured prefix.
- This call test is done on all SIP interfaces.

40.4 Test Call Configuration Examples

Below are a few examples of test call configurations.

- **Single Test Call Scenario:** This example describes the configuration of a simple test call scenario that includes a single test call between a simulated test endpoint on the device and a remote endpoint.

Figure 40-6: Single Test Call Example

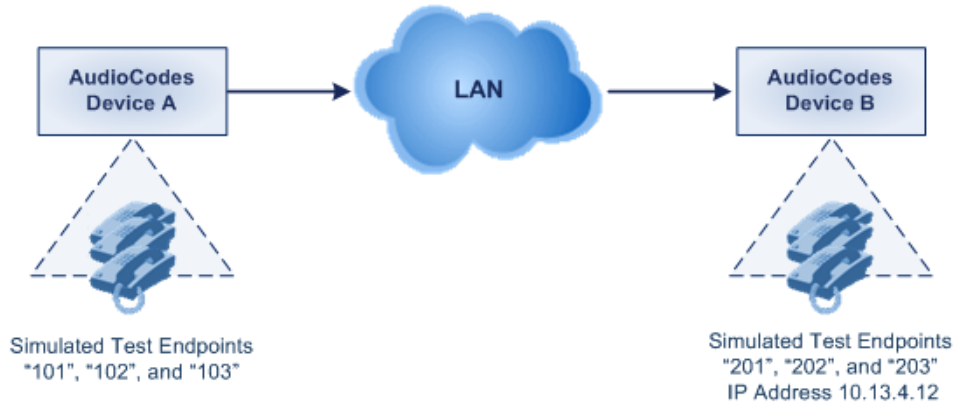


- Test Call table configuration:
 - ◆ Endpoint URI: "101"
 - ◆ Called URI: "201"
 - ◆ Route By: Dest Address

- ◆ Destination Address: "10.30.40.01"
- ◆ Call Party: Caller
- ◆ Test Mode: Once (default)

- **Batch Test Call Scenario:** This example describes the configuration of a batch test call setup for scheduled and continuous call testing of multiple endpoints. The test call is done between two AudioCodes devices - Device A and Device B - with simulated test endpoints. This eliminates the need for phone users, who would otherwise need to answer and end calls many times for batch testing. The calls are initiated from Device A, where Device B serves as the remote answering endpoint.

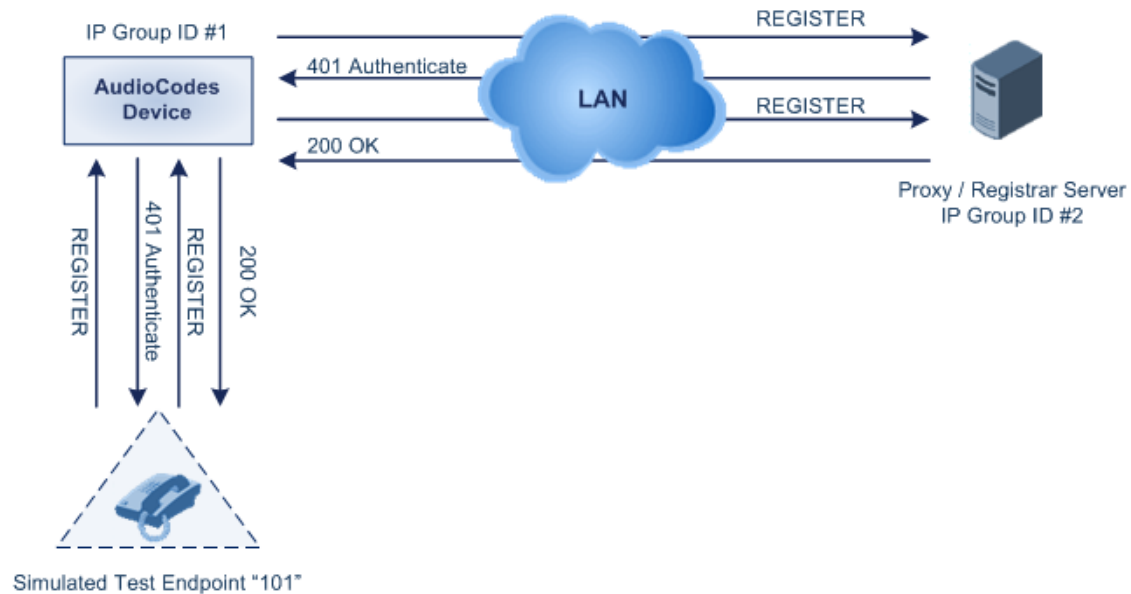
Figure 40-7: Batch Test Call Example



- Test Call table configuration at Device A:
 - ◆ Endpoint URI: "101"
 - ◆ Called URI: "201"
 - ◆ Route By: Dest Address
 - ◆ Destination Address: "10.13.4.12"
 - ◆ Call Party: Caller
 - ◆ Maximum Channels for Session: "3" (this setting configures three endpoints - "101", "102" and "103")
 - ◆ Call Duration: "5" (seconds)
 - ◆ Calls per Sec: "1"
 - ◆ Test Mode: Continuous
 - ◆ Test Duration: "3" (minutes)
 - ◆ Schedule Interval: "180" (minutes)
- Test Call table configuration at Device B:
 - ◆ Endpoint URI: "201"
 - ◆ Call Party: Caller
 - ◆ Maximum Channels for Session: "3" (this setting configures three endpoints - "201", "202" and "203")

- **Registration Test Call Scenario:** This example describes the configuration for testing the registration and authentication (i.e., username and password) process of a simulated test endpoint on the device with an external proxy/registrar server. This is useful, for example, for verifying that endpoints located in the LAN can register with an external proxy and subsequently, communicate with one another.

Figure 40-8: Test Call Registration Example



This example assumes that you have configured your device for communication between LAN phone users such as IP Groups to represent the device (10.13.4.12) and the proxy server, and IP-to-IP routing rules to route calls between these IP Groups.

- Test Call table configuration:
 - ◆ Endpoint URI: "101"
 - ◆ Called URI: "itsp"
 - ◆ Route By: Dest Address
 - ◆ Destination Address: "10.13.4.12" (this is the IP address of the device itself)
 - ◆ Auto Register: Enable
 - ◆ User Name: "testuser"
 - ◆ Password: "12345"
 - ◆ Call Party: Caller

Part XII

Appendix

41 Dialing Plan Notation for Routing and Manipulation

The device supports flexible dialing plan notations for denoting the prefix and/or suffix source and/or destination numbers and SIP URI user names in the routing and manipulation tables.

Table 41-1: Dialing Plan Notations for Prefixes and Suffixes

Notation	Description
x (letter "x")	Denotes any single digit.
# (pound symbol)	<ul style="list-style-type: none"> When used at the end of a prefix, it denotes the end of a number. For example, 54324xx# represents a 7-digit number that starts with the digits 54324. When used anywhere in the suffix, it is part of the number. For example, (3#45) can represent the number string, 123#45.
* (asterisk symbol)	<ul style="list-style-type: none"> When used in the prefix, it denotes any number. When used in the suffix, it is part of the number. For example, (3*45) can represent the number string, 123*45.
\$ (dollar sign)	<p>Denotes an empty prefix for incoming IP calls that do not have a user part in the Request-URI, or for incoming Tel calls that do not have a called or calling number. This is used for the following matching criteria:</p> <ul style="list-style-type: none"> Source and Destination Phone Prefix Source and Destination Username Source and Destination Calling Name Prefix
Range of Digits Notes: <ul style="list-style-type: none"> Dial plans denoting a prefix that is a range must be enclosed in square brackets, e.g., [4-8] or 23xx[456]. Dial plans denoting a prefix that is not a range is not enclosed, e.g., 12345#. Dial plans denoting a suffix must be enclosed in parenthesis, e.g., (4) and (4-8). Dial plans denoting a suffix that include multiple ranges, the range must be enclosed in square brackets, e.g., (23xx[4,5,6]). An example for entering a combined prefix and suffix dial plan - assume you want to match a rule whose destination phone prefix is 4 to 8, and suffix is 234, 235, or 236. The entered value would be the following: [4-8](23[4,5,6]). 	
[n-m] or (n-m)	<p>Represents a range of numbers, for example:</p> <ul style="list-style-type: none"> To depict numbers from 5551200 to 5551300: <ul style="list-style-type: none"> ✓ Prefix: [5551200-5551300]# ✓ Suffix: (5551200-5551300) To depict numbers from 123100 to 123200: <ul style="list-style-type: none"> ✓ Prefix: 123[100-200] ✓ Suffix: (123[100-200]) To depict prefix and suffix numbers together: <ul style="list-style-type: none"> ✓ 03(100): for any number that starts with 03 and ends with 100. ✓ [100-199](100,101,105): for a number that starts with 100 to 199 and ends with 100, 101 or 105. ✓ 03(abc): for any number that starts with 03 and ends with abc. ✓ 03(5xx): for any number that starts with 03 and ends with 5xx. ✓ 03(400,401,405): for any number that starts with 03 and ends with 400 or 401 or 405.

Notation	Description
	Notes: <ul style="list-style-type: none"> The value n must be less than the value m. Only numerical ranges are supported (not alphabetical letters). For suffix ranges, the starting (n) and ending (m) numbers in the range must have the same number of digits. For example, (23-34) is correct, but (3-12) is not.
[n,m,\dots] or (n,m,\dots)	<p>Represents multiple numbers. For example, to depict a one-digit number starting with 2, 3, 4, 5, or 6:</p> <ul style="list-style-type: none"> Prefix: [2,3,4,5,6]# Suffix: (2,3,4,5,6) Prefix with Suffix: [2,3,4,5,6](8,7,6) - prefix is denoted in square brackets; suffix in parenthesis <p>For prefix only, the notations $d[n,m]e$ and $d[n-m]e$ can also be used:</p> <ul style="list-style-type: none"> To depict a five-digit number that starts with 11, 22, or 33: [11,22,33]xxx# To depict a six-digit number that starts with 111 or 222: [111,222]xxx# <p>Note: Up to three digits can be used to denote each number.</p>
[$n1-m1,n2-m2,a,b,c,n3-m3$] or ($n1-m1,n2-m2,a,b,c,n3-m3$)	<p>Represents a mixed notation of single numbers and multiple ranges. For example, to depict numbers 123 to 130, 455, 766, and 780 to 790:</p> <ul style="list-style-type: none"> Prefix: [123-130,455,766,780-790] Suffix: (123-130,455,766,780-790) <p>Note: The ranges and the single numbers used in the dial plan must have the same number of digits. For example, each number range and single number in the dialing plan example above consists of three digits.</p>



Note: When configuring phone numbers or prefixes in the Web interface, enter them only as digits without any other characters. For example, if you wish to enter the phone number 555-1212, it must be entered as 5551212 without the hyphen (-). If the hyphen is entered, the entry is invalid.

42 Configuration Parameters Reference

The device's configuration parameters, default values, and their descriptions are documented in this section.



Notes: Parameters and values enclosed in square brackets [...] represent the *ini* file parameters and their enumeration values.

42.1 Networking Parameters

This subsection describes the device's networking parameters.

42.1.1 Ethernet Parameters

The Ethernet parameters are described in the table below.

Table 42-1: Ethernet Parameters

Parameter	Description
Physical Ports Settings Table	
Web: Physical Ports Settings [PhysicalPortsTable]	<p>This table parameter configures the physical Ethernet ports</p> <p>The format of this parameter is as follows:</p> <p>[PhysicalPortsTable] FORMAT PhysicalPortsTable_Index = PhysicalPortsTable_Port, PhysicalPortsTable_Mode, PhysicalPortsTable_NativeVlan, PhysicalPortsTable_SpeedDuplex, PhysicalPortsTable_PortDescription, PhysicalPortsTable_GroupMember, PhysicalPortsTable_GroupStatus; [\PhysicalPortsTable]</p> <p>For example:</p> <p>PhysicalPortsTable 0 = GE_4_1, 1, 1, 4, "User Port #0", GROUP_1, Active; PhysicalPortsTable 1 = GE_4_2, 1, 1, 4, "User Port #1", GROUP_1, Redundant;</p> <p>Note: For a description of this parameter, see Configuring Physical Ethernet Ports on page 95.</p>
Ethernet Group Settings Table	
Web: Ethernet Group Settings [EtherGroupTable]	<p>Defines the transmit (Tx) and receive (Rx) settings for the Ethernet port groups. The format of this parameter is as follows:</p> <p>[EtherGroupTable] FORMAT EtherGroupTable_Index = EtherGroupTable_Group, EtherGroupTable_Mode, EtherGroupTable_Member1, EtherGroupTable_Member2; [\EtherGroupTable]</p> <p>Notes:</p> <ul style="list-style-type: none"> For this parameter to take effect, a device reset is required. For a description of this parameter, see Configuring Tx/Rx Ethernet Port-Pair Groups on page 97.

42.1.2 Multiple VoIP Network Interfaces and VLAN Parameters

The IP network interfaces and VLAN parameters are described in the table below.

Table 42-2: IP Network Interfaces and VLAN Parameters

Parameter	Description
Multiple Interface Table	
Web: Multiple Interface Table EMS: IP Interface Settings [InterfaceTable]	<p>This table parameter configures the Multiple Interface table. The format of this parameter is as follows:</p> <p>[InterfaceTable] FORMAT InterfaceTable_Index = InterfaceTable_ApplicationTypes, InterfaceTable_InterfaceMode, InterfaceTable_IPAddress, InterfaceTable_PrefixLength, InterfaceTable_Gateway, InterfaceTable_VlanID, InterfaceTable_InterfaceName, InterfaceTable_PrimaryDNSServerIPAddress, InterfaceTable_SecondaryDNSServerIPAddress, InterfaceTable_UnderlyingInterface; [InterfaceTable]</p> <p>For example: InterfaceTable 0 = 0, 0, 192.168.85.14, 16, 0.0.0.0, 1, Management; InterfaceTable 1 = 2, 0, 200.200.85.14, 24, 0.0.0.0, 200, Control; InterfaceTable 2 = 1, 0, 211.211.85.14, 24, 211.211.85.1, 211, Media;</p> <p>Notes:</p> <ul style="list-style-type: none"> For this parameter to take effect, a device reset is required. For a description of this parameter, see 'Configuring IP Network Interfaces' on page 98.
[EnableNTPasOAM]	<p>Defines the application type for Network Time Protocol (NTP) services.</p> <ul style="list-style-type: none"> [1] = OAMP (default) [0] = Control <p>Note: For this parameter to take effect, a device reset is required.</p>

42.1.3 Routing Parameters

The IP network routing parameters are described in the table below.

Table 42-3: IP Network Routing Parameters

Parameter	Description
Web: Disable ICMP Redirects [DisableICMPRedirects]	<p>Determines whether the device accepts or ignores ICMP Redirect messages.</p> <ul style="list-style-type: none"> [0] Disable = (Default) ICMP Redirect messages are handled by the device. [1] Enable = ICMP Redirect messages are ignored.
Static IP Routing Table	
Web/EMS: IP Routing Table [StaticRouteTable]	<p>Defines up to 30 static IP routing rules for the device. These rules can be associated with IP interfaces defined in the Multiple Interface table (InterfaceTable parameter). The routing decision for sending the outgoing IP packet is based on the source subnet/VLAN. If not associated with an IP interface, the static IP rule is based on destination IP address.</p> <p>When the destination of an outgoing IP packet does not match one of the</p>

Parameter	Description
	<p>subnets defined in the Multiple Interface table, the device searches this table for an entry that matches the requested destination host/network. If such an entry is found, the device sends the packet to the indicated router (i.e., next hop). If no explicit entry is found, the packet is sent to the default gateway according to the source interface of the packet (if defined).</p> <p>The format of this parameter is as follows:</p> <p>[StaticRouteTable]</p> <p>FORMAT StaticRouteTable_Index = StaticRouteTable_InterfaceName, StaticRouteTable_Destination, StaticRouteTable_PrefixLength, StaticRouteTable_Gateway, StaticRouteTable_Description;</p> <p>[\StaticRouteTable]</p> <p>Note: For a description of this parameter, see 'Configuring Static IP Routing' on page 108.</p>

42.1.4 Quality of Service Parameters

The Quality of Service (QoS) parameters are described in the table below.

Table 42-4: QoS Parameters

Parameter	Description
Layer-2 Class Of Service (CoS) Parameters (VLAN Tag Priority Field)	
Web: DiffServ Table EMS: QoS Settings – DSCP to QoS Mapping [DiffServToVlanPriority]	<p>This table parameter configures DiffServ-to-VLAN Priority mapping. For each packet sent to the LAN, the VLAN Priority of the packet is set according to the DiffServ value in the IP header of the packet. The format of this ini file is as follows:</p> <p>[DiffServToVlanPriority]</p> <p>FORMAT DiffServToVlanPriority_Index = DiffServToVlanPriority_DiffServ, DiffServToVlanPriority_VlanPriority;</p> <p>[\DiffServToVlanPriority]</p> <p>For example:</p> <p>DiffServToVlanPriority 0 = 46, 6; DiffServToVlanPriority 1 = 40, 6; DiffServToVlanPriority 2 = 26, 4; DiffServToVlanPriority 3 = 10, 2;</p> <p>Notes:</p> <ul style="list-style-type: none"> For this parameter to take effect, a device reset is required. For a description of this table, see Configuring Quality of Service on page 111
Layer-3 Class of Service (TOS/DiffServ) Parameters	
Web: Media Premium QoS EMS: Premium Service Class Media Diff Serv [PremiumServiceClassMediaDiffServ]	<p>Defines the DiffServ value for Premium Media CoS content.</p> <p>The valid range is 0 to 63. The default is 46.</p> <p>Note: The value for the Premium Control DiffServ is determined by the following (according to priority):</p>

Parameter	Description
	<ul style="list-style-type: none"> IPDiffServ value in the selected IP Profile (IPProfile parameter). PremiumServiceClassMediaDiffServ.
Web: Control Premium QoS EMS: Premium Service Class Control Diff Serv [PremiumServiceClassControlDiffServ]	Defines the DiffServ value for Premium Control CoS content (Call Control applications). The valid range is 0 to 63. The default is 40. Notes: <ul style="list-style-type: none"> The value for the Premium Control DiffServ is determined by the following (according to priority): <ul style="list-style-type: none"> ✓ SigIPDiffServ value in the selected IP Profile (IPProfile parameter). ✓ PremiumServiceClassControlDiffServ.
Web: Gold QoS EMS: Gold Service Class Diff Serv [GoldServiceClassDiffServ]	Defines the DiffServ value for the Gold CoS content (Streaming applications). The valid range is 0 to 63. The default is 26.
Web: Bronze QoS EMS: Bronze Service Class Diff Serv [BronzeServiceClassDiffServ]	Defines the DiffServ value for the Bronze CoS content (OAMP applications). The valid range is 0 to 63. The default is 10.

42.1.5 NAT and STUN Parameters

The Network Address Translation (NAT) parameters are described in the table below.

Table 42-5: NAT Parameters

Parameter	Description
NAT Parameters	
Web/EMS: NAT Traversal [DisableNAT]	Enables the NAT mechanism. For more information, see 'First Incoming Packet Mechanism' on page 122 . <ul style="list-style-type: none"> [0] Enable [1] Disable (default)
Web: NAT IP Address EMS: Static NAT IP Address [StaticNatIP]	Defines the global (public) IP address of the device to enable static NAT between the device and the Internet. Note: For this parameter to take effect, a device reset is required.
Web/EMS: Inbound Media Latch Mode [InboundMediaLatchMode]	Enables the receipt of media streams whose IP address/port are not configured for the channel. <ul style="list-style-type: none"> [0] Strict = Accepts only the media stream configured for the channel. [1] Dynamic = (Default) Accepts any media stream.

42.1.6 NFS Parameters

The Network File Systems (NFS) configuration parameters are described in the table below.

Table 42-6: NFS Parameters

Parameter	Description
[NFSBasePort]	Defines the start of the range of numbers used for local UDP ports used by the NFS client. The maximum number of local ports is maximum channels plus maximum NFS servers. The valid range is 0 to 65535. The default is 47000.
NFS Table	
Web: NFS Table EMS: NFS Settings [NFSServers]	This table parameter defines up to 16 NFS file systems so that the device can access a remote server's shared files and directories for loading cmp, ini, and auxiliary files (using the Automatic Update mechanism). The format of this table ini file parameter is as follows: [NFSServers] FORMAT NFSServers_Index = NFSServers_HostOrIP, NFSServers_RootPath, NFSServers_NfsVersion, NFSServers_AuthType, NFSServers_UID, NFSServers_GID, NFSServers_VlanType; [NFSServers] For example: NFSServers 1 = 101.1.13, /audio1, 3, 1, 0, 1, 1; Note: For a detailed description of this table, see 'Configuring NFS Settings' on page 117.

42.1.7 DNS Parameters

The Domain name System (DNS) parameters are described in the table below.

Table 42-7: DNS Parameters

Parameter	Description
Internal DNS Table	
Web: Internal DNS Table EMS: DNS Information [DNS2IP]	This table parameter defines the internal DNS table for resolving host names into IP addresses. Up to four different IP addresses (in dotted-decimal notation) can be assigned to a host name. The format of this parameter is as follows: [Dns2Ip] FORMAT Dns2Ip_Index = Dns2Ip_DomainName, Dns2Ip_FirstIpAddress, Dns2Ip_SecondIpAddress, Dns2Ip_ThirdIpAddress, Dns2Ip_FourthIpAddress; [Dns2Ip] For example: Dns2Ip 0 = DnsName, 1.1.1.1, 2.2.2.2, 3.3.3.3, 4.4.4.4; Note: For a detailed description of this table parameter, see 'Configuring the Internal DNS Table' on page 114.

Parameter	Description
Internal SRV Table	
Web: Internal SRV Table EMS: DNS Information [SRV2IP]	<p>This table parameter defines the internal SRV table for resolving host names into DNS A-Records. Three different A-Records can be assigned to a host name. Each A-Record contains the host name, priority, weight, and port. The format of this parameter is as follows:</p> <p>[SRV2IP] FORMAT SRV2IP_Index = SRV2IP_InternalDomain, SRV2IP_TransportType, SRV2IP_Dns1, SRV2IP_Priority1, SRV2IP_Weight1, SRV2IP_Port1, SRV2IP_Dns2, SRV2IP_Priority2, SRV2IP_Weight2, SRV2IP_Port2, SRV2IP_Dns3, SRV2IP_Priority3, SRV2IP_Weight3, SRV2IP_Port3; [SRV2IP]</p> <p>For example: SRV2IP 0 = SrvDomain,0,Dnsname1,1,1,500,Dnsname2,2,2,501,\$\$,0,0,0;</p> <p>Note: For a detailed description of this table parameter, see 'Configuring the Internal SRV Table' on page 116.</p>

42.1.8 DHCP Parameters

The Dynamic Host Control Protocol (DHCP) parameters are described in the table below.

Table 42-8: DHCP Parameters

Parameter	Description
Web: Enable DHCP EMS: DHCP Enable [DHCPEnable]	<p>Enables Dynamic Host Control Protocol (DHCP) functionality.</p> <ul style="list-style-type: none"> ▪ [0] Disable (default) ▪ [1] Enable <p>Notes:</p> <ul style="list-style-type: none"> ▪ For this parameter to take effect, a device reset is required. ▪ After you enable the DHCP server, do the following: <ol style="list-style-type: none"> Enable DHCP and save the configuration. Perform a cold reset using the device's hardware reset button (soft reset using the Web interface doesn't trigger the DHCP procedure and this parameter reverts to 'Disable'). ▪ This parameter is a special 'Hidden' parameter. Once defined and saved in flash memory, its assigned value doesn't revert to its default even if the parameter doesn't appear in the <i>ini</i> file.
EMS: DHCP Speed Factor [DHCPSpeedFactor]	<p>Defines the DHCP renewal speed.</p> <ul style="list-style-type: none"> ▪ [0] = Disable ▪ [1] = (Default) Normal ▪ [2] to [10] = Fast <p>When set to 0, the DHCP lease renewal is disabled. Otherwise, the renewal time is divided by this factor. Some DHCP-enabled routers perform better when set to 4.</p> <p>Note: For this parameter to take effect, a device reset is required.</p>

42.1.9 NTP and Daylight Saving Time Parameters

The Network Time Protocol (NTP) and daylight saving time parameters are described in the table below.

Table 42-9: NTP and Daylight Saving Time Parameters

Parameter	Description
NTP Parameters	
Note: For more information on Network Time Protocol (NTP), see 'Simple Network Time Protocol Support' on page 91.	
Web: NTP Server DN/IP EMS: Server IP Address [NTPServerIP]	Defines the IP address (in dotted-decimal notation or as an FQDN) of the NTP server. The advantage of using an FQDN is that multiple IP addresses can be resolved from the DNS server, providing NTP server redundancy. The default IP address is 0.0.0.0 (i.e., internal NTP client is disabled).
Web: NTP Secondary Server IP [NTPSecondaryServerIP]	Defines a second NTP server's address as an FQDN or an IP address (in dotted-decimal notation). This NTP is used for redundancy; if the primary NTP server fails, then this NTP server is used. The default IP address is 0.0.0.0.
Web: NTP UTC Offset EMS: UTC Offset [NTPServerUTCOffset]	Defines the Universal Time Coordinate (UTC) offset (in seconds) from the NTP server. The default offset is 0. The offset range is -43200 to 43200.
Web: NTP Update Interval EMS: Update Interval [NTPUpdateInterval]	Defines the time interval (in seconds) that the NTP client requests for a time update. The default interval is 86400 (i.e., 24 hours). The range is 0 to 214783647. Note: It is not recommend to set this parameter to beyond one month (i.e., 2592000 seconds).
Daylight Saving Time Parameters	
Web: Day Light Saving Time EMS: Mode [DayLightSavingTimeEnable]	Enables daylight saving time. <ul style="list-style-type: none"> [0] Disable (default) [1] Enable
Web: Start Time or Day of Month Start EMS: Start [DayLightSavingTimeStart]	Defines the date and time when daylight saving begins. This value can be configured using any of the following formats: <ul style="list-style-type: none"> Day of year - <i>mm:dd:hh:mm</i>, where: <ul style="list-style-type: none"> ✓ <i>mm</i> denotes month ✓ <i>dd</i> denotes date of the month ✓ <i>hh</i> denotes hour ✓ <i>mm</i> denotes minutes For example, "05:01:08:00" denotes daylight saving starting from May 1 at 8 A.M. Day of month - <i>mm:day/wk:hh:mm</i>, where: <ul style="list-style-type: none"> ✓ <i>mm</i> denotes month (e.g., 04) ✓ <i>day</i> denotes day of week (e.g., FRI) ✓ <i>wk</i> denotes week of the month (e.g., 03) ✓ <i>hh</i> denotes hour (e.g., 23)

Parameter	Description
	<p>✓ <i>mm</i> denotes minutes (e.g., 10)</p> <p>For example, "04:FRI/03:23:00" denotes Friday, the third week of April, at 11 P.M. The week field can be 1-5, where 5 denotes the last occurrence of the specified day in the specified month. For example, "04:FRI/05:23:00" denotes the last Friday of April, at 11 P.M.</p>
Web: End Time or Day of Month End EMS: End [DayLightSavingTimeEnd]	Defines the date and time when daylight saving ends. For a description of the format of this value, see the DayLightSavingTimeStart parameter.
Web/EMS: Offset [DayLightSavingTimeOffset]	Defines the daylight saving time offset (in minutes). The valid range is 0 to 120. The default is 60.

42.2 Management Parameters

This subsection describes the device's Web and Telnet parameters.

42.2.1 General Parameters

The general management parameters are described in the table below.

Table 42-10: General Management Parameters

Parameter	Description
Web: Web and Telnet Access List Table EMS: Web Access Addresses [WebAccessList_x]	<p>This table configures up to ten IP addresses that are permitted to access the device's Web interface and Telnet interfaces. Access from an undefined IP address is denied. When no IP addresses are defined in this table, this security feature is inactive (i.e., the device can be accessed from any IP address).</p> <p>The default is 0.0.0.0 (i.e., the device can be accessed from any IP address).</p> <p>For example: WebAccessList_0 = 10.13.2.66 WebAccessList_1 = 10.13.77.7</p> <p>For a description of this parameter, see 'Configuring Web and Telnet Access List' on page 60.</p>

42.2.2 Web Parameters

The Web parameters are described in the table below.

Table 42-11: Web Parameters

Parameter	Description
Web: Enable web access from all interfaces [EnableWebAccessFromAllInterfaces]	<p>Enables Web access from any of the device's IP network interfaces (i.e., OAMP, Control, and/or Media). This feature applies to HTTP and HTTPS protocols.</p> <ul style="list-style-type: none"> ▪ [0] = (Default) Disable – Web access is only through the OAMP interface.

Parameter	Description
	<ul style="list-style-type: none"> [1] = Enable - Web access is through any network interface.
Web: Password Change Interval [WebUserPassChangeInterval]	<p>Defines the duration (in minutes) of the validity of Web login passwords. When this duration expires, the password of the Web user must be changed.</p> <p>The valid value is 0 to 100000, where 0 means that the password is always valid. The default is 1140.</p> <p>Note: This parameter is applicable only when using the Web Users table, where the default value of the 'Password Age' parameter in the Web Users table inherits this parameter's value.</p>
Web: User inactivity timer [UserInactivityTimer]	<p>Defines the duration (in days) for which a user has not logged in to the Web interface, after which the status of the user becomes inactive and can no longer access the Web interface. These users can only log in to the Web interface if their status is changed (to New or Valid) by a System Administrator or Master user.</p> <p>The valid value is 0 to 10000, where 0 means inactive. The default is 90.</p> <p>Note: This parameter is applicable only when using the Web Users table.</p>
Web: Session Timeout [WebSessionTimeout]	<p>Defines the duration (in minutes) of Web inactivity of a logged-in user, after which the user is automatically logged off the Web interface.</p> <p>The valid value is 0-100000, where 0 means no timeout. The default is 15.</p> <p>Note: This parameter can apply to all users, or per user when set in the Web Users table.</p>
Web: Deny Access On Fail Count [DenyAccessOnFailCount]	<p>Defines the maximum number of failed login attempts, after which the requesting IP address is blocked.</p> <p>The valid value range is 0 to 10. The values 0 and 1 mean immediate block. The default is 3.</p>
Web: Deny Authentication Timer EMS: WEB Deny Authentication Timer [DenyAuthenticationTimer]	<p>Defines the duration (in seconds) for which login to the Web interface is denied from a specific IP address (for all users) when the number of failed login attempts has exceeded the maximum. This maximum is defined by the DenyAccessOnFailCount parameter. Only after this time expires can users attempt to login from this same IP address.</p> <p>The valid value is 0 to 100000, where 0 means that login is not denied regardless of number of failed login attempts. The default is 60.</p>
Web: Display Login Information [DisplayLoginInformation]	<p>Enables display of user's login information on each successful login attempt.</p> <ul style="list-style-type: none"> [0] = Disable (default) [1] = Enable
[EnableMgmtTwoFactorAuthentication]	<p>Enables Web login authentication using a third-party, smart card.</p> <ul style="list-style-type: none"> [0] = Disable (default) [1] = Enable <p>When enabled, the device retrieves the Web user's login username from the smart card, which is automatically displayed (read-only) in</p>

Parameter	Description
	<p>the Web Login screen; the user is then required to provide only the login password.</p> <p>Typically, a TLS connection is established between the smart card and the device's Web interface, and a RADIUS server is implemented to authenticate the password with the username. Thus, this feature implements a two-factor authentication - what the user has (the physical card) and what the user knows (i.e., the login password).</p>
EMS: HTTPS Port [HTTPport]	<p>Defines the HTTP port for Web management (default is 80). To enable Web management from the LAN, configure the desired port.</p> <p>Note: For this parameter to take effect, a device reset is required.</p>
EMS: Disable WEB Config [DisableWebConfig]	<p>Determines whether the entire Web interface is read-only.</p> <ul style="list-style-type: none"> [0] = (Default) Enables modifications of parameters. [1] = Web interface is read-only. <p>When in read-only mode, parameters can't be modified. In addition, the following pages can't be accessed: 'Web User Accounts', 'Certificates', 'Regional Settings', 'Maintenance Actions' and all file-loading pages ('Load Auxiliary Files', 'Software Upgrade Wizard', and 'Configuration File').</p> <p>Note: For this parameter to take effect, a device reset is required.</p>
[ResetWebPassword]	<p>Resets the username and password of the primary ("Admin") and secondary ("User") accounts to their default settings ("Admin" and "Admin" respectively), and deletes all other users that may have been configured.</p> <ul style="list-style-type: none"> [0] = (Default) Password and username retain their values. [1] = Password and username are reset. <p>Notes:</p> <ul style="list-style-type: none"> For this parameter to take effect, a device reset is required. You cannot reset the username and password through the Web interface (by loading an ini file or on the AdminPage). To reset the username and password, use SNMP: <ul style="list-style-type: none"> a. Set acSysGenericINILine to WEBPasswordControlViaSNMP = 1, and reset the device with a flash burn (set acSysActionSetResetControl to 1 and acSysActionSetReset to 1). b. Change the username and password in the acSysWEBAccessEntry table. Use the following format: Username acSysWEBAccessUserName: old/pass/new Password acSysWEBAccessUserCode: username/old/new
[ScenarioFileName]	<p>Defines the file name of the Scenario file to be loaded to the device. The file name must have the .dat extension and can be up to 47 characters. For loading a Scenario using the Web interface, see Loading a Scenario to the Device on page 45.</p>
[WelcomeMessage]	<p>Enables and defines a Welcome message that appears on the Web Login page for logging in to the Web interface.</p> <p>The format of this parameter is as follows:</p> <p>[WelcomeMessage] FORMAT WelcomeMessage_Index = WelcomeMessage_Text [WelcomeMessage]</p> <p>For Example:</p>

Parameter	Description
	<p>FORMAT WelcomeMessage_Index = WelcomeMessage_Text WelcomeMessage 1 = "*****" ; WelcomeMessage 2 = "***** This is a Welcome message ***" ; WelcomeMessage 3 = "*****" ;</p> <p>Notes:</p> <ul style="list-style-type: none"> Each index row represents a line of text in the Welcome message box. Up to 20 lines (or rows) of text can be defined. The configured text message must be enclosed in double quotation marks (i.e., "..."). If this parameter is not configured, no Welcome message is displayed.
Web: HA Device Name [HAGUnitIdName]	<p>Defines a name for the device, which is displayed on the Home page to indicate the active device.</p> <p>The valid value is a string of up to 128 characters. For the default value, the device assigns either "Device 1" or "Device 2", so that active and redundant devices have different default names.</p>

42.2.3 Telnet Parameters

The Telnet parameters are described in the table below.

Table 42-12: Telnet Parameters

Parameter	Description
Web: Embedded Telnet Server EMS: Server Enable [TelnetServerEnable]	<p>Enables the device's embedded Telnet server. Telnet is disabled by default for security.</p> <ul style="list-style-type: none"> [0] Disable (default) [1] Enable Unsecured [2] Enable Secured (SSL) <p>Note: Only the primary Web User Account (which has Security Administration access level) can access the device using Telnet (see 'Configuring Web User Accounts' on page 51).</p>
Web: Telnet Server TCP Port EMS: Server Port [TelnetServerPort]	<p>Defines the port number for the embedded Telnet server.</p> <p>The valid range is all valid port numbers. The default port is 23.</p>
Web: Telnet Server Idle Timeout EMS: Server Idle Disconnect [TelnetServerIdleDisconnect]	<p>Defines the timeout (in minutes) for disconnection of an idle Telnet session. When set to zero, idle sessions are not disconnected.</p> <p>The valid range is any value. The default is 0.</p> <p>Note: For this parameter to take effect, a device reset is required.</p>

42.2.4 SNMP Parameters

The SNMP parameters are described in the table below.

Table 42-13: SNMP Parameters

Parameter	Description
Web: Enable SNMP [DisableSNMP]	Enables SNMP. <ul style="list-style-type: none"> [0] Enable = (Default) SNMP is enabled. [1] Disable = SNMP is disabled and no traps are sent.
[SNMPPort]	Defines the device's local (LAN) UDP port used for SNMP Get/Set commands. The range is 100 to 3999. The default port is 161. Note: For this parameter to take effect, a device reset is required.
EMS: Keep Alive Trap Port [KeepAliveTrapPort]	Defines the port to which keep-alive traps are sent. The valid range is 0 - 65534. The default is port 162.
[SendKeepAliveTrap]	Enables keep-alive traps and sends them every 9/10 of the time as defined by the NATBindingDefaultTimeout parameter. <ul style="list-style-type: none"> [0] = Disable [1] = Enable Note: For this parameter to take effect, a device reset is required.
[SNMPSysOid]	Defines the base product system OID. The default is eSNMP_AC_PRODUCT_BASE_OID_D. Note: For this parameter to take effect, a device reset is required.
[SNMPTrapEnterpriseOid]	Defines the Trap Enterprise OID. The default is eSNMP_AC_ENTERPRISE_OID. The inner shift of the trap in the AcTrap subtree is added to the end of the OID in this parameter. Note: For this parameter to take effect, a device reset is required.
[acUserInputAlarmDescription]	Defines the description of the input alarm.
[acUserInputAlarmSeverity]	Defines the severity of the input alarm.
[AlarmHistoryTableMaxSize]	Defines the maximum number of rows in the Alarm History table. This parameter can be controlled by the Config Global Entry Limit MIB (located in the Notification Log MIB). The valid range is 50 to . The default is . Note: For this parameter to take effect, a device reset is required.
[SNMPEngineIDString]	Defines the SNMP engine ID for SNMPv2/SNMPv3 agents. This is used for authenticating a user attempting to access the SNMP agent on the device. The ID can be a string of up to 36 characters. The default is 00:00:00:00:00:00:00:00:00:00:00:00 (12 Hex octets characters). The provided key must be set with 12 Hex values delimited by a colon (":") in the format xx:xx:...:xx. For example, 00:11:22:33:44:55:66:77:88:99:aa:bb Notes: <ul style="list-style-type: none"> For this parameter to take effect, a device reset is required. Before setting this parameter, all SNMPv3 users must be

Parameter	Description
	<p>deleted; otherwise, the parameter setting is ignored.</p> <ul style="list-style-type: none"> If the supplied key does not pass validation of the 12 Hex values input or it is set with the default value, the engine ID is generated according to RFC 3411.
Web: SNMP Trap Destination Parameters EMS: Network > SNMP Managers Table Note: Up to five SNMP trap managers can be defined.	
SNMP Manager [SNMPManagerIsUsed_x]	Determines the validity of the parameters (IP address and port number) of the corresponding SNMP Manager used to receive SNMP traps. <ul style="list-style-type: none"> [0] (Check box cleared) = Disabled (default) [1] (Check box selected) = Enabled
Web: IP Address EMS: Address [SNMPManagerTableIP_x]	Defines the IP address of the remote host used as an SNMP Manager. The device sends SNMP traps to this IP address. Enter the IP address in dotted-decimal notation, e.g., 108.10.1.255.
Web: Trap Port EMS: Port [SNMPManagerTrapPort_x]	Defines the port number of the remote SNMP Manager. The device sends SNMP traps to this port. The valid SNMP trap port range is 100 to 4000. The default port is 162.
Web: Trap Enable [SNMPManagerTrapSendingEnable_x]	Enables the sending of traps to the corresponding SNMP manager. <ul style="list-style-type: none"> [0] Disable = Sending is disabled. [1] Enable = (Default) Sending is enabled.
Web: Trap User [SNMPManagerTrapUser_x]	Defines the SNMPv3 USM user or SNMPv2 user to associate with the trap destination. This determines the trap format, authentication level, and encryption level. By default, it is associated with the SNMPv2 user (SNMP trap community string). The valid value is a string.
Web: Trap Manager Host Name [SNMPTrapManagerHostName]	Defines an FQDN of the remote host used as an SNMP manager. The resolved IP address replaces the last entry in the Trap Manager table (defined by the SNMPManagerTableIP parameter) and the last trap manager entry of snmpTargetAddrTable in the snmpTargetMIB. For example: 'mngtr.corp.mycompany.com'. The valid range is a string of up to 99 characters.
SNMP Community String Parameters	
Community String [SNMPReadOnlyCommunityString_x]	Defines up to five read-only SNMP community strings (up to 19 characters each). The default string is 'public'.
Community String [SNMPReadWriteCommunityString_x]	Defines up to five read/write SNMP community strings (up to 19 characters each). The default string is 'private'.

Parameter	Description
Trap Community String [SNMPTrapCommunityString]	Defines the Community string used in traps (up to 19 characters). The default string is 'trapuser'.
SNMP Trusted Managers Table	
Web: SNMP Trusted Managers [SNMPTrustedMgr_x]	<p>Defines up to five IP addresses of remote trusted SNMP managers from which the SNMP agent accepts and processes SNMP Get and Set requests.</p> <p>Notes:</p> <ul style="list-style-type: none"> By default, the SNMP agent accepts SNMP Get and Set requests from any IP address, as long as the correct community string is used in the request. Security can be enhanced by using Trusted Managers, which is an IP address from which the SNMP agent accepts and processes SNMP requests. If no values are assigned to these parameters any manager can access the device. Trusted managers can work with all community strings.
SNMP V3 Users Table	
Web/EMS: SNMP V3 Users [SNMPUsers]	<p>This <i>parameter</i> table defines SNMP v3 users. The format of this parameter is as follows:</p> <p>[SNMPUsers] FORMAT SNMPUsers_Index = SNMPUsers_Username, SNMPUsers_AuthProtocol, SNMPUsers_PrivProtocol, SNMPUsers_AuthKey, SNMPUsers_PrivKey, SNMPUsers_Group; [SNMPUsers]</p> <p>For example: SNMPUsers 1 = v3admin1, 1, 0, myauthkey, -, 1; The example above configures user 'v3admin1' with security level authNoPriv(2), authentication protocol MD5, authentication text password 'myauthkey', and ReadWriteGroup2.</p> <p>Note: For a description of this table, see 'Configuring SNMP V3 Users' on page 74.</p>

42.2.5 Serial Parameters

The RS-232 serial parameters are described in the table below.

Table 42-14: Serial Parameters

Parameter	Description
[DisableRS232]	<p>Enables the device's RS-232 (serial) port.</p> <ul style="list-style-type: none"> [0] = Enabled [1] = (Default) Disabled <p>The RS-232 serial port can be used to change the networking parameters and view error/notification messages. For how to establish a serial communication with the device, refer to the <i>Installation Manual</i>.</p> <p>Note: For this parameter to take effect, a device reset is required.</p>
EMS: Baud Rate [SerialBaudRate]	<p>Defines the RS-232 baud rate.</p> <p>The valid values include the following: 1200, 2400, 9600, 14400, 19200, 38400, 57600, or 115200.</p>

Parameter	Description
	Note: For this parameter to take effect, a device reset is required.
EMS: Data [SerialData]	<p>Defines the RS-232 data bit.</p> <ul style="list-style-type: none"> ▪ [7] = 7-bit ▪ [8] = (Default) 8-bit <p>Note: For this parameter to take effect, a device reset is required.</p>
EMS: Parity [SerialParity]	<p>Defines the RS-232 polarity.</p> <ul style="list-style-type: none"> ▪ [0] = (Default) None ▪ [1] = Odd ▪ [2] = Even <p>Note: For this parameter to take effect, a device reset is required.</p>
EMS: Stop [SerialStop]	<p>Defines the RS-232 stop bit.</p> <ul style="list-style-type: none"> ▪ [1] = (Default) 1-bit (default) ▪ [2] = 2-bit <p>Note: For this parameter to take effect, a device reset is required.</p>
EMS: Flow Control [SerialFlowControl]	<p>Defines the RS-232 flow control.</p> <ul style="list-style-type: none"> ▪ [0] = (Default) None ▪ [1] = Hardware <p>Note: For this parameter to take effect, a device reset is required.</p>

42.3 Debugging and Diagnostics Parameters

This subsection describes the device's debugging and diagnostic parameters.

42.3.1 General Parameters

The general debugging and diagnostic parameters are described in the table below.

Table 42-15: General Debugging and Diagnostic Parameters

Parameter	Description
EMS: Enable Diagnostics [EnableDiagnostics]	<p>Determines the method for verifying correct functioning of the different hardware components on the device. On completion of the check and if the test fails, the device sends information on the test results of each hardware component to the Syslog server.</p> <ul style="list-style-type: none"> ▪ [0] = (Default) Rapid and Enhanced self-test mode. ▪ [1] = Detailed self-test mode (full test of DSPs, PCM, Switch, LAN, PHY and Flash). ▪ [2] = A quicker version of the Detailed self-test mode (full test of DSPs, PCM, Switch, LAN, PHY, but partial test of Flash). <p>Note: For this parameter to take effect, a device reset is required.</p>
Web: Delay After Reset [sec] [GWAppDelayTime]	<p>Defines the time interval (in seconds) that the device's operation is delayed after a reset.</p> <p>The valid range is 0 to 45. The default is 7 seconds.</p> <p>Note: This feature helps overcome connection problems caused by some LAN routers or IP configuration parameters' modifications by a DHCP server.</p>

Parameter	Description
[EnableAutoRAITransmitBER]	<p>Enables the device to send a remote alarm indication (RAI) when the bit error rate (BER) is greater than 0.001.</p> <ul style="list-style-type: none"> [0] Disable (default) [1] Enable

42.3.2 SIP Test Call Parameters

The SIP Signaling Test Call parameters are described in the table below.

Table 42-16: SIP Test Call Parameters

Parameter	Description
Web: Test Call DTMF String [TestCallDtmfString]	<p>Defines the DTMF tone that is played for answered test calls (incoming and outgoing).</p> <p>The DTMF string can be up to 15 strings. The default is "3212333". An empty string means that no DTMF is played.</p>
Web: Test Call ID [TestCallID]	<p>Defines the test call prefix number (<i>ID</i>) of the simulated phone on the device. Incoming calls received with this called prefix number are identified as test calls.</p> <p>This can be any string of up to 15 characters. By default, no number is defined.</p> <p>Note: This parameter is only for testing incoming calls destined to this prefix number.</p>
Web: SBC Test ID CLI: sbc-test-id [SBCtestID]	<p>Defines the SBC test call prefix (ID) for identifying SBC test calls that traverse the device to register with an external routing entity such as an IP PBX or proxy server.</p> <p>This parameter functions together with the TestCallID parameter, which defines the prefix of the simulated endpoint. Upon receiving an incoming call with this prefix, the device removes the prefix, enabling it to forward the test call to the external entity. Upon receiving the call from the external entity, the device identifies the call as a test call according to its prefix, defined by the TestCallID, and then sends the call to the simulated endpoint.</p> <p>For example, assume SBCtestID is set to 4 and TestCallID to 2. If a call is received with called destination 4200, the device removes the prefix 4 and routes the call to the IP PBX. When it receives the call from the IP PBX, it identifies the call as a test call (i.e., prefix 2) and therefore, sends it to the simulated endpoint.</p> <p>The valid value can be any string of up to 15 characters. By default, no number is defined.</p> <p>Note: This feature is applicable only to the SBC application.</p>
Test Call Table	
Web: Test Call Table [Test_Call]	<p>Defines the local and remote endpoints to be tested.</p> <p>[Test_Call]</p> <p>FORMAT Test_Call_Index = Test_Call_EndpointURI, Test_Call_CalledURI, Test_Call_RouteBy, Test_Call_IPGroupID, Test_Call_DestAddress, Test_Call_DestTransportType, Test_Call_SRD, Test_Call_ApplicationType, Test_Call_AutoRegister, Test_Call_UserName, Test_Call_Password, Test_Call_CallParty, Test_Call_MaxChannels, Test_Call_CallDuration,</p>

Parameter	Description
	Test_Call_CallsPerSecond, Test_Call_TestMode, Test_Call_TestDuration, Test_Call_Play, Test_Call_ScheduleInterval; [Test_Call] Note: For a description of this table, see 'Configuring Test Calls' on page 423.

42.3.3 Syslog, CDR and Debug Parameters

The Syslog, CDR and debug parameters are described in the table below.

Table 42-17: Syslog, CDR and Debug Parameters

Parameter	Description
Web: Enable Syslog EMS: Syslog enable [EnableSyslog]	Determines whether the device sends logs and error messages (e.g., CDRs) generated by the device to a Syslog server. <ul style="list-style-type: none"> ▪ [0] Disable (default) ▪ [1] Enable Notes: <ul style="list-style-type: none"> ▪ If you enable Syslog, you must enter an IP address of the Syslog server (using the SyslogServerIP parameter). ▪ Syslog messages may increase the network traffic. ▪ To configure Syslog SIP message logging levels, use the GwDebugLevel parameter.
Web/EMS: Syslog Server IP Address [SyslogServerIP]	Defines the IP address (in dotted-decimal notation) of the computer on which the Syslog server is running. The Syslog server is an application designed to collect the logs and error messages generated by the device. The default IP address is 0.0.0.0.
Web: Syslog Server Port EMS: Syslog Server Port Number [SyslogServerPort]	Defines the UDP port of the Syslog server. The valid range is 0 to 65,535. The default port is 514.
[MaxBundleSyslogLength]	Defines the maximum size (in bytes) threshold of logged Syslog messages bundled into a single UDP packet, after which they are sent to a Syslog server. The valid value range is 0 to 1220 (where 0 indicates that no bundling occurs). The default is 1220. Note: This parameter is applicable only if the GwDebugLevel parameter is set to 7.
Web: CDR Server IP Address EMS: IP Address of CDR Server [CDRSyslogServerIP]	Defines the destination IP address to where CDR logs are sent. The default is a null string, which causes CDR messages to be sent with all Syslog messages to the Syslog server. Notes: <ul style="list-style-type: none"> ▪ The CDR messages are sent to UDP port 514 (default Syslog port). ▪ This mechanism is active only when Syslog is enabled (i.e., the parameter EnableSyslog is set to 1).
Web/EMS: CDR Report Level	Enables signaling-related CDRs to be sent to a Syslog server and

Parameter	Description
[CDRReportLevel]	<p>determines the call stage at which they are sent.</p> <ul style="list-style-type: none"> [0] None = (Default) CDRs are not used. [1] End Call = CDR is sent to the Syslog server at the end of each call. [2] Start & End Call = CDR report is sent to Syslog at the start and end of each call. [3] Connect & End Call = CDR report is sent to Syslog at connection and at the end of each call. [4] Start & End & Connect Call = CDR report is sent to Syslog at the start, at connection, and at the end of each call. <p>Notes:</p> <ul style="list-style-type: none"> The CDR Syslog message complies with RFC 3161 and is identified by: Facility = 17 (local1) and Severity = 6 (Informational). This mechanism is active only when Syslog is enabled (i.e., the parameter EnableSyslog is set to 1).
Web: Media CDR Report Level [MediaCDRReportLevel]	<p>Enables media-related CDRs of SBC calls to be sent to a Syslog server and determines the call stage at which they are sent.</p> <ul style="list-style-type: none"> [0] None = (Default) No media-related CDR is sent. [1] End Media = Sends a CDR only at the end of the call. [2] Start & End Media = Sends a CDR once the media starts. In some calls it may only be after the call is established, but in other calls the media may start at ringback tone. A CDR is also sent upon termination (end) of the media in the call. [3] Update & End Media = Sends a CDR when an update occurs in the media of the call. For example, a call starts and a ringback tone occurs, a re-INVITE is sent for a fax call and as a result, a CDR with the MediaReportType field set to "Update" is sent, as the media was changed from voice to T.38. A CDR is also sent upon termination (end) of the media in the call. [4] Start & End & Update Media = Sends a CDR at the start of the media, upon an update in the media (if occurs), and at the end of the media. <p>Note: To enable CDR generation as well as enable signaling-related CDRs, use the CDRReportLevel parameter.</p>
Web/EMS: Debug Level [GwDebugLevel]	<p>Defines the Syslog debug logging level.</p> <ul style="list-style-type: none"> [0] 0 = (Default) Debug is disabled. [1] 1 = Flow debugging is enabled. [5] 5 = Flow, device interface, stack interface, session manager, and device interface expanded debugging are enabled. [7] 7 = This option is recommended when the device is running under "heavy" traffic. In this mode: <ul style="list-style-type: none"> ✓ The Syslog debug level automatically changes between level 5, level 1, and level 0, depending on the device's CPU consumption so that VoIP traffic isn't affected. ✓ Syslog messages are bundled into a single UDP packet, after which they are sent to a Syslog server (bundling size is determined by the MaxBundleSyslogLength parameter). Bundling reduces the number of UDP Syslog packets, thereby improving CPU utilization. <p>Note that when this option is used, in order to read Syslog messages with Wireshark, a special plug-in (i.e., acsyslog.dll) must be used. Once the plug-in is installed, the Syslog messages are</p>

Parameter	Description
	<p>decoded as "AC SYSLOG" and are displayed using the 'acsyslog' filter instead of the regular 'syslog' filter.</p> <p>Notes:</p> <ul style="list-style-type: none"> This parameter is typically set to 5 if debug traces are required. However, in cases of heavy traffic, option 7 is recommended. Options 2, 3, 4, and 6 are not recommended.
Web: Syslog Facility Number EMS: SyslogFacility [SyslogFacility]	<p>Defines the Facility level (0 through 7) of the device's Syslog messages, according to RFC 3164. This allows you to identify Syslog messages generated by the device. This is useful, for example, if you collect the device's and other equipments' Syslog messages, at one single server. The device's Syslog messages can easily be identified and distinguished from other Syslog messages by its Facility level. Therefore, in addition to filtering Syslog messages according to IP address, the messages can be filtered according to Facility level.</p> <ul style="list-style-type: none"> [16] = (Default) local use 0 (local0) [17] = local use 1 (local1) [18] = local use 2 (local2) [19] = local use 3 (local3) [20] = local use 4 (local4) [21] = local use 5 (local5) [22] = local use 6 (local6) [23] = local use 7 (local7)
Web: Activity Types to Report via Activity Log Messages [ActivityListToLog]	<p>Defines the Activity Log mechanism of the device, which sends log messages to a Syslog server for reporting certain types of Web operations according to the below user-defined filters.</p> <ul style="list-style-type: none"> [pvc] Parameters Value Change = Changes made on-the-fly to parameters. Note that the <i>ini</i> file parameter, EnableParametersMonitoring can also be used to set this option, using values [0] (disable) or [1] (enable). [afl] Auxiliary Files Loading = Loading of auxiliary files. [dr] Device Reset = Reset of device via the 'Maintenance Actions' page. Note: For this option to take effect, a device reset is required. [fb] Flash Memory Burning = Burning of files or parameters to flash (in 'Maintenance Actions' page). [swu] Device Software Update = cmp file loading via the Software Upgrade Wizard. [ard] Access to Restricted Domains = Access to restricted domains, which include the following Web pages: <ul style="list-style-type: none"> ✓ (1) ini parameters (AdminPage) ✓ (2) General Security Settings ✓ (3) Configuration File ✓ (5) Software Upgrade Key Status ✓ (7) Web & Telnet Access List ✓ (8) WEB User Accounts [naa] Non-Authorized Access = Attempt to access the Web interface with a false or empty user name or password. [spc] Sensitive Parameters Value Change = Changes made to sensitive parameters: <ul style="list-style-type: none"> ✓ (1) IP Address ✓ (2) Subnet Mask

Parameter	Description
	<ul style="list-style-type: none"> ✓ (3) Default Gateway IP Address ✓ (4) ActivityListToLog ▪ [II] Login and Logout = Every login and logout attempt. <p>For example: ActivityListToLog = 'pvc', 'afl', 'dr', 'fb', 'swu', 'ard', 'naa', 'spc'</p> <p>Note: For the <i>ini</i> file, values must be enclosed in single quotation marks.</p>
Web: Debug Recording Destination IP [DebugRecordingDestIP]	Defines the IP address of the server for capturing debug recording.
Web: Debug Recording Destination Port [DebugRecordingDestPort]	Defines the UDP port of the server for capturing debug recording. The default is 925.
Debug Recording Status [DebugRecordingStatus]	<p>Activates or de-activates debug recording.</p> <ul style="list-style-type: none"> ▪ [0] Stop (default) ▪ [1] Start
Logging Filters Table	
Web: Logging Filters Table [LoggingFilters]	<p>This table parameter defines logging filtering rules for Syslog messages and debug recordings. The format of this parameter is as follows:</p> <p>[LoggingFilters] FORMAT LoggingFilters_Index = LoggingFilters_Type, LoggingFilters_Value, LoggingFilters_Syslog, LoggingFilters_CaptureType; [\LoggingFilters]</p> <p>Note: For a detailed description of this table, see 'Filtering Syslog Messages and Debug Recordings' on page 416.</p>

42.3.4 HA Ping Parameters

The High Availability (HA) Ping parameters are described in the table below.

HA Ping Parameters

Parameter	Description
Enable HA Network reachability [HAPingEnabled]	<p>Enables the pinging of an active IP network destination in HA mode to test reachability from one of the device's IP network interfaces. If no reply is received from a ping and the previous ping was successful, a switchover occurs to the redundant device.</p> <ul style="list-style-type: none"> ▪ [0] Disabled (default) ▪ [1] Enabled
HA Network reachability destination address [HAPingDestination]	<p>Defines the IP address of the destination that the device pings. The default is 0.0.0.0.</p>
HA Network reachability source If name [HAPingSourceIfName]	<p>Defines the device's IP network interface from where the ping is sent. The valid value is the name of the IP interface as configured in the 'Interface Name' field of the Multiple Interface table. By default, no IP network is defined.</p>

Parameter	Description
HA Network reachability ping timeout [HAPingTimeout]	Defines the timeout (in seconds) for which the ping request waits for a reply. The valid value is 1 to 60. The default is 1.
HA Network reachability ping retries [HAPingRetries]	Defines the number of ping requests that the device sends after no response is received from the destination, before the destination is declared unavailable. For example, if you specify 2, the destination is declared as down after three consecutive ping requests fail to evoke a response from the destination. The valid value is 0 to 100. The default 2.

42.3.5 Resource Allocation Indication Parameters

The Resource Allocation Indication (RAI) parameters are described in the table below.

Table 42-18: RAI Parameters

Parameter	Description
[EnableRAI]	Enables RAI alarm generation if the device's busy endpoints exceed a user-defined threshold. <ul style="list-style-type: none"> ▪ [0] = (Default) Disable RAI (Resource Available Indication) service. ▪ [1] = RAI service enabled and an SNMP 'acBoardCallResourcesAlarm' Alarm Trap is sent.
[RAIHighThreshold]	Defines the high threshold percentage of total calls that are active (busy endpoints). When the percentage of the device's busy endpoints exceeds this high threshold, the device sends the SNMP acBoardCallResourcesAlarm alarm trap with a 'major' alarm status. The range is 0 to 100. The default is 90. Note: The percentage of busy endpoints is calculated by dividing the number of busy endpoints by the total number of "enabled" endpoints.
[RAILowThreshold]	Defines the low threshold percentage of total calls that are active (busy endpoints). When the percentage of the device's busy endpoints falls below this low threshold, the device sends an SNMP acBoardCallResourcesAlarm alarm trap with a 'cleared' alarm status. The range is 0 to 100%. The default is 90%.
[RAILoopTime]	Defines the time interval (in seconds) that the device periodically checks call resource availability. The valid range is 1 to 200. The default is 10.

42.4 Security Parameters

This subsection describes the device's security parameters.

42.4.1 General Parameters

The general security parameters are described in the table below.

Table 42-19: General Security Parameters

Parameter	Description
Firewall Table	
Web/EMS: Internal Firewall Parameters [AccessList]	<p>This table parameter defines the device's access list (firewall), which defines network traffic filtering rules.</p> <p>The format of this parameter is as follows: [AccessList] FORMAT AccessList_Index = AccessList_Source_IP, AccessList_Source_Port, AccessList_PrefixLen, AccessList_Source_Port, AccessList_Start_Port, AccessList_End_Port, AccessList_Protocol, AccessList_Use_Specific_Interface, AccessList_Interface_ID, AccessList_Packet_Size, AccessList_Byte_Rate, AccessList_Byte_Burst, AccessList_Allow_Type; [AccessList]</p> <p>For example: AccessList 10 = mgmt.customer.com, , , 32, 0, 80, tcp, 1, OAMP, 0, 0, 0, allow; AccessList 22 = 10.4.0.0, , , 16, 4000, 9000, any, 0, , 0, 0, 0, block;</p> <p>In the example above, Rule #10 allows traffic from the host 'mgmt.customer.com' destined to TCP ports 0 to 80 on interface OAMP (OAMP). Rule #22 blocks traffic from the subnet 10.4.xxx.yyy destined to ports 4000 to 9000.</p> <p>Note: For a description of this table, see 'Configuring Firewall Settings' on page 125.</p>

42.4.2 HTTPS Parameters

The Secure Hypertext Transport Protocol (HTTPS) parameters are described in the table below.

Table 42-20: HTTPS Parameters

Parameter	Description
Web: Secured Web Connection (HTTPS) EMS: HTTPS Only [HTTPSOnly]	<p>Determines the protocol used to access the Web interface.</p> <ul style="list-style-type: none"> [0] HTTP and HTTPS (default). [1] HTTPS Only = Unencrypted HTTP packets are blocked. <p>Note: For this parameter to take effect, a device reset is required.</p>
EMS: HTTPS Port [HTTPSPort]	<p>Defines the local Secured HTTPS port of the device. This parameter allows secure remote device Web management from the LAN. To enable secure Web management from the LAN, configure the desired port.</p>

Parameter	Description
	<p>The valid range is 1 to 65535 (other restrictions may apply within this range). The default port is 443.</p> <p>Note: For this parameter to take effect, a device reset is required.</p>
Web/EMS: HTTPS Cipher String [HTTPSCipherString]	<p>Defines the Cipher string for HTTPS (in OpenSSL cipher list format). For the valid range values, refer to URL http://www.openssl.org/docs/apps/ciphers.html.</p> <p>The default is 'RC4:EXP' (Export encryption algorithms). For example, use 'ALL' for all ciphers suites (e.g., for ARIA encryption for TLS). The only ciphers available are RC4 and DES, and the cipher bit strength is limited to 56 bits.</p> <p>Notes:</p> <ul style="list-style-type: none"> For this parameter to take effect, a device reset is required. If the "Strong Encryption" Software License Key is enabled, the default of this parameter is changed to 'RC4:EXP', enabling RC-128bit encryption. The value 'ALL' can be configured only if the "Strong Encryption" Software License Key is enabled.
Web: HTTP Authentication Mode EMS: Web Authentication Mode [WebAuthMode]	<p>Determines the authentication mode used for the Web interface.</p> <ul style="list-style-type: none"> [0] Basic Mode = (Default) Basic authentication (clear text) is used. [1] Web Based Authentication = Digest authentication (MD5) is used. <p>Note: If you enable RADIUS login (i.e., the WebRADIUSLogin parameter is set to 1), you must set the WebAuthMode parameter to Basic Mode [0].</p>
Web: Requires Client Certificates for HTTPS connection [HTTPSRequireClientCertificate]	<p>Determines whether client certificates are required for HTTPS connection.</p> <ul style="list-style-type: none"> [0] Disable = (Default) Client certificates are not required. [1] Enable = Client certificates are required. The client certificate must be preloaded to the device and its matching private key must be installed on the managing PC. Time and date must be correctly set on the device for the client certificate to be verified. <p>Notes:</p> <ul style="list-style-type: none"> For this parameter to take effect, a device reset is required. For a description on implementing client certificates, see 'Client Certificates' on page 88.

42.4.3 SRTP Parameters

The Secure Real-Time Transport Protocol (SRTP) parameters are described in the table below.

Table 42-21: SRTP Parameters

Parameter	Description
Web: Media Security EMS: Enable Media Security	<p>Enables Secure Real-Time Transport Protocol (SRTP).</p> <ul style="list-style-type: none"> [0] Disable = (Default) SRTP is disabled.

Parameter	Description
[EnableMediaSecurity]	<ul style="list-style-type: none"> [1] Enable = SRTP is enabled. <p>Note: For this parameter to take effect, a device reset is required.</p>
Web/EMS: Media Security Behavior [MediaSecurityBehaviour]	<p>Determines the device's mode of operation when SRTP is used (i.e., when the parameter EnableMediaSecurity is set to 1).</p> <ul style="list-style-type: none"> [0] Preferable = (Default) The device initiates encrypted calls. However, if negotiation of the cipher suite fails, an unencrypted call is established. Incoming calls that don't include encryption information are accepted. [1] Mandatory = The device initiates encrypted calls, but if negotiation of the cipher suite fails, the call is terminated. Incoming calls that don't include encryption information are rejected. [2] Disable = The IP Profile for which this parameter is set does not support encrypted calls (i.e., SRTP). [3] Preferable - Single Media = The device sends SDP with a single media ('m=') line only (e.g., m=audio 6000 RTP/AVP 4 0 70 96) with RTP/AVP and crypto keys. The remote UA can respond with SRTP or RTP parameters: <ul style="list-style-type: none"> ✓ If the remote SIP UA does not support SRTP, it uses RTP and ignores the crypto lines. ✓ In the opposite direction, if the device receives an SDP offer with a single media (as shown above), it responds with SRTP (RTP/SAVP) if the EnableMediaSecurity parameter is set to 1. If SRTP is not supported (i.e., EnableMediaSecurity is set to 0), it responds with RTP. <p>Notes:</p> <ul style="list-style-type: none"> Before configuring this parameter, set the EnableMediaSecurity parameter to 1. If this parameter is set to Preferable [3] and two 'm=' lines are received in the SDP offer, the device prefers the SAVP (secure audio video profile) regardless of the order in the SDP. Option [2] Disable is applicable only to IP Profiles. This parameter can also be configured per IP Profile, using the IPProfile parameter (see 'Configuring IP Profiles' on page 221).
Web: Master Key Identifier (MKI) Size EMS: Packet MKI Size [SRTPTxPacketMKISize]	<p>Defines the size (in bytes) of the Master Key Identifier (MKI) in SRTP Tx packets.</p> <p>The range is 0 to 4. The default is 0 (i.e., new keys are generated without MKI).</p> <p>Notes:</p> <ul style="list-style-type: none"> You can also configure MKI size in an IP Profile. The device can forward MKI size as is for SRTP-to-SRTP flows or override the MKI size during negotiation, using IP Profiles. This can be done on the inbound or outbound leg.
Web:symmetric MKI Negotiation EMS: Enable Symmetric MKI [EnableSymmetricMKI]	<p>Enables symmetric MKI negotiation.</p> <ul style="list-style-type: none"> [0] Disable = (Default) The device includes the MKI in its 200 OK response according to the SRTPTxPacketMKISize parameter (if set to 0, then it is not included; if set to any other value, it is included with this value). [1] Enable = The answer crypto line contains (or excludes) an MKI value according to the selected crypto line in the offer. For example, assume that the device receives an INVITE containing the following two crypto lines in SDP:

Parameter	Description
	<pre>a=crypto:2 AES_CM_128_HMAC_SHA1_80 inline:TAaxNnQt8/qLQMnDuG4vxYfWl6K7eBK/ufk04pR4 2 ^31 1:1 a=crypto:3 AES_CM_128_HMAC_SHA1_80 inline:bnuYZnMxSfUiGitviWJZmzr7OF3AiRO0l5Vnh0kH 2 ^31</pre> <p>The first crypto line includes the MKI parameter "1:1". In the 200 OK response, the device selects one of the crypto lines (i.e., '2' or '3'). Typically, it selects the first line that supports the crypto suite. However, for SRTP-to-SRTP in SBC sessions, it can be determined by the remote side on the outgoing leg. If the device selects crypto line '2', it includes the MKI parameter in its answer SDP, for example:</p> <pre>a=crypto:2 AES_CM_128_HMAC_SHA1_80 inline:RlVyAlxV/qwBjkEklu4kSJyl3wCtYeZLq1/QFuxw 2 ^31 1:1</pre> <p>If the device selects a crypto line that does not contain the MKI parameter, then the MKI parameter is not included in the crypto line in the SDP answer (even if the SRTPTxPacketMKISize parameter is set to any value other than 0).</p> <p>Notes:</p> <ul style="list-style-type: none"> To enable symmetric MKI, the SRTPTxPacketMKISize parameter must be set to any value other than 0. You can also enable MKI negotiation per IP Profile.
Web/EMS: SRTP offered Suites [SRTPofferedSuites]	<p>Defines the offered crypto suites (cipher encryption algorithms) for SRTP.</p> <ul style="list-style-type: none"> [0] = (Default) All available crypto suites. [1] CIPHER SUITES AES CM 128 HMAC SHA1 80 = device uses AES-CM encryption with a 128-bit key and HMAC-SHA1 message authentication with a 80-bit tag. [2] CIPHER SUITES AES CM 128 HMAC SHA1 32 = device uses AES-CM encryption with a 128-bit key and HMAC-SHA1 message authentication with a 32-bit tag. <p>Note: This parameter also affects the selection of the crypto in the device's answer. For example, if the device receives an offer with two crypto lines containing HMAC_SHA1_80 and HMAC_SHA_32, it uses the HMAC_SHA_32 key in its SIP 200 OK response if the parameter is set to 2.</p>
Web: Disable Authentication On Transmitted RTP Packets EMS: RTP AuthenticationDisable Tx [RTPAuthenticationDisableTx]	<p>Enables authentication on transmitted RTP packets in a secured RTP session.</p> <ul style="list-style-type: none"> [0] Enable (default) [1] Disable
Web: Disable Encryption On Transmitted RTP Packets EMS: RTP EncryptionDisable Tx [RTPEncryptionDisableTx]	<p>Enables encryption on transmitted RTP packets in a secured RTP session.</p> <ul style="list-style-type: none"> [0] Enable (default) [1] Disable
Web: Disable Encryption On Transmitted RTCP Packets EMS: RTCP	<p>Enables encryption on transmitted RTCP packets in a secured RTP session.</p>

Parameter	Description
EncryptionDisable Tx [RTCPEncryptionDisableTx]	<ul style="list-style-type: none"> [0] Enable (default) [1] Disable
[ResetSRTPStateUponRekey]	<p>Enables synchronization of the SRTP state between the device and a server when a new SRTP key is generated upon a SIP session expire. This feature ensures that the roll-over counter (ROC), one of the parameters used in the SRTP encryption/decryption process of the SRTP packets, is synchronized on both sides for transmit and receive packets.</p> <ul style="list-style-type: none"> [0] = (Default) Disabled. ROC is not reset on the device side. [1] = Enabled. If the session expires causing a session refresh through a re-INVITE, the device or server generates a new key and the device resets the ROC index (and other SRTP fields) as done by the server, resulting in a synchronized SRTP. <p>Notes:</p> <ul style="list-style-type: none"> This feature can also be configured for an IP Profile. If this feature is disabled and the server resets the ROC upon a re-key generation, one-way voice may occur.

42.4.4 TLS Parameters

The Transport Layer Security (TLS) parameters are described in the table below.

Table 42-22: TLS Parameters

Parameter	Description
Web/EMS: TLS Version [TLSVersion]	<p>Determines the supported versions of SSL/TLS (Secure Socket Layer/Transport Layer Security).</p> <ul style="list-style-type: none"> [0] SSL 2.0-3.0 and TLS 1.0 = (Default) SSL 2.0, SSL 3.0, and TLS 1.0 are supported. [1] TLS 1.0 Only = only TLS 1.0 is used. <p>When set to 0, SSL/TLS handshakes always start with SSL 2.0 and switch to TLS 1.0 if both peers support it. When set to 1, TLS 1.0 is the only version supported; clients attempting to contact the device using SSL 2.0 are rejected.</p> <p>Note: For this parameter to take effect, a device reset is required.</p>
Web: TLS Client Re-Handshake Interval EMS: TLS Re Handshake Interval [TLSReHandshakeInterval]	<p>Defines the time interval (in minutes) between TLS Re-Handshakes initiated by the device.</p> <p>The interval range is 0 to 1,500 minutes. The default is 0 (i.e., no TLS Re-Handshake).</p>
Web: TLS Mutual Authentication EMS: SIPS Require Client Certificate [SIPSRequireClientCertificate]	<p>Determines the device's behavior when acting as a server for TLS connections.</p> <ul style="list-style-type: none"> [0] Disable = (Default) The device does not request the client certificate. [1] Enable = The device requires receipt and verification of the client certificate to establish the TLS connection. <p>Notes:</p> <ul style="list-style-type: none"> For this parameter to take effect, a device reset is required. The SIPS certificate files can be changed using the parameters HTTPSCertFileName and

Parameter	Description
	HTTPSRootFileName.
Web/EMS: Peer Host Name Verification Mode [PeerHostNameVerificationMode]	<p>Determines whether the device verifies the Subject Name of a remote certificate when establishing TLS connections.</p> <ul style="list-style-type: none"> ▪ [0] Disable (default). ▪ [1] Server Only = Verify Subject Name only when acting as a client for the TLS connection. ▪ [2] Server & Client = Verify Subject Name when acting as a server or client for the TLS connection. <p>When a remote certificate is received and this parameter is not disabled, the value of SubjectAltName is compared with the list of available Proxies. If a match is found for any of the configured Proxies, the TLS connection is established.</p> <p>The comparison is performed if the SubjectAltName is either a DNS name (DNSName) or an IP address. If no match is found and the SubjectAltName is marked as 'critical', the TLS connection is not established. If DNSName is used, the certificate can also use wildcards (*) to replace parts of the domain name.</p> <p>If the SubjectAltName is not marked as 'critical' and there is no match, the CN value of the SubjectName field is compared with the parameter TLSRemoteSubjectName. If a match is found, the connection is established. Otherwise, the connection is terminated.</p> <p>Note: If you set this parameter to [2] (Server & Client), for this functionality to operate, you also need to set the SIPSRequireClientCertificate parameter to [1] (Enable).</p>
Web: TLS Client Verify Server Certificate EMS: Verify Server Certificate [VerifyServerCertificate]	<p>Determines whether the device, when acting as a client for TLS connections, verifies the Server certificate. The certificate is verified with the Root CA information.</p> <ul style="list-style-type: none"> ▪ [0] Disable (default) ▪ [1] Enable <p>Note: If Subject Name verification is necessary, the parameter PeerHostNameVerificationMode must be used as well.</p>
Web: Strict Certificate Extension Validation [RequireStrictCert]	<p>Enables the validation of the extensions (keyUsage and extendedKeyUsage) of peer certificates. This validation ensures that the signing CA is authorized to sign certificates and that the end-entity certificate is authorized to negotiate a secure TLS connection.</p> <ul style="list-style-type: none"> ▪ [0] Disable (default) ▪ [1] Enable
Web/EMS: TLS Remote Subject Name [TLSRemoteSubjectName]	<p>Defines the Subject Name that is compared with the name defined in the remote side certificate when establishing TLS connections.</p> <p>If the SubjectAltName of the received certificate is not equal to any of the defined Proxies Host names/IP addresses and is not marked as 'critical', the Common Name (CN) of the Subject field is compared with this value. If not equal, the TLS connection is not established. If the CN uses a domain name, the certificate can also use wildcards (*) to replace parts of the domain name.</p>

Parameter	Description
	<p>The valid range is a string of up to 49 characters.</p> <p>Note: This parameter is applicable only if the parameter PeerHostNameVerificationMode is set to 1 or 2.</p>
Web: Client Cipher String [TLSCClientCipherString]	<p>Defines the cipher-suite string for TLS clients.</p> <p>The valid value is up to 255 strings. The default is "ALL:!ADH".</p> <p>For example: TLSCClientCipherString = 'EXP'</p> <p>This parameter complements the HTTPSCipherString parameter (which affects TLS servers). For possible values and additional details, refer to: http://www.openssl.org/docs/apps/ciphers.html</p>
[TLSPkeySize]	<p>Defines the key size (in bits) for RSA public-key encryption for newly self-signed generated keys for SSH.</p> <ul style="list-style-type: none"> ▪ [512] ▪ [768] ▪ [1024] (default) ▪ [2048]

42.4.5 SSH Parameters

Secure Shell (SSH) parameters are described in the table below.

Table 42-23: SSH Parameters

Parameter	Description
Web/EMS: Enable SSH Server [SSHServerEnable]	<p>Enables the device's embedded SSH server.</p> <ul style="list-style-type: none"> ▪ [0] Disable (default) ▪ [1] Enable
Web/EMS: Server Port [SSHServerPort]	<p>Defines the port number for the embedded SSH server.</p> <p>Range is any valid port number. The default port is 22.</p>
Web/EMS: SSH Admin Key [SSHAdminKey]	<p>Defines the RSA public key for strong authentication for logging in to the SSH interface (if enabled).</p> <p>The value should be a base64-encoded string. The value can be a maximum length of 511 characters.</p>
Web: Require Public Key EMS: EMS: SSH Require Public Key [SSHRequirePublicKey]	<p>Enables RSA public keys for SSH.</p> <ul style="list-style-type: none"> ▪ [0] = (Default) RSA public keys are optional if a value is configured for the parameter SSHAdminKey. ▪ [1] = RSA public keys are mandatory. <p>Note: To define the key size, use the TLSPkeySize parameter.</p>
Web: Max Payload Size EMS: SSH Max Payload Size [SSHMaxPayloadSize]	<p>Defines the maximum uncompressed payload size (in bytes) for SSH packets.</p> <p>The valid value is 550 to 32768. The default is 32768.</p>
Web: Max Binary Packet Size EMS: SSH Max Binary Packet Size [SSHMaxBinaryPacketSize]	<p>Defines the maximum packet size (in bytes) for SSH packets.</p> <p>The valid value is 582 to 35000. The default is 35000.</p>
EMS: Telnet SSH Max Sessions	<p>Defines the maximum number of simultaneous SSH sessions.</p>

Parameter	Description
[SSHMaxSessions]	The valid range is 1 to 2. The default is 2 sessions.
Web: Enable Last Login Message [SSHEnableLastLoginMessage]	Enables message display in SSH sessions of the time and date of the last SSH login. The SSH login message displays the number of unsuccessful login attempts since the last successful login. <ul style="list-style-type: none"> ▪ [0] Disable ▪ [1] Enable (default) Note: The last SSH login information is cleared when the device is reset.
Web: Max Login Attempts [SSHMaxLoginAttempts]	Defines the maximum SSH login attempts allowed for entering an incorrect password by an administrator before the SSH session is rejected. The valid range is 1 to 3. the default is 3.

42.4.6 OCSP Parameters

The Online Certificate Status Protocol (OCSP) parameters are described in the table below.

Table 42-24: OCSP Parameters

Parameter	Description
Web: Enable OCSP Server EMS: OCSP Enable [OCSPEnable]	Enables or disables certificate checking using OCSP. <ul style="list-style-type: none"> ▪ [0] Disable (default) ▪ [1] Enable
Web: Primary Server IP EMS: OCSP Server IP [OCSPServerIP]	Defines the IP address of the OCSP server. The default IP address is 0.0.0.0.
Web: Secondary Server IP [OCSPSecondaryServerIP]	Defines the IP address (in dotted-decimal notation) of the secondary OCSP server (optional). The default IP address is 0.0.0.0.
Web: Server Port EMS: OCSP Server Port [OCSPServerPort]	Defines the OCSP server's TCP port number. The default port number is 2560.
Web: Default Response When Server Unreachable EMS: OCSP Default Response [OCSPDefaultResponse]	Determines the default OCSP behavior when the server cannot be contacted. <ul style="list-style-type: none"> ▪ [0] Reject = (Default) Rejects peer certificate. ▪ [1] Allow = Allows peer certificate.

42.4.7 IDS Parameters

The Intrusion Detection System (IDS) parameters are described in the table below.

Table 42-25: IDS Parameters

Parameter	Description
-----------	-------------

Parameter	Description
Web: Intrusion Detection System (IDS) CLI: enable-ids [EnableIDS]	Enables the IDS feature. <ul style="list-style-type: none"> [0] Disable (default) [1] Enable Note: For this parameter to take effect, a device reset is required.
CLI: ids-clear-period [IDSArmClearPeriod]	Defines the interval (in seconds) after which an IDS alarm is cleared from the Active Alarms table if no thresholds are crossed during this time. However, this "quiet" period must be at least twice the Threshold Window value. For example, if IDSArmClearPeriod is set to 20 sec and the Threshold Window is set to 15 sec, the IDSArmClearPeriod parameter is ignored and the alarm is cleared only after 30 seconds (2 x 15 sec). The valid value is 0 to 86400. The default is 300.
IDS Policy Table	
Web: IDS Policy Table [IDSPolicy]	Defines IDS Policies. The format of the ini file parameter is: [IDSPolicy] FORMAT IDSPolicy_Index = IDSPolicy_Name, IDSPolicy_Description; [\IDSPolicy] For a detailed description of this table, see 'Configuring IDS Policies' on page 131 .
IDS Rule Table	
Web: IDS Rule Table [IDSRule]	Defines rules for the IDS Policies. The format of the ini file parameter is: [IDSRule] FORMAT IDSRule_Index = IDSRule_Policy, IDSRule_RuleID, IDSRule_Reason, IDSRule_ThresholdScope, IDSRule_ThresholdWindow, IDSRule_MinorAlarmThreshold, IDSRule_MajorAlarmThreshold, IDSRule_CriticalAlarmThreshold; [\IDSRule] For a detailed description of this table, see 'Configuring IDS Policies' on page 131 .
IDS Match Table	
Web: IDS Match Table [IDSMatch]	Defines target rules per IDS Policy. The format of the ini file parameter is: [IDSMatch] FORMAT IDSMatch_Index = IDSMatch_SIPInterface, IDSMatch_ProxySet, IDSMatch_Subnet, IDSMatch_Policy; [\IDSMatch] For a detailed description of this table, see 'Assigning IDS Policies' on page 134 .

42.5 RADIUS Parameters

The RADIUS parameters are described in the table below. For supported RADIUS attributes, see 'RADIUS Accounting CDR Attributes' on page 406.

Table 42-26: RADIUS Parameters

Parameter	Description
RADIUS Accounting Parameters	
Web: Enable RADIUS Access Control [EnableRADIUS]	<p>Enables the RADIUS application.</p> <ul style="list-style-type: none"> ▪ [0] Disable (Default) ▪ [1] Enable <p>Note: For this parameter to take effect, a device reset is required.</p>
Web: Accounting Server IP Address [RADIUSAccServerIP]	Defines the IP address of the RADIUS accounting server.
Web: Accounting Port [RADIUSAccPort]	<p>Defines the port of the RADIUS accounting server.</p> <p>The default is 1646.</p>
Web/EMS: RADIUS Accounting Type [RADIUSAccountingType]	<p>Determines when the RADIUS accounting messages are sent to the RADIUS accounting server.</p> <ul style="list-style-type: none"> ▪ [0] At Call Release = (Default) Sent at call release only. ▪ [1] At Connect & Release = Sent at call connect and release. ▪ [2] At Setup & Release = Sent at call setup and release.
Web: AAA Indications EMS: Indications [AAAIndications]	<p>Determines the Authentication, Authorization and Accounting (AAA) indications.</p> <ul style="list-style-type: none"> ▪ [0] None = (Default) No indications. ▪ [3] Accounting Only = Only accounting indications are used.
General RADIUS Parameters	
Web: Use RADIUS for Web/Telnet Login EMS: Web Use Radius Login [WebRADIUSLogin]	<p>Enables RADIUS queries for Web and Telnet login authentication. When enabled, logging into the device's Web and Telnet embedded servers is done through a RADIUS server. The device communicates with a user-defined RADIUS server and verifies the given username and password against a remote database, in a secure manner.</p> <ul style="list-style-type: none"> ▪ [0] Disable (default) ▪ [1] Enable <p>Notes:</p> <ul style="list-style-type: none"> ▪ For RADIUS login authentication to function, you also need to set the following parameters: <ul style="list-style-type: none"> ✓ EnableRADIUS = 1 (Enable) ✓ WebAuthMode = 0 (Basic Mode) ▪ RADIUS authentication requires HTTP basic authentication, where the username and password are transmitted in clear text over the network. Therefore, it's recommended to set the HTTPSONly parameter to 1 in order to force the use of HTTPS, since the transport is encrypted. ▪ If using RADIUS authentication to log into the CLI, only the primary Web User Account, which has Security Administration

Parameter	Description
	access level, can access the device's CLI (see 'Configuring Web User Accounts' on page 51).
Web: RADIUS Authentication Server IP Address EMS: RADIUS Auth Server IP [RADIUSAuthServerIP]	Defines the IP address of the RADIUS authentication server. Note: For this parameter to take effect, a device reset is required.
Web: RADIUS Authentication Server Port EMS: RADIUS Auth Server Port [RADIUSAuthPort]	Defines the port of the RADIUS Authentication Server. Note: For this parameter to take effect, a device reset is required.
Web: RADIUS Shared Secret EMS: RADIUS Auth Server Secret [SharedSecret]	Defines the 'Secret' used to authenticate the device to the RADIUS server. This should be a cryptically strong password.
RADIUS Authentication Parameters	
Web: Default Access Level [DefaultAccessLevel]	Defines the default access level for the device when the RADIUS (authentication) response doesn't include an access level attribute. The valid range is 0 to 255. The default is 200 (i.e., Security Administrator).
Web: Device Behavior Upon RADIUS Timeout [BehaviorUponRadiusTimeout]	Defines the device's response upon a RADIUS timeout. <ul style="list-style-type: none"> [0] Deny Access = Denies access. [1] Verify Access Locally = (Default) Checks password locally.
Web: Local RADIUS Password Cache Mode [RadiusLocalCacheMode]	Determines the device's mode of operation regarding the timer (configured by the parameter RadiusLocalCacheTimeout) that determines the validity of the user name and password (verified by the RADIUS server). <ul style="list-style-type: none"> [0] Absolute Expiry Timer = When you access a Web page, the timeout doesn't reset, instead it continues decreasing. [1] Reset Timer Upon Access = (Default) Upon each access to a Web page, the timeout always resets (reverts to the initial value configured by RadiusLocalCacheTimeout).
Web: Local RADIUS Password Cache Timeout [RadiusLocalCacheTimeout]	Defines the time (in seconds) the locally stored user name and password (verified by the RADIUS server) are valid. When this time expires, the user name and password become invalid and a must be re-verified with the RADIUS server. The valid range is 1 to 0xFFFFFFFF. The default is 300 (5 minutes). <ul style="list-style-type: none"> [-1] = Never expires. [0] = Each request requires RADIUS authentication.
Web: RADIUS VSA Vendor ID [RadiusVSAVendorID]	Defines the vendor ID that the device accepts when parsing a RADIUS response packet. The valid range is 0 to 0xFFFFFFFF. The default is 5003.
Web: RADIUS VSA Access Level Attribute [RadiusVSAAccessAttribute]	Defines the code that indicates the access level attribute in the Vendor Specific Attributes (VSA) section of the received RADIUS packet. The valid range is 0 to 255. The default is 35.

Parameter	Description
[MaxRADIUSSessions]	Defines the number of concurrent calls that can communicate with the RADIUS server (optional). The valid range is 0 to 240. The default is 240.
EMS: RADIUS Auth Number of Retries [RADIUSRetransmission]	Defines the number of retransmission retries. The valid range is 1 to 10. The default is 3.
[RadiusTO]	Defines the time interval (measured in seconds) that the device waits for a response before a RADIUS retransmission is issued. The valid range is 1 to 30. The default is 10.

42.6 SIP Media Realm Parameters

The Media Realm parameters are described in the table below.

Table 42-27: Media Realm Parameters

Parameter	Description
Media Realm Table	
Web: Media Realm Table EMS: VoIP > Media > Media Realm [CpMediaRealm]	<p>This table parameter defines the Media Realm table. The Media Realm table allows you to divide a Media-type interface (defined in the Multiple Interface table) into several realms, where each realm is specified by a UDP port range.</p> <p>The format of this parameter is as follows:</p> <p>[CpMediaRealm] FORMAT CpMediaRealm_Index = CpMediaRealm_MediaRealmName, CpMediaRealm_IPv4IF, CpMediaRealm_IPv6IF, CpMediaRealm_PortRangeStart, CpMediaRealm_MediaSessionLeg, CpMediaRealm_PortRangeEnd, CpMediaRealm_TransRateRatio, CpMediaRealm_IsDefault; [\CpMediaRealm]</p> <p>For example, CpMediaRealm 1 = Mrealm1, Voice, , 6600, 20, 6790, , 1; CpMediaRealm 2 = Mrealm2, Voice, , 6800, 10, 6890; , 0;</p> <p>Notes:</p> <ul style="list-style-type: none"> For this parameter to take effect, a device reset is required. For a detailed description of this table, see 'Configuring Media Realms' on page 157.
Bandwidth Management per Media Realm Table	
Web: Bandwidth Management [BWManagement]	<p>This table parameter defines bandwidth management rules per Media Realm.</p> <p>The format of this parameter is as follows:</p> <p>[BWManagement] FORMAT BWManagement_Index = BWManagement_MediaRealmIndex, BWManagement_ThresholdIndex, BWManagement_RuleAction, BWManagement_Threshold, BWManagement_Hysteresis;</p>

Parameter	Description
	<p>[\BWManagement]</p> <p>Where ThresholdIndex is the bandwidth threshold rule type:</p> <ul style="list-style-type: none"> [0] High Threshold Rule [1] Critical Threshold Rule <p>Notes:</p> <ul style="list-style-type: none"> For this parameter to take effect, a device reset is required. This table can include up to two row entries (where 0 is the first index). For a detailed description of this table, see 'Configuring Bandwidth Management per Media Realm' on page 162.
Quality of Experience Parameters	
Web: Server IP CLI: server-ip [QOEServerIP]	Defines the IP address of AudioCodes Session Experience Manager (SEM) server to where the quality experience reports are sent. Note: For this parameter to take effect, a device reset is required.
Web: Port [QOEPort]	Defines the port of the SEM server. The valid value range is 0 to 65534. The default is 5000.
Web: Interface Name [QOEInterfaceName]	Defines the IP network interface on which the quality experience reports are sent. The default is the OAMP interface. Note: For this parameter to take effect, a device reset is required.
Web: Connection Mode [QOEConnectionMode]	Defines the connection between the device and the SEM. <ul style="list-style-type: none"> [0] Server = The device receives connection from the server. [1] Client (default) = The device connects to the SEM. [2] None Note: Currently, only the client connection is supported.
Web: Information Level [QOEInformationLevel]	Defines the level (i.e., amount of detail) of voice quality information that is sent to the SEM server. <ul style="list-style-type: none"> [0] Standard (default) [1] Enhanced [2] Debug
Web: Use Mos LQ [QOEUseMosLQ]	Enables the reporting of the MOS-LQ (listening quality). If disabled, the MOS-CQ (conversational quality) is reported. MOS-LQ measures the quality of audio for listening purposes only. MOS-LQ does not take into account bi-directional effects such as delay and echo. MOS-CQ takes into account listening quality in both directions, as well as the bi-directional effects. <ul style="list-style-type: none"> [0] Disable (default) [1] Enable
Media Realm > Quality of Experience Table	
Web: Media Realm > Quality Of Experience EMS: Media > Media Realm > Voice Quality Rules [QOERules]	This table configures Quality of Experience parameters per Media Realm. [QOERules] ORMAT QOERules_Index = QOERules_MediaRealmIndex, QOERules_RuleIndex, QOERules_MonitoredParam, QOERules_Direction, QOERules_Profile, QOERules_GreenYellowThreshold, QOERules_GreenYellowHysteresis, QOERules_YellowRedThreshold, QOERules_YellowRedHysteresis,

Parameter	Description
	QOERules_GreenYellowOperation, QOERules_GreenYellowOperationDetails, QOERules_YellowRedOperation, QOERules_YellowRedOperationDetails; [\QOERules] Note: For a detailed description of this table, see Configuring Quality of Experience Parameters per Media Realm on page 159.

42.7 Control Network Parameters

42.7.1 IP Group, Proxy, Registration and Authentication Parameters

The proxy server, registration and authentication SIP parameters are described in the table below.

Table 42-28: Proxy, Registration and Authentication SIP Parameters

Parameter	Description
IP Group Table	
Web: IP Group Table EMS: Endpoints > IP Group [IPGroup]	This table configures IP Groups. The ini file format of this parameter is as follows: [IPGroup] FORMAT IPGroup_Index = IPGroup_Type, IPGroup_Description, IPGroup_ProxySetId, IPGroup_SIPGroupName, IPGroup_ContactUser, IPGroup_EnableSurvivability, IPGroup_ServingIPGroup, IPGroup_SipReRoutingMode, IPGroup_AlwaysUseRouteTable, IPGroup_RoutingMode, IPGroup_SRD, IPGroup_MediaRealm, IPGroup_ClassifyByProxySet, IPGroup_ProfileId, IPGroup_MaxNumOfRegUsers, IPGroup_InboundManSet, IPGroup_OutboundManSet, IPGroup_RegistrationMode, IPGroup_AuthenticationMode, IPGroup_MethodList, IPGroup_EnableSBCCClientForking, IPGroup_SourceUriInput, IPGroup_DestUriInput, IPGroup_ContactName; [/IPGroup] Notes: <ul style="list-style-type: none"> For this parameter to take effect, a device reset is required. For a description of this table, see 'Configuring IP Groups' on page 191.
Account Table	
Web: Account Table EMS: SIP Endpoints > Account [Account]	This table parameter configures the Account table for registering and/or authenticating (digest) IP Groups (e.g., an IP-PBX) to another IP Group (e.g., an Internet Telephony Service Provider - ITSP). The format of this parameter is as follows: [Account] FORMAT Account_Index = Account_ServedTrunkGroup, Account_ServedIPGroup, Account_ServingIPGroup, Account_Username, Account_Password, Account_HostName,

Parameter	Description
	Account_Register, Account_ContactUser, Account_ApplicationType; [Account] For example: Account 1 = 1, -1, 1, user, 1234, acl, 1, ITSP1; Note: For a detailed description of this table, see 'Configuring Account Table' on page 203.
Proxy Registration Parameters	
Web: Use Default Proxy EMS: Proxy Used [IsProxyUsed]	Enables the use of a SIP proxy server. <ul style="list-style-type: none"> [0] No = (Default) Proxy isn't used and instead, the internal routing table is used. [1] Yes = Proxy server is used. Define the IP address of the proxy server in the Proxy Sets table (see 'Configuring Proxy Sets Table' on page 197). Note: If you are not using a proxy server, you must define outbound IP call routing rules in the IP-to-IP Routing table.
Web/EMS: Proxy Name [ProxyName]	Defines the Home Proxy domain name. If specified, this name is used as the Request-URI in REGISTER, INVITE and other SIP messages, and as the host part of the To header in INVITE messages. If not specified, the Proxy IP address is used instead. The valid value is a string of up to 49 characters. Note: This parameter functions together with the UseProxyIPasHost parameter.
Web: Use Proxy IP as Host [UseProxyIPasHost]	Enables the use of the proxy server's IP address (in dotted-decimal notation) as the host name in SIP From and To headers in REGISTER requests. <ul style="list-style-type: none"> [0] Disable (default) [1] Enable If this parameter is disabled and the device registers to an IP Group (i.e., proxy server), it uses the string configured by the ProxyName parameter as the host name in the REGISTER's Request-URI and uses the string configured by the IP Group table parameter, SIPGroupName as the host name in the To and From headers. If the IP Group is configured with a Proxy Set that has multiple IP addresses, all the REGISTER messages sent to these proxies are sent with the same host name. Note: If this parameter is disabled and the ProxyName parameter is not configured, the proxy's IP address is used as the host name in the REGISTER Request-URI.
Web: Redundancy Mode EMS: Proxy Redundancy Mode [ProxyRedundancyMode]	Determines whether the device switches back to the primary Proxy after using a redundant Proxy. <ul style="list-style-type: none"> [0] Parking = (Default) The device continues working with a redundant (now active) Proxy until the next failure, after which it works with the next redundant Proxy. [1] Homing = The device always tries to work with the primary Proxy server (i.e., switches back to the primary Proxy whenever it's available). Note: To use this Proxy Redundancy mechanism, you need to enable the keep-alive with Proxy option, by setting the

Parameter	Description
	parameter EnableProxyKeepAlive to 1 or 2.
Web: Proxy IP List Refresh Time EMS: IP List Refresh Time [ProxyIPListRefreshTime]	<p>Defines the time interval (in seconds) between each Proxy IP list refresh.</p> <p>The range is 5 to 2,000,000. The default interval is 60.</p>
Web/EMS: Always Use Proxy [AlwaysSendToProxy]	<p>Determines whether the device sends SIP messages and responses through a Proxy server.</p> <ul style="list-style-type: none"> ▪ [0] Disable = (Default) Use standard SIP routing rules. ▪ [1] Enable = All SIP messages and responses are sent to the Proxy server. <p>Note: This parameter is applicable only if a Proxy server is used (i.e., the parameter IsProxyUsed is set to 1).</p>
Web/EMS: DNS Query Type [DNSQueryType]	<p>Enables the use of DNS Naming Authority Pointer (NAPTR) and Service Record (SRV) queries to resolve Proxy and Registrar servers and to resolve all domain names that appear in the SIP Contact and Record-Route headers.</p> <ul style="list-style-type: none"> ▪ [0] A-Record (default) ▪ [1] SRV ▪ [2] NAPTR <p>If set to A-Record [0], no NAPTR or SRV queries are performed.</p> <p>If set to SRV [1] and the Proxy/Registrar IP address parameter, Contact/Record-Route headers, or IP address defined in the Routing tables contain a domain name, an SRV query is performed. The device uses the first host name received from the SRV query. The device then performs a DNS A-record query for the host name to locate an IP address.</p> <p>If set to NAPTR [2], an NAPTR query is performed. If it is successful, an SRV query is sent according to the information received in the NAPTR response. If the NAPTR query fails, an SRV query is performed according to the configured transport type.</p> <p>If the Proxy/Registrar IP address parameter, the domain name in the Contact/Record-Route headers, or the IP address defined in the Routing tables contain a domain name with port definition, the device performs a regular DNS A-record query.</p> <p>If a specific Transport Type is defined, a NAPTR query is not performed.</p> <p>Note: To enable NAPTR/SRV queries for Proxy servers only, use the parameter ProxyDNSQueryType.</p>
Web: Proxy DNS Query Type [ProxyDNSQueryType]	<p>Enables the use of DNS Naming Authority Pointer (NAPTR) and Service Record (SRV) queries to discover Proxy servers.</p> <ul style="list-style-type: none"> ▪ [0] A-Record (default) ▪ [1] SRV ▪ [2] NAPTR <p>If set to A-Record [0], no NAPTR or SRV queries are performed.</p> <p>If set to SRV [1] and the Proxy IP address parameter contains a domain name without port definition (e.g., ProxyIP = domain.com), an SRV query is performed. The SRV query returns up to four Proxy host names and their weights. The</p>

Parameter	Description
	<p>device then performs DNS A-record queries for each Proxy host name (according to the received weights) to locate up to four Proxy IP addresses. Therefore, if the first SRV query returns two domain names and the A-record queries return two IP addresses each, no additional searches are performed.</p> <p>If set to NAPTR [2], an NAPTR query is performed. If it is successful, an SRV query is sent according to the information received in the NAPTR response. If the NAPTR query fails, an SRV query is performed according to the configured transport type.</p> <p>If the Proxy IP address parameter contains a domain name with port definition (e.g., ProxyIP = domain.com:5080), the device performs a regular DNS A-record query.</p> <p>If a specific Transport Type is defined, a NAPTR query is not performed.</p> <p>Note: When enabled, NAPTR/SRV queries are used to discover Proxy servers even if the parameter DNSQueryType is disabled.</p>
Web/EMS: Password [Password]	<p>Defines the password for Basic/Digest authentication with a Proxy/Registrar server. A single password is used for all device ports.</p> <p>The default is 'Default_Passwd'.</p>
Web/EMS: Cnonce [Cnonce]	<p>Defines the Cnonce string used by the SIP server and client to provide mutual authentication.</p> <p>The value is free format, i.e., 'Cnonce = 0a4f113b'. The default is 'Default_Cnonce'.</p>
Web/EMS: Mutual Authentication Mode [MutualAuthenticationMode]	<p>Determines the device's mode of operation when Authentication and Key Agreement (AKA) Digest Authentication is used.</p> <ul style="list-style-type: none"> [0] Optional = (Default) Incoming requests that don't include AKA authentication information are accepted. [1] Mandatory = Incoming requests that don't include AKA authentication information are rejected.
Proxy IP Table	
Web: Proxy IP Table EMS: Proxy IP [ProxyIP]	<p>This table parameter configures the Proxy Set table with Proxy Set IDs, each with up to five Proxy server IP addresses (or fully qualified domain name/FQDN). Each Proxy Set can be defined with a transport type (UDP, TCP, or TLS). The format of this parameter is as follows:</p> <p>[ProxyIP] FORMAT ProxyIp_Index = ProxyIp_IpAddress, ProxyIp_TransportType, ProxyIp_ProxySetId; [ProxyIP]</p> <p>For example: ProxyIp 0 = 10.33.37.77, -1, 0; ProxyIp 1 = 10.8.8.10, 0, 2; ProxyIp 2 = 10.5.6.7, -1, 1;</p> <p>Notes:</p> <ul style="list-style-type: none"> To assign various attributes (such as Proxy Load Balancing) per Proxy Set ID, use the parameter ProxySet. For a description of this table, see 'Configuring Proxy Sets

Parameter	Description
	Table' on page 197.
Proxy Set Table	
Web: Proxy Set Table EMS: Proxy Set [ProxySet]	<p>This table parameter configures the Proxy Set ID table. It is used in conjunction with the ProxyIP table ini file parameter, which defines the IP addresses per Proxy Set ID.</p> <p>The ProxySet table ini file parameter defines additional attributes per Proxy Set ID. This includes, for example, Proxy keep-alive and load balancing and redundancy mechanisms (if a Proxy Set contains more than one proxy address).</p> <p>The format of this parameter is as follows: [ProxySet] FORMAT ProxySet_Index = ProxySet_EnableProxyKeepAlive, ProxySet_ProxyKeepAliveTime, ProxySet_ProxyLoadBalancingMethod, ProxySet_IsProxyHotSwap, ProxySet_SRD, ProxySet_ClassificationInput, ProxySet_ProxyRedundancyMode; [ProxySet]</p> <p>For example: ProxySet 0 = 0, 60, 0, 0, 0, , 1; ProxySet 1 = 1, 60, 1, 0, 1, , 0;</p> <p>Notes:</p> <ul style="list-style-type: none"> For configuring the Proxy Set IDs and their IP addresses, use the parameter ProxyIP. For a description of this table, see 'Configuring Proxy Sets Table' on page 197.
Registrar Parameters	
Web/EMS: Registration Time [RegistrationTime]	<p>Defines the time interval (in seconds) for registering to a Proxy server. The value is used in the SIP Expires header. This parameter also defines the time interval between Keep-Alive messages when the parameter EnableProxyKeepAlive is set to 2 (REGISTER).</p> <p>Typically, the device registers every 3,600 sec (i.e., one hour). The device resumes registration according to the parameter RegistrationTimeDivider.</p> <p>The valid range is 10 to 2,000,000. The default is 180.</p>
Web: Re-registration Timing [%] EMS: Time Divider [RegistrationTimeDivider]	<p>Defines the re-registration timing (in percentage). The timing is a percentage of the re-register timing set by the Registrar server.</p> <p>The valid range is 50 to 100. The default is 50.</p> <p>For example: If this parameter is set to 70% and the Registration Expires time is 3600, the device re-sends its registration request after 3600 x 70% (i.e., 2520 sec).</p> <p>Note: This parameter may be overridden if the parameter RegistrationTimeThreshold is greater than 0.</p>
Web/EMS: Registration Retry Time [RegistrationRetryTime]	<p>Defines the time interval (in seconds) after which a registration request is re-sent if registration fails with a 4xx response or if there is no response from the Proxy/Registrar server.</p> <p>The default is 30 seconds. The range is 10 to 3600.</p>

Parameter	Description
Web: Registration Time Threshold EMS: Time Threshold [RegistrationTimeThreshold]	<p>Defines a threshold (in seconds) for re-registration timing. If this parameter is greater than 0, but lower than the computed re-registration timing (according to the parameter RegistrationTimeDivider), the re-registration timing is set to the following: timing set by the Registration server in the SIP Expires header minus the value of the parameter RegistrationTimeThreshold.</p> <p>The valid range is 0 to 2,000,000. The default is 0.</p>
Web: Re-register On INVITE Failure EMS: Register On Invite Failure [RegisterOnInviteFailure]	<p>Enables immediate re-registration if no response is received for an INVITE request sent by the device.</p> <ul style="list-style-type: none"> [0] Disable (default) [1] Enable <p>When enabled, the device immediately expires its re-registration timer and commences re-registration to the same Proxy upon any of the following scenarios:</p> <ul style="list-style-type: none"> The response to an INVITE request is 407 (Proxy Authentication Required) without an authentication header included. The remote SIP UA abandons a call before the device has received any provisional response (indicative of an outbound proxy server failure). The remote SIP UA abandons a call and the only provisional response the device has received for the call is 100 Trying (indicative of a home proxy server failure, i.e., the failure of a proxy in the route after the outbound proxy). The device terminates a call due to the expiration of RFC 3261 Timer B or due to the receipt of a 408 (Request Timeout) response and the device has not received any provisional response for the call (indicative of an outbound proxy server failure). The device terminates a call due to the receipt of a 408 (Request Timeout) response and the only provisional response the device has received for the call is the 100 Trying provisional response (indicative of a home proxy server failure).
Web: ReRegister On Connection Failure EMS: Re Register On Connection Failure [ReRegisterOnConnectionFailure]	<p>Enables the device to perform SIP re-registration upon TCP/TLS connection failure.</p> <ul style="list-style-type: none"> [0] Disable (default) [1] Enable
[UnregistrationMode]	<p>Enables the device to perform explicit unregisters.</p> <ul style="list-style-type: none"> [0] Disable (default) [1] Enable = The device sends an asterisk ("*") value in the SIP Contact header, instructing the Registrar server to remove all previous registration bindings. The device removes SIP User Agent (UA) registration bindings in a Registrar, according to RFC 3261. Registrations are soft state and expire unless refreshed, but they can also be explicitly removed. A client can attempt to influence the expiration interval selected by the Registrar. A UA requests the immediate removal of a binding by specifying an expiration interval of "0" for that contact address in a REGISTER request. UA's should support this mechanism

Parameter	Description
	<p>so that bindings can be removed before their expiration interval has passed. Use of the "*" Contact header field value allows a registering UA to remove all bindings associated with an address-of-record (AOR) without knowing their precise values.</p> <p>Note: The REGISTER-specific Contact header field value of "*" applies to all registrations, but it can only be used if the Expires header field is present with a value of "0".</p>
Web/EMS: Add Empty Authorization Header [EmptyAuthorizationHeader]	<p>Enables the inclusion of the SIP Authorization header in initial registration (REGISTER) requests sent by the device.</p> <ul style="list-style-type: none"> ▪ [0] Disable (default) ▪ [1] Enable <p>The Authorization header carries the credentials of a user agent (UA) in a request to a server. The sent REGISTER message populates the Authorization header with the following parameters:</p> <ul style="list-style-type: none"> ▪ username - set to the value of the private user identity ▪ realm - set to the domain name of the home network ▪ uri - set to the SIP URI of the domain name of the home network ▪ nonce - set to an empty value ▪ response - set to an empty value <p>For example:</p> <pre>Authorization: Digest username=alice_private@home1.net, realm="home1.net", nonce="", response="e56131d19580cd833064787ecc"</pre> <p>Note: This registration header is according to the IMS 3GPP TS24.229 and PKT-SP-24.220 specifications.</p>
Web: Add initial Route Header [InitialRouteHeader]	<p>Enables the inclusion of the SIP Route header in initial registration or re-registration (REGISTER) requests sent by the device.</p> <ul style="list-style-type: none"> ▪ [0] Disable (default) ▪ [1] Enable <p>When the device sends a REGISTER message, the Route header includes either the Proxy's FQDN, or IP address and port according to the configured Proxy Set, for example:</p> <pre>Route: <sip:10.10.10.10;lr;transport=udp></pre> <p>or</p> <pre>Route: <sip: pcscf- gm.ims.rr.com;lr;transport=udp></pre>
EMS: Ping Pong Keep Alive [UsePingPongKeepAlive]	<p>Enables the use of the carriage-return and line-feed sequences (CRLF) Keep-Alive mechanism, according to RFC 5626 "Managing Client-Initiated Connections in the Session Initiation Protocol (SIP)" for reliable, connection-orientated transport types such as TCP.</p> <ul style="list-style-type: none"> ▪ [0] Disable (default) ▪ [1] Enable <p>The SIP user agent/client (i.e., device) uses a simple periodic message as a keep-alive mechanism to keep their flow to the</p>

Parameter	Description
	<p>proxy or registrar alive (used for example, to keep NAT bindings open). For connection-oriented transports such as TCP/TLS this is based on CRLF. This mechanism uses a client-to-server "ping" keep-alive and a corresponding server-to-client "pong" message. This ping-pong sequence allows the client, and optionally the server, to tell if its flow is still active and useful for SIP traffic. If the client does not receive a pong in response to its ping, it declares the flow "dead" and opens a new flow in its place. In the CRLF Keep-Alive mechanism the client periodically (defined by the PingPongKeepAliveTime parameter) sends a double-CRLF (the "ping") then waits to receive a single CRLF (the "pong"). If the client does not receive a "pong" within an appropriate amount of time, it considers the flow failed.</p> <p>Note: The device sends a CRLF message to the Proxy Set only if the Proxy Keep-Alive feature (EnableProxyKeepAlive parameter) is enabled and its transport type is set to TCP or TLS. The device first sends a SIP OPTION message to establish the TCP/TLS connection and if it receives any SIP response, it continues sending the CRLF keep-alive sequences.</p>
EMS: Ping Pong Keep Alive Time [PingPongKeepAliveTime]	<p>Defines the periodic interval (in seconds) after which a "ping" (double-CRLF) keep-alive is sent to a proxy/registrar, using the CRLF Keep-Alive mechanism.</p> <p>The default range is 5 to 2,000,000. The default is 120.</p> <p>The device uses the range of 80-100% of this user-defined value as the actual interval. For example, if the parameter value is set to 200 sec, the interval used is any random time between 160 to 200 seconds. This prevents an "avalanche" of keep-alive by multiple SIP UAs to a specific server.</p>

42.7.2 Network Application Parameters

The SIP network application parameters are described in the table below.

Table 42-29: SIP Network Application Parameters

Parameter	Description
Signaling Routing Domain Table	
Web: SRD Settings EMS: SRD Table [SRD]	<p>This table parameter configures the Signaling Routing Domain (SRD) table. The format of this parameter is as follows:</p> <p>[SRD] FORMAT SRD_Index = SRD_Name, SRD_MediaRealm, SRD_IntraSRDMediaAnchoring, SRD_BlockUnRegUsers, SRD_MaxNumOfRegUsers, SRD_EnableUnAuthenticatedRegistrations; [SRD]</p> <p>For example: SRD 1 = LAN1_SRD, Mrealm1, 0, 1, 15, 1; SRD 2 = LAN2_SRD, Mrealm2, 0, 1, 15, 1;</p> <p>Note: For a detailed description of this table, see 'Configuring SRD Table' on page 185.</p>
SIP Interface Table	

Parameter	Description
Web: SIP Interface Table EMS: SIP Interfaces Table [SIPInterface]	<p>This table parameter configures the SIP Interface table. The SIP Interface represents a SIP signaling entity, comprising ports (UDP, TCP, and TLS) and associated with a specific IP interface and an SRD ID. The format of this parameter is as follows:</p> <p>[SIPInterface] FORMAT SIPInterface_Index = SIPInterface_NetworkInterface, SIPInterface_ApplicationType, SIPInterface_UDPPort, SIPInterface_TCPPort, SIPInterface_TLSPort, SIPInterface_SRD, SIPInterface_MessagePolicy, SIPInterface_TLSMutualAuthentication, SIPInterface_TCPKeepaliveEnable, SIPInterface_ClassificationFailureResponseType; [\SIPInterface]</p> <p>Note: For a detailed description of this table, see 'Configuring SIP Interface Table' on page 188.</p>
TCP Keep Alive Idle Time [TCPKeepAliveTime]	<p>Defines the interval (in sec) between the last data packet sent and the first keep-alive probe to send.</p> <p>The valid value is 10 to 65,000. The default is 60.</p> <p>Notes:</p> <ul style="list-style-type: none"> Simple ACKs such as keep-alives are not considered data packets. TCP keepalive is enabled per SIP Interface in the SIP Interface table.
TCP Keep Alive Interval Time [TCPKeepAliveInterval]	<p>Defines the interval (in sec) between consecutive keep-alive probes, regardless of what the connection has exchanged in the meantime.</p> <p>The valid value is 10 to 65,000. The default is 10.</p> <p>Note: TCP keepalive is enabled per SIP Interface in the SIP Interface table.</p>
TCP Keep Alive Retry Number [TCPKeepAliveRetry]	<p>Defines the number of unacknowledged keep-alive probes to send before considering the connection down.</p> <p>The valid value is 1 to 100. The default is 5.</p> <p>Note: TCP keepalive is enabled per SIP Interface in the SIP Interface table.</p>
NAT Translation Table	
Web: NAT Translation Table [NATTranslation]	<p>This table parameter defines NAT rules for translating source IP addresses per VoIP interface (SIP control and RTP media traffic) into NAT IP addresses. This allows, for example, the separation of VoIP traffic between different ISTP's, and topology hiding (of internal IP addresses to the "public" network). Each IP interface (configured in the Multiple Interface table - InterfaceTable parameter) can be associated with a NAT rule in this table, translating the source IP address and port of the outgoing packet into the NAT address (IP address and port range).</p> <p>The format of this parameter is as follows:</p> <p>[NATTranslation] FORMAT NATTranslation_Index = NATTranslation_SourceIPInterfaceName, NATTranslation_TargetIPAddress, NATTranslation_SourceStartPort, NATTranslation_SourceEndPort, NATTranslation_TargetStartPort, NATTranslation_TargetEndPort; [\NATTranslation]</p> <p>Note: For a detailed description of this table, see 'Configuring NAT Translation per IP Interface' on page 120.</p>

42.8 General SIP Parameters

The general SIP parameters are described in the table below.

Table 42-30: General SIP Parameters

Parameter	Description
Web: SIP Remote Reset CLI: sip-remote-reset [EnableSIPRemoteReset]	<p>Enables a specific device action upon the receipt of a SIP NOTIFY request, where the action depends on the value received in the Event header.</p> <ul style="list-style-type: none"> [0] Disable (default) [1] Enable <p>The action depends on the Event header value:</p> <ul style="list-style-type: none"> 'check-sync;reboot=false': triggers the regular Automatic Update feature (if Automatic Update has been enabled on the device) 'check-sync;reboot=true': triggers a device reset <p>Note: The Event header value is proprietary to AudioCodes.</p>
Web/EMS: Max SIP Message Length [KB] [MaxSIPMessageLength]	<p>Defines the maximum size (in Kbytes) for each SIP message that can be sent over the network. The device rejects messages exceeding this user-defined size.</p> <p>The valid value range is 1 to 50. The default is 50.</p>
[SIPForceRport]	<p>Determines whether the device sends SIP responses to the UDP port from where SIP requests are received even if the 'rport' parameter is not present in the SIP Via header.</p> <ul style="list-style-type: none"> [0] = (Default) Disabled. The device sends the SIP response to the UDP port defined in the Via header. If the Via header contains the 'rport' parameter, the response is sent to the UDP port from where the SIP request is received. [1] = Enabled. SIP responses are sent to the UDP port from where SIP requests are received even if the 'rport' parameter is not present in the Via header.
Web: Reject Cancel after Connect CLI: reject-cancel-after-connect [RejectCancelAfterConnect]	<p>Determines whether the device accepts or rejects a SIP CANCEL request received after the receipt of a 200 OK, during an established call.</p> <ul style="list-style-type: none"> [0] = (Default) Accepts the CANCEL, by responding with a 200 OK and terminating the call session. [1] = Rejects the CANCEL, by responding with a SIP 481 Call/Transaction Does Not Exist, and maintaining the call session.
Web: Verify Received RequestURI CLI: verify-rcvd-requri [VerifyReceivedRequestUri]	<p>Enables the device to reject SIP requests (such as ACK, BYE, or re-INVITE) whose user part in the Request-URI is different from the user part received in the Contact header of the last sent SIP request.</p> <ul style="list-style-type: none"> [0] Disable = (Default) Even if the user is different, the device accepts the SIP request. [1] Enable = If the user is different, the device rejects the SIP request (BYE is responded with 481; re-INVITE is responded with 404; ACK is ignored).
Web: Max Number of Active Calls EMS: Maximum Concurrent Calls	<p>Defines the maximum number of simultaneous active calls supported by the device. If the maximum number of calls is reached, new calls are not established.</p> <p>The valid range is 1 to the maximum number of supported channels. The default is the maximum available channels (i.e., no restriction on</p>

Parameter	Description
[MaxActiveCalls]	the maximum number of calls).
Web: QoS statistics in SIP Release Call [QoSStatistics]	<p>Enables the device to include call quality of service (QoS) statistics in SIP BYE and SIP 200 OK response to BYE, using the proprietary SIP header X-RTP-Stat.</p> <ul style="list-style-type: none"> ▪ [0] = Disable (default) ▪ [1] = Enable <p>The X-RTP-Stat header provides the following statistics:</p> <ul style="list-style-type: none"> ▪ Number of received and sent voice packets ▪ Number of received and sent voice octets ▪ Received packet loss, jitter (in ms), and latency (in ms) <p>The X-RTP-Stat header contains the following fields:</p> <ul style="list-style-type: none"> ▪ PS=<voice packets sent> ▪ OS=<voice octets sent> ▪ PR=<voice packets received> ▪ OR=<voice octets received> ▪ PL=<receive packet loss> ▪ JI=<jitter in ms> ▪ LA=<latency in ms> <p>Below is an example of the X-RTP-Stat header in a SIP BYE message:</p> <pre> BYE sip:302@10.33.4.125 SIP/2.0 Via: SIP/2.0/UDP 10.33.4.126;branch=z9hG4bKac2127550866 Max-Forwards: 70 From: <sip:401@10.33.4.126;user=phone>;tag=1c2113553324 To: <sip:302@company.com>;tag=1c991751121 Call-ID: 991750671245200001912@10.33.4.125 CSeq: 1 BYE X-RTP-Stat: PS=207;OS=49680;;PR=314;OR=50240;PL=0;JI=600;LA=40; Supported: em,timer,replaces,path,resource-priority Allow: REGISTER,OPTIONS,INVITE,ACK,CANCEL,BYE,NOTIFY,PRACK ,REFER,INFO,SUBSCRIBE,UPDATE User-Agent: Sip-Gateway-/v.6.2A.008.006 Reason: Q.850 ;cause=16 ;text="local" Content-Length: 0 </pre>
Web/EMS: PRACK Mode [PrackMode]	<p>Determines the PRACK (Provisional Acknowledgment) mechanism mode for SIP 1xx reliable responses.</p> <ul style="list-style-type: none"> ▪ [0] Disable ▪ [1] Supported (default) ▪ [2] Required <p>Notes:</p> <ul style="list-style-type: none"> ▪ The Supported and Required headers contain the '100rel' tag. ▪ The device sends PRACK messages if 180/183 responses are received with '100rel' in the Supported or Required headers.
Web/EMS: Enable Early Media	<p>Enables the Early Media feature.</p> <ul style="list-style-type: none"> ▪ [0] Disable (default)

Parameter	Description
[EnableEarlyMedia]	<ul style="list-style-type: none"> [1] Enable <p>Note: This feature can also be configured as an IP Profile and/or Tel Profile.</p>
Web: 183 Message Behavior EMS: SIP 183 Behaviour [SIP183Behaviour]	<ul style="list-style-type: none"> [0] Progress = (Default) [1] Alert =
Web: Session-Expires Time EMS: Sip Session Expires [SIPSessionExpires]	<p>Defines the numerical value sent in the Session-Expires header in the first INVITE request or response (if the call is answered).</p> <p>The valid range is 1 to 86,400 sec. The default is 0 (i.e., the Session-Expires header is disabled).</p>
Web: Minimum Session-Expires EMS: Minimal Session Refresh Value [MinSE]	<p>Defines the time (in seconds) that is used in the Min-SE header. This header defines the minimum time that the user agent refreshes the session.</p> <p>The valid range is 10 to 100,000. The default is 90.</p>
Web/EMS: Session Expires Disconnect Time CLI: session-exp-disconnect-time [SessionExpiresDisconnectTime]	<p>Defines a session expiry timeout. The device disconnects the session (sends a SIP BYE) if the refresher does not send a refresh request before one-third (1/3) of the session expires time, or before the time configured by this parameter (the minimum of the two).</p> <p>The valid range is 0 to 32 (in seconds). The default is 32.</p>
Web/EMS: Session Expires Method [SessionExpiresMethod]	<p>Determines the SIP method used for session-timer updates.</p> <ul style="list-style-type: none"> [0] Re-INVITE = (Default) Uses Re-INVITE messages for session-timer updates. [1] UPDATE = Uses UPDATE messages. <p>Notes:</p> <ul style="list-style-type: none"> The device can receive session-timer refreshes using both methods. The UPDATE message used for session-timer is excluded from the SDP body.
[RemoveToTagInFailureResponse]	<p>Determines whether the device removes the 'to' header tag from final SIP failure responses to INVITE transactions.</p> <ul style="list-style-type: none"> [0] = (Default) Do not remove tag. [1] = Remove tag.
[EnableRTCPAttribute]	<p>Enables the use of the 'rtcp' attribute in the outgoing SDP.</p> <ul style="list-style-type: none"> [0] = Disable (default) [1] = Enable
EMS: Options User Part [OPTIONSUserPart]	<p>Defines the user part value of the Request-URI for outgoing SIP OPTIONS requests. If no value is configured, the configuration parameter 'Username' value is used.</p> <p>A special value is 'empty', indicating that no user part in the Request-URI (host part only) is used.</p> <p>The valid range is a 30-character string. The default is an empty string ("").</p>
Web: Fax Signaling Method EMS: Fax Used	<p>Determines the SIP signaling method for establishing and transmitting a fax session after a fax is detected.</p> <ul style="list-style-type: none"> [0] No Fax = (Default) No fax negotiation using SIP signaling. Fax

Parameter	Description
[IsFaxUsed]	<p>transport method is according to the parameter FaxTransportMode.</p> <ul style="list-style-type: none"> ▪ [1] T.38 Relay = Initiates T.38 fax relay. ▪ [2] G.711 Transport = Initiates fax/modem using the coder G.711 A-law/Mu-law with adaptations (see Note below). ▪ [3] Fax Fallback = Initiates T.38 fax relay. If the T.38 negotiation fails, the device re-initiates a fax session using the coder G.711 A-law/μ-law with adaptations (see the Note below). <p>Notes:</p> <ul style="list-style-type: none"> ▪ Fax adaptations (for options 2 and 3): <ul style="list-style-type: none"> ✓ Echo Canceller = On ✓ Silence Compression = Off ✓ Echo Canceller Non-Linear Processor Mode = Off ✓ Dynamic Jitter Buffer Minimum Delay = 40 ✓ Dynamic Jitter Buffer Optimization Factor = 13 ▪ If the device initiates a fax session using G.711 (option 2 and possibly 3), a 'gpmid' attribute is added to the SDP in the following format: <ul style="list-style-type: none"> ✓ For A-law: 'a=gpmid:8 vbd=yes;ecan=on' ✓ For μ-law: 'a=gpmid:0 vbd=yes;ecan=on' ▪ When this parameter is set to 1, 2, or 3, the parameter FaxTransportMode is ignored. ▪ When this parameter is set to 0, T.38 might still be used without the control protocol's involvement. To completely disable T.38, set FaxTransportMode to a value other than 1. ▪ This parameter can also be configured per IP Profile (using the IPProfile parameter). ▪ For more information on fax transport methods, see 'Fax/Modem Transport Modes' on page 143.
[HandleG711asVBD]	<p>Enables the handling of G.711 as G.711 VBD coder.</p> <ul style="list-style-type: none"> ▪ [0] = (Default) Disable. The device negotiates G.711 as a regular audio coder and sends an answer only with G.729 coder. For example, if the device is configured with G.729 and G.711 VBD coders and it receives an INVITE with an SDP offer containing G.729 and "regular" G.711 coders, it sends an SDP answer containing only the G.729 coder. ▪ [1] = Enable. The device assumes that the G.711 coder received in the INVITE SDP offer is a VBD coder. For example, if the device is configured with G.729 and G.711 VBD coders and it receives an INVITE with an SDP offer containing G.729 and "regular" G.711 coders, it sends an SDP answer containing G.729 and G.711 VBD coders, allowing a subsequent bypass (passthrough) session if fax/modem signals are detected during the call. <p>Note: This parameter is applicable only if G.711 VBD coder(s) with regular G.711 payload types 0 or 8 are configured for the device (using the CodersGroup parameter).</p>
[FaxVBDBehavior]	<p>Determines the device's fax transport behavior when G.711 VBD coder is negotiated at call start.</p> <ul style="list-style-type: none"> ▪ [0] = (Default) If the device is configured with a VBD coder (see the CodersGroup parameter) and is negotiated OK at call start, then both fax and modem signals are sent over RTP using the bypass payload type (and no mid-call VBD or T.38 Re-INVITEs occur). ▪ [1] = If the IsFaxUsed parameter is set to 1, the channel opens with

Parameter	Description
	<p>the FaxTransportMode parameter set to 1 (relay). This is required to detect mid-call fax tones and to send T.38 Re-INVITE messages upon fax detection. If the remote party supports T.38, the fax is relayed over T.38.</p> <p>Notes:</p> <ul style="list-style-type: none"> If VBD coder negotiation fails at call start and if the IsFaxUsed parameter is set to 1 (or 3), then the channel opens with the FaxTransportMode parameter set to 1 (relay) to allow future detection of fax tones and sending of T.38 Re-INVITES. In such a scenario, the FaxVBDBehavior parameter has no effect. This feature can be used only if the remote party supports T.38 fax relay; otherwise, the fax fails.
[NoAudioPayloadType]	<p>Defines the payload type of the outgoing SDP offer. The valid value range is 96 to 127 (dynamic payload type). The default is 0 (i.e. NoAudio is not supported). For example, if set to 120, the following is added to the INVITE SDP:</p> <pre>a=rtpmap:120 NoAudio/8000\r\n</pre> <p>Note: For incoming SDP offers, NoAudio is always supported.</p>
Web: SIP Transport Type EMS: Transport Type [SIPTransportType]	<p>Determines the default transport layer for outgoing SIP calls initiated by the device.</p> <ul style="list-style-type: none"> [0] UDP (default) [1] TCP [2] TLS (SIPS) <p>Notes:</p> <ul style="list-style-type: none"> It's recommended to use TLS for communication with a SIP Proxy and not for direct device-to-device communication. For received calls (i.e., incoming), the device accepts all these protocols. The value of this parameter is also used by the SAS application as the default transport layer for outgoing SIP calls.
Web: SIP UDP Local Port EMS: Local SIP Port [LocalSIPPort]	<p>Defines the local UDP port for SIP messages. The valid range is 1 to 65534. The default is 5060.</p>
Web: SIP TCP Local Port EMS: TCP Local SIP Port [TCPLocalSIPPort]	<p>Defines the local TCP port for SIP messages. The valid range is 1 to 65535. The default is 5060.</p>
Web: SIP TLS Local Port EMS: TLS Local SIP Port [TLSLocalSIPPort]	<p>Defines the local TLS port for SIP messages. The valid range is 1 to 65535. The default is 5061.</p> <p>Note: The value of this parameter must be different from the value of the parameter TCPLocalSIPPort.</p>
Web/EMS: Enable SIPS [EnableSIPS]	<p>Enables secured SIP (SIPS URI) connections over multiple hops.</p> <ul style="list-style-type: none"> [0] Disable (default) [1] Enable <p>When the SIPTransportType parameter is set to 2 (i.e., TLS) and the parameter EnableSIPS is disabled, TLS is used for the next network hop only. When the parameter SIPTransportType is set to 2 or 1 (i.e., TCP or TLS) and EnableSIPS is enabled, TLS is used through the entire connection (over multiple hops).</p> <p>Note: If this parameter is enabled and the parameter SIPTransportType</p>

Parameter	Description
	is set to 0 (i.e., UDP), the connection fails.
Web/EMS: Enable TCP Connection Reuse [EnableTCPConnectionReuse]	<p>Enables the reuse of the same TCP connection for all calls to the same destination.</p> <ul style="list-style-type: none"> ▪ [0] Disable = Uses a separate TCP connection for each call. ▪ [1] Enable = (Default) Uses the same TCP connection for all calls. <p>Note: For the SAS application, this feature is configured using the SASConnectionReuse parameter.</p>
Web: Fake TCP alias [FakeTCPAlias]	<p>Enables the re-use of the same TCP/TLS connection for sessions with the same user, even if the "alias" parameter is not present in the SIP Via header of the first INVITE.</p> <ul style="list-style-type: none"> ▪ [0] Disable = (Default) TCP/TLS connection reuse is done only if the "alias" parameter is present in the Via header of the first INVITE. ▪ [1] Enable <p>Note: To enable TCP/TLS connection re-use, set the EnableTCPConnectionReuse parameter to 1.</p>
Web/EMS: Reliable Connection Persistent Mode [ReliableConnectionPersistentMode]	<p>Enables setting of all TCP/TLS connections as persistent and therefore, not released.</p> <ul style="list-style-type: none"> ▪ [0] = (Default) Disable. All TCP connections (except those that are set to a proxy IP) are released if not used by any SIP dialog/transaction. ▪ [1] = Enable - TCP connections to all destinations are persistent and not released unless the device reaches 70% of its maximum TCP resources. <p>While trying to send a SIP message connection, reuse policy determines whether live connections to the specific destination are re-used.</p> <p>Persistent TCP connection ensures less network traffic due to fewer setting up and tearing down of TCP connections and reduced latency on subsequent requests due to avoidance of initial TCP handshake. For TLS, persistent connection may reduce the number of costly TLS handshakes to establish security associations, in addition to the initial TCP connection set up.</p> <p>Note: If the destination is a Proxy server, the TCP/TLS connection is persistent regardless of the settings of this parameter.</p>
Web/EMS: TCP Timeout [SIPTCPTimeout]	<p>Defines the Timer B (INVITE transaction timeout timer) and Timer F (non-INVITE transaction timeout timer), as defined in RFC 3261, when the SIP Transport Type is TCP.</p> <p>The valid range is 0 to 40 sec. The default is 64 multiplied by the SipT1Rtx parameter value. For example, if SipT1Rtx is set to 500 msec, then the default of SIPTCPTimeout is 32 sec.</p>
Web: SIP Destination Port EMS: Destination Port [SIPDestinationPort]	<p>Defines the SIP destination port for sending initial SIP requests.</p> <p>The valid range is 1 to 65534. The default port is 5060.</p> <p>Note: SIP responses are sent to the port specified in the Via header.</p>
Web: Use user=phone in SIP URL EMS: Is User Phone [IsUserPhone]	<p>Determines whether the 'user=phone' string is added to the SIP URI and SIP To header.</p> <ul style="list-style-type: none"> ▪ [0] No = 'user=phone' string is not added. ▪ [1] Yes = (Default) 'user=phone' string is part of the SIP URI and SIP To header.
Web: Use user=phone in	Determines whether the 'user=phone' string is added to the From and

Parameter	Description											
From Header EMS: Is User Phone In From [IsUserPhoneInFrom]	Contact SIP headers. <ul style="list-style-type: none"> [0] No = (Default) Doesn't add 'user=phone' string. [1] Yes = 'user=phone' string is part of the From and Contact headers. 											
Web: Use Tel URI for Asserted Identity [UseTelURIForAssertedID]	Determines the format of the URI in the P-Asserted-Identity and P-Preferred-Identity headers. <ul style="list-style-type: none"> [0] Disable = (Default) 'sip:' [1] Enable = 'tel:' 											
Web: Tel to IP No Answer Timeout EMS: IP Alert Timeout [IPAlertTimeout]	Defines the time (in seconds) that the device waits for a 200 OK response from the called party (IP side) after sending an INVITE message. If the timer expires, the call is released. The valid range is 0 to 3600. The default is 180.											
Web: Enable Remote Party ID EMS: Enable RPI Header [EnableRPIheader]	Enables Remote-Party-Identity headers for calling and called numbers for Tel-to-IP calls. <ul style="list-style-type: none"> [0] Disable (default). [1] Enable = Remote-Party-Identity headers are generated in SIP INVITE messages for both called and calling numbers. 											
Web: Enable History-Info Header EMS: Enable History Info [EnableHistoryInfo]	<p>Enables usage of the History-Info header.</p> <ul style="list-style-type: none"> [0] Disable (default) [1] Enable <p>User Agent Client (UAC) Behavior:</p> <ul style="list-style-type: none"> Initial request: The History-Info header is equal to the Request-URI. If a PSTN Redirect number is received, it is added as an additional History-Info header with an appropriate reason. Upon receiving the final failure response, the device copies the History-Info as is, adds the reason of the failure response to the last entry, and concatenates a new destination to it (if an additional request is sent). The order of the reasons is as follows: <ol style="list-style-type: none"> Q.850 Reason SIP Reason SIP Response code Upon receiving the final response (success or failure), the device searches for a Redirect reason in the History-Info (i.e., 3xx/4xx SIP reason). If found, it is passed to ISDN according to the following table: <table border="1"> <thead> <tr> <th>SIP Reason Code</th><th>ISDN Redirecting Reason</th></tr> </thead> <tbody> <tr> <td>302 - Moved Temporarily</td><td>Call Forward Universal (CFU)</td></tr> <tr> <td>408 - Request Timeout</td><td rowspan="3">Call Forward No Answer (CFNA)</td></tr> <tr> <td>480 - Temporarily Unavailable</td></tr> <tr> <td>487 - Request Terminated</td></tr> <tr> <td>486 - Busy Here</td><td rowspan="2">Call Forward Busy (CFB)</td></tr> <tr> <td>600 - Busy Everywhere</td></tr> </tbody> </table> <ul style="list-style-type: none"> If history reason is a Q.850 reason, it is translated to the SIP reason (according to the SIP-ISDN tables) and then to ISDN Redirect reason according to the table above. <p>User Agent Server (UAS) Behavior:</p> <ul style="list-style-type: none"> The History-Info header is sent only in the final response. 	SIP Reason Code	ISDN Redirecting Reason	302 - Moved Temporarily	Call Forward Universal (CFU)	408 - Request Timeout	Call Forward No Answer (CFNA)	480 - Temporarily Unavailable	487 - Request Terminated	486 - Busy Here	Call Forward Busy (CFB)	600 - Busy Everywhere
SIP Reason Code	ISDN Redirecting Reason											
302 - Moved Temporarily	Call Forward Universal (CFU)											
408 - Request Timeout	Call Forward No Answer (CFNA)											
480 - Temporarily Unavailable												
487 - Request Terminated												
486 - Busy Here	Call Forward Busy (CFB)											
600 - Busy Everywhere												

Parameter	Description
Web/EMS: Enable GRUU [EnableGRUU]	<ul style="list-style-type: none"> Upon receiving a request with History-Info, the UAS checks the policy in the request. If a 'session', 'header', or 'history' policy tag is found, the (final) response is sent without History-Info; otherwise, it is copied from the request. <p>Determines whether the Globally Routable User Agent URIs (GRUU) mechanism is used, according to RFC 5627. This is used for obtaining a GRUU from a registrar and for communicating a GRUU to a peer within a dialog.</p> <ul style="list-style-type: none"> [0] Disable (default) [1] Enable <p>A GRUU is a SIP URI that routes to an instance-specific UA and can be reachable from anywhere. There are a number of contexts in which it is desirable to have an identifier that addresses a single UA (using GRUU) rather than the group of UA's indicated by an Address of Record (AOR). For example, in call transfer where user A is talking to user B, and user A wants to transfer the call to user C. User A sends a REFER to user C:</p> <pre>REFER sip:C@domain.com SIP/2.0 From: sip:A@domain.com;tag=99asd To: sip:C@domain.com Refer-To: (URI that identifies B's UA)</pre> <p>The Refer-To header needs to contain a URI that user C can use to place a call to user B. This call needs to route to the specific UA instance that user B is using to talk to user A. User B should provide user A with a URI that has to be usable by anyone. It needs to be a GRUU.</p> <ul style="list-style-type: none"> Obtaining a GRUU: The mechanism for obtaining a GRUU is through registrations. A UA can obtain a GRUU by generating a REGISTER request containing a Supported header field with the value "gruu". The UA includes a "+sip.instance" Contact header parameter of each contact for which the GRUU is desired. This Contact parameter contains a globally unique ID that identifies the UA instance. The global unique ID is created from one of the following: <ul style="list-style-type: none"> ✓ If the REGISTER is per the device's client (endpoint), it is the MAC address concatenated with the phone number of the client. ✓ If the REGISTER is per device, it is the MAC address only. ✓ When using TP, "User Info" can be used for registering per endpoint. Thus, each endpoint can get a unique id – its phone number. The globally unique ID in TP is the MAC address concatenated with the phone number of the endpoint. <p>If the remote server doesn't support GRUU, it ignores the parameters of the GRUU. Otherwise, if the remote side also supports GRUU, the REGISTER responses contain the "gruu" parameter in each Contact header. This parameter contains a SIP or SIPS URI that represents a GRUU corresponding to the UA instance that registered the contact. The server provides the same GRUU for the same AOR and instance-id when sending REGISTER again after registration expiration. RFC 5627 specifies that the remote target is a GRUU target if its' Contact URL has the "gr" parameter with or without a value.</p> <ul style="list-style-type: none"> Using GRUU: The UA can place the GRUU in any header field that can contain a URI. It must use the GRUU in the following messages: INVITE request, its 2xx response, SUBSCRIBE request, its 2xx response, NOTIFY request, REFER request and its 2xx response.

Parameter	Description
EMS: Is CISCO Sce Mode [IsCiscoSCEMode]	<p>Determines whether a Cisco gateway exists at the remote side.</p> <ul style="list-style-type: none"> [0] = (Default) No Cisco gateway exists at the remote side. [1] = A Cisco gateway exists at the remote side. <p>When a Cisco gateway exists at the remote side, the device must set the value of the 'annexb' parameter of the fmtp attribute in the SDP to 'no'. This logic is used if the parameter EnableSilenceCompression is set to 2 (enable without adaptation). In this case, Silence Suppression is used on the channel but not declared in the SDP.</p> <p>Note: The IsCiscoSCEMode parameter is applicable only when the selected coder is G.729.</p>
Web: User-Agent Information EMS: User Agent Display Info [UserAgentDisplayInfo]	<p>Defines the string that is used in the SIP User-Agent and Server response headers. When configured, the string <UserAgentDisplayInfo value>/software version' is used, for example:</p> <pre>User-Agent: myproduct/v.6.40.010.006</pre> <p>If not configured, the default string, <AudioCodes product-name>/software version' is used, for example:</p> <pre>User-Agent: Audiocodes-Sip-Gateway-Mediant 4000 E-SBC/v.6.40.010.006</pre> <p>The maximum string length is 50 characters.</p> <p>Note: The software version number and preceding forward slash (/) cannot be modified. Therefore, it is recommended not to include a forward slash in the parameter's value (to avoid two forward slashes in the SIP header, which may cause problems).</p>
Web/EMS: SDP Session Owner [SIPSDPSessionOwner]	<p>Defines the value of the Owner line ('o' field) in outgoing SDP messages.</p> <p>The valid range is a string of up to 39 characters. The default is 'AudiocodesGW'.</p> <p>For example:</p> <pre>o=AudiocodesGW 1145023829 1145023705 IN IP4 10.33.4.126</pre>
[EnableSDPVersionNegotiation]	<p>Enables the device to ignore new SDP re-offers (from the media negotiation perspective) in certain scenarios (such as session expires). According to RFC 3264, once an SDP session is established, a new SDP offer is considered a new offer only when the SDP origin value is incremented. In scenarios such as session expires, SDP negotiation is irrelevant and thus, the origin field is not changed.</p> <p>Even though some SIP devices don't follow this behavior and don't increment the origin value even in scenarios where they want to re-negotiate, the device can assume that the remote party operates according to RFC 3264, and in cases where the origin field is not incremented, the device does not re-negotiate SDP capabilities.</p> <ul style="list-style-type: none"> [0] Disable = (Default) The device negotiates any new SDP re-offer, regardless of the origin field. [1] Enable = The device negotiates only an SDP re-offer with an incremented origin field.
Web/EMS: Subject [SIPSubject]	<p>Defines the Subject header value in outgoing INVITE messages. If not specified, the Subject header isn't included (default).</p> <p>The maximum length is up to 50 characters.</p>

Parameter	Description
Web: Multiple Packetization Time Format EMS: Multi Ptime Format [MultiPtimeFormat]	<p>Determines whether the 'mptime' attribute is included in the outgoing SDP.</p> <ul style="list-style-type: none"> [0] None = (Default) Disabled. [1] PacketCable = Includes the 'mptime' attribute in the outgoing SDP - PacketCable-defined format. <p>The 'mptime' attribute enables the device to define a separate packetization period for each negotiated coder in the SDP. The 'mptime' attribute is only included if this parameter is enabled even if the remote side includes it in the SDP offer. Upon receipt, each coder receives its 'ptime' value in the following precedence: from 'mptime' attribute, from 'ptime' attribute, and then from default value.</p>
EMS: Enable P Time [EnablePtime]	<p>Determines whether the 'ptime' attribute is included in the SDP.</p> <ul style="list-style-type: none"> [0] = Remove the 'ptime' attribute from SDP. [1] = (Default) Include the 'ptime' attribute in SDP.
Web/EMS: 3xx Behavior [3xxBehavior]	<p>Determines the device's behavior regarding call identifiers when a 3xx response is received for an outgoing INVITE request. The device can either use the same call identifiers (Call-ID, To, and From tags) or change them in the new initiated INVITE.</p> <ul style="list-style-type: none"> [0] Forward = (Default) Use different call identifiers for a redirected INVITE message. [1] Redirect = Use the same call identifiers.
Web/EMS: Enable P-Charging Vector [EnablePChargingVector]	<p>Enables the inclusion of the P-Charging-Vector header to all outgoing INVITE messages.</p> <ul style="list-style-type: none"> [0] Disable (default) [1] Enable
Web/EMS: Retry-After Time [RetryAfterTime]	<p>Defines the time (in seconds) used in the Retry-After header when a 503 (Service Unavailable) response is generated by the device.</p> <p>The time range is 0 to 3,600. The default is 0.</p>
Web/EMS: Fake Retry After [sec] [FakeRetryAfter]	<p>Determines whether the device, upon receipt of a SIP 503 response without a Retry-After header, behaves as if the 503 response included a Retry-After header and with the period (in seconds) specified by this parameter.</p> <ul style="list-style-type: none"> [0] Disable (default) Any positive value (in seconds) for defining the period <p>When enabled, this feature allows the device to operate with Proxy servers that do not include the Retry-After SIP header in SIP 503 (Service Unavailable) responses to indicate an unavailable service.</p> <p>The Retry-After header is used with the 503 (Service Unavailable) response to indicate how long the service is expected to be unavailable to the requesting SIP client. The device maintains a list of available proxies, by using the Keep-Alive mechanism. The device checks the availability of proxies by sending SIP OPTIONS every keep-alive timeout to all proxies.</p> <p>If the device receives a SIP 503 response to an INVITE, it also marks that the proxy is out of service for the defined "Retry-After" period.</p>
Web/EMS: Enable P-Associated-URI Header [EnablePAssociatedURIHeader]	<p>Determines the device usage of the P-Associated-URI header. This header can be received in 200 OK responses to REGISTER requests. When enabled, the first URI in the P-Associated-URI header is used in subsequent requests as the From/P-Asserted-Identity headers value.</p>

Parameter	Description
	<ul style="list-style-type: none"> ▪ [0] Disable (default) ▪ [1] Enable <p>Note: P-Associated-URIs in registration responses is handled only if the device is registered per endpoint (using the User Information file).</p>
Web/EMS: Source Number Preference [SourceNumberPreference]	<p>Determines from which SIP header the source (calling) number is obtained in incoming INVITE messages.</p> <ul style="list-style-type: none"> ▪ If not configured (i.e., empty string) or if any string other than "From" or "Pai2" is configured, the calling number is obtained from a specific header using the following logic: <ul style="list-style-type: none"> a. P-Preferred-Identity header. b. If the above header is not present, then the first P-Asserted-Identity header is used. c. If the above header is not present, then the Remote-Party-ID header is used. d. If the above header is not present, then the From header is used. ▪ "From" = The calling number is obtained from the From header. ▪ "Pai2" = The calling number is obtained using the following logic: <ul style="list-style-type: none"> a. If a P-Preferred-Identity header is present, the number is obtained from it. b. If no P-Preferred-Identity header is present and two P-Asserted-Identity headers are present, the number is obtained from the second P-Asserted-Identity header. c. If only one P-Asserted-Identity header is present, the calling number is obtained from it. <p>Notes:</p> <ul style="list-style-type: none"> ▪ The "From" and "Pai2" values are not case-sensitive. ▪ Once a URL is selected, all the calling party parameters are set from this header. If P-Asserted-Identity is selected and the Privacy header is set to 'id', the calling number is assumed restricted.
[SelectSourceHeaderForCalledNumber]	<p>Determines the SIP header used for obtaining the called number (destination) for IP-to-Tel calls.</p> <ul style="list-style-type: none"> ▪ [0] Request-URI header = (Default) Obtains the destination number from the user part of the Request-URI. ▪ [1] To header = Obtains the destination number from the user part of the To header. ▪ [2] P-Called-Party-ID header = Obtains the destination number from the P-Called-Party-ID header.
Web/EMS: Forking Handling Mode [ForkingHandlingMode]	<p>Determines how the device handles the receipt of multiple SIP 18x forking responses for Tel-to-IP calls. The forking 18x response is the response with a different SIP to-tag than the previous 18x response. These responses are typically generated (initiated) by Proxy / Application servers that perform call forking, sending the device's originating INVITE (received from SIP clients) to several destinations, using the same CallID.</p> <ul style="list-style-type: none"> ▪ [0] Parallel handling = (Default) If SIP 18x with SDP is received, the device opens a voice stream according to the received SDP and disregards any subsequently received 18x forking responses (with or without SDP). If the first response is 180 without SDP, the device responds according to the PlayRBTone2TEL parameter and disregards the subsequent forking 18x responses. ▪ [1] Sequential handling = If 18x with SDP is received, the device opens a voice stream according to the received SDP. The device re-

Parameter	Description
	<p>opens the stream according to subsequently received 18x responses with SDP, or plays a ringback tone if 180 response without SDP is received. If the first received response is 180 without SDP, the device responds according to the PlayRBTone2TEL parameter and processes the subsequent 18x forking responses.</p> <p>Note: Regardless of this parameter setting, once a SIP 200 OK response is received, the device uses the RTP information and re-opens the voice stream, if necessary.</p>
Web: Forking Timeout [ForkingTimeOut]	<p>Defines the timeout (in seconds) that is started after the first SIP 2xx response has been received for a User Agent when a Proxy server performs call forking (Proxy server forwards the INVITE to multiple SIP User Agents). The device sends a SIP ACK and BYE in response to any additional SIP 2xx received from the Proxy within this timeout. Once this timeout elapses, the device ignores any subsequent SIP 2xx.</p> <p>The number of supported forking calls per channel is 20. In other words, for an INVITE message, the device can receive up to 20 forking responses from the Proxy server.</p> <p>The valid range is 0 to 30. The default is 30.</p>
Web: Tel2IP Call Forking Mode [Tel2IPCallForkingMode]	<p>Enables Tel-to-IP call forking, whereby a Tel call can be routed to multiple IP destinations.</p> <ul style="list-style-type: none"> ▪ [0] Disable (default) ▪ [1] Enable <p>Note: Once enabled, routing rules must be assigned Forking Groups in the Outbound IP Routing table.</p>
Web/EMS: Enable Reason Header [EnableReasonHeader]	<p>Enables the usage of the SIP Reason header.</p> <ul style="list-style-type: none"> ▪ [0] Disable ▪ [1] Enable (default)
Web/EMS: Gateway Name CLI: gw-name [SIPGatewayName]	<p>Defines a name for the device (e.g., device123.com). This name is used as the host part of the SIP URI in the From header. If not specified, the device's IP address is used instead (default).</p> <p>Notes:</p> <ul style="list-style-type: none"> ▪ Ensure that the parameter value is the one with which the Proxy has been configured with to identify the device. ▪ This parameter can also be configured for an IP Group (in the IP Group table).
[ZeroSDPHandling]	<p>Determines the device's response to an incoming SDP that includes an IP address of 0.0.0.0 in the SDP's Connection Information field (i.e., "c=IN IP4 0.0.0.0").</p> <ul style="list-style-type: none"> ▪ [0] = (Default) Sets the IP address of the outgoing SDP's c= field to 0.0.0.0. ▪ [1] = Sets the IP address of the outgoing SDP c= field to the IP address of the device. If the incoming SDP doesn't contain the "a=inactive" line, the returned SDP contains the "a=recvonly" line.
Web/EMS: Enable Delayed Offer [EnableDelayedOffer]	<p>Determines whether the device sends the initial INVITE message with or without an SDP. Sending the first INVITE without SDP is typically done by clients for obtaining the far-end's full list of capabilities before sending their own offer. (An alternative method for obtaining the list of supported capabilities is by using SIP OPTIONS, which is not supported by every SIP agent.)</p> <ul style="list-style-type: none"> ▪ [0] Disable = (Default) The device sends the initial INVITE message

Parameter	Description
	<p>with an SDP.</p> <ul style="list-style-type: none"> [1] Enable = The device sends the initial INVITE message without an SDP.
[DisableCryptoLifeTimeInSDP]	<p>Enables the device to send "a=crypto" lines without the lifetime parameter in the SDP. For example, if the SDP contains "a=crypto:12 AES_CM_128_HMAC_SHA1_80 inline:hhQe10yZRcRcpIFPkH5xYY9R1de37ogh9G1MpvNp 2^31", it removes the lifetime parameter "2^31".</p> <ul style="list-style-type: none"> [0] Disable (default) [1] Enable
Web/EMS: Enable Contact Restriction [EnableContactRestriction]	<p>Determines whether the device sets the Contact header of outgoing INVITE requests to 'anonymous' for restricted calls.</p> <ul style="list-style-type: none"> [0] Disable (default) [1] Enable
[AnonymousMode]	<p>Determines whether the device's IP address is used as the URI host part instead of "anonymous.invalid" in the INVITE's From header for Tel-to-IP calls.</p> <ul style="list-style-type: none"> [0] = (Default) If the device receives a call from the Tel with blocked caller ID, it sends an INVITE with From: "anonymous"<anonymous@anonymous.invalid> [1] = The device's IP address is used as the URI host part instead of "anonymous.invalid". <p>This parameter may be useful, for example, for service providers who identify their SIP Trunking customers by their source phone number or IP address, reflected in the From header of the SIP INVITE. Therefore, even customers blocking their Caller ID can be identified by the service provider. Typically, if the device receives a call with blocked Caller ID from the PSTN side (e.g., Trunk connected to a PBX), it sends an INVITE to the IP with a From header as follows: From: "anonymous"<anonymous@anonymous.invalid>. This is in accordance with RFC 3325. However, when this parameter is set to 1, the device replaces the "anonymous.invalid" with its IP address.</p>
EMS: P Asserted User Name [PAssertedUserName]	<p>Defines a 'representative number' (up to 50 characters) that is used as the user part of the Request-URI in the P-Asserted-Identity header of an outgoing INVITE for Tel-to-IP calls.</p> <p>The default is null.</p>
EMS: Use URL In Refer To Header [UseAORInReferToHeader]	<p>Defines the source for the SIP URI set in the Refer-To header of outgoing REFER messages.</p> <ul style="list-style-type: none"> [0] = (Default) Use SIP URI from Contact header of the initial call. [1] = Use SIP URI from To/From header of the initial call.
Web: Enable User-Information Usage [EnableUserInfoUsage]	<p>Enables the usage of the User Information, which is loaded to the device in the User Information auxiliary file. For a description on User Information, see 'Loading Auxiliary Files' on page 363.</p> <ul style="list-style-type: none"> [0] Disable (default) [1] Enable <p>Note: For this parameter to take effect, a device reset is required.</p>
[HandleReasonHeader]	<p>Determines whether the device uses the value of the incoming SIP Reason header for Release Reason mapping.</p> <ul style="list-style-type: none"> [0] = Disregard Reason header in incoming SIP messages. [1] = (Default) Use the Reason header value for Release Reason

Parameter	Description
	mapping.
[EnableSilenceSuppInSDP]	<p>Determines the device's behavior upon receipt of SIP Re-INVITE messages that include the SDP's 'silencesupp:off' attribute.</p> <ul style="list-style-type: none"> ▪ [0] = (Default) Disregard the 'silencesupp' attribute. ▪ [1] = Handle incoming Re-INVITE messages that include the 'silencesupp:off' attribute in the SDP as a request to switch to the Voice-Band-Data (VBD) mode. In addition, the device includes the attribute 'a=silencesupp:off' in its SDP offer. <p>Note: This parameter is applicable only if the G.711 coder is used.</p>
[EnableRport]	<p>Enables the usage of the 'rport' parameter in the Via header.</p> <ul style="list-style-type: none"> ▪ [0] = Disabled (default) ▪ [1] = Enabled <p>The device adds an 'rport' parameter to the Via header of each outgoing SIP message. The first Proxy that receives this message sets the 'rport' value of the response to the actual port from where the request was received. This method is used, for example, to enable the device to identify its port mapping outside a NAT.</p> <p>If the Via header doesn't include the 'rport' parameter, the destination port of the response is obtained from the host part of the Via header. If the Via header includes the 'rport' parameter without a port value, the destination port of the response is the source port of the incoming request.</p> <p>If the Via header includes 'rport' with a port value (e.g., rport=1001), the destination port of the response is the port indicated in the 'rport' parameter.</p>
EMS: X Channel Header [XChannelHeader]	<p>Determines whether the SIP X-Channel header is added to SIP messages for providing information on the physical channel on which the call is received or placed.</p> <ul style="list-style-type: none"> ▪ [0] Disable = (Default) X-Channel header is not used. ▪ [1] Enable = X-Channel header is generated by the device and sent in INVITE messages and 180, 183, and 200 OK SIP responses. The header includes the channel, and the device's IP address. For example, 'x-channel: DS/DS1-8;IP=192.168.13.1', where: <ul style="list-style-type: none"> ✓ 'DS/DS-1' is a constant string ✓ '8' is the channel ✓ 'IP=192.168.13.1' is the device's IP address
[EnableRekeyAfter181]	<p>Enables the device to send a re-INVITE with a new (different) SRTP key (in the SDP) if a SIP 181 response is received ("call is being forwarded"). The re-INVITE is sent immediately upon receipt of the 200 OK (when the call is answered).</p> <ul style="list-style-type: none"> ▪ [0] = Disable (default) ▪ [1] = Enable <p>Note: This parameter is applicable only if SRTP is used.</p>
[NumberOfActiveDialogs]	<p>Defines the maximum number of concurrent, outgoing SIP REGISTER dialogs. This parameter is used to control the registration rate. The valid range is 1 to 20. The default is 20.</p> <p>Notes:</p> <ul style="list-style-type: none"> ▪ Once a 200 OK is received in response to a REGISTER message, the REGISTER message is not considered in this maximum count

Parameter	Description
	<p>limit.</p> <ul style="list-style-type: none"> This parameter applies only to outgoing REGISTER messages (i.e., incoming is unlimited).
Web/EMS: Default Release Cause [DefaultReleaseCause]	<p>Defines the default Release Cause (sent to IP) for IP-to-Tel calls when the device initiates a call release and an explicit matching cause for this release is not found.</p> <p>The default release cause is NO_ROUTE_TO_DESTINATION (3). Other common values include NO_CIRCUIT_AVAILABLE (34), DESTINATION_OUT_OF_ORDER (27), etc.</p> <p>Notes:</p> <ul style="list-style-type: none"> The default release cause is described in the Q.931 notation and is translated to corresponding SIP 40x or 50x values (e.g., 3 to SIP 404, and 34 to SIP 503). For a list of SIP responses-Q.931 release cause mapping, see Alternative Routing to Trunk upon Q.931 Call Release Cause Code.
Web: Enable Microsoft Extension [EnableMicrosoftExt]	<p>Enables the modification of the called and calling number for numbers received with Microsoft's proprietary "ext=xxx" parameter in the SIP INVITE URI user part. Microsoft Office Communications Server sometimes uses this proprietary parameter to indicate the extension number of the called or calling party.</p> <ul style="list-style-type: none"> [0] Disable (default) [1] Enable <p>For example, if a calling party makes a call to telephone number 622125519100 Ext. 104, the device receives the SIP INVITE (from Microsoft's application) with the URI user part as INVITE sip:622125519100;ext=104@10.1.1.10 (or INVITE tel:622125519100;ext=104). If the parameter EnableMicrosoftExt is enabled, the device modifies the called number by adding an "e" as the prefix, removing the "ext=" parameter, and adding the extension number as the suffix (e.g., e622125519100104). Once modified, the device can then manipulate the number further, using the Number Manipulation tables to leave only the last 3 digits (for example) for sending to a PBX.</p>
EMS: Use SIP URI For Diversion Header [UseSIPURIForDiversionHeader]	<p>Defines the URI format in the SIP Diversion header.</p> <ul style="list-style-type: none"> [0] = 'tel:' (default) [1] = 'sip:'
[TimeoutBetween100And18x]	<p>Defines the timeout (in msec) between receiving a 100 Trying response and a subsequent 18x response. If a 18x response is not received within this timeout period, the call is disconnected.</p> <p>The valid range is 0 to 180,000 (i.e., 3 minutes). The default is 32000 (i.e., 32 sec).</p>
[IgnoreRemoteSDPMKI]	<p>Determines whether the device ignores the Master Key Identifier (MKI) if present in the SDP received from the remote side.</p> <ul style="list-style-type: none"> [0] Disable (default) [1] Enable
Web: Comfort Noise Generation Negotiation EMS: Comfort Noise Generation [ComfortNoiseNegotiation]	<p>Enables negotiation and usage of Comfort Noise (CN).</p> <ul style="list-style-type: none"> [0] Disable [1] Enable (default) <p>The use of CN is indicated by including a payload type for CN on the media description line of the SDP. The device can use CN with a codec whose RTP time stamp clock rate is 8,000 Hz (G.711/G.726). The static</p>

Parameter	Description
	<p>payload type 13 is used. The use of CN is negotiated between sides. Therefore, if the remote side doesn't support CN, it is not used. Regardless of the device's settings, it always attempts to adapt to the remote SIP UA's request for CNG, as described below.</p> <p>To determine CNG support, the device uses the ComfortNoiseNegotiation parameter and the codec's SCE (silence suppression setting) using the CodersGroup parameter.</p> <p>If the ComfortNoiseNegotiation parameter is enabled, then the following occurs:</p> <ul style="list-style-type: none"> ▪ If the device is the initiator, it sends a "CN" in the SDP only if the SCE of the codec is enabled. If the remote UA responds with a "CN" in the SDP, then CNG occurs; otherwise, CNG does not occur. ▪ If the device is the receiver and the remote SIP UA does not send a "CN" in the SDP, then no CNG occurs. If the remote side sends a "CN", the device attempts to be compatible with the remote side and even if the codec's SCE is disabled, CNG occurs. <p>If the ComfortNoiseNegotiation parameter is disabled, then the device does not send "CN" in the SDP. However, if the codec's SCE is enabled, then CNG occurs.</p>
[SDPEcanFormat]	<p>Defines the echo canceller format in the outgoing SDP. The 'ecan' attribute is used in the SDP to indicate the use of echo cancellation.</p> <ul style="list-style-type: none"> ▪ [0] = (Default) The 'ecan' attribute appears on the 'a=gpmid' line. ▪ [1] = The 'ecan' attribute appears as a separate attribute. ▪ [2] = The 'ecan' attribute is not included in the SDP. ▪ [3] = The 'ecan' attribute and the 'vbd' parameter are not included in the SDP. <p>Note: This parameter is applicable only when the IsFaxUsed parameter is set to 2, and for re-INVITE messages generated by the device as result of modem or fax tone detection.</p>
Web/EMS: First Call Ringback Tone ID [FirstCallIRBTId]	<p>Defines the index of the first ringback tone in the CPT file. This option enables an Application server to request the device to play a distinctive ringback tone to the calling party according to the destination of the call. The tone is played according to the Alert-Info header received in the 180 Ringing SIP response (the value of the Alert-Info header is added to the value of this parameter).</p> <p>The valid range is -1 to 1,000. The default is -1 (i.e., play standard ringback tone).</p> <p>Notes:</p> <ul style="list-style-type: none"> ▪ It is assumed that all ringback tones are defined in sequence in the CPT file. ▪ In case of an MLPP call, the device uses the value of this parameter plus 1 as the index of the ringback tone in the CPT file (e.g., if this value is set to 1, then the index is 2, i.e., 1 + 1).
Web: PSTN Alert Timeout EMS: Trunk PSTN Alert Timeout [PSTNAlertTimeout]	<p>The valid value range is 1 to 600 (in seconds). The default is 180.</p>
Web/EMS: RTP Only Mode [RTPOnlyMode]	<p>Enables the device to send and receive RTP packets to and from remote endpoints without the need to establish a SIP session. The remote IP address is determined according to the Outbound IP Routing table (Prefix parameter). The port is the same port as the local RTP port</p>

Parameter	Description
	<p>(configured by the BaseUDPPort parameter and the channel on which the call is received).</p> <ul style="list-style-type: none"> [0] Disable (default) [1] Transmit & Receive = Send and receive RTP packets. [2] Transmit Only= Send RTP packets only. [3] Receive Only= Receive RTP packets only. <p>Notes:</p> <ul style="list-style-type: none"> To configure the RTP Only mode per trunk, use the RTPOnlyModeForTrunk_ID parameter. If per trunk configuration (using the RTPOnlyModeForTrunk_ID parameter) is set to a value other than the default, the RTPOnlyMode parameter value is ignored.
Web/EMS: Media IP Version Preference [MediaIPVersionPreference]	<p>Determines the preferred RTP media IP addressing version for outgoing SIP calls. This is indicated in the "c=" field (Connection Information) of the SDP.</p> <ul style="list-style-type: none"> [0] Only IPv4 = (Default) offer includes only IPv4 media IP addresses. [1] Only IPv6 = offer includes only IPv6 media IPs addresses. [2] Prefer IPv4 = offer includes both IPv4 and IPv6 media IP addresses, but the first media is IPv4. [3] Prefer IPv6 = offer includes both IPv4 and IPv6 media IP addresses, but the first media is IPv6. <p>Notes:</p> <ul style="list-style-type: none"> This parameter is applicable only when the device offers an SDP. The IP addressing version is determined according to the first SDP "m=" field. This parameter can be configured per IP Profile, using the parameter IPProfile (see Configuring IP Profiles on page 221).
Retransmission Parameters	
Web: SIP T1 Retransmission Timer [msec] EMS: T1 RTX [SipT1Rtx]	<p>Defines the time interval (in msec) between the first transmission of a SIP message and the first retransmission of the same message.</p> <p>The default is 500.</p> <p>Note: The time interval between subsequent retransmissions of the same SIP message starts with SipT1Rtx. For INVITE requests, it is multiplied by two for each new retransmitted message. For all other SIP messages, it is multiplied by two until SipT2Rtx. For example, assuming SipT1Rtx = 500 and SipT2Rtx = 4000:</p> <ul style="list-style-type: none"> The first retransmission is sent after 500 msec. The second retransmission is sent after 1000 (2*500) msec. The third retransmission is sent after 2000 (2*1000) msec. The fourth retransmission and subsequent retransmissions until SIPMaxRtx are sent after 4000 (2*2000) msec.
Web: SIP T2 Retransmission Timer [msec] EMS: T2 RTX [SipT2Rtx]	<p>Defines the maximum interval (in msec) between retransmissions of SIP messages (except for INVITE requests).</p> <p>The default is 4000.</p> <p>Note: The time interval between subsequent retransmissions of the same SIP message starts with SipT1Rtx and is multiplied by two until SipT2Rtx.</p>

Parameter	Description
Web: SIP Maximum RTX EMS: Max RTX [SIPMaxRtx]	Defines the maximum number of UDP transmissions of SIP messages (first transmission plus retransmissions). The range is 1 to 30. The default is 7.
Web: Number of RTX Before Hot-Swap EMS: Proxy Hot Swap Rtx [HotSwapRtx]	Defines the number of retransmitted INVITE/REGISTER messages before the call is routed (hot swap) to another Proxy/Registrar. The valid range is 1 to 30. The default is 3. Note: This parameter is also used for alternative routing using the SBC IP-to-IP Routing table. If a domain name in the table is resolved into two IP addresses, and if there is no response for HotSwapRtx retransmissions to the INVITE message that is sent to the first IP address, the device immediately initiates a call to the second IP address.
SIP Message Manipulations Table	
Web: Message Manipulations EMS: Message Manipulations [MessageManipulations]	This table parameter defines manipulation rules for SIP header messages. The format of this parameter is as follows: [MessageManipulations] FORMAT MessageManipulations_Index = MessageManipulations_ManSetID, MessageManipulations_MessageType, MessageManipulations_Condition, MessageManipulations_ActionSubject, MessageManipulations_ActionType, MessageManipulations_ActionValue, MessageManipulations_RowRole; [MessageManipulations] For example, the below configuration changes the user part of the SIP From header to 200: MessageManipulations 1 = 0, Invite.Request, , Header.From.Url.User, 2, 200, 0; Note: For a detailed description of this table, see 'Configuring SIP Message Manipulation' on page 208.
Message Policy Table	
Web: Message Policy Table [MessagePolicy]	This table parameter configures SIP message policy rules for blocking (blacklist) unwanted incoming SIP messages or allowing (whitelist) receipt of desired messages. The format of this parameter is as follows: [MessagePolicy] FORMAT MessagePolicy_Index = MessagePolicy_Policy, MessagePolicy_MaxMessageLength, MessagePolicy_MaxHeaderLength, MessagePolicy_MaxBodyLength, MessagePolicy_MaxNumHeaders, MessagePolicy_MaxNumBodies, MessagePolicy_SendRejection, MessagePolicy_MethodListType, MessagePolicy_MethodList, MessagePolicy_BodyListType, MessagePolicy_BodyList; [MessagePolicy] Note: For a detailed description of this table, see 'Configuring SIP Message Policy Rules'.

42.9 Coders and Profile Parameters

The profile parameters are described in the table below.

Table 42-31: Profile Parameters

Parameter	Description
Coders Table / Coders Groups Table	
Web: Coders Table/Coder Group Settings EMS: Coders Group [CodersGroup0] [CodersGroup1] [CodersGroup2] [CodersGroup3] [CodersGroup4] [CodersGroup5] [CodersGroup6] [CodersGroup7] [CodersGroup8] [CodersGroup9]	<p>This table parameter defines the device's coders. Each group can consist of up to 10 coders. The first Coders Group is the default coder list and the default Coders Group.</p> <p>The format of this parameter is as follows:</p> <p>[CodersGroup<0-9>] FORMAT CodersGroup0_Index = CodersGroup0_Name, CodersGroup0_pTime, CodersGroup0_rate, CodersGroup0_PayloadType, CodersGroup0_Sce; [\CodersGroup<0-9>]</p> <p>For example, below are defined two Coders Groups (0 and 1):</p> <p>[CodersGroup0] FORMAT CodersGroup0_Index = CodersGroup0_Name, CodersGroup0_pTime, CodersGroup0_rate, CodersGroup0_PayloadType, CodersGroup0_Sce; CodersGroup0 0 = g711Alaw64k, 20, 0, 255, 0; CodersGroup0 1 = eg711Ulaw, 10, 0, 71, 0; CodersGroup0 2 = eg711Ulaw, 10, 0, 71, 0; [\CodersGroup0]</p> <p>[CodersGroup1] FORMAT CodersGroup1_Index = CodersGroup1_Name, CodersGroup1_pTime, CodersGroup1_rate, CodersGroup1_PayloadType, CodersGroup1_Sce; CodersGroup1 0 = Transparent, 20, 0, 56, 0; CodersGroup1 1 = g726, 20, 0, 23, 0; [\CodersGroup1]</p> <p>Notes:</p> <ul style="list-style-type: none"> For a list of supported coders and a detailed description of this table, see Configuring Coders on page 217. The coder name is case-sensitive.
IP Profile Table	
Web: IP Profile Settings EMS: Protocol Definition > IP Profile [IPProfile]	<p>This table parameter configures the IP Profile table. Each IP Profile ID includes a set of parameters (which are typically configured separately using their individual "global" parameters). You can later assign these IP Profiles to outbound IP routing rules (Prefix parameter), inbound IP routing rules and IP Groups.</p> <p>The format of this parameter is as follows:</p> <p>[IPProfile] FORMAT IpProfile_Index = IpProfile_ProfileName, IpProfile_IpPreference, IpProfile_CodersGroupID, IpProfile_IsFaxUsed, IpProfile_JitterBufMinDelay, IpProfile_JitterBufOptFactor, IpProfile_IPDiffServ, IpProfile_SigIPDiffServ, IpProfile_SCE, IpProfile_RTPRedundancyDepth, IpProfile_RemoteBaseUDPPort, IpProfile_CNGmode, IpProfile_VxxTransportType, IpProfile_NSEMode, IpProfile_IsDTMFUsed, IpProfile_PlayRBTone2IP, IpProfile_EnableEarlyMedia, IpProfile_ProgressIndicator2IP, IpProfile_EnableEchoCanceller, IpProfile_CopyDest2RedirectNumber,</p>

Parameter	Description
	<p> IpProfile_MediaSecurityBehaviour, IpProfile_CallLimit, IpProfile_DisconnectOnBrokenConnection, IpProfile_FirstTxDtmfOption, IpProfile_SecondTxDtmfOption, IpProfile_RxDTMFOption, IpProfile_EnableHold, IpProfile_InputGain, IpProfile_VoiceVolume, IpProfile_AddIEInSetup, IpProfile_SBCExtensionCodersGroupID, IpProfile_MediaIPVersionPreference, IpProfile_TranscodingMode, IpProfile_SBCAllowedCodersGroupID, IpProfile_SBCAllowedCodersMode, IpProfile_SBCMediaSecurityBehaviour, IpProfile_SBCRFC2833Behavior, IpProfile_SBCAlternativeDTMFMethod, IpProfile_SBCAssertIdentity, IpProfile_AMDSensitivityParameterSuit, IpProfile_AMDSensitivityLevel, IpProfile_AMDMaxGreetingTime, IpProfile_AMDMaxPostSilenceGreetingTime, IpProfile_SBCDiversionsMode, IpProfile_SBCHistoryInfoMode, IpProfile_EnableQSIGTunneling, IpProfile_SBCFaxCodersGroupID, IpProfile_SBCFaxBehavior, IpProfile_SBCFaxOfferMode, IpProfile_SBCFaxAnswerMode, IpProfile_SbcPrackMode, IpProfile_SBCSessionExpiresMode, IpProfile_SBCRemoteUpdateSupport, IpProfile_SBCRemoteReinviteSupport, IpProfile_SBCRemoteDelayedOfferSupport, IpProfile_SBCRemoteReferBehavior, IpProfile_SBCRemote3xxBehavior, IpProfile_SBCRemoteMultiple18xSupport, IpProfile_SBCRemoteEarlyMediaResponseType, IpProfile_SBCRemoteEarlyMediaSupport, IpProfile_EnableSymmetricMKI, IpProfile_MKISize, IpProfile_SBCEnforceMKISize, IpProfile_SBCRemoteEarlyMediaRTP, IpProfile_SBCRemoteSupportsRFC3960, IpProfile_SBCRemoteCanPlayRingback, IpProfile_EnableEarly183, IpProfile_EarlyAnswerTimeout, IpProfile_SBC2833DTMFPayloadType, IpProfile_SBCUserRegistrationTime, IpProfile_ResetSRTPStateUponRekey, IpProfile_AmdMode, IpProfile_SBCReliableHeldToneSource, IpProfile_SBCPlayHeldTone, IpProfile_SBCRemoteHoldFormat, IpProfile_DelayTimeForInvite; [IPProfile] </p> <p>Note: For a description of this table, see 'Configuring IP Profiles' on page 221.</p>

42.10 Channel Parameters

This subsection describes the device's channel parameters.

42.10.1 Voice Parameters

The voice parameters are described in the table below.

Table 42-32: Voice Parameters

Parameter	Description
Web/EMS: Input Gain [InputGain]	<p>Defines the pulse-code modulation (PCM) input gain control (in decibels). This parameter sets the level for the received (-to-IP) signal.</p> <p>The valid range is -32 to 31 dB. The default is 0 dB.</p> <p>Note: This parameter can also be configured in an IP Profile and/or a Tel Profile.</p>
Web: Voice Volume EMS: Volume (dB) [VoiceVolume]	<p>Defines the voice gain control (in decibels). This parameter sets the level for the transmitted (IP-to-) signal.</p> <p>The valid range is -32 to 31 dB. The default is 0 dB.</p> <p>Note: This parameter can also be configured in an IP Profile and/or a Tel Profile.</p>
EMS: Payload Format [VoicePayloadFormat]	<p>Determines the bit ordering of the G.726/G.727 voice payload format.</p> <ul style="list-style-type: none"> [0] = (Default) Little Endian [1] = Big Endian <p>Note: To ensure high voice quality when using G.726/G.727, both communicating ends should use the same endianness format. Therefore, when the device communicates with a third-party entity that uses the G.726/G.727 voice coder and voice quality is poor, change the settings of this parameter (between Big Endian and Little Endian).</p>
Web: MF Transport Type [MFTransportType]	Currently, not supported.
Web: Enable Answer Detector [EnableAnswerDetector]	Currently, not supported.
Web: Answer Detector Activity Delay [AnswerDetectorActivityDelay]	<p>Defines the time (in 100-msec resolution) between activating the Answer Detector and the time that the detector actually starts to operate.</p> <p>The valid range is 0 to 1023. The default is 0.</p>
Web: Answer Detector Silence Time [AnswerDetectorSilenceTime]	Currently, not supported.
Web: Answer Detector Redirection [AnswerDetectorRedirection]	Currently, not supported.
Web: Answer Detector Sensitivity	Defines the Answer Detector sensitivity.

Parameter	Description
EMS: Sensitivity [AnswerDetectorSensitivity]	The range is 0 (most sensitive) to 2 (least sensitive). The default is 0.
Web: Silence Suppression EMS: Silence Compression Mode [EnableSilenceCompression]	<p>Determines the Silence Suppression support. Silence Suppression is a method for conserving bandwidth on VoIP calls by not sending packets when silence is detected.</p> <ul style="list-style-type: none"> ▪ [0] Disable = (Default) Silence Suppression is disabled. ▪ [1] Enable = Silence Suppression is enabled. ▪ [2] Enable without Adaptation = A single silence packet is sent during a silence period (applicable only to G.729). <p>Note: If the selected coder is G.729, the value of the 'annexb' parameter of the fmtp attribute in the SDP is determined by the following rules:</p> <ul style="list-style-type: none"> ▪ If EnableSilenceCompression is 0: 'annexb=no'. ▪ If EnableSilenceCompression is 1: 'annexb=yes'. ▪ If EnableSilenceCompression is 2 and IsCiscoSCEMode is 0: 'annexb=yes'. ▪ If EnableSilenceCompression is 2 and IsCiscoSCEMode is 1: 'annexb=no'. <p>Note: This parameter can also be configured in an IP Profile.</p>
Web: Echo Canceller EMS: Echo Canceller Enable [EnableEchoCanceller]	<p>Enables echo cancellation (i.e., echo from voice calls is removed).</p> <ul style="list-style-type: none"> ▪ [0] Disable ▪ [1] Enable (default) <p>Note: This parameter can also be configured in an IP Profile and/or a Tel Profile.</p>
Web: Network Echo Suppressor Enable [AcousticEchoSuppressorSupport]	<p>Enables the network Acoustic Echo Suppressor feature on SBC calls. This feature removes echoes and sends only the near-end's desired speech signal to the network (i.e., to the far-end party).</p> <ul style="list-style-type: none"> ▪ [0] Disable (default) ▪ [1] Enable <p>Note: For this parameter to take effect, a device reset is required.</p>
Web: Echo Canceller Type [EchoCancellerType]	<p>Defines the echo canceller type.</p> <ul style="list-style-type: none"> ▪ [0] Line echo canceller = (Default) Echo canceller for Tel side. ▪ [1] Acoustic Echo suppressor - netw = Echo canceller for IP side.
Web: Attenuation Intensity [AcousticEchoSuppAttenuationIntensity]	<p>Defines the acoustic echo suppressor signals identified as echo attenuation intensity.</p> <p>The valid range is 0 to 3. The default is 0.</p>
Web: Max ERL Threshold - DB [AcousticEchoSuppMaxERLThreshold]	<p>Defines the acoustic echo suppressor maximum ratio between signal level and returned echo from the phone (in decibels).</p> <p>The valid range is 0 to 60. The default is 10.</p>
Web: Min Reference Delay x10 msec [AcousticEchoSuppMinRefDelayx10ms]	<p>Defines the acoustic echo suppressor minimum reference delay (in 10-ms units).</p> <p>The valid range is 0 to 40. The default is 0.</p>

Parameter	Description
Web: Max Reference Delay x10 msec [AcousticEchoSuppMaxRefDelayx10ms]	Defines the acoustic echo suppressor maximum reference delay (in 10-ms units). The valid range is 0 to 40. The default is 40 (i.e., 40 x 10 = 400 ms).
EMS: Echo Canceller Hybrid Loss [ECHybridLoss]	Defines the four-wire to two-wire worst-case Hybrid loss, the ratio between the signal level sent to the hybrid and the echo level returning from the hybrid. <ul style="list-style-type: none"> [0] = (Default) 6 dB [1] = N/A [2] = 0 dB [3] = 3 dB
EMS: ECN Ip Mode [ECNLPMODE]	Defines the echo cancellation Non-Linear Processing (NLP) mode. <ul style="list-style-type: none"> [0] = (Default) NLP adapts according to echo changes [1] = Disables NLP [2] = Silence output NLP Note: This parameter can also be configured in a Tel Profile.
[EchoCancellerAggressiveNLP]	Enables the Aggressive NLP at the first 0.5 second of the call. <ul style="list-style-type: none"> [0] = Disable [1] = (Default) Enable. The echo is removed only in the first half of a second of the incoming IP signal. Note: For this parameter to take effect, a device reset is required.
[RTPSIDCoeffNum]	Defines the number of spectral coefficients added to an SID packet being sent according to RFC 3389. The valid values are [0] (default), [4] , [6] , [8] and [10] .

42.10.2 Coder Parameters

The coder parameters are described in the table below.

Table 42-33: Coder Parameters

Parameter	Description
Web: AMR Payload Format [AmrOctetAlignedEnable]	Defines the AMR payload format type. <ul style="list-style-type: none"> [0] Bandwidth Efficient [1] Octet Aligned (default)
Silk Tx Inband FEC [SilkTxInbandFEC]	Enables forward error correction (FEC) for the SILK coder. <ul style="list-style-type: none"> [0] Disable (default) [1] Enable
Silk Max Average Bit Rate [SilkMaxAverageBitRate]	Defines the maximum average bit rate for the SILK coder. The valid value range is 5000 to 30000. The default is 16000. Note: The SILK coder is Skype's default audio codec used for Skype-to-Skype calls.

42.10.3 DTMF Parameters

The dual-tone multi-frequency (DTMF) parameters are described in the table below.

Table 42-34: DTMF Parameters

Parameter	Description
Web/EMS: DTMF Transport Type [DTMFTransportType]	<p>Determines the DTMF transport type.</p> <ul style="list-style-type: none"> ▪ [0] Mute DTMF = DTMF digits are removed from the voice stream and are not relayed to remote side. ▪ [2] Transparent DTMF = DTMF digits remain in the voice stream. ▪ [3] RFC 2833 Relay DTMF = (Default) DTMF digits are removed from the voice stream and are relayed to remote side according to RFC 2833. ▪ [7] RFC 2833 Relay Decoder Mute = DTMF digits are sent according to RFC 2833 and muted when received. <p>Note: This parameter is automatically updated if the parameters TxDTMFOption or RxDTMFOption are configured.</p>
Web: DTMF Volume (-31 to 0 dB) EMS: DTMF Volume (dBm) [DTMFVolume]	<p>Defines the DTMF gain control value (in decibels) to the side. The valid range is -31 to 0 dB. The default is -11 dB.</p> <p>Note: This parameter can also be configured in a Tel Profile.</p>
Web: DTMF Generation Twist EMS: DTMF Twist Control [DTMFGenerationTwist]	<p>Defines the range (in decibels) between the high and low frequency components in the DTMF signal. Positive decibel values cause the higher frequency component to be stronger than the lower one. Negative values cause the opposite effect. For any parameter value, both components change so that their average is constant. The valid range is -10 to 10 dB. The default is 0 dB.</p> <p>Note: For this parameter to take effect, a device reset is required.</p>
EMS: DTMF Inter Interval (msec) [DTMFInterDigitInterval]	<p>Defines the time (in msec) between generated DTMF digits to PSTN side (if TxDTMFOption = 1, 2 or 3). The default is 100 msec. The valid range is 0 to 32767.</p>
EMS: DTMF Length (msec) [DTMFDigitLength]	<p>Defines the time (in msec) for generating DTMF tones to the PSTN side (if TxDTMFOption = 1, 2 or 3). It also configures the duration that is sent in INFO (Cisco) messages. The valid range is 0 to 32767. The default is 100.</p>
EMS: Rx DTMF Relay Hang Over Time (msec) [RxDTMFHangOverTime]	<p>Defines the Voice Silence time (in msec) after playing DTMF or MF digits to the Tel/PSTN side that arrive as Relay from the IP side. Valid range is 0 to 2,000 msec. The default is 1,000 msec.</p>
EMS: Tx DTMF Relay Hang Over Time (msec) [TxDTMFHangOverTime]	<p>Defines the Voice Silence time (in msec) after detecting the end of DTMF or MF digits at the Tel/PSTN side when the DTMF Transport Type is either Relay or Mute. Valid range is 0 to 2,000 msec. The default is 1,000 msec.</p>
Web/EMS: NTE Max Duration [NTEMaxDuration]	<p>Defines the maximum time for sending Named Telephony Events / NTEs (RFC 4733/2833 DTMF relay) to the IP side, regardless of the DTMF signal duration on the TDM side. The range is -1 to 200,000,000 msec. The default is -1 (i.e., NTE stops only upon detection of an End event).</p>

42.10.4 RTP, RTCP and T.38 Parameters

The RTP, RTCP and T.38 parameters are described in the table below.

Table 42-35: RTP/RTCP and T.38 Parameters

Parameter	Description
Web: Dynamic Jitter Buffer Minimum Delay EMS: Minimal Delay (dB) [DJBufMinDelay]	<p>Defines the minimum delay (in msec) for the Dynamic Jitter Buffer.</p> <p>The valid range is 0 to 150. The default delay is 10.</p> <p>Notes:</p> <ul style="list-style-type: none"> This parameter can also be configured in an IP Profile and/or a Tel Profile. For more information on Jitter Buffer, see Dynamic Jitter Buffer Operation on page 152.
Web: Dynamic Jitter Buffer Optimization Factor EMS: Opt Factor [DJBufOptFactor]	<p>Defines the Dynamic Jitter Buffer frame error/delay optimization factor.</p> <p>The valid range is 0 to 12. The default factor is 10.</p> <p>Notes:</p> <ul style="list-style-type: none"> For data (fax and modem) calls, set this parameter to 12. This parameter can also be configured in an IP Profile and/or a Tel Profile. For more information on Jitter Buffer, see Dynamic Jitter Buffer Operation on page 152.
Web: RTP Redundancy Depth EMS: Redundancy Depth [RTPRedundancyDepth]	<p>Enables the device to generate RFC 2198 redundant packets. This can be used for packet loss where the missing information (audio) can be reconstructed at the receiver's end from the redundant data that arrives in subsequent packets. This is required, for example, in wireless networks where a high percentage (up to 50%) of packet loss can be experienced.</p> <ul style="list-style-type: none"> [0] 0 = (Default) Disable. [1] 1 = Enable - previous voice payload packet is added to current packet. <p>Notes:</p> <ul style="list-style-type: none"> When enabled, you can configure the payload type, using the RFC2198PayloadType parameter. The RTP redundancy dynamic payload type can be included in the SDP, by using the EnableRTPRedundancyNegotiation parameter. This parameter can also be configured in an IP Profile.
Web: Enable RTP Redundancy Negotiation [EnableRTPRedundancyNegotiation]	<p>Enables the device to include the RTP redundancy dynamic payload type in the SDP, according to RFC 2198.</p> <ul style="list-style-type: none"> [0] Disable (default) [1] Enable <p>When enabled, the device includes in the SDP message the RTP payload type "RED" and the payload type configured by the parameter RFC2198PayloadType.</p> <pre>a=rtpmap:<PT> RED/8000</pre> <p>Where <PT> is the payload type as defined by</p>

Parameter	Description
	<p>RFC2198PayloadType. The device sends the INVITE message with "a=rtpmap:<PT> RED/8000" and responds with a 18x/200 OK and "a=rtpmap:<PT> RED/8000" in the SDP.</p> <p>Notes:</p> <ul style="list-style-type: none"> For this feature to be functional, you must also set the parameter RTPRedundancyDepth to 1 (i.e., enabled). Currently, the negotiation of "RED" payload type is not supported and therefore, it should be configured to the same PT value for both parties.
Web: RFC 2198 Payload Type EMS: Redundancy Payload Type [RFC2198PayloadType]	<p>Defines the RTP redundancy packet payload type according to RFC 2198.</p> <p>The range is 96 to 127. The default is 104.</p> <p>Note: This parameter is applicable only if the parameter RTPRedundancyDepth is set to 1.</p>
Web: Packing Factor EMS: Packetization Factor [RTPPackFactor]	N/A. Controlled internally by the device according to the selected coder.
Web/EMS: Basic RTP Packet Interval [BasicRTPPacketInterval]	N/A. Controlled internally by the device according to the selected coder.
Web: RTP Directional Control [RTPDirectionControl]	N/A. Controlled internally by the device according to the selected coder.
Web/EMS: RFC 2833 TX Payload Type [RFC2833TxPayloadType]	<p>Defines the Tx RFC 2833 DTMF relay dynamic payload type.</p> <p>The valid range is 96 to 99, and 106 to 127. The default is 96. The 100, 102 to 105 range is allocated for proprietary usage.</p> <p>Notes:</p> <ul style="list-style-type: none"> Certain vendors (e.g., Cisco) use payload type 101 for RFC 2833. When RFC 2833 payload type negotiation is used (i.e., the parameter TxDTMFOption is set to 4), this payload type is used for the received DTMF packets. If negotiation isn't used, this payload type is used for receive and for transmit.
Web/EMS: RFC 2833 RX Payload Type [RFC2833RxPayloadType]	<p>Defines the Rx RFC 2833 DTMF relay dynamic payload type.</p> <p>The valid range is 96 to 99, and 106 to 127. The default is 96. The 100, 102 to 105 range is allocated for proprietary usage.</p> <p>Notes:</p> <ul style="list-style-type: none"> Certain vendors (e.g., Cisco) use payload type 101 for RFC 2833. When RFC 2833 payload type negotiation is used (i.e., the parameter TxDTMFOption is set to 4), this payload type is used for the received DTMF packets. If negotiation isn't used, this payload type is used for receive and for transmit.

Parameter	Description
[EnableDetectRemoteMACChange]	<p>Determines whether the device changes the RTP packets according to the MAC address of received RTP packets and according to Gratuitous Address Resolution Protocol (GARP) messages.</p> <ul style="list-style-type: none"> [0] = Nothing is changed. [1] = If the device receives RTP packets with a different source MAC address (than the MAC address of the transmitted RTP packets), then it sends RTP packets to this MAC address and removes this IP entry from the device's ARP cache table. [2] = (Default) The device uses the received GARP packets to change the MAC address of the transmitted RTP packets. [3] = Options 1 and 2 are used. <p>Notes:</p> <ul style="list-style-type: none"> For this parameter to take effect, a device reset is required. If the device is located in a network subnet which is connected to other gateways using a router that uses Virtual Router Redundancy Protocol (VRRP) for redundancy, then set this parameter to 0 or 2.
Web: RTP Base UDP Port EMS: Base UDP Port [BaseUDPport]	<p>Defines the lower boundary of the UDP port used for RTP, RTCP (RTP port + 1) and T.38 (RTP port + 2). For example, if the Base UDP Port is set to 6000, then one channel may use the ports RTP 6000, RTCP 6001, and T.38 6002, while another channel may use RTP 6010, RTCP 6011, and T.38 6012, and so on.</p> <p>The range of possible UDP ports is 6,000 to 64,000. The default base UDP port is 6000.</p> <p>Once this parameter is configured, the UDP port range (lower to upper boundary) is calculated as follows:</p> <ul style="list-style-type: none"> BaseUDPport to BaseUDPport + 4030*10 <p>Notes:</p> <ul style="list-style-type: none"> For this parameter to take effect, a device reset is required. The UDP ports are allocated randomly to channels. You can define a UDP port range per Media Realm (see Configuring Media Realms on page 157). If RTP Base UDP Port is not a factor of 10, the following message is generated: 'invalid local RTP port'.
EMS: No Op Enable [NoOpEnable]	<p>Enables the transmission of RTP or T.38 No-Op packets.</p> <ul style="list-style-type: none"> [0] = Disable (default) [1] = Enable <p>This mechanism ensures that the NAT binding remains open during RTP or T.38 silence periods.</p>
EMS: No Op Interval [NoOpInterval]	<p>Defines the time interval in which RTP or T.38 No-Op packets are sent in the case of silence (no RTP/T.38 traffic) when No-Op packet transmission is enabled.</p> <p>The valid range is 20 to 65,000 msec. The default is 10,000.</p> <p>Note: To enable No-Op packet transmission, use the NoOpEnable parameter.</p>

Parameter	Description
EMS: No Op Payload Type [RTPNoOpPayloadType]	Defines the payload type of No-Op packets. The valid range is 96 to 127 (for the range of Dynamic RTP Payload Type for all types of non hard-coded RTP Payload types, refer to RFC 3551). The default is 120. Note: When defining this parameter, ensure that it doesn't cause collision with other payload types.
[RTCPActivationMode]	Disables RTCP traffic when there is no RTP traffic. This feature is useful, for example, to stop RTCP traffic that is typically sent when calls are put on hold (by an INVITE with 'a=inactive' in the SDP). <ul style="list-style-type: none"> [0] Active Always = (Default) RTCP is active even during inactive RTP periods, i.e., when the media is in 'recvonly' or 'inactive' mode. [1] Inactive Only If RTP Inactive = No RTCP is sent when RTP is inactive.
RTP Control Protocol Extended Reports (RTCP XR) Parameters	
Web: Enable RTCP XR EMS: RTCP XR Enable [VQMonEnable]	Enables voice quality monitoring and RTCP XR, according to Internet-Draft draft-ietf-sipping-rtcp-summary-13. <ul style="list-style-type: none"> [0] Disable (default) [1] Enable Note: For this parameter to take effect, a device reset is required.
Web: Minimum Gap Size EMS: GMin [VQMonGMin]	Defines the voice quality monitoring - minimum gap size (number of frames). The default is 16.
Web/EMS: Burst Threshold [VQMonBurstHR]	Defines the voice quality monitoring - excessive burst alert threshold. The default is -1 (i.e., no alerts are issued).
Web/EMS: Delay Threshold [VQMonDelayTHR]	Defines the voice quality monitoring - excessive delay alert threshold. The default is -1 (i.e., no alerts are issued).
Web: R-Value Delay Threshold EMS: End of Call Rval Delay Threshold [VQMonEOCRValTHR]	Defines the voice quality monitoring - end of call low quality alert threshold. The default is -1 (i.e., no alerts are issued).
Web: RTCP XR Packet Interval EMS: Packet Interval [RTCPInterval]	Defines the time interval (in msec) between adjacent RTCP reports. The valid value range is 0 to 65,535. The default is 5,000.
Web: Disable RTCP XR Interval Randomization EMS: Disable Interval Randomization [DisableRTCPRandomize]	Determines whether RTCP report intervals are randomized or whether each report interval accords exactly to the parameter RTCPInterval. <ul style="list-style-type: none"> [0] Disable = (Default) Randomize [1] Enable = No Randomize
EMS: RTCP XR Collection Server Transport Type [RTCPXRESCTransportType]	Defines the transport layer used for outgoing SIP dialogs initiated by the device to the RTCP XR Collection Server. <ul style="list-style-type: none"> [-1] Not Configured (default)

Parameter	Description
	<ul style="list-style-type: none"> ▪ [0] UDP ▪ [1] TCP ▪ [2] TLS <p>Note: When set to [-1], the value of the SIPTransportType parameter is used.</p>
Web: RTCP XR Collection Server EMS: Esc IP [RTCPXREscIP]	Defines the IP address of the Event State Compositor (ESC). The device sends RTCP XR reports to this server, using SIP PUBLISH messages. The address can be configured as a numerical IP address or as a domain name.
Web: RTCP XR Report Mode EMS: Report Mode [RTCPXRReportMode]	<p>Determines whether RTCP XR reports are sent to the Event State Compositor (ESC) and defines the interval at which they are sent.</p> <ul style="list-style-type: none"> ▪ [0] Disable = (Default) RTCP XR reports are not sent to the ESC. ▪ [1] End Call = RTCP XR reports are sent to the ESC at the end of each call. ▪ [2] End Call & Periodic = RTCP XR reports are sent to the ESC at the end of each call and periodically according to the RTCPInterval parameter.

42.11 Least Cost Routing Parameters

The Least Cost Routing (LCR) parameters are described in the table below.

Table 42-36: LCR Parameters

Parameter	Description
Web: Routing Rule Groups Table [RoutingRuleGroups]	<p>This table parameter enables the LCR feature and configures the average call duration and default call cost. The default call cost determines whether routing rules that are not configured with a Cost Group are considered as a higher or lower cost route compared to other matching routing rules that are assigned Cost Groups.</p> <p>[RoutingRuleGroups] FORMAT RoutingRuleGroups_Index = RoutingRuleGroups_LCREnable, RoutingRuleGroups_LCRAverageCallLength, RoutingRuleGroups_LCRDefaultCost; [\RoutingRuleGroups]</p> <p>Note: For a detailed description of this table, see 'Enabling LCR and Configuring Default LCR' on page 178.</p>
Web: Cost Group Table EMS: Cost Group Provisioning > Cost Group [CostGroupTable]	<p>This table parameter configures the Cost Groups for LCR, where each Cost Group is configured with a name, fixed call connection charge, and a call rate (charge per minute).</p> <p>[CostGroupTable] FORMAT CostGroupTable_Index = CostGroupTable_CostGroupName, CostGroupTable_DefaultConnectionCost, CostGroupTable_DefaultMinuteCost; [\CostGroupTable]</p> <p>For example: CostGroupTable 2 = "Local Calls", 2, 1;</p> <p>Note: For a detailed description of this table, see 'Configuring Cost</p>

Parameter	Description
	Groups' on page 179.
Web: Cost Group > Time Band Table EMS: Time Band Provisioning > Time Band [CostGroupTimebands]	<p>This table parameter configures time bands and associates them with Cost Groups.</p> <p>[CostGroupTimebands] FORMAT CostGroupTimebands_TimebandIndex = CostGroupTimebands_StartTime, CostGroupTimebands_EndTime, CostGroupTimebands_ConnectionCost, CostGroupTimebands_MinuteCost; [CostGroupTimebands]</p> <p>Note: For a detailed description of this table, see 'Configuring Time Bands for Cost Groups' on page 180.</p>

42.12 LDAP Parameters

The Lightweight Directory Access Protocol (LDAP) parameters are described in the table below. For more information on routing based on LDAP, see 'Routing Based on LDAP Active Directory Queries' on page 167.

Table 42-37: LDAP Parameters

Parameter	Description
Web: LDAP Service [LDAPServiceEnable]	<p>Enables the LDAP feature.</p> <ul style="list-style-type: none"> [0] Disable (default) [1] Enable <p>Note: For this parameter to take effect, a device reset is required.</p>
Web: LDAP Server IP EMS: Server Ip [LDAPServerIP]	<p>Defines the LDAP server's address as an IP address (in dotted-decimal notation, e.g., 192.10.1.255).</p> <p>The default is 0.0.0.0.</p>
Web: LDAP Server Port EMS: Server Port [LDAPServerPort]	<p>Defines the LDAP server's port number.</p> <p>The valid value range is 0 to 65535. The default port number is 389.</p>
Web: LDAP Server Domain Name EMS: Server Domain Name [LDAPServerDomainName]	<p>Defines the host name (FQDN) of the LDAP server. The device tries to connect to the LDAP server according to the IP address list received in the DNS query. If there is no connection to the LDAP server or the connection to the LDAP server fails, the device tries to connect to the LDAP server with the next IP address in the DNS query list.</p> <p>Note: The 'LDAP Server IP' parameter takes precedence over this parameter. Thus, if you want to use an FQDN, keep the 'LDAP Server IP' parameter empty.</p>
Web: LDAP Password EMS: Password [LDAPPassword]	<p>Defines the LDAP server's user password.</p>
Web: LDAP Bind DN EMS: Bind DN [LDAPBindDN]	<p>Defines the LDAP server's bind Distinguished Name (DN). This is used as the username during connection and binding to the server.</p> <p>For example: LDAPBindDN = "CN=Search</p>

Parameter	Description
	<p>user,OU=Labs,DC=OCSR2,DC=local"</p> <p>Note: The DN is used to uniquely name an Active Directory object.</p>
Web: LDAP Search Dn EMS: Search DN [LDAPSearchDN]	<p>Defines up to three search DN's for LDAP search queries. These are the DN subtrees where the search is done. This parameter is mandatory for the search.</p> <p>The format of this parameter is as follows:</p> <p>[LdapSearchDNs] FORMAT LdapSearchDNs_Index = LdapSearchDNs_Base_Path; [\LdapSearchDNs]</p> <p>For example: LdapSearchDNs 0 = "CN=Search user,OU=NY,DC=OCSR2,DC=local"; LdapSearchDNs 1 = "CN=Search user,OU=SF,DC=OCSR2,DC=local";</p> <p>In this example, the DN path is defined by the LDAP names, cn (common name), ou (organizational unit) and dc (domain component).</p> <p>Note: If you configure multiple DN's, you can specify whether the search is done sequentially or in parallel, using the LDAPSearchDNsinParallel parameter.</p>
[LDAPSearchDNsinParallel]	<p>Defines the LDAP query DN search method in the AD database if multiple search DN's are configured, using the LDAPSearchDNs parameter.</p> <ul style="list-style-type: none"> [0] Sequential = If the first DN search fails, the search is done on the next configured DN, and so on. [1] Parallel (Default)
Web: LDAP Server Max Respond Time EMS: Server Max Respond Time [LDAPServerMaxRespondTime]	<p>Defines the time (in seconds) that the device waits for LDAP server responses.</p> <p>The valid value range is 0 to 86400. The default is 3000.</p>
[LDAPDebugMode]	<p>Determines whether to enable the LDAP task debug messages. This is used for providing debug information regarding LDAP tasks.</p> <p>The valid value range is 0 to 3. The default is 0.</p>
Web: MS LDAP OCS Number attribute name EMS: LDAP ocs Number Attribute Name [MSLDAPOCSNumAttributeName]	<p>Defines the name of the attribute that represents the user OCS number in the Microsoft AD database.</p> <p>The valid value is a string of up to 49 characters. The default is "msRTCSIP-PrimaryUserAddress".</p>
Web: MS LDAP PBX Number attribute name [MSLDAPPBXNumAttributeName]	<p>Defines the name of the attribute that represents the user PBX number in the Microsoft AD database.</p> <p>The valid value is a string of up to 49 characters. The default is "telephoneNumber".</p>
Web: MS LDAP MOBILE Number attribute name [MSLDAPMobileNumAttributeName]	<p>Defines the name of the attribute that represents the user Mobile number in the Microsoft AD database.</p> <p>The valid value is a string of up to 49 characters. The default is "mobile".</p>

Parameter	Description
[MSLDAPPrivateNumAttributeName]	<p>Defines the name of the attribute that represents the user's private number in the AD. If this value equals the value of the MSLDAPPrimaryKey or MSLDAPSecondaryKey parameter, then the device queries the AD for the destination number in this private attribute name; otherwise, this parameter is not used as a search key.</p> <p>The default is "msRTCSIP-PrivateLine".</p>
[MSLDAPPrimaryKey]	<p>Defines the name of the attribute used as a query search key for the destination number in the AD. This is used instead of the "PBX" attribute name (configured by the MSLDAPPBXNumAttributeName parameter).</p> <p>The default is not configured.</p>
[MSLDAPSecondaryKey]	<p>Defines the name of the attribute used as the second query search key for the destination number in the AD, if the primary search key or PBX search is not found.</p>
LDAP Cache Service [LDAPCacheEnable]	<p>Enables the LDAP cache service.</p> <ul style="list-style-type: none"> ▪ [0] Disable (default) ▪ [1] Enable <p>Notes:</p> <ul style="list-style-type: none"> ▪ For this parameter to take effect, a device reset is required. ▪ For more information on LDAP caching, see 'Configuring the Device's LDAP Cache' on page 168.
LDAP Cache Entry Timeout [LDAPCacheEntryTimeout]	<p>Defines the duration (in minutes) that an entry in the LDAP cache is valid. If the timeout expires, the cached entry is only used if there is no connectivity with the LDAP server.</p> <p>The default is 1200.</p>
LDAP Cache Entry Removal Timeout [LDAPCacheEntryRemovalTimeout]	<p>Defines the duration (in hours) after which the LDAP entry is removed from the cache.</p> <p>The default is 0.</p>

42.13 SBC and CRP Parameters

The SBC and CRP parameters are described in the table below.

Table 42-38: SBC and CRP Parameters

Parameter	Description
CRP-Specific Parameters	
Web: CRP Application [EnableCRPApplication]	<p>Enables the CRP application.</p> <ul style="list-style-type: none"> ▪ [0] Disable (default) ▪ [1] Enable <p>Note: For this parameter to take effect, a device reset is required.</p>
Web: CRP Survivability Mode [CRPSurvivabilityMode]	<p>Defines the CRP mode.</p> <ul style="list-style-type: none"> ▪ [0] Standard Mode (default) ▪ [1] Always Emergency Mode ▪ [2] Auto-answer REGISTER
SBC-Specific Parameters	
Web/EMS: Enable SBC [EnableSBCApplication]	<p>Enables the Session Border Control (SBC) application.</p> <ul style="list-style-type: none"> ▪ [0] Disable (default) ▪ [1] Enable <p>Notes:</p> <ul style="list-style-type: none"> ▪ For this parameter to take effect, a device reset is required. ▪ In addition to enabling this parameter, the number of maximum SBC/IP-to-IP sessions must be defined in the Software License Key.
SBC and CRP Parameters	
Web: Allow Unclassified Calls [AllowUnclassifiedCalls]	<p>Determines whether incoming calls that cannot be classified (i.e. classification process fails) to a Source IP Group are rejected or processed.</p> <ul style="list-style-type: none"> ▪ [0] Reject = Call is rejected if classification fails. ▪ [1] Allow = (Default) If classification fails, the incoming packet is assigned to a source IP Group (and subsequently processed) as follows: <ul style="list-style-type: none"> ✓ The source SRD is determined according to the SIP Interface to where the SIP-initiating dialog request is sent. The source IP Group is set to the default IP Group associated with this SRD. ✓ If the source SRD is ID 0, then source IP Group ID 0 is chosen. In case of any other SRD, then the first IP Group associated with this SRD is chosen as the source IP Group or the call. If no IP Group is associated with this SRD, the call is rejected.
Web: SBC No Answer Timeout [SBCAlertTimeout]	<p>Defines the timeout (in seconds) for SBC outgoing (outbound IP routing) SIP INVITE messages. If the called IP party does not answer the call within this user-defined interval, the device disconnects the session. The device starts the timeout count upon receipt of a SIP 180 Ringing response from the called party. If no other SIP response (for example, 200 OK) is received thereafter within this timeout, the call is released.</p> <p>The valid range is 0 to 3600 seconds. the default is 600.</p>

Parameter	Description
Web: SBC Max Forwards Limit [SBCMaxForwardsLimit]	<p>Defines the Max-Forwards SIP header value. The Max-Forwards header is used to limit the number of servers (such as proxies) that can forward the SIP request. The Max-Forwards value indicates the remaining number of times this request message is allowed to be forwarded. This count is decremented by each server that forwards the request.</p> <p>This parameter affects the Max-Forwards header in the received message as follows:</p> <ul style="list-style-type: none"> ▪ If the received header's original value is 0, the message is not passed on and is rejected. ▪ If the received header's original value is less than this parameter's value, the header's value is decremented before being sent on. ▪ If the received header's original value is greater than the parameter's value, the header's value is replaced by the user-defined parameter's value. <p>The valid value range is 1-70. The default is 10.</p>
Web: SBC Session-Expires [SBCSessionExpires]	<p>Defines the SBC session refresh timer (in seconds) in the Session-Expires header of outgoing INVITE messages.</p> <p>The valid value range is 90 (according to RFC 4028) to 86400. The default is 180.</p>
Web: Minimum Session-Expires [SBCMinSE]	<p>Defines the minimum amount of time (in seconds) between session refresh requests in a dialog before the session is considered timed out. This value is conveyed in the SIP Min-SE header.</p> <p>The valid range is 0 (default) to 1,000,000 (where 0 means that the device does not limit Session-Expires).</p>
Web/EMS: Handle P-Asserted-Identity [SBCAssertIdentity]	<p>Determines the device's privacy handling of the P-Asserted-Identity header. This indicates how the outgoing SIP message asserts identity.</p> <ul style="list-style-type: none"> ▪ [0] Don't Care = (Default) P-Asserted Identity header is not affected. ▪ [1] Add P-Asserted-Identity Header = Adds a P-Asserted-Identity header. The header's values are taken from the source URL. ▪ [2] Remove P-Asserted-Identity Header = Removes the P-Asserted-Identity header. <p>Notes:</p> <ul style="list-style-type: none"> ▪ This parameter affects only the initial INVITE request. ▪ The configuration of privacy handling in the IP Group table takes precedence over the settings of this global parameter. <ul style="list-style-type: none"> ✓ If in the IP Group this parameter is set to 'Don't care', then the settings of this global parameter is used. ✓ If this global parameter and the IP Group are set to 'Don't care', the device uses the same P-Asserted-Identity header (if present) in the incoming message for the outgoing message. ▪ This parameter can also be configured in an IP Profile.
Web: Keep original user in Register	<p>Determines whether the device replaces the Contact user with a unique Contact user in the outgoing message in response to a</p>

Parameter	Description
[SBCKeepContactUserinRegister]	<p>REGISTER request.</p> <ul style="list-style-type: none"> ▪ [0] Disable = (Default) The device replaces the original Contact user with a unique Contact user, for example: <ul style="list-style-type: none"> ✓ Received Contact: <sip:123@domain.com> ✓ Outgoing (unique) Contact: <sip:FEU1_7_1@SBC> ▪ [1] Enable = The original Contact user is retained and used in the outgoing REGISTER request. <p>Note: This parameter is applicable only to REGISTER messages received from User-type IP Groups and that are sent to Server-type IP Groups.</p>
Web: SBC Remote Refer Behavior [SBCReferBehavior]	<p>Determines the device's handling of REFER requests.</p> <ul style="list-style-type: none"> ▪ [0] Transparent = (Default) Refer-To header is unchanged and the device forwards the REFER as is. ▪ [1] DB URL = Changes the Refer-To header so that the re-routed INVITE is sent through the SBC: <ol style="list-style-type: none"> a. Before forwarding the REFER request, the device changes the host part to the device's IP address and adds a special prefix ("T~&R_") to the Contact user part. b. The incoming INVITE is identified as a REFER-resultant INVITE according to this special prefix. c. The device replaces the host part in the Request-URI with the host from the REFER contact. The special prefix remains in the user part for regular classification, manipulation, and routing. The special prefix can also be used for specific routing rules for REFER-resultant INVITES. d. The special prefix is removed before the resultant INVITE is sent to the destination. ▪ [2] IP Group Name = Sets the host part in the REFER message to the name defined for the IP Group (in the IP Group table). ▪ [3] Handle Locally = Handles the incoming REFER request itself without forwarding the REFER. The device generates a new INVITE to the alternative destination according to the rules in the IP-to-IP Routing table (the 'Call Trigger' field must be set to REFER). <p>Note: This parameter can be configured in an IP Profile.</p>
[SBCXferPrefix]	<p>When the SBCReferBehavior is set to 1, the device, while interworking the SIP REFER message, adds the prefix "T~&R-" to the user part of the URI in the Refer-To header. After this, the device can receive an INVITE with such a prefix (the INVITE is sent by the UA that receives the REFER message or 302 response). If the device receives an INVITE with such a prefix, it replaces the prefix with the value defined for the SBCXferPrefix parameter.</p> <p>The default is empty ("").</p> <p>Note: This feature is also applicable to 3xx redirect responses. The device adds the prefix "T~&R-" to the URI user part in the Contact header if the SBC3xxBehavior parameter is set to 1.</p>
[SBC3xxBehavior]	<p>Determines the device's handling of SIP 3xx responses. When enabled, the device handles SIP redirections between different subnets (e.g., between LAN and WAN sides). This is required where the new address provided by the redirector (Redirect sever) may not be reachable by the far-end user (FEU) located in another</p>

Parameter	Description
	<p>subnet. For example, a far-end user (FEU) in the WAN sends a SIP request via the device to a Redirect server in the LAN, and the Redirect server replies with a SIP 3xx response to a PBX in the LAN in the Contact header. If the device sends this response as is (i.e., with the original Contact header), the FEU is unable to reach the new destination.</p> <ul style="list-style-type: none"> ▪ [0] Transparent = (Default) The device forwards the received SIP 3xx response as is, without changing the Contact header (i.e., transparent handling). ▪ [1] DB URL = The device changes the Contact header so that the re-route request is sent through the the device. The device changes the URI in the Contact header of the received SIP 3xx response to its own URI and adds a special user prefix ("T-&R_"), which is then sent to the FEU. The FEU then sends a new INVITE to the device, which the device then sends to the correct destination. ▪ [2] Handle Locally = The device handles SIP 3xx responses on behalf of the dialog-initiating UA and retries the request (e.g., INVITE) using one or more alternative URIs included in the 3xx response. The device sends the new request to the alternative destination according to the IP-to-IP Routing table (the 'Call Trigger' field must be set to 3xx). <p>Notes:</p> <ul style="list-style-type: none"> ▪ When this parameter is changed from 1 to 0, new 3xx Contact headers remain unchanged. However, requests with the special prefix continue using the device's database to locate the new destination. ▪ Only one database entry is supported for the same host, port, and transport combination. For example, the following URLs cannot be distinguished by the device: <ul style="list-style-type: none"> ✓ sip:10.10.10.10:5060;transport=tcp;param=a ✓ sip:10.10.10.10:5060;transport=tcp;param=b ▪ The database entry expires two hours after the last use. ▪ The maximum number of destinations (i.e., database entries) is 50. ▪ This parameter can also be configured as an IP Profile. ▪ For more information on SIP 3xx Redirect response handling, see 'Handling SIP 3xx Redirect Responses' on page 253.
Web: Enforce Media Order [SBCEnforceMediaOrder]	<p>Enables the device to arrange media lines ('m=' line) in the SDP offer according to the previous offer-answer exchange (RFC 3264).</p> <ul style="list-style-type: none"> ▪ [0] Disable ▪ [1] Enable (default)
Web: Lifetime of the nonce in seconds [AuthNonceDuration]	<p>Defines the lifetime (in seconds) that the current nonce is valid for server-based authentication. The device challenges a message that attempts to use a server nonce beyond this period. This parameter is used to provide replay protection (i.e., ensures that old communication streams are not used in replay attacks).</p> <p>The valid value range is 30 to 600. The default is 300.</p>

Parameter	Description
Web: Authentication Challenge Method [AuthChallengeMethod]	<p>Defines the type of server-based authentication challenge.</p> <ul style="list-style-type: none"> [0] 0 = (Default) Send SIP 401 "Unauthorized" with a WWW-Authenticate header as the authentication challenge response. [1] 1 = Send SIP 407 "Proxy Authentication Required" with a Proxy-Authenticate header as the authentication challenge response.
Web: Authentication Quality of Protection [AuthQOP]	<p>Defines the authentication and integrity level of quality of protection (QoP) for digest authentication offered to the client. When the device challenges a SIP request (e.g., INVITE), it sends a SIP 401 response with the Proxy-Authenticate header or WWW-Authenticate header containing the 'qop' parameter. The QoP offered in the 401 response can be 'auth', 'auth-int', both 'auth' and 'auth-int', or the 'qop' parameter can be omitted from the 401 response. In response to the 401, the client needs to send the device another INVITE with the MD5 hash of the INVITE message and indicate the selected auth type.</p> <ul style="list-style-type: none"> [0] 0 = The device sends 'qop=auth' in the SIP response, requesting authentication (i.e., validates user by checking user name and password). This option does not authenticate the message body (i.e., SDP). [1] 1 = The device sends 'qop=auth-int' in the SIP response, indicating required authentication and authentication with integrity (e.g., checksum). This option restricts the client to authenticating the entire SIP message, including the body, if present. [2] 2 = (Default) The device sends 'qop=auth, auth-int' in the SIP response, indicating either authentication or integrity. This enables the client to choose 'auth' or 'auth-int'. If the client chooses 'auth-int', then the body is included in the authentication. If the client chooses 'auth', then the body is not authenticated. [3] 3 = No 'qop' parameter is offered by the device in the SIP 401 challenge message.
Web: SBC User Registration Time [SBCUserRegistrationTime]	<p>Defines the duration (in seconds) of the periodic registrations between the user and the device (the device responds with this value to the user). When set to 0, the device does not change the Expires header's value received in the user's REGISTER request. If no Expires header is received in the REGISTER message and the SBCUserRegistrationTime parameter is set to 0, then by default, the Expires header's value is set to 180 seconds.</p> <p>The valid range is 0 to 2,000,000 seconds. The default is 0.</p> <p>Note: This parameter can also be configured for an IP Profile (in the IP Profile table).</p>
Web: SBC Proxy Registration Time [SBCProxyRegistrationTime]	<p>Defines the duration (in seconds) for which the user is registered in the proxy database (after the device forwards the REGISTER message). When set to 0, the device sends the Expires header's value as received from the user to the proxy.</p> <p>The valid range is 0 to 2,000,000 seconds. The default is 0.</p>
Web: SBC Survivability Registration Time [SBCSurvivabilityRegistrationTime]	<p>Defines the duration of the periodic registrations between the user and the device, when the device is in survivability state (i.e., when REGISTER requests cannot be forwarded to the proxy and are terminated by the device). When set to 0, the device uses the value set by the SBCUserRegistrationTime parameter for the</p>

Parameter	Description
	<p>device's response.</p> <p>The valid range is 0 to 2,000,000 seconds. The default is 0.</p>
[SBCEnableAASTRASurvivabilityNotice]	<p>Enables the device to notify Aastra IP phones that the device is currently operating in Survivability mode. When this occurs, the Aastra IP phones display the message, "Stand Alone Mode" on their LCD screens. Survivability mode occurs when connectivity with the WAN fails and as a result, the device enables communication between IP phone users within the LAN enterprise.</p> <ul style="list-style-type: none"> ▪ [0] = Disable ▪ [1] = Enable <p>When this feature is enabled and the SBC device is in Survivability mode, it responds to SIP REGISTER messages from the IP phones with a SIP 200 OK containing the following XML body:</p> <pre>Content-Type: application/xml <?xml version="1.0" encoding="utf-8"?> <LMIDocument version="1.0"> <LocalModeStatus> <LocalModeActive>true</LocalModeActive> <LocalModeDisplay>StandAlone Mode</LocalModeDisplay> </LocalModeStatus> </LMIDocument></pre>
Web: SBC GRUU Mode [SBCGruuMode]	<p>Determines the Globally Routable User Agent (UA) URI (GRUU) support, according to RFC 5627.</p> <ul style="list-style-type: none"> ▪ [0] None = No GRUU is supplied to users. ▪ [1] As Proxy = (Default) The device provides same GRUU types as the proxy provided the device's GRUU clients. ▪ [2] Temporary only = Supply only temporary GRUU to users. (Currently not supported.) ▪ [3] Public only = The device provides only public GRUU to users. ▪ [4] Both = The device provides temporary and public GRUU to users. (Currently not supported.) <p>This parameter allows the device to act as a GRUU server for its SIP UA clients, providing them with public GRUU's, according to RFC 5627. The public GRUU provided to the client is denoted in the SIP Contact header parameters, "pub-gruu". Public GRUU remains the same over registration expirations. On the other SBC leg communicating with the Proxy/Registrar, the device acts as a GRUU client.</p> <p>The device creates a GRUU value for each of its registered clients, which is mapped to the GRUU value received from the Proxy server. In other words, the created GRUU value is only used between the device and its clients (endpoints).</p> <pre>Public-GRUU: sip:userA@domain.com;gr=unique-id</pre>
Web: Bye Authentication [SBCEnableByeAuthentication]	<p>Enables authenticating a SIP BYE request before disconnecting the call. This feature prevents, for example, a scenario in which the SBC SIP client receives a BYE request from a third-party imposer assuming the identity of a participant in the call and as a consequence, the call between the first and second parties is</p>

Parameter	Description
	<p>inappropriately disconnected.</p> <ul style="list-style-type: none"> ▪ [0] Disable (default) ▪ [1] Enable = The device forwards the SIP authentication response (for the BYE request) to the request sender and waits for the user to authenticate it. The call is disconnected only if the authenticating server responds with a 200 OK.
[SBCExtensionsProvisioning Mode]	<p>Enables SBC user registration for interoperability with BroadSoft's BroadWorks server, to provide call survivability in case of connectivity failure with the BroadWorks server.</p> <ul style="list-style-type: none"> ▪ [0] = (Default) Normal processing of REGISTER messages. ▪ [1] = Registration method for BroadWorks server. In a failure scenario with BroadWorks, the device acts as a backup SIP proxy server, maintaining call continuity between the enterprise LAN users (subscribers) and between the subscribers and the PSTN (if provided). <p>Note: For a detailed description of this feature, see 'Auto-Provisioning of Subscriber-Specific Information for BroadWorks Server' on page 260.</p>
Web: SBC Direct Media [SBCDirectMedia]	<p>Enables the No Media Anchoring feature (i.e., direct media) for all SBC calls. No Media Anchoring uses SIP signaling capabilities without handling the RTP/SRTP (media) flow between remote SIP user agents (UA). The RTP packets do not traverse the device, instead, the two SIP UAs establish a direct RTP/SRTP flow between one another. Signaling continues to traverse the device with minimal intermediation and involvement to enable certain SBC abilities such as routing</p> <ul style="list-style-type: none"> ▪ [0] Disable = (Default) All SRD calls via SBC are not direct media - internal SRD calls are according to SRD configuration. ▪ [1] Enable = All SBC calls use the No Media Anchoring feature (i.e., direct media). <p>Notes:</p> <ul style="list-style-type: none"> ▪ For more information on No Media Anchoring, see 'No Media Anchoring (Anti Tromboning)' on page 245. ▪ When No Media Anchoring is enabled: <ul style="list-style-type: none"> ✓ Manipulation is not done on SDP data (offer/answer transaction) such as ports and IP addresses. ✓ Opening voice channels and allocation of IP media ports are not required. ✓ Forced Transcoding and Extension Coders features are disabled for No Media Anchoring calls. ✓ The Coder Restriction feature (Allowed Coders List) operates simultaneously with No Media Anchoring calls. Restricted coders are removed from the SDP offer message. ▪ No Media Anchoring is typically implemented in the following scenarios: <ul style="list-style-type: none"> ✓ SBC device is located in the LAN. ✓ Calls between two SIP UAs in the same LAN and signals are sent to a SIP proxy server that is located in the WAN. ✓ SBC device does not do NAT traversal (for media) and all the users are in the same domain. ▪ The benefits of implementing the No Media Anchoring feature includes the following: saves network bandwidth, reduces CPU

Parameter	Description
	<p>usage (no RTP/SRTP handling), and avoids interference in SDP negotiation and header manipulation on RTP/SRTP.</p> <ul style="list-style-type: none"> ▪ The process for handling the No Media Anchoring feature is as follows: <ul style="list-style-type: none"> ✓ Identifying a No Media Anchoring call according to configuration and the call's properties (such as source, destination, IP Group, and SRD). ✓ Handling the identified No Media Anchoring call. ▪ You can enable No Media Anchoring per SRD (using the IntraSRDMediaAnchoring parameter), whereby calls between two UAs that pertain to the same SRD (source and destination) are handled as a No Media Anchoring (direct media) call. ▪ Chosen configuration can't handle call from any UA to a foreign UA (vice versa) but both UAs belong to the same SRD and the parameter IntraSRDMediaAnchoring for that specific SRD is > 0. ▪ When this parameter is disabled, No Media Anchoring calls between two UAs that belong to separate SRDs cannot be configured. No Media Anchoring calls between two UAs that belong to the same SRD is configurable only (in this case).
Web: Transcoding Mode CLI: transcoding-mode [TranscodingMode]	<p>Defines the voice transcoding mode (media negotiation) between the two SBC legs for the SBC application.</p> <ul style="list-style-type: none"> ▪ [0] Only if Required = (Default) Do not force transcoding. Many of the media settings (such as gain control) are not implemented on the voice stream. The SBC application passes packets RTP to RTP without any processing. ▪ [1] Force = Forces transcoding on the outgoing SBC leg. The device's SBC application interworks the media by implementing DSP transcoding. <p>Note: This parameter can also be configured in an IP Profile.</p>
Web: SBC Preferences Mode [SBCPreferencesMode]	<p>Determines the order of the Extension coders (coders added if there are no common coders between SDP offered coders and Allowed coders) and Allowed coders (defined in the Allowed Coders Group table) in the outgoing SIP message (in the SDP).</p> <ul style="list-style-type: none"> ▪ [0] Doesn't Include Extensions = (Default) Extension coders are added at the end of the coder list. ▪ [1] Include Extensions = Extension coders and Allowed coders are arranged according to their order of appearance in the Allowed Coders Group table. <p>Note: If the SBCExtensionCodersGroupID parameter of the IP Profile table is set to None, then this parameter is not applicable.</p>
Web: SBC Send Invite To All Contacts CLI: sbc-send-invite-to-all-contacts [SBCSendInviteToAllContacts]	<p>Enables call forking of INVITE message received with a Request-URI for a specific contact registered in the device's database, to all users under the same AOR as the contact.</p> <ul style="list-style-type: none"> ▪ [0] Disable (default) = Sends the INVITE only to the contact of the received Request-URI. ▪ [1] Enable <p>Note: To configure call forking initiated by the device, see Initiating SIP Call Forking.</p>

Parameter	Description
Web: SBC Shared Line Registration Mode CLI: sbc-shared-line-reg-mode [SBCSharedLineRegMode]	<p>Enables the termination on the device of SIP REGISTER messages from secondary lines pertaining to the Shared Line feature.</p> <ul style="list-style-type: none"> [0] Disable = (Default) Device forwards the REGISTER messages as is (i.e., not terminated on the device). [1] Enable = REGISTER messages of secondary lines are terminated on the device. <p>Note: The device always forwards REGISTER messages of the primary line.</p>
Web: SBC Forking Handling Mode [SBCForkingHandlingMode]	<p>Defines the handling of SIP 18x responses received due to call forking of an INVITE.</p> <ul style="list-style-type: none"> [0] Latch On First = (Default) Only the first 18x is forwarded to the INVITE-initiating UA. If SIP 18x with SDP is received, the device opens a voice stream according to the received SDP and disregards any subsequent 18x forking responses (with or without SDP). If the first response is 180 without SDP, the device responds according to the PlayRBTone2TEL parameter and disregards the subsequent forking 18x responses. [1] Sequential = All 18x responses are forwarded, one at a time (sequentially) to the INVITE-initiating UA. If a 18x arrives with an offer only, then only the first offer is forwarded to the INVITE-initiating UA and subsequent 18x responses are discarded.
[EnableSBCMediaSync]	<p>Enables SBC media synchronization process for calls established from SIP forking that is initiated by external proxy servers. It is possible that a call is established with the media not synchronized between the SBC legs. Media synchronization resolves this issue.</p> <ul style="list-style-type: none"> [0] Disable [1] Enable (default)
Admission Control Table	
Web: Admission Control EMS: Call Admission Control [SBCAdmissionControl]	<p>This table parameter defines limitations on the number of allowed concurrent calls (SIP dialogs). This is useful for controlling bandwidth utilization between Voice and Data traffic. The format of this parameter is as follows:</p> <p>[SBCAdmissionControl] FORMAT SBCAdmissionControl_Index = SBCAdmissionControl_LimitType, SBCAdmissionControl_IPGroupID, SBCAdmissionControl_SRIDID, SBCAdmissionControl_RequestType, SBCAdmissionControl_RequestDirection, SBCAdmissionControl_Limit, SBCAdmissionControl_LimitPerUser, SBCAdmissionControl_Rate, SBCAdmissionControl_MaxBurst; [SBCAdmissionControl]</p> <p>For example, the below configuration allows a maximum of 10 concurrent SIP INVITEs for IP Group 1: SBCAdmissionControl 1 = 0, 1, -1, 1, 0, 10, -1, 0, 0;</p> <p>Note: For a detailed description of this table, see 'Configuring Admission Control' on page 268.</p>
Allowed Audio Coders Table	

Parameter	Description
Web: Allowed Audio Coders [AllowedCodersGroup0] [AllowedCodersGroup1] [AllowedCodersGroup2] [AllowedCodersGroup3] [AllowedCodersGroup4]	<p>This table parameter configures Allowed Coders Groups, which determines the coders that can be used for a specific SBC leg. Coders excluded from the Allowed Coders Group are removed from the SDP offer (only coders common between SDP offered coders and Allowed Coders are used). In addition, coders defined in top entries in the Allowed Coders Group are assigned higher priority than those entered in lower entries.</p> <p>[AllowedCodersGroupx] FORMAT AllowedCodersGroup_Index = AllowedCodersGroup_Name; [AllowedCodersGroup]</p> <p>For example, below represents two configured Allowed Coders Groups. Group 0 has two coders; Group 1 has one coder. The highest priority coder is G.723.1.</p> <p>[AllowedCodersGroup0] FORMAT AllowedCodersGroup0_Index = AllowedCodersGroup0_Name; AllowedCodersGroup0 0 = g7231; AllowedCodersGroup0 1 = g711Alaw64k; [AllowedCodersGroup0]</p> <p>[AllowedCodersGroup1] FORMAT AllowedCodersGroup1_Index = AllowedCodersGroup0_Name; AllowedCodersGroup1 0 = g711Ulaw64k; [AllowedCodersGroup1]</p> <p>Note: For a detailed description of this table, see 'Configuring Allowed Coder Groups' on page 271.</p>
Classification Table	
Web: Classification Table EMS: SBC Classification [Classification]	<p>This table parameter configures the Classification table. This table classifies incoming SIP dialogs to Source IP Groups. The format of this parameter is as follows:</p> <p>[Classification] FORMAT Classification_Index = Classification_MessageCondition, Classification_SrcSRDID, Classification_SrcAddress, Classification_SrcPort, Classification_SrcTransportType, Classification_SrcUsernamePrefix, Classification_SrcHost, Classification_DestUsernamePrefix, Classification_DestHost, Classification_ActionType, Classification_SrcIPGroupID; [\Classification]</p> <p>For example: Classification 1 = 1, , 10.8.6.15, 5060, 2, *, *, *, *, 1, 4;</p> <p>Note: For a detailed description of this table, see 'Configuring Classification Rules' on page 272.</p>
Condition Table	
Web: Condition Table [ConditionTable]	<p>This table parameter configures Condition rules for SIP messages using the same syntax as used in the SIP Message Manipulation table. These Condition rules are later assigned to Classification rules in the Classification table for enhancing the process of classifying incoming SIP dialogs to an IP Groups.</p> <p>[ConditionTable] FORMAT ConditionTable_Index = ConditionTable_Condition,</p>

Parameter	Description
	ConditionTable_Description; [\ConditionTable] Note: For a detailed description of this table, see 'Configuring Condition Rules' on page 278 .
SBC IP-to-IP Routing Table	
Web: IP-to-IP Routing Table EMS: IP to IP Routing [IP2IPRouting]	This table parameter configures the SBC IP-to-IP Routing table for routing incoming SIP messages such as INVITE messages to an IP destination. The format of this parameter is as follows: [IP2IPRouting] FORMAT IP2IPRouting_Index = IP2IPRouting_SrcIPGroupID, IP2IPRouting_SrcUsernamePrefix, IP2IPRouting_SrcHost, IP2IPRouting_DestUsernamePrefix, IP2IPRouting_DestHost, IP2IPRouting_RequestType, IP2IPRouting_MessageCondition, IP2IPRouting_ReRouteIPGroupID, IP2IPRouting_Trigger, IP2IPRouting_DestType, IP2IPRouting_DestIPGroupID, IP2IPRouting_DestSRDID, IP2IPRouting_DestAddress, IP2IPRouting_DestPort, IP2IPRouting_DestTransportType, IP2IPRouting_AltRouteOptions, IP2IPRouting_CostGroup; [\IP2IPRouting] For example: IP2IPRouting 1 = -1, *, *, *, *, 0, , -1, 0, 0, 1, , , 0, -1, 0,; Note: For a detailed description of this table, see 'Configuring SBC IP-to-IP Routing' on page 279 .
SBC Alternative Routing Reasons Table	
Web: SBC Alternative Routing Reasons EMS: Alternative Routing Reasons [SBCAlternativeRoutingReasons]	This table parameter configures the SBC Alternative Routing Reasons table. This table is used for alternative IP-to-IP routing. If 4xx, 5xx, or 6xx SIP responses are received as a result of outgoing SIP dialog-initiating methods (e.g., INVITE, OPTIONS, and SUBSCRIBE messages), the device re-sends the messages to an alternative route if the response is defined in this table and if there are alternative routes configured in the IP-to-IP Routing table. The format of this parameter is as follows: [SBCAlternativeRoutingReasons] FORMAT SBCAlternativeRoutingReasons_Index = SBCAlternativeRoutingReasons_ReleaseCause; [\SBCAlternativeRoutingReasons] For example: SBCAlternativeRoutingReasons 0 = 403; SBCAlternativeRoutingReasons 1 = 404; Note: For a detailed description of this table, see 'Configuring Alternative Routing Reasons' on page 286 .
IP to IP Inbound Manipulation Table	
Web: IP to IP Inbound Manipulation EMS: IP to IP Inbound Manipulation [IPInboundManipulation]	This table parameter configures the IP to IP Inbound Manipulation table. This table allows you to manipulate the SIP URI user part (source and/or destination) of the inbound SIP dialog message. The format of this parameter is as follows: [IPInboundManipulation] FORMAT IPInboundManipulation_Index =

Parameter	Description
	<p>IPInboundManipulation_IsAdditionalManipulation, IPInboundManipulation_ManipulatedURI, IPInboundManipulation_ManipulationPurpose, IPInboundManipulation_SrcIPGroupID, IPInboundManipulation_SrcUsernamePrefix, IPInboundManipulation_SrcHost, IPInboundManipulation_DestUsernamePrefix, IPInboundManipulation_DestHost, IPInboundManipulation_RequestType, IPInboundManipulation_RemoveFromLeft, IPInboundManipulation_RemoveFromRight, IPInboundManipulation_LeaveFromRight, IPInboundManipulation_Prefix2Add, IPInboundManipulation_Suffix2Add; [IPInboundManipulation]</p> <p>For example: IPInboundManipulation 1 = 0, 0, 0, -1, *, abc, *, *, 0, 0, 0, 255, , ;</p> <p>Note: For a detailed description of this table, see 'Configuring IP-to-IP Inbound Manipulations' on page 289.</p>
IP to IP Outbound Manipulation Table	
<p>Web: IP to IP Outbound Manipulation EMS: IP to IP Outbound Manipulation [IPOutboundManipulation]</p>	<p>This table parameter configures the IP to IP Outbound Manipulation table. This table allows you to manipulate the SIP URI user part (source and/or destination) of the outbound SIP dialog message. The format of this parameter is as follows:</p> <p>FORMAT IPOutboundManipulation_Index = IPOutboundManipulation_IsAdditionalManipulation, IPOutboundManipulation_SrcIPGroupID, IPOutboundManipulation_DestIPGroupID, IPOutboundManipulation_SrcUsernamePrefix, IPOutboundManipulation_SrcHost, IPOutboundManipulation_DestUsernamePrefix, IPOutboundManipulation_DestHost, IPOutboundManipulation_RequestType, IPOutboundManipulation_ReRouteIPGroupID, IPOutboundManipulation_Trigger, IPOutboundManipulation_ManipulatedURI, IPOutboundManipulation_RemoveFromLeft, IPOutboundManipulation_RemoveFromRight, IPOutboundManipulation_LeaveFromRight, IPOutboundManipulation_Prefix2Add, IPOutboundManipulation_Suffix2Add, IPOutboundManipulation_PrivacyRestrictionMode;</p> <p>For example: IPOutboundManipulation 1 = 0, 2, 4, "", "", "", "", 0, -1, 0, 0, 0, 0, 255, "", "", 0;</p> <p>Note: For a detailed description of this table, see 'Configuring IP-to-IP Outbound Manipulations' on page 292.</p>

42.14 Standalone Survivability Parameters

The Stand-alone Survivability (SAS) parameters are described in the table below.

Table 42-39: SAS Parameters

Parameter	Description
Web: Enable SAS EMS: Enable [EnableSAS]	<p>Enables the Stand-Alone Survivability (SAS) feature.</p> <ul style="list-style-type: none"> [0] Disable (default) [1] Enable <p>When enabled, the device receives the registration requests from different SIP entities in the local network and then forwards them to the defined proxy. If the connection to the proxy fails ('Emergency Mode'), the device serves as a proxy by allowing calls internal to the local network or outgoing to PSTN.</p> <p>Note: For this parameter to take effect, a device reset is required.</p>
Web: SAS Local SIP UDP Port EMS: Local SIP UDP [SASLocalSIPUDPPort]	<p>Defines the local UDP port for sending and receiving SIP messages for SAS. The SIP entities in the local network need to send the registration requests to this port. When forwarding the requests to the proxy ('Normal Mode'), this port serves as the source port.</p> <p>The valid range is 1 to 65,534. The default is 5080.</p>
Web: SAS Default Gateway IP EMS: Default Gateway IP [SASDefaultGatewayIP]	<p>Defines the Default Gateway used in SAS 'Emergency Mode'. When an incoming SIP INVITE is received and the destination Address-Of-Record is not included in the SAS database, the request is immediately sent to this default gateway.</p> <p>The address can be configured as an IP address (dotted-decimal notation) or as a domain name (up to 49 characters). You can also configure the IP address with a destination port, e.g., "10.1.2.3:5060". The default is a null string, i.e., the local IP address of the gateway.</p>
Web: SAS Registration Time EMS: Registration Time [SASRegistrationTime]	<p>Defines the value of the SIP Expires header that is sent in a 200 OK response to an incoming REGISTER message when in SAS 'Emergency Mode'.</p> <p>The valid range is 10 to 2,000,000. The default is 20.</p>
Web: SAS Local SIP TCP Port EMS: Local SIP TCP Port [SASLocalSIPTCPPort]	<p>Defines the local TCP port used to send/receive SIP messages for the SAS application. The SIP entities in the local network need to send the registration requests to this port. When forwarding the requests to the proxy ('Normal Mode'), this port serves as the source port.</p> <p>The valid range is 1 to 65,534. The default is 5080.</p>
Web: SAS Local SIP TLS Port EMS: Local SIP TLS Port [SASLocalSIPTLSPort]	<p>Defines the local TLS port used to send/receive SIP messages for the SAS application. The SIP entities in the local network need to send the registration requests to this port. When forwarding the requests to the proxy ('Normal Mode'), this port serves as the source port.</p> <p>The valid range is 1 to 65,534. The default is 5081.</p>
Web: SAS Connection Reuse [SASConnectionReuse]	<p>Enables the re-use of the same TCP connection for sessions with the same user in the SAS application.</p> <ul style="list-style-type: none"> [0] Disable

Parameter	Description
	<ul style="list-style-type: none"> [1] Enable (default) <p>The device can use the same TCP connection for multiple SIP requests / responses for a specific SIP UA. The benefits of this feature include less CPU and memory usage because fewer TCP connections are open and reduced network congestion. For example, assume the following:</p> <ul style="list-style-type: none"> User A sends a REGISTER message to SAS with transport=TCP. User B sends an INVITE message to A using SAS. <p>In this scenario, the SAS application forwards the INVITE request using the TCP connection that User A initially opened with the REGISTER message.</p>
Web/EMS: Enable Record-Route [SASEnableRecordRoute]	<p>Determines whether the device's SAS application adds the SIP Record-Route header to SIP requests. This ensures that SIP messages traverse the device's SAS agent by including the SAS IP address in the Record-Route header.</p> <ul style="list-style-type: none"> [0] Disable (default) [1] Enable <p>The Record-Route header is inserted in a request by a SAS proxy to force future requests in the dialog session to be routed through the SAS agent. Each traversed proxy in the path can insert this header, causing all future dialogs in the session to pass through it as well.</p> <p>When this feature is enabled, the SIP Record-Route header includes the URI "lr" parameter, indicating loose routing, for example:</p> <pre>Record-Route: <sip:server10.biloxi.com;lr></pre>
Web: SAS Proxy Set EMS: Proxy Set [SASProxySet]	<p>Defines the Proxy Set (index number) used in SAS Normal mode to forward REGISTER and INVITE requests from users that are served by the SAS application.</p> <p>The valid range is 0 to 5. The default is 0 (i.e., default Proxy Set).</p>
Web: Redundant SAS Proxy Set EMS: Redundant Proxy Set [RedundantSASProxySet]	<p>Defines the Proxy Set (index number) used in SAS Emergency mode for fallback when the user is not found in the Registered Users database. Each time a new SIP request arrives, the SAS application checks whether the user is listed in the registration database. If the user is located in the database, the request is sent to the user. If the user is not found, the request is forwarded to the next redundant SAS defined in the Redundant SAS Proxy Set. If that SAS Proxy IP appears in the Via header of the request, it is not forwarded (thereby, preventing loops in the request's course). If no such redundant SAS exists, the SAS sends the request to its default gateway (configured by the parameter SASDefaultGatewayIP).</p> <p>The valid range is -1 to 5. The default is -1 (i.e., no redundant Proxy Set).</p>
Web/EMS: SAS Block Unregistered Users [SASBlockUnRegUsers]	<p>Determines whether the device rejects SIP INVITE requests received from unregistered SAS users. This applies to SAS Normal and Emergency modes.</p> <ul style="list-style-type: none"> [0] Un-Block = (Default) Allow INVITE from unregistered SAS users.

Parameter	Description
	<ul style="list-style-type: none"> [1] Block = Reject dialog-establishment requests from un-registered SAS users.
[SASEnableContactReplace]	<p>Enables the device to change the SIP Contact header so that it points to the SAS host and therefore, the top-most SIP Via header and the Contact header point to the same host.</p> <ul style="list-style-type: none"> [0] (default) = Disable - when relaying requests, the SAS agent adds a new Via header (with the SAS IP address) as the top-most Via header and retains the original Contact header. Thus, the top-most Via header and the Contact header point to different hosts. [1] = Enable - the device changes the Contact header so that it points to the SAS host and therefore, the top-most Via header and the Contact header point to the same host. <p>Note: Operating in this mode causes all incoming dialog requests to traverse the SAS, which may cause load problems.</p>
Web: SAS Survivability Mode EMS: Survivability Mode [SASSurvivabilityMode]	<p>Determines the Survivability mode used by the SAS application.</p> <ul style="list-style-type: none"> [0] Standard = (Default) Incoming INVITE and REGISTER requests are forwarded to the defined Proxy list of SASProxySet in Normal mode and handled by the SAS application in Emergency mode. [1] Always Emergency = The SAS application does not use Keep-Alive messages towards the SASProxySet, instead it always operates in Emergency mode (as if no Proxy in the SASProxySet is available). [2] Ignore Register = Use regular SAS Normal/Emergency logic (same as option [0]), but when in Normal mode incoming REGISTER requests are ignored. [3] Auto-answer REGISTER = When in Normal mode, the device responds to received REGISTER requests by sending a SIP 200 OK (instead of relaying the registration requests to a Proxy), and enters the registrations in its SAS database. [4] Use Routing Table only in Normal mode = The device uses the IP-to-IP Routing table to route IP-to-IP SAS calls only when in SAS Normal mode (and is unavailable when SAS is in Emergency mode). This allows routing of SAS IP-to-IP calls to different destinations (and not only to the SAS Proxy Set).
Web: SAS Subscribe Response [SASSubscribeResponse]	<p>Defines the SIP response upon receipt of a SUBSCRIBE message when SAS is in Emergency mode. For example, if this parameter is set to "200", then SAS sends a SIP 200 OK in response to a SUBSCRIBE message, when in Emergency mode.</p> <p>The valid value is 200 to 699. The default is 489.</p>
Web: Enable ENUM [SASEnableENUM]	<p>Enables SAS to perform ENUM (E.164 number to URI mapping) queries when receiving INVITE messages in SAS emergency mode.</p> <ul style="list-style-type: none"> [0] Disable (default) [1] Enable
Web: SAS Binding Mode EMS: Binding Mode [SASBindingMode]	<p>Determines the SAS application database binding mode.</p> <ul style="list-style-type: none"> [0] URI = (Default) If the incoming AoR in the INVITE requests is using a 'tel:' URI or 'user=phone' is defined, the binding is performed according to the user part of the URI

Parameter	Description
	<p>only. Otherwise, the binding is according to the entire URI, i.e., User@Host.</p> <ul style="list-style-type: none"> ▪ [1] User Part only = The binding is always performed according to the User Part only.
Web: SAS Emergency Numbers [SASEmergencyNumbers]	<p>Defines emergency numbers for the device's SAS application. When the device's SAS agent receives a SIP INVITE (from an IP phone) that includes one of the emergency numbers (in the SIP user part), it forwards the INVITE to the default gateway (configured by the parameter SASDefaultGatewayIP), i.e., the device itself, which sends the call directly to the PSTN. This is important for routing emergency numbers such as 911 (in North America) directly to the PSTN. This is applicable to SAS operating in Normal and Emergency modes.</p> <p>Up to four emergency numbers can be defined, where each number can be up to four digits.</p>
[SASEmergencyPrefix]	<p>Defines a prefix that is added to the Request-URI user part of the INVITE message that is sent by the device's SAS agent when in Emergency mode to the default gateway or to any other destination (using the IP-to-IP Routing table). This parameter is required to differentiate between normal SAS calls routed to the default gateway and emergency SAS calls. Therefore, this allows you to define different manipulation rules for normal and emergency calls.</p> <p>This valid value is a character string. The default is an empty "" string.</p>
Web: SAS Entering Emergency Mode [SASEnteringEmergencyMode]	<p>Determines for which sent SIP message types the device enters SAS Emergency mode if no response is received for them from the proxy server.</p> <ul style="list-style-type: none"> ▪ [0] = (Default) SAS enters Emergency mode only if no response is received from sent SIP OPTIONS messages. ▪ [1] = SAS enters Emergency mode if no response is received from sent SIP OPTIONS, INVITE, or REGISTER messages. <p>Note: If the keep-alive mechanism is disabled for the Proxy Set (in the Proxy Set table) and this parameter is set to [1], SAS enters Emergency mode only if no response is received from sent INVITE or REGISTER messages.</p>
Web: SAS Inbound Manipulation Mode [SASInboundManipulationMode]	<p>Enables destination number manipulation of incoming INVITE messages when SAS is in Emergency mode. The manipulation rule is done in the IP to IP Inbound Manipulation table.</p> <ul style="list-style-type: none"> ▪ [0] None (default) ▪ [1] Emergency Only <p>Notes:</p> <ul style="list-style-type: none"> ▪ Inbound manipulation applies only to INVITE requests. ▪ For more information on SAS inbound manipulation, see 'Manipulating Destination Number of Incoming INVITE' on page 323.
SAS Registration Manipulation Table	
Web: SAS Registration Manipulation EMS: Stand-Alone Survivability	<p>This table parameter configures the SAS Registration Manipulation table. This table is used by the SAS application to manipulate the SIP Request-URI user part of incoming INVITE</p>

Parameter	Description
[SASRegistrationManipulation]	<p>messages and of incoming REGISTER request AoR (To header), before saving it to the registered users database. The format of this table parameter is as follows:</p> <p>[SASRegistrationManipulation] FORMAT SASRegistrationManipulation_Index = SASRegistrationManipulation_RemoveFromRight, SASRegistrationManipulation_LeaveFromRight; [SASRegistrationManipulation]</p> <p>For example, the manipulation rule below routes an INVITE with Request-URI header "sip:7184002@10.33.4.226" to user "4002@10.33.4.226" (i.e., keep only four digits from right of user part):</p> <pre>SASRegistrationManipulation 0 = 0, 4;</pre> <p>Note: For a detailed description of this table, see 'Manipulating URI user part of Incoming REGISTER' on page 322.</p>
Web: SAS IP-to-IP Routing Table	
[IP2IPRouting]	<p>This table parameter configures the IP-to-IP Routing table for SAS routing rules. The format of this parameter is as follows:</p> <p>[IP2IPRouting] FORMAT IP2IPRouting_Index = IP2IPRouting_SrcIPGroupID, IP2IPRouting_SrcUsernamePrefix, IP2IPRouting_SrcHost, IP2IPRouting_DestUsernamePrefix, IP2IPRouting_DestHost, IP2IPRouting_DestType, IP2IPRouting_DestIPGroupID, IP2IPRouting_DestSRDID, IP2IPRouting_DestAddress, IP2IPRouting_DestPort, IP2IPRouting_DestTransportType, IP2IPRouting_AltRouteOptions; [IP2IPRouting]</p> <p>For example:</p> <pre>IP2IPRouting 1 = -1, *, *, *, *, 0, -1, -1, , 0, -1, 0;</pre> <p>Note: For a detailed description of this table parameter, see 'SAS Routing Based on IP-to-IP Routing Table' on page 326.</p>

42.15 IP Media Parameters

The IP media parameters are described in the table below.

Table 42-40: IP Media Parameters

Parameter	Description
Web: Number of Media Channels EMS: Media Channels [MediaChannels]	<p>Defines the maximum number of DSP channels allocated for various functionalities.</p> <p>The default is 0.</p> <p>Notes:</p> <ul style="list-style-type: none"> For this parameter to take effect, a device reset is required. The SBC application does not require DSP channels. The SBC application uses DSP channels only if media transcoding is needed, where two DSP channels are used per transcoding session.

42.16 Auxiliary and Configuration File Name Parameters

The configuration files (i.e., auxiliary files) can be loaded to the device using the Web interface. For loading these files using the *ini* file, you need to configure these files in the *ini* file and configured whether they must be stored in the non-volatile memory. The table below lists the *ini* file parameters associated with these auxiliary files. For more information on the auxiliary files, see 'Loading Auxiliary Files' on page 363.

Table 42-41: Auxiliary and Configuration File Parameters

Parameter	Description
General Parameters	
[SetDefaultOnIniFileProcess]	<p>Determines if all the device's parameters are set to their defaults before processing the updated <i>ini</i> file.</p> <ul style="list-style-type: none"> ▪ [0] = Disable - parameters not included in the downloaded <i>ini</i> file are not returned to default settings (i.e., retain their current settings). ▪ [1] = Enable (default). <p>Note: This parameter is applicable only for automatic HTTP update or Web <i>ini</i> file upload (not applicable if the <i>ini</i> file is loaded using BootP).</p>
[SaveConfiguration]	<p>Determines if the device's configuration (parameters and files) is saved to flash (non-volatile memory).</p> <ul style="list-style-type: none"> ▪ [0] = Configuration isn't saved to flash memory. ▪ [1] = (Default) Configuration is saved to flash memory.
Auxiliary and Configuration File Name Parameters	
Web/EMS: Call Progress Tones File [CallProgressTonesFilename]	<p>Defines the name of the file containing the Call Progress Tones definitions. For more information on how to create and load this file, refer to <i>DConvert Utility User's Guide</i>.</p> <p>Note: For this parameter to take effect, a device reset is required.</p>
Web: Dial Plan File EMS: Dial Plan File Name [DialPlanFileName]	<p>Defines the name (and path) of the Dial Plan file. This file should be created using AudioCodes DConvert utility (refer to <i>DConvert Utility User's Guide</i>).</p>
[UserInfoFileName]	<p>Defines the name (and path) of the file containing the User Information data.</p>

42.17 Automatic Update Parameters

The automatic update of software and configuration files parameters are described in the table below.

Table 42-42: Automatic Update of Software and Configuration Files Parameters

Parameter	Description
General Automatic Update Parameters	
[AutoUpdateCmpFile]	<p>Enables the Automatic Update mechanism for the cmp file.</p> <ul style="list-style-type: none"> [0] = (Default) The Automatic Update mechanism doesn't apply to the cmp file. [1] = The Automatic Update mechanism includes the cmp file. <p>Note: For this parameter to take effect, a device reset is required.</p>
[AutoUpdateFrequency]	<p>Defines the number of minutes that the device waits between automatic updates. The default is 0 (i.e., the update at fixed intervals mechanism is disabled).</p> <p>Note: For this parameter to take effect, a device reset is required.</p>
[AutoUpdatePredefinedTime]	<p>Defines schedules (time of day) for automatic updates. The format of this parameter is: 'HH:MM', where <i>HH</i> denotes the hour and <i>MM</i> the minutes, for example, 20:18.</p> <p>Notes:</p> <ul style="list-style-type: none"> For this parameter to take effect, a device reset is required. The actual update time is randomized by five minutes to reduce the load on the Web servers.
EMS: AUPD Verify Certificates [AUPDVerifyCertificates]	<p>Determines whether the Automatic Update mechanism verifies server certificates when using HTTPS.</p> <ul style="list-style-type: none"> [0] = Disable (default) [1] = Enable
[AUPDCheckIfIniChanged]	<p>Determines whether the Automatic Update mechanism performs CRC checking to determine if the <i>ini</i> file has changed prior to processing.</p> <ul style="list-style-type: none"> [0] = (Default) Do not check CRC. The <i>ini</i> file is loaded whenever the server provides it. [1] = Check CRC for the entire file. Any change, including line order, causes the <i>ini</i> file to be re-processed. [2] = Check CRC for individual lines. Use this option when the HTTP server scrambles the order of lines in the provided <i>ini</i> file.
[ResetNow]	<p>Invokes an immediate device reset. This option can be used to activate offline (i.e., not on-the-fly) parameters that are loaded using the parameter IniFileUrl.</p> <ul style="list-style-type: none"> [0] = (Default) The immediate restart mechanism is disabled. [1] = The device immediately resets after an <i>ini</i> file with this parameter set to 1 is loaded.
Software/Configuration File URL Path for Automatic Update Parameters	
[CmpFileURL]	<p>Defines the name of the <i>cmp</i> file and the path to the server (IP address or FQDN) from where the device can load the <i>cmp</i> file and update itself. The <i>cmp</i> file can be loaded using HTTP/HTTPS. For example: http://192.168.0.1/filename</p>

Parameter	Description
	<p>Notes:</p> <ul style="list-style-type: none"> For this parameter to take effect, a device reset is required. When this parameter is configured, the device always loads the <i>cmp</i> file after it is reset. The <i>cmp</i> file is validated before it's burned to flash. The checksum of the <i>cmp</i> file is also compared to the previously burnt checksum to avoid unnecessary resets. The maximum length of the URL address is 255 characters.
[IniFileURL]	<p>Defines the name of the <i>ini</i> file and the path to the server (IP address or FQDN) on which it is located. The <i>ini</i> file can be loaded using HTTP/HTTPS.</p> <p>For example: http://192.168.0.1/filename <a href="http://192.8.77.13/config<MAC>">http://192.8.77.13/config<MAC> <a href="https://<username>:<password>@<IP address>/<file name>">https://<username>:<password>@<IP address>/<file name></p> <p>Notes:</p> <ul style="list-style-type: none"> For this parameter to take effect, a device reset is required. When using HTTP or HTTPS, the date and time of the <i>ini</i> file are validated. Only more recently dated <i>ini</i> files are loaded. The optional string <MAC> is replaced with the device's MAC address. Therefore, the device requests an <i>ini</i> file name that contains its MAC address. This option allows the loading of specific configurations for specific devices. The maximum length of the URL address is 99 characters.
[CptFileURL]	<p>Defines the name of the CPT file and the path to the server (IP address or FQDN) on which it is located.</p> <p>For example: http://server_name/file, https://server_name/file.</p> <p>Note: The maximum length of the URL address is 99 characters.</p>
[TLSRootFileUrl]	<p>Defines the name of the TLS trusted root certificate file and the URL from where it can be downloaded.</p> <p>Note: For this parameter to take effect, a device reset is required.</p>
[TLSCertFileUrl]	<p>Defines the name of the TLS certificate file and the URL from where it can be downloaded.</p> <p>Note: For this parameter to take effect, a device reset is required.</p>
[TLSPkeyFileUrl]	<p>Defines the URL for downloading a TLS private key file using the Automatic Update facility.</p>
[UserInfoFileURL]	<p>Defines the name of the User Information file and the path to the server (IP address or FQDN) on which it is located.</p> <p>For example: http://server_name/file, https://server_name/file</p> <p>Note: The maximum length of the URL address is 99 characters.</p>

Reader's Notes

43 DSP Templates

The maximum number of supported SBC sessions is mentioned in 'Technical Specifications' on page 535. These SBC sessions also support SRTP and RTCP XR. When DSP capabilities are required, the number of sessions that can use DSP capabilities is reduced, as shown in the table below:

Table 43-1: Channel Capacity and Capabilities for Mediant 4000 SBC

Basic DSP Capabilities	Advanced DSP Capabilities					Min. DSP Sessions	
	SBC Enhancements	G.722	AMR WB	SILK	SILK WB	MPM	No MPM
✓	-	-	-	-	-	1050	350
✓	-	✓	-	-	-	950	300
✓	-	✓	-	✓	-	700	200
✓	-	✓	✓	✓	✓	550	150
✓	✓	✓	✓	✓	✓	450	150
✓	✓	✓	-	✓	-	600	200
✓	✓	✓	-	-	-	800	250
✓	✓	-	-	-	-	850	250



Notes:

- *Min. DSP Sessions* refers to the minimum number of sessions available when using the selected DSP capabilities. When using only some of the DSP capabilities or using them only on some channels, the number of available DSP sessions may increase.
- *Basic DSP Capabilities* refers to the Network Acoustic Echo Suppressor.
- *SBC Enhancements* refers to the following channel capabilities that require DSP resources: IBS, Silence Compression, T.38, G.711, G.726, G.729, G.723.1, and AMR.
- *MPM* refers to the optional Media Processing Module.

Reader's Notes

44 Technical Specifications

The device's technical specifications are listed in the table below.



Notes:

- All specifications in this document are subject to change without prior notice.
- The compliance and regulatory information can be downloaded from AudioCodes Web site at <http://www.audiocodes.com/library>.

Table 44-1: Technical Specifications

Function	Specification
Capacity	
SBC Sessions	<ul style="list-style-type: none"> ▪ RTP-RTP: 4,000 ▪ SRTP-RTP: 2,000
Transcoding Sessions	1,092
Registered Users (SBC, CRP, SAS)	20,000
Networking Interfaces	
LAN/WAN	<ul style="list-style-type: none"> ▪ Eight 1000Base-T (Gigabit Ethernet) Ethernet ports operating in pairs (i.e., 4 port pairs) for 1+1 redundancy (where one port is active and the other standby). ▪ Physical port separation by selecting port group per network interface.
High Availability (HA)	
Full HA	Two deployed devices for 1+1 high availability, communicating through a Maintenance network interface. Upon failure of the active device, all functionality is switched over to the redundant device.
Media Processing	
IP Transport	VoIP (RTP/RTCP) per IETF RFC 3550 and 3551, IPv6
Control and Management	
Control Protocols	<ul style="list-style-type: none"> ▪ SIP-TCP, UDP, TLS and MSCML ▪ Stand Alone Survivability for service continuity
Operations & Management	<ul style="list-style-type: none"> ▪ Embedded HTTP Web Server, Telnet, SNMP V2/V3 ▪ Remote configuration and software download via TFTP, HTTP, HTTPS, DHCP ▪ RADIUS, Syslog (for events, alarms and CDRs)
IP/VoIP Quality of Service	
	<ul style="list-style-type: none"> ▪ IEEE 802.1p, TOS, DiffServ ▪ IEEE 802.1Q VLAN tagging ▪ Shaping, Policing, Queuing, Bandwidth Reservation ▪ RTCP XR Report Publish (RFC 3611)

Function	Specification
Session Border Controller	
	<ul style="list-style-type: none"> SIP Header conversion: IP to IP Routing translations of SIP, UDP, TCP, TLS. Translation of RTP, SRTP; Support SIP trunk with multi-ITSP (Registrations to ITSPs is invoked independently); Topology hiding; Call Admission Control; Call Black/White list. Intrusion detection/prevention (NIDS); Anti SPIT & SPAM mechanisms.
Hardware Specifications	
E-SBC CPU AMC Module	<ul style="list-style-type: none"> CPU. Sub-module providing eight 1000Base-T (Gigabit Ethernet) ports, operating in pairs for 1+1 port redundancy (one active; the other standby). Supports auto-negotiation, half- and full-duplex modes, and straight-through and crossover cable detection. Reset pinhole button for resetting the device. Serial interface. LEDs for indicating status of power, Ethernet, and operation.
Media Processing Module (MPM)	Optional module for providing DSP resources for additional IP-to-IP transcoding sessions
Fan Tray	Includes six integrated fans for cooling the internal chassis components.
Power Supply	<ul style="list-style-type: none"> Up to two hot swappable, power supply modules for power load sharing and AC power redundancy in case of failure of one of the modules. Input ratings: Single universal power supply 100-240V, 50-60 Hz, 2.5A max Output ratings: 12V / 10A max. Connection to the electrical outlet through an AC power supply inlet.
Enclosure	4/8-slot, 1U chassis
Dimensions (HxWxD)	1U x 19" (444 mm) x 14" (355 mm)
Weight Approx.	11.7 lbs (5.3 kg)

Reader's Notes



User's Manual Ver. 6.6